

Efficient Parameter Extraction of Solar Cell using Modified ABC

Maitreyi Ketkar
E & TC Engineering
Vishwakarma Institute of Technology
Pune, India

A. M. Chopde
HOD E & TC Engineering
Vishwakarma Institute of Technology
Pune, India

ABSTRACT

In this paper an efficient method for parameter extraction of solar cell double diode model using Artificial Bee Colony (ABC) Algorithm is presented. ABC packs solid features such as simplicity of implementation, promising optimization capability, fewer control parameters, etc. In this paper we have implemented various methods suggested to boost traditional ABC performance to solve multi-parameter optimization problem. The results demonstrate ability of modified algorithm to be primary candidate for the parameter extraction in wide search space.

Keywords:

Parameter Extraction, Solar Cell, Artificial Bee Colony, ABC, Double Diode Model

1. INTRODUCTION

Solar energy has widely gained importance because of several promising features like renewability and nature friendly source of energy. Solar cells are basic module of photovoltaic systems (PV systems) in the generation of solar power. PV systems consist of different parts centered around a solar panel that typically has arrays of interconnected solar cells. Hence, it is very important to choose a model which will closely emulate characteristics of PV cell. This is not only important for analytical purpose but also important in designing of large scale solar energy systems. The use of lumped circuit models is a convenient and widely used method for simulating solar-cell performance. In practice, two main equivalent circuit models are primarily used: Single and Double diode models. For both the models main parameters that describe the behavior are the generated photocurrent (I_{ph}), saturation current (I_s), series resistance (R_s), shunt resistance (R_{sh}), and ideality factor (η). Accurate estimation of these parameters is always required to provide better modeling and efficient performance evaluation of a given solar system[2].

The traditional extraction methods were based on direct approaches on the use of I-V curve features such as axis intercepts and the gradients at selected points, to determine some cell parameters. However accuracy of these techniques was limited due to non-linearity of measured I-V data, multi variable and multi modal problem which has many local optima[1][4]. The newly developed algorithms are based on nature inspired ideas, such as ant colony optimisation, evolutionary algorithms, particle swarm op-

timisation etc[8]. Most of these algorithms are meta-heuristic and they have applicability to a wide range of problems. In a similar context Artificial Bee Colony (ABC) algorithm was initially proposed by Karaboga in 2005 as a technical report for numerical optimisation problem[6]. Artificial Bee Colony Algorithm (ABC) is nature-inspired meta-heuristic, which imitates the foraging behavior of bees.

The algorithm is composed of three important elements: employed and unemployed foragers, and food sources. The employed and unemployed foragers are the first two elements, while the third element is the rich food sources close to their hive. The behaviour of the algorithm can be described by two modes[3];

1. Recruitment of forager bees to rich food sources, resulting into positive feedback and
2. Simultaneously, the abandonment of poor sources by foragers, which causes negative feedback.

The positions of the food sources are randomly selected by the bees at the initialization stage and their nectar qualities are measured. The bee colony consists of three groups of artificial bees[10]: Employed bees, Onlooker bees and scout bees. The objective of bee colony is to select food source with rich nectar quality and maximize nectar amount in hive. The different phases of the foraging are:

- (i). Initially there are some scout bees and onlooker bees in colony. The unemployed scout bees starts searching for a food sources randomly near their hives. Once scout bees find a food source they are transformed into employed bees. So number of employed bees is equal to number of food sources and is equal to half the colony size.
- (ii). Employed bees collect nectar, calculates the nectar quality and go back to hive. Employed bees shares the information about nectar to onlooker bees by performing waggle dance. The direction of bees indicates the direction of the food source relative to the Sun, the intensity of the waggles indicates how far away it is and the duration of the dance indicates the amount of nectar on food source[12]. After sharing this information, every employed bees returns to the previously visited food source, since the position of the food source had been memorized and then selects another food source in the neighborhood of the present one and again share this information to onlooker bees.
- (iii). Onlooker bees use the information of food sources obtained from the employed bees to select a food source. The food sources with highest amount of nectar are selected by onlooker bees. The probability of selecting food source increases with increase in its

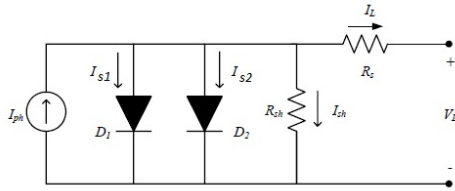


Fig. 1. Double diode solar cell model

nectar quality. Then onlooker bee performs neighbourhood search around the current food source in her memory and the process continues until the best food source is found (Global Optima).
(iv). If the food source is exhausted (not improved within the specified limit), then corresponding employed bees will abandon it, transform into scout bees and search for new food source randomly.

2. PROBLEM FORMULATION

The two main lumped parameter equivalent circuits are used in practice to describe solar cells. The first one is the single diode model and second is double diode model. Under illumination and normal operating conditions, the single diode model is suitable for solar cells. However, the single diode model is particularly inaccurate in describing cell behavior at low illuminations [2]. Hence, the double diode model is the most popular and accurate equivalent circuit to simulate solar cell. This paper describes the parameter extraction of solar cell double diode model using ABC.

2.1 Double Diode Model

In this model, the solar cell is modeled as a current source connected in parallel with a rectifying diode. However, in practice the current source is also shunted by another diode that models the space charge recombination current and a shunt leakage resistor to account for the partial short circuit current path near the cells edges due to the semiconductor impurities and non-idealities. In addition, the solar cell metal contacts and the semiconductor material bulk resistance are represented by a resistor connected in series with the cell shunt elements. The double diode model can be formulated as follows:

$$I_L = I_{ph} - I_{D1} - I_{D2} - I_{sh} \quad (1)$$

where I_L is cell output current and V_L is cell output voltage, I_{ph} indicates photogenerated current. I_{sh} is current through R_{sh} . I_{D1} and I_{D2} are diode currents and can be calculated using Shockley equation as shown below:

$$I = I_{ph} - I_{s1} \left[\exp\left(\frac{q(V_L + R_s I_L)}{\eta_1 K T}\right) - 1 \right] - I_{s2} \left[\exp\left(\frac{q(V_L + R_s I_L)}{\eta_2 K T}\right) - 1 \right] - \frac{(V_L + R_s I_L)}{R_{sh}} \quad (2)$$

where I_{s1} and I_{s2} are diffusion and saturation currents respectively. η_1 and η_2 denotes diffusion and recombination diode ideality factors. q is an electronic charge (1.6×10^{-19} C), K is Boltzmann constant ($1.3806503 \times 10^{-23}$ J/K) and T is junction Temperature in Kelvin.

In double diode model, the seven parameters (I_{ph} , I_{s1} , I_{s2} , η_1 , η_2 , R_s , R_{sh}) are extracted from the IV data of solar cell.

2.2 Function to be minimized

The objective function is derived from double diode model current equation (2). The root mean square error (RMSE) is used as objective function to be minimized [4], as shown below:

$$F(X) = \sqrt{\frac{1}{N} \sum_{k=1}^N f_k(V_L, I_L, X)^2} \quad (3)$$

where N is number of experimental data. X is a decision vector which consists of parameters to be extracted. To put in simple words, we will need to minimize difference between $I_{L_{Exp}}$ and $I_{L_{Calc}}$ on the basis of RMSE function. Every parameter is bounded in a search space called as upper and lower bounds. These are shown in table (1).

Table 1. Double diode model parameters with range

Parameter	Lower bound	Upper bound
I_{ph}	0.2	1
I_{s1}	1e-10	1e-05
I_{s2}	1e-10	1e-05
η_1	1	2
η_2	1	2
R_s	0.001	10
R_{sh}	1	1000

3. ABC ALGORITHM

The procedure of ABC could be described in the following steps [6][9]:

3.1 Initialization

The number of colony size (NP) and scout bee triggering threshold ($Limit$) are key parameters of ABC. Maximum cycle number ($maxCycle$) is set for terminating criteria. The parameters to be optimized are D . The lower bound LB and upper bound UB is specified for each parameter. The food sources ($FoodNumber$) are initialized and there are $NP/2$. The Food Source Memory (Foods) is an augmented matrix of size $FoodNumber \times D$ comprised in each row, a vector representing food source (Foods matrix) as shown below.

$$Foods = \begin{bmatrix} x_1(1) & \cdots & x_1(D) \\ x_2(1) & \cdots & x_2(D) \\ \cdots & \ddots & \cdots \\ x_{FoodNumber}(1) & \cdots & x_{FoodNumber}(D) \end{bmatrix} \quad (4)$$

Each vector is generated as follows

$$x_j(i) = LB_j(i) + \Phi(UB_j(i) - LB_j(i)) \quad (5)$$

$$\forall j \in (1, 2, \dots, FoodNumber), \forall i \in (1, 2, \dots, D)$$

Note that $\Phi \sim (0, 1)$ is any random number between 0 and 1.

3.2 Employed Bee Phase

In this step, each employee bee is assigned to its food source and in turn, a new one is generated from its neighbouring solution, using equation

$$x'_j(i) = x_j(i) \pm \Phi(x_j(i) - x_k(i)) \quad (6)$$

$$\forall k \in (1, 2, \dots, FoodNumber)$$

where $x'_j(i)$ is new food source generated and $x_k(i)$ is a randomly chosen food source taken from the current population. The fitness of original and modified food source is estimated and the food source is replaced by new food source if its fitness is better than original one. If fitness of EB is not improved after modification, *trial* count will be incremented by 1.

3.3 Onlooker Bee Phase

The OB has the same number of food sources as the EB. It initially calculates the selection probability of each food source generated by the EB in the previous step. The fittest food source is selected by the onlooker, using Roulette Wheel selection mechanism. The process of selection at the onlooker phase works as follows:

$$Fitness(j) = \frac{1}{1 + f(x)} \quad (7)$$

$$p_j = \frac{0.9(Fitness(j))}{max(Fitness(j))} + 0.1 \quad (8)$$

The food source of the EB with the highest fitness is selected by the OB, based on its selection probability. Then each solution will be modified again by OB, using same steps as employed bee phase.

3.4 Scout Bee phase

If the *trial* count of any food source exceeds the scout bee triggering threshold *Limit*, then that corresponding food source is abandoned and it is replaced by new food source using equation (5).

3.5 Memorizing the best food source

This involves memorizing the fitness and position of the best food source, x^{best} found so far in *Foods* matrix. Above steps are repeated until a stop criterion is met. If counter is reached to *maxCycle* then algorithm is over.

4. MODIFIED ABC ALGORITHM

The traditional ABC algorithm suffers from the drawbacks like slow convergence, converging to local optima and inability to optimize multiple parameters simultaneously. The drawbacks arise from the fact that, in traditional ABC simultaneously two processes occur; one is replacement of food sources and another is optimisation of current food sources. The two processes should be balanced well in order to converge faster and avoid local minima. Below are few modifications suggested to traditional ABC for countering the above mentioned problems.

4.1 Pre-filtering of food sources based on fitness:

While testing the effectiveness of the algorithm, to carry out multiple parameter optimization, we chose to provide wide search space for each parameter as shown in table(1). Random food sources generated during initialization phase proved to be deviating the algorithm from optimum startpoint. This was mainly because of small

colonysize operating on huge search space. If colony size was to be increased, it would have caused algorithm to take a much longer time for convergence. A filtering algorithm was introduced to keep the colony size small but still provide effective startpoint from large search space. At the initialization phase, colony size was taken to be 20 or 25 times the actual *FoodSources*. The *Fitness* is evaluated for each colony member. The colony sources are later arranged in descending order of their fitness (food source with maximum fitness is most likely candidate for good startpoint). The top members equal to *FoodSources* were selected as the final food source for initialisation of algorithm. This also propelled algorithm to converge faster by filtering out sub-optimal food sources at early stages.

4.2 Dynamic ABC

Overall idea of ABC algorithm implies that global searching is performed by employed bees and local searching is implemented by onlooker bees. However, this idea is not prominent in traditional ABC algorithms, because the modifying extent of employed bees and onlooker bees is similar and relatively fixed. In order to improve the efficiency of algorithm, the authors of [10] proposed the concept of dynamic ABC. In order to make the activity of bees dynamic in different stages of the algorithm, global searching should be dominant in early cycles and local searching should be dominant in the posterior cycles. This could be more consistent with actions of real bees: Employed bees become main force in the early cycles, then more and more onlookers follow, they play the major role afterwards. Specifically, in early stages of optimization, audaciously modify more elements of food vector in employed bees phase. That makes the bees approaching better solution by a greater probability. Furthermore, onlooker bees becomes active in posterior stages, and they modify more elements of food vector. That provides more opportunities to jump out of local optimal solution. Following scheme is suggested for implementation of D-ABC algorithm [10].

1. In early stages, for employed bees phase the number of parameters to be modified (out of D parameters) of a food source under consideration N_{EB} is defined by formula below. N_{EB} reduces as current cycle (*iter*) increases, and N_{EB} elements are randomly picked to be modified. For onlooker bees phase, MR method (modification rate) is recommended [7]. MR is constant and its optimum value is found to be between 0.3 to 0.4 through various experiments [7].

$$\text{If } iter \leq \frac{maxCycle}{2}$$

—Employed bees stage:

$$N_{EB} = (1 - \frac{iter}{maxCycle}) \times D \quad (9)$$

—Onlooker bees stage:

$$\begin{aligned} x'_j(i) &= x_j(i) \pm 2\phi(x_j(i) - x_k(i)) & \text{if } \phi < MR \\ x'_j(i) &= x_j(i) & \text{otherwise} \end{aligned} \quad (10)$$

2. In posterior stages, for employed bees phase, MR method is reused[7],for employed bees phase. For onlooker bee phase, the number of parameters to be modified of a food source under consideration N_{OB} is defined by formula below. N_{OB} increases as current cycle (*iter*) increases, and N_{OB} elements are randomly picked to be modified.

If $iter > \frac{maxCycle}{2}$

—Employed bees stage:

$$\begin{aligned} x'_j(i) &= x_j(i) \pm 2\phi(x_j(i) - x_k(i)) & \text{if } \phi < MR \\ x'_j(i) &= x_j(i) & \text{otherwise} \end{aligned} \quad (11)$$

—Onlooker bees stage:

$$N_{OB} = \frac{iter}{maxCycle} \times D \quad (12)$$

Furthermore, D-ABC algorithm is closely related to the length D of solution vector. When D is small, there is practically little difference between modified ABC algorithm and D-ABC algorithm. And for larger D, the advantages of D-ABC algorithm are prominent on convergence rate and improving the quality of solutions[10].

4.3 Modified Employed Bee Operator

a. Authors of [9] suggested this modification at employed bee stage. Acting on every food source, the employed bee modifies its food source by using the modified operator in Equation

$$x'_j(i) = x_j(i) \pm 2\phi(x_j(i) - x_k(i)) \quad (13)$$

$k \in \{1, 2, \dots, FoodNumber\}$

ϕ is multiplied by 2 in order to increase step size of operator(wider search space). This ϕ value is within range $[0, 1]$.

b. It is useful to memorise best food source obtained in previous iteration and then utilise it during next employed bee phase. This provides locally skewed search space near best food source during employed bee phase.

$$\begin{aligned} x'_j(i) &= x_j(i) + 2\phi(x_j(i) - x_{k1}(i)) \\ &\quad - 2\phi(x_{best-prev-iter} - x_{k2}(i)) \end{aligned} \quad (14)$$

$k \in \{1, 2, \dots, FoodNumber\}$

c. In traditional ABC algorithm, if the modified parameter evaluated at employed bee phase exceeds its upper bound then it was replaced by its upper bound value and if it falls below lower bound, then it was replaced by its lower bound value. But the ranges that we have provided are quite wide and replacing food sources by upper or lower bounds will diverge the algorithm from optimum solution. Hence, its value is replaced by random value using equation

$$x'_j(i) = LB(i) + \phi(UB(i) - LB(i)) \quad (15)$$

for all $j \in (1, 2, \dots, FoodNumber)$, for all $i \in (1, 2, \dots, D)$

4.4 Modified Onlooker Bee Operator

a. Authors of [9] suggested this modification at onlooker bee stage. Acting on every food source, the onlooker bee modifies its food source by using the modified operator in Equation

$$x'_j(i) = x_j(i) \pm 2\phi(x_j(i) - x_B) \quad (16)$$

$k \in \{1, 2, \dots, FoodNumber\}$

Where, x_B is a best food source found so far. This will speed up the convergence rate of algorithm. ϕ is multiplied by 2 in order to increase step size of operator(wider search space). This ϕ value is within range $[0, 1]$.

b. In traditional ABC algorithm, if the modified parameter evaluated at employed bee phase exceeds its upper bound then it was

replaced by its upper bound value and if it falls below lower bound, then it was replaced by its lower bound value. But the ranges that we have provided are quite wide and replacing food sources by upper or lower bounds will diverge the algorithm from optimum solution. Hence, its value is replaced by random value using equation

$$x'_j(i) = LB(i) + \phi(UB(i) - LB(i)) \quad (17)$$

for all $j \in (1, 2, \dots, FoodNumber)$, for all $i \in (1, 2, \dots, D)$

4.5 Modified Scout Bee Operator

Instead of generating a random food source(as in the original ABC), the scout bee will generate a new food source $x'_j(i)$ by using the food source subject to be replaced $x_j(i)$ as a base to generate a new search direction biased by the best food source so far $x_B(i)$ and a randomly chosen food source $x_k(i)$ as indicated in Equation below[9], where the ϕ value is generated and fixed per each food source $k \in (1, 2, \dots, FoodNumber)$.

$$\begin{aligned} x'_j(i) &= x_j(i) + \phi(x_j(i) - x_k(i)) \\ &\quad + (1 - \phi)(x_B(i) - x_j(i)) \end{aligned} \quad (18)$$

The aim of this modified operator is to increase the capabilities of the algorithm to sample solutions within the range of search defined by the current population[9].

5. EXPERIMENTAL RESULTS AND ANALYSIS

We verified the proposed method of parameter extraction of solar cell using both synthetic and real world experimental I-V data. The entire algorithm of modified ABC was implemented in MATLAB. The effectiveness of proposed algorithm was compared with the traditional algorithm in terms of obtained I-V characteristics, RMSE value and relative error of extracted parameters.

5.1 Parameter extraction with synthetic I-V data

At first, we consider the parameter extraction of solar cell with synthetic I-V data in order to test accuracy of modified ABC algorithm against Traditional ABC algorithm. Table(2) shows default values of parameters of solar cell double diode model used to calculate synthetic I-V data.

Table 2. Default parameter values for double diode model

Name	I_{ph}	I_{s1}	I_{s2}	n_1	n_2	R_s	R_{sh}
Synthetic data	0.9072	2.83E-05	2.47E-09	2	1	0.03117	19.92

The convergence rate of traditional and modified ABC are compared in fig(2). It is clear that, initially traditional ABC converges faster than modified ABC. However, as the algorithm progresses, traditional ABC is seen to be difficult to converge to a global optimum solution. The modified ABC shows a much better convergence rate in posterior cycles. If the number of iterations are large enough, then modified ABC can generate better solutions than traditional ABC.

The extracted parameter values of solar cell are summarised in table(3). As can be seen the extracted values of solar cell by modified ABC method are in better synchronization with default parameters. The RMSE value of modified ABC is also very less compared to traditional ABC which corresponds to good parameter extraction. Also, it can be seen that with traditional ABC the relative errors

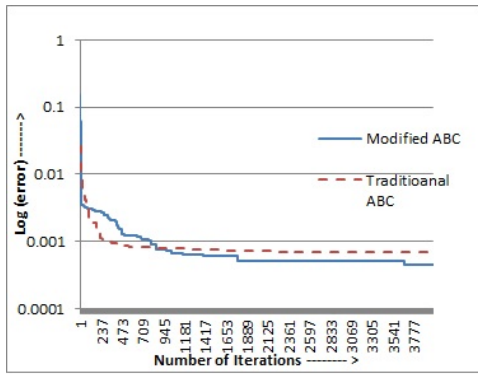


Fig. 2. Convergence graph of ABC variants for synthetic data

Table 3. Results of parameter extraction for synthetic data

Data	Default values	Traditional ABC	Modified ABC
I_{ph}	0.9072	0.9075	0.9072
I_{s1}	2.831E-05	1.000E-05	2.406E-05
I_{s2}	2.466E-09	4.429E-06	3.188E-09
n_1	2.0000	1.9997	1.8849
n_2	1.0000	1.5636	1.0000
R_s	0.03117	0.018402	0.03018
R_{sh}	19.92	15.78	18.38
RMSE	—	8.108E-04	4.2735E-04

of all extracted parameters are much more than those obtained by modified ABC.

The results clearly show that with the use of modified ABC, the errors in extracted values of solar cell parameters decrease significantly. This improves accuracy of determined parameters. This is by virtue of the memory function (ability to remember best food source so far) incorporated in modified ABC due to which the solution moves towards near optimal solution.

5.2 Parameter extraction with experimental I-V data

To further validate the efficiency of modified ABC, it is now applied to experimental I-V data of PV module containing 36 solar cells in series. The extracted parameters which show least RMSE between I_{exp} and I_{calc} are chosen as final values of extracted parameters. Figures(3) and (4) shows comparison between simple and modified ABC for values of I_{exp} and I_{calc} .

From the figures 3 and 4, we can say that, in both the methods, calculated current is closely following the experimental value of current. This might suggests that, both the methods are converging to accurate optimal values. However, since we do not have a comparison point between two methods, as this is an experimental data, we deduce whether one set of parameters is better than the other. The real comparison point here can only be the RMSE values between I_{exp} and I_{calc} plotted over number of cycles. Figure(5) shows RMSE curve over number of cycles. It is evident from the above plot that when run over enough number of cycles, modified ABC produces less RMSE than traditional ABC. This also indicates that, the parameters extracted by modified ABC might be closer to real world values. Traditional ABC fails to achieve similar performance as shown by modified ABC. The table(4) summarizes the final values extracted by both the methods for experimental data. The last row of the table shows least RMSE obtained by both the methods.

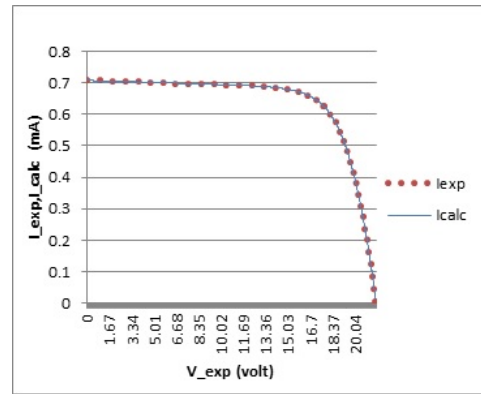


Fig. 3. Comparison on the I-V characteristics between the experimental data and simulated data obtained by Traditional ABC algorithm

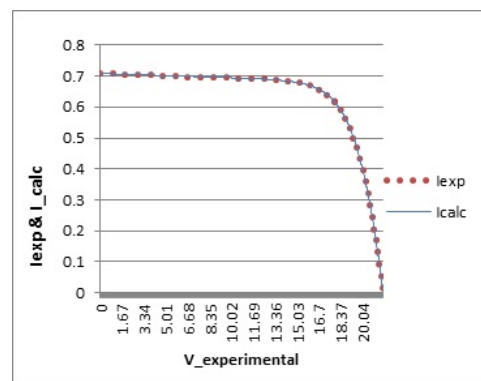


Fig. 4. Comparison on the I-V characteristics between the experimental data and simulated data obtained by Modified ABC algorithm

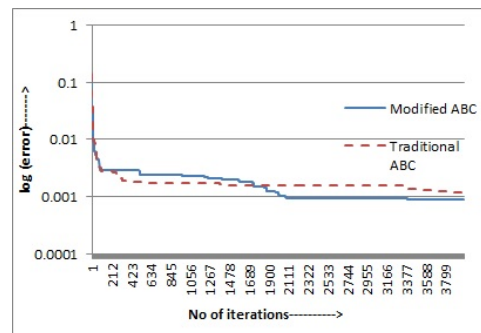


Fig. 5. Convergence graph of ABC variants for experimental data

The closer analysis of the table proves that results obtained by modified ABC outperform those obtained by traditional ABC, when a comparison based on minimum value of RMSE is taken into consideration. Looking at ranges of parameters, we can say that modified algorithm does not necessarily needs a search space within proximity of optimal solutions. This also makes algorithm robust while making initial guesses.

Table 4. Results of parameter extraction for experimental data

Data	Ranges	Traditional ABC	Modified ABC
I_{ph}	[0.2,1]	0.7081	0.7086
I_{s1}	[1E-05,1E-09]	2.156E-07	4.272E-09
I_{s2}	[1E-05,1E-09]	1.461E-08	1.221E-07
n_1	[1,2]	1.5710	1.9235
n_2	[1,2]	1.2466	1.3886
R_s	[0.0001,1]	0.0290	0.0237
R_{sh}	[1,1000]	22.1253	20.1980
RMSE	—	1.427E-03	8.819E-04

6. CONCLUSION

The problem of solar cell parameters identification using modified ABC algorithm is presented in this study. The double diode model of solar cell is used to validate the performance of the proposed approach. The validity of the proposed method has been confirmed by applying it to both experimental and synthetic I-V data. It is evident from the results that modifications implemented in traditional ABC algorithm certainly brings about more accuracy and convergence rate to parameter extraction. This can be measured in terms of root mean square value(RMSE), precision with which parameters are extracted and convergence graphs provided in this paper. The modified ABC algorithm is also able to converge on global optimum values despite the search space being very wide as seen from upper and lower bound of parameters. This makes algorithm more suitable for cases where even the anticipated parameter ranges are unknown. Hence, the proposed technique is accurate, fast, and easily applicable for the parameter extraction of solar cells from I-V characteristics.

7. REFERENCES

- [1] Daniel S. H. Chan and Jacob C H. Phang, "Analytical methods for the extraction of solar cell single and double diode model parameters from I-V characteristics," *IEEE transaction of electronics*,1987
- [2] M. R. AlRashidi, K. M. El-Naggar, and M. F. AlHajri, "Parameters Estimation of Double Diode Solar Cell Model," *World Academy of Science, Engineering and Technology* 74, 2013
- [3] Meiyong Ye, Xiaodong Wang, and Yousheng Xu, "Parameter extraction of solar cells using particle swarm optimization," *Journal of Applied Physics*,pp. 105, 2009
- [4] Wenyin Gong, Zhihua Caia, "Parameter extraction of solar cell models using repaired adaptive differential evolution,"*China University of Geosciences*,pp.1-16, 2013
- [5] Tomas Zahradnick, "MOSFET Parameter Extraction," *Dissertation Thesis Proposal Czech Technical University in Prague,2005-04*,pp.1-138
- [6] D. Karaboga , "An idea based on honey bee swarm for numerical optimization," *Erciyes University Press*,Turkey,2005
- [7] D. Karaboga,B Akay "A modified artificial bee colony(ABC) algorithm for real parameter optimization," *Information science* 192,pp.120-142,2012
- [8] A. L. Bolaji, A. T. Khader, M. A. Al-Betar and M. A. Awadallah, "Artificial Bee Colony Algorithm, its variants and applications: A survey," *Journal of Theoretical and Applied Information Technology* Vol.47 No.2,pp 434-459,20th January 2013
- [9] Efren Mezura-Montes and Ramiro Ernesto Velez-Koepfel, "Elitist Artificial Bee Colony for Constrained Real-Parameter Optimization," *IEEE Conference*,pp.1-7,2010
- [10] Yiming Yan, Ye Zhang and Fengjiao Gao, "Dynamic artificial bee colony algorithm for multi-parameters optimization of support vector machine-based soft margin classifier," *EURASIP Journal on Advances in Signal Processing*,2012,pp.146
- [11] Defeng Wu,Wanneng Yu and Zibin Yin , "Parameter Estimation of Rational Models Based on Artificial Bee Colony Algorithm," *Proceedings of 2011 International Conference on Modelling, Identification and control*, Shanghai,China,pp.219-224,June 26-29,2011
- [12] Adil Baykasoglu,Lale zbakr and Pnar Tapkan, "Artificial Bee Colony Algorithm and Its Application to Generalized Assignment Problem," *Swarm Intellenge:Focus on ant and PSO*,pp.532-560,December 2007
- [13] A. Aderhold, K. Diwold, A. Scheidler, and M. Middendorf , "Artificial bee colony optimization: A new selection scheme and its performance," *Nature Inspired Cooperative Strategies for Optimization*,pp.283-294, 2010
- [14] M. S. Kiram, Mesut Gunduz, "A novel artificial Bee Colony Algorithm for solving the numerical optimization problems," *International journal of Innovative Computing Information and control*,Volume 8,pp.8107-8121, sep 2012