Hindawi
International Journal of Photoenergy
Volume 2019, Article ID 4692108, 16 pages
https://doi.org/10.1155/2019/4692108

Research Article

Adaptive Electromagnetic Field Optimization Algorithm for the Solar Cell Parameter Identification Problem

Ilker Kucukoglu

Department of Industrial Engineering, Bursa Uludag University, Bursa, Turkey

Correspondence should be addressed to Ilker Kucukoglu; ikucukoglu@uludag.edu.tr

Received 21 January 2019; Revised 18 April 2019; Accepted 28 April 2019; Published 3 June 2019

Academic Editor: Huiqing Wen

Solar cell parameter identification problem (SCPIP) is one of the most studied optimization problems in the field of renewable energy since accurate estimation of model parameters plays an important role to increase their efficiency. The SCPIP is aimed at optimizing the performance of solar cells by estimating the best parameter values of the solar cells that produce an accurate approximation between the current vs. voltage ($I - V$) measurements. To solve the SCPIP efficiently, this paper introduces an adaptive variant of the electromagnetic field optimization (EFO) algorithm, named adaptive EFO (AEFO). The EFO simulates the attraction-repulsion mechanism between particles of electromagnets having different polarities. The main idea behind the EFO is to guide electromagnetic particles towards global optimum by the attraction-repulsion forces and the golden ratio. Distinct from the EFO, the AEFO searches the solution space with an adaptive search procedure. In the adaptive search strategy, the selection probability of a better solution is increased adaptively whereas the selection probability of worse solutions is reduced throughout the search progress. By employing the adaptive strategy, the AEFO is able to maintain the balance between exploration and exploitation more efficiently. Further, new boundary control and randomization procedures for the candidate electromagnets are presented. To identify the performance of the proposed algorithm, two different benchmark problems are taken into account in the computational studies. First, the AEFO is performed on global optimization benchmark functions and compared to the EFO. The efficiency of the AEFO is identified by statistical significance tests. Then, the AEFO is implemented on a well-known SCPIP benchmark problem set formed as a result of real-life physical experiments based on single- and double-diode models. To validate the performance of the AEFO on the SCPIP, extensive experiments are carried out, where the AEFO is tested against the original EFO, AEFO variants, and novel metaheuristic algorithms. Results of the computational studies reveal that the AEFO exhibits superior performance and outperforms other competitor algorithms.

1. Introduction

Renewable energy has experienced a tremendous increase in recent decades because of the depletion of conventional sources oil, coal, or natural gas. Among various kinds of renewable energy sources such as wind, wave, nuclear, and biomass, solar or photovoltaic (PV) energy is the most important source due to its properties such as effectiveness, wide-scale availability, unlimited capacity, and safe-use [1]. Furthermore, PV, which is able to provide power for specific purposes, is an emission-free system with direct conversion from solar energy to electricity [2]. Since solar cell installation has received great attention, numerous researchers have focused to maximize the efficiency of PV systems. In order to control and optimize PV systems, it is required to accurately simulate the characteristics of the PV system before installation. The accuracy of the PV systems mainly depends on the parameters of solar cells, which are generally not provided by the cell manufacturers [3]. Therefore, it is vital to identify the parameters of solar cells or modules based on nonlinear mathematical models. Among a variety of existing models in the literature, the main ones are the single-diode model (SD), the double-diode model (DD), and the PV module model [4–6]. The problem of extracting the parameters of solar cells from the experimental data is called the solar cell parameter identification problem (SCPIP) in literature.

To solve the SCPIP, there exist several solution approaches in the literature, which are mainly divided into

two groups: deterministic and heuristic solution approaches. Regarding the deterministic approaches, a number of methods are employed by the researchers, such as nonlinear least squares based on the Newton model [7], iterative curve fitting [8], Lambert W-function [9], and J-V model [10]. However, these deterministic solution approaches are not efficient to solve the SCPIP since they need continuity, convexity, and differentiability conditions for being applicable and involve heavy computations [4, 11]. To cope with the complexity of the SCPIP, heuristic methods are used as an alternative to deterministic solution approaches.

Regarding the popular metaheuristic algorithms, simulated annealing algorithm [12], genetic algorithm [13, 14], particle swarm optimization algorithm [15, 16], differential evolution algorithm [17–20], pattern search [21], artificial bee colony algorithm [22] are widely used for the SCPIP. In addition to these well-known heuristic algorithms, there exist several papers in the literature which consider more recent approaches, such as bacterial foraging algorithm [23, 24], teaching-learning-based optimization algorithm [25–27], biogeography-based optimization algorithm [28], chaos optimization algorithm [29], artificial fish swarm algorithm [30], bird mating optimizer approach [31], artificial immune system [32], evolutionary algorithm [1], cat swarm optimization algorithm [33], moth-flame optimization algorithm [5], JAYA optimization algorithm [34, 35], chaotic whale optimization algorithm [36], imperialist competitive algorithm [37], bee pollinator flower pollination algorithm [38], shuffled complex evolution algorithm [39], memetic algorithm [40], interior search algorithm [41], collaborative swarm intelligence approach [42], and cuckoo search algorithm [43]. On the other hand, it has been proven by No-Free-Lunch theorem [44] that none of these algorithms is able to solve all type of optimization problems. As a result of No-Free-Lunch theorem, it should be denoted that a new algorithm is always likely to exhibit better performance on the SCPIP compared to the existing solution methodologies.

Based on the aforementioned motivation, this study considers electromagnetic field optimization (EFO) algorithm to solve the SCPIP. The EFO is a relatively new and effective algorithm on global optimization problems, and it has been shown that the EFO outperforms other optimization algorithms and effectively balance the exploration and exploitation performance [45]. Conversely, it is also reported that the traditional EFO tends to suffer poor exploitation performance on specific optimization problems [46]. Therefore, this study introduces an adaptive version of electromagnetic field optimization to solve the SCPIP efficiently, which is called the adaptive EFO (AEFO). The proposed AEFO adaptively controls the algorithm parameters and explores the search space effectively, especially in the early stages of the search process, whereas exploitation is emphasized in the latter phases. In addition to the adaptive control of parameters, boundary control and randomization procedures are modified in the algorithm. In computational studies, performance of the proposed algorithm is tested into two parts. First, the AEFO is performed on a recently introduced global optimization benchmark problem set and compared to the EFO solutions to identify the efficiency of the adaptive control mechanism of the proposed algorithm. In the second part, the AEFO is tested on the well-known PV models and compared to the original version of the EFO, artificial bee colony algorithm (ABC), particle swarm optimization (PSO), and differential evolution algorithm (DE) in identical test conditions. The AEFO is further tested against recent metaheuristic algorithms, which are presented to solve the SCPIP. Computational results and statistical tests show that the AEFO significantly achieves superior performance to competitor algorithms.

The main contributions of the proposed study are as follows:

(i) To the best of the author's knowledge, the EFO has not been considered in the literature to solve the SCPIP until now

(ii) An adaptive version of the EFO is introduced by enriching the algorithm employing an adaptive search strategy. Additionally, modified boundary check and randomization procedures are used for the candidate solution generation. By these novel modifications, the performance of the traditional EFO is improved

(iii) Detailed comparisons between the EFO and the AEFO variants and also between the AEFO and the other recent algorithms are presented. The outperforming performance of the AEFO is proved by statistical significance tests

The remainder of the paper is organized as follows. In Section 2, the SCPIP is described and the mathematical formulation of the problem is given. Section 3 presents the details of the EFO. Section 4 introduces the proposed AEFO for the SCPIP. Computational results are given in Section 5. Finally, a conclusion part with future research perspectives is provided in Section 6.

2. Problem Definition

To describe the $I - V$ characteristics of the solar cells, there exist several models in the literature. In this study, the SD and DD models, which are the most commonly used models since their practical usage for the solar cells, are taken into account [4, 11]. In this section, the SD and DD models are introduced, and the SCPIP based on these models is defined.

2.1. Single-Diode Model. The SD model consists of a current source in parallel with a diode, a shunt resistor to represent the leakage current, and a series resistor to denote the losses of load current. This model has commonly used to describe the static characteristics of solar cells because of its simplicity and accuracy [34]. Figure 1 represents the equivalent circuit for the SD model, where V_t is the terminal voltage, R_s is the series resistance, R_{sh} is the shunt resistance, I_t is the terminal current, I_{ph} is the photo-generated current, I_d is the diode current, and I_{sh} is the shunt resistor current.

By using the Shockley equation for the diode currents, the single-diode model can be formulated as shown in equation

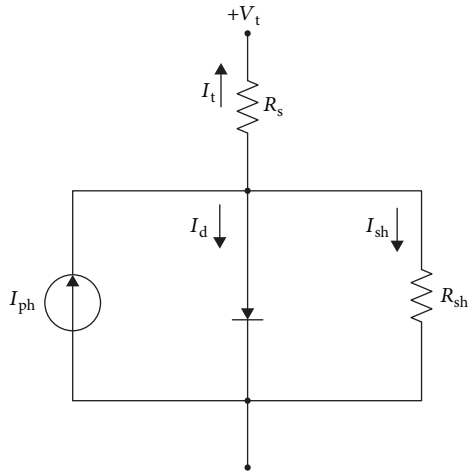

Figure 1: The equivalent circuit of the single-diode model.

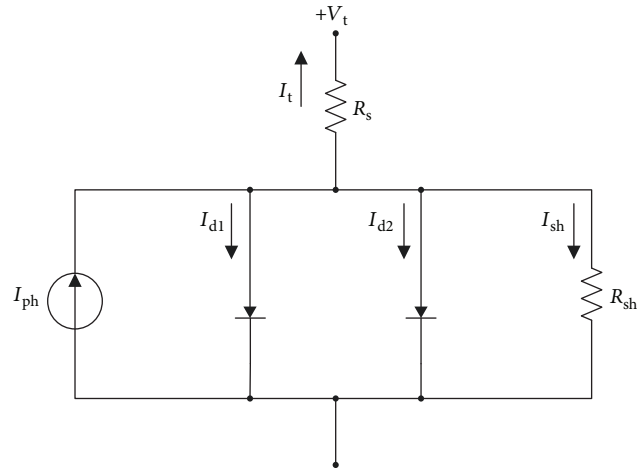

Figure 2: The equivalent circuit of the double-diode model.

(1), where n is the ideality factor of the diode and I_{sd} is the reverse saturation current of the diode [22, 27, 36]. According to the Shockley equation, q is the magnitude of charge on an electron ($1.60217646 \times 10^{-19}$ coulombs), k is the Boltzmann constant ($1.3806503 \times 10^{-23}\ J/K$), and T is the cell temperature in Kelvin [22].

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

2.2. Double-Diode Model. The DD model consists of two diodes in parallel with the current source and a shunt resistance to consider the effect of recombination current loss in the depletion region [34]. The DD model provides more precise solution with regard to the consideration of this loss, especially at low voltage [27]. In Figure 2, the equivalent circuit of the DD model is represented. Similar to the SD model, the output current is described as follows:

$$\begin{aligned} I_t = {} & I_{ph} - I_{sd1}\left[\exp\left(\frac{q(V_t + R_s.I_t)}{n_1.k.T}\right) - 1\right] \\ & - I_{sd2}\left[\exp\left(\frac{q(V_t + R_s.I_t)}{n_2.k.T}\right) - 1\right] - \left(\frac{V_t + R_s.I_t}{R_{sh}}\right) \end{aligned} \quad (2)$$

where I_{sd1} and I_{sd2} are, respectively, the diffusion and saturation currents, and n_1 and n_1 are, respectively, the diffusion and recombination diode ideal factors. The other parameters are the same as defined for the SD model.

2.3. Parameter Estimation Problem. Regarding the SD and DD models described above, the SCPIP can be defined as identifying the parameters of equation (1) for the SD model and equation (2) for the DD model within their lower and upper bounds. The aim of the problem is to estimate the best parameter values for the solar cell models that produce an accurate approximation between the $I - V$ measurements from the physical experiments and the values from the mathematical model. Hence, equations (1) and (2) can be rewritten as an error function as shown in equations (3) and (4) for SD and DD models, respectively.

$$\begin{aligned} & f\left(V_t, I_t, R_s, R_{sh}, I_{ph}, I_{sd}, n\right) \\ & \quad = I_t - I_{ph} + I_{sd}\left[\exp\left(\frac{q(V_t + R_s.I_t)}{n.k.T}\right) - 1\right] \\ & \quad + \left(\frac{V_t + R_s.I_t}{R_{sh}}\right) \end{aligned} \quad (3)$$

$$\begin{aligned} & f\left(V_t, I_t, R_s, R_{sh}, I_{ph}, I_{sd1}, I_{sd1}, n_1, n_2\right) \\ & \quad = I_t - I_{ph} + I_{sd1}\left[\exp\left(\frac{q(V_t + R_s.I_t)}{n_1.k.T}\right) - 1\right] \\ & \quad + I_{sd2}\left[\exp\left(\frac{q(V_t + R_s.I_t)}{n_2.k.T}\right) - 1\right] + \left(\frac{V_t + R_s.I_t}{R_{sh}}\right) \end{aligned} \quad (4)$$

In the sense of optimization problem for accurate estimation, the error function for the SCPIP can be transformed into equations (5) and (6) for the SD and DD, respectively. The $\mathbf{X}$ in these equations represents the decision variables to be optimized for the SCPIP, where Table 1 shows the descriptions of the decision variables and their lower and upper bounds for the SD and DD models [22, 27, 36].

$$\begin{aligned} f(V_t, I_t, \mathbf{X}) = {} & I_t - x_3 + x_4\left[\exp\left(\frac{q(V_t + x_1.I_t)}{x_5.k.T}\right) - 1\right] \\ & + \left(\frac{V_t + x_1.I_t}{x_2}\right) \end{aligned} \quad (5)$$

$$\begin{aligned} f(V_t, I_t, \mathbf{X}) = {} & I_t - x_3 + x_4\left[\exp\left(\frac{q(V_t + x_1.I_t)}{x_6.k.T}\right) - 1\right] \\ & + x_5\left[\exp\left(\frac{q(V_t + x_1.I_t)}{x_7.k.T}\right) - 1\right] + \left(\frac{V_t + x_1.I_t}{x_2}\right) \end{aligned} \quad (6)$$

With regard to the error functions described for the SD and DD above, the mathematical formulation of the SCPIP can be defined as follows:

Table 1: Lower and upper bounds of the solar cell parameters.

SD model				DD model			
Solar cell parameters	Decision variable vector $\mathbf{X}$	Lover bound	Upper bound	Solar cell parameters	Decision variable vector $\mathbf{X}$	Lover bound	Upper bound
$R_s(\Omega)$	x_1	0	0.5	$R_s(\Omega)$	x_1	0	0.5
$R_{sh}(\Omega)$	x_2	0	100	$R_{sh}(\Omega)$	x_2	0	100
$I_{ph}(\mathrm{A})$	x_3	0	1	$I_{ph}(\mathrm{A})$	x_3	0	1
$I_{sd}(\mu\mathrm{A})$	x_4	0	1	$I_{sd1}(\mu\mathrm{A})$	x_4	0	1
n	x_5	1	2	$I_{sd2}(\mu\mathrm{A})$	x_5	0	1
				n_1	x_6	1	2
				n_2	x_7	1	2

(1) Parameters

M:Number of experiments considered for the SCPIP

N:Number of solar cell parameters (5 for SD and 7 for DD models)

LB_j:Lower bound value of solar cell parameter j; $j = 1, \cdots, N$

UB_j:Upper bound value of solar cell parameter j; $j = 1, \cdots, N$

(2) Decision variables

x_j:Positive continuous variable and identifies the value of solar cell parameter j; $j = 1, \cdots, N$

(3) Model

$$\mathrm{Min}\, Z = \mathrm{RMSE}(\mathbf{X}) \tag{7}$$

Subject to

$$\mathrm{LB}_j \le x_j \le \mathrm{UB}_j, \quad j = 1, \cdots, N \tag{8}$$

$$x_j \ge 0, \quad j = 1, \cdots, N. \tag{9}$$

The objective function (7) aims to minimize RMSE (root mean square error) of the experiments, which is determined by using equation (10). The constraints (8) describe the boundaries of the decision variables. Finally, the decision variables are described in constraint (9).

$$\mathrm{RMSE}(\mathbf{X}) = \sqrt{\frac{1}{M}\sum_{i=1}^{M}\left(f_i(V_t, I_t, \mathbf{X})\right)^2} \tag{10}$$

3. Electromagnetic Field Optimization Algorithm

EFO is proposed by Abedinpourshotorban et al. [45], which is a relatively new population-based physics-inspired metaheuristic algorithm. EFO simulates the attraction-repulsion mechanisms between electromagnets having different polarities, where each candidate solution is associated with an electromagnetic particle (EMP) made of electromagnets. EMP is represented by a real-coded vector with a dimension of D, where D denotes the problem size, i.e., number of electromagnets. The quality of a solution determines the polarity of the corresponding EMP.

EFO classifies the EMP population into three groups as positive, neutral, and negative polarities according to the predetermined ratios, and the attraction-repulsion mechanisms among those EMPs will guide the population to the global optimum. The main idea behind the search mechanism of the EFO is that the negative EMPs will repel, whereas EMPs having positive polarities will attract the neutral EMPs. Furthermore, EFO employs the golden ratio to balance the attraction and repulsion forces to favor the exploitation behavior of the population [45].

In the initialization step, the EFO generates a randomly distributed population of N, which is made of electromagnets within the search space, where N represents the number of the EMPs, i.e., population size. Then, the fitness of the initial population is evaluated, and the population is sorted regarding their fitness in descending order. Then, the population is classified into three groups: (i) fittest EMPs as positive part, (ii) EMPs having the lowest fitness as negative part, and (iii) the remaining EMPs as neutral part. Afterwards, one EMP is selected from each group, and a new candidate EMP is formed using the search mechanism of the EFO. In order to keep the population diversity and to improve the exploration ability of the algorithm, one electromagnet of the candidate solution has a chance to be updated in the random search phase. Last, the fitness of the candidate EMP is evaluated, and if it is better than the worst particle in the current population, then the candidate will be inserted into the population, whereas the worst EMP will be eliminated.

The main steps of the EFO are as follows:

In the initialization step, an initial population of N EMPs is randomly generated within the search space. Let $x_i = \{x_{i,1}, x_{i,2}, \cdots, x_{i,D}\}$ represent the ith EMP, where D is the problem size. Each EMP is generated by

$$x_{i,j} = x_j^{\min} + \mathrm{rand} * \left(x_j^{\max} - x_j^{\min}\right) \tag{11}$$

where $i = 1, 2, \cdots, N$, $j = 1, 2, \cdots, D$. $x_j^{\min}$ and $x_j^{\max}$ are the

(1): Initialization
(2): Fitness evaluation and sorting
(3): **Repeat**
(4): Classification
(5): **Repeat**
(6): Candidate EMP generation
(7): **Until** all electromagnets are generated
(8): Random search/Mutation
(9): Selection and re-sorting
(10): **Until** termination criterion is satisfied

Algorithm 1:

lower and upper bounds for the index j, respectively. Last, rand is a random real number within the range $[0, 1]$. Then, the fitness of the EMPs in the initial population is determined, and the population is sorted based on the fitness values in nonincreasing order.

In the classification phase of the EFO, the population is divided into three groups with different polarities. In this point, two different control parameters as P_field and N_field are employed in order to determine the EMPs for each group. The P_field and N_field represent the percentage of the allocated solution for positive and negative part, respectively. The remaining solutions form the neutral part.

The candidate solution generation step of the EFO is the most important part of the algorithm [47]. In this step, one EMP from each group is selected randomly. Then, a random number between 0 and 1 is generated. If this number is lower than the predetermined control parameter, *Ps_rate*, then the corresponding electromagnet of the candidate EMP is set as the electromagnet of the EMP from the positive field. Otherwise, the electromagnet of the candidate solution is determined as follows:

$$v_j = x_{k,j} + (\varphi * \text{rand})\left(x_{p,j} - x_{k,j}\right) - \text{rand} * \left(x_{n,j} - x_{k,j}\right) \quad (12)$$

where v is the candidate solution and j is the corresponding electromagnet (index). p, n, and k are the indexes of the selected EMPs from positive, negative, and neutral parts, respectively. φ is the golden ratio constant, which is used to guide the candidate solutions towards the positive part as can be seen from equation (12); the candidate EMP is generated based on the randomly selected neutral particle, where the positive EMP will attract and the negative EMP will repel the candidate solution. The algorithm of the candidate solution generation step of the EFO is given in Algorithm 2.

In the random search step of the EFO, the random real number is generated between 0 and 1. If the generated random number is lower than the control parameter, i.e., R_rate, then one selected electromagnet of the candidate EMP (v) is reinitialized within the search space randomly. The electromagnet selection in the random search follows the order of the electromagnets, which is controlled by a counter in the algorithm, and the subsequent electromagnet is updated in each random search step. If the end of the EMP is reached, then the random search starts from the beginning in the next time.

As the last step, the fitness of the v is evaluated, and if it is better than the worst solution of the current population, it is inserted to the population, and the worst solution is eliminated. The processes of classification, candidate EMP generation, random search, and selection are repeated until some termination criterion is satisfied.

From the aforementioned descriptions, the following characteristics of the EFO are observed:

(i) In each electromagnet generation, new EMPs are selected from each group. Therefore, EFO utilizes various information sources during the candidate solution generation

(ii) The search mechanism in equation (12) shows strong exploitation characteristics. First, better solutions attract and the worse solution will repel the selected neutral particle. Second, the golden ratio is employed, where more weight is given to the attraction force. Moreover, by utilizing the Ps_rate parameter, candidate electromagnet has a chance to be copied directly from the positive EMP

(iii) EFO generates one candidate EMP in each cycle, and the fitness of the candidate is only compared with the worst solution in the current population. If the candidate solution is not better than the worst, there will be no chance to be accepted to the population

(iv) The random search part is the only process in the EFO, which is responsible for exploration

It is well known that the compromise between exploration and exploitation throughout a run is critical to the success of a metaheuristic algorithm. Exploration means the ability of an algorithm to search for unvisited points in the search region, whereas exploitation is the process of refining those points within the neighborhood of previously visited locations to improve the solution quality. In a word, it can be concluded from the above observations that the EFO algorithm is good at exploitation but may have poor exploration behavior. Therefore, an adaptive version of the EFO is presented in this study.

4. Proposed Algorithm

In this section, an adaptive version of traditional EFO, which is named AEFO, is presented. In the AEFO, two main algorithmic parameters, i.e., Ps_rate and R_rate, are updated adaptively at run time to improve the balance of exploration and exploitation. As mentioned in the previous section, R_rate favors exploration whereas Ps_rate is responsible for the exploitation. In the EFO, Ps_rate and R_rate are determined at time = 0 and not updated throughout the search process. However, in the AEFO, Ps_rate has increased adaptively from Ps_rate_min to Ps_rate_max during the search, where Ps_rate_min and Ps_rate_max represent the minimum and maximum allowable values for Ps_rate, respectively.

```
(1):  set r = random real number within the range of (0,1).
(2):  for j = 1 to D do //for each electromagnet of the candidate
(3):     set p = randomly selected index of the EMP from the positive field.
(4):     set n = randomly selected index of the EMP from the negative field.
(5):     set k = randomly selected index of the EMP from the neutral field.
(6):     if rand < Ps_rate then
(7):        set v(j) as the electromagnet of the positive EMP (x^p).
(8):     else
(9):        set v(j) using the Eqn. (12) //Use r as rand in Eqn. (12).
(10):    end if
(11):    Randomly Update v(j) if it is outside the boundary.
(12): end for
```

Algorithm 2 : Candidate solution generation

```
(1):  set Popsize = Number of EMPs
(2):  set N_var = Number of variables in the problem //number of electromagnets
(3):  set Termination condition
(4):  set R_rate_max = Maximum probability of random search
(5):  set R_rate_min = Minimum probability of random search
(6):  set Ps_rate_max = Maximum probability of selecting the electromagnet from positive field
(7):  set Ps_rate_min = Minimum probability of selecting the electromagnet from positive field
(8):  set R_rate = R_rate_max //R_rate will decrease from R_rate_max to R_rate_min
(9):  set Ps_rate = Ps_rate_min //Ps_rate will increase from Ps_rate_min to Ps_rate_max
(10): set UB and LB = Upper and lower bounds of electromagnets
(11): set phi = Golden ratio
(12): set P_field = ratio of the positive field
(13): set N_field = ratio of the negative field
(14): Generate initial population as pop. //use Eqn (11)
(15): Evaluate initial population //calculate fitness of the initial population
(16): Sort population according to the fitness values
(17): do while Termination condition is not satisfied //Main loop
      //******* Candidate EMP generation ******** //
      //v will be the candidate EMP, the size of v is N_var
(18):    for i = 1 to N_var do //for each electromagnets of the generated particle
(19):       Determine positive_index, negative_index and neutral_index randomly
(20):       if rand(0,1) ≤ Ps_rate then
(21):          v(i) = pop(positive_index,i) //generated EMPs electromagnet will be equal to positive field EMP
(22):       else
(23):          Determine v(i) using Eqn (12)
(24):       end if
(25):    end for
(26):    Check the boundary limits. //if v is outside of the boundary values, set the corresponding electromagnets to the boundary values
      //******* Random search ********//
(27):    if rand(0,1) ≤ R_rate then
(28):       Update randomly selected electromagnet of v within limits randomly
(29):    end if
(30):    set fitv = fitness value of the generated EMP
(31):    if fitv is better than the worst EMP in pop then
(32):       Delete the worst EMP and insert v to the pop
(33):       Re-sort the pop according to the fitness values
(34):    end if
(35):    Update Ps_rate and R_rate using the Eqn (13) and Eqn (14)
(36):    Save necessary information // Memorize global best
(37): end while
```

Algorithm 3 : AEFO algorithm

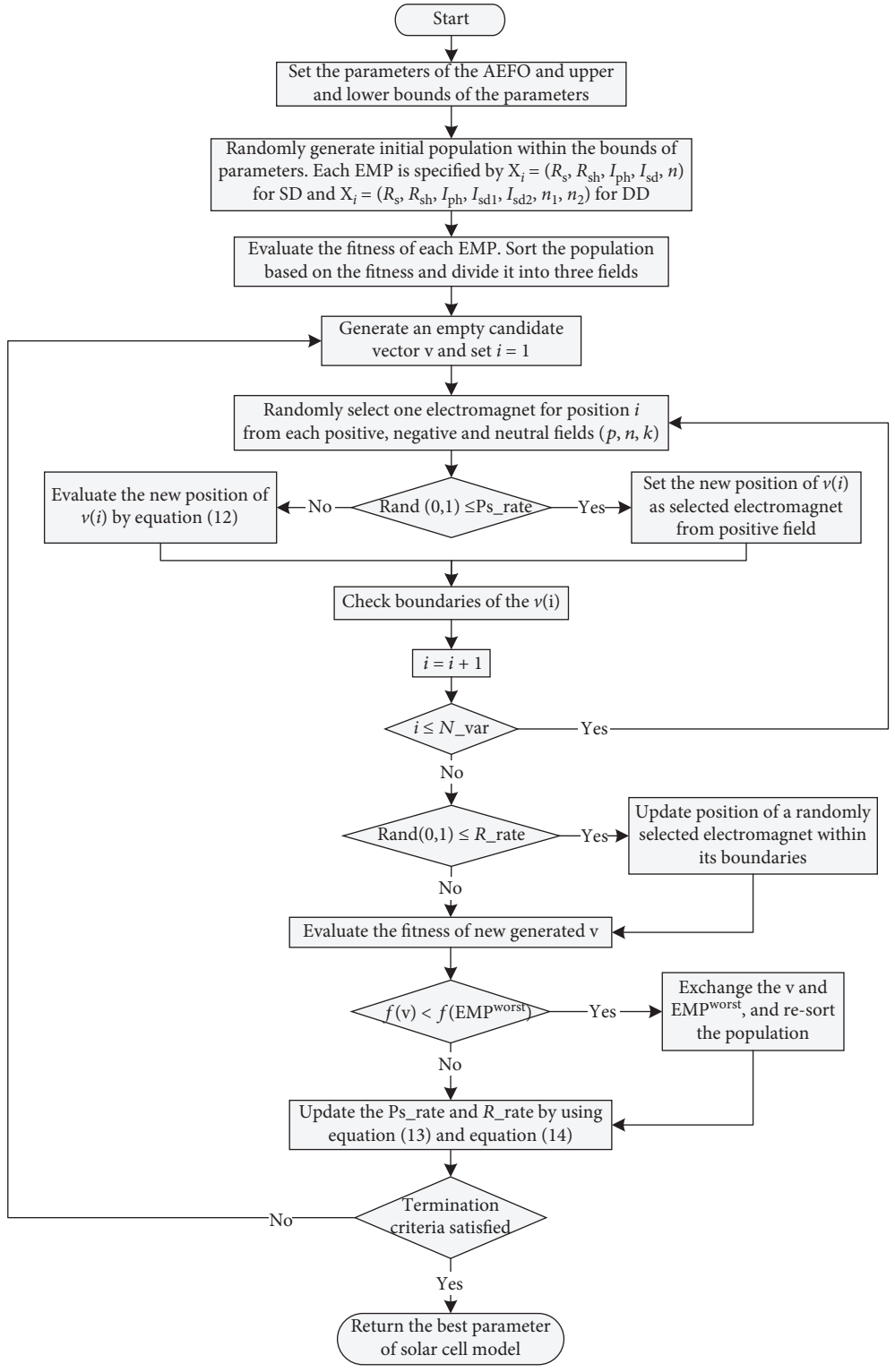

Figure 3: Flow chart of the proposed AEFO.

Similarly, *R*_rate is decreased from *R*_rate_max to *R*_rate_min as the algorithm proceeds. In other words, AEFO achieves better exploration performance, specifically in the early phases of the search, whereas favors exploitation in the latter phase. As Ps_rate increases from Ps_rate_min to Ps_rate_max, the candidate EMPs will more likely to move towards better solutions, i.e., positive field in the latter phases of the search process. At the end of each iteration, Ps_rate and *R*_rate are updated according to equations (13) and (14), respectively. In equations (13)

Table 2: Comparisons of the AEFO with the EFO on CEC 2017 benchmark functions ($D = 30$).

Func.	EFO				AEFO				t-test
	Best	Worst	Mean	StdDev	Best	Worst	Mean	StdDev	
$F1$	$5.71E-01$	$1.92E+04$	$3.69E+03$	$4.72E+03$	$5.15E-01$	$1.64E+04$	$2.74E+03$	$3.88E+03$	≈
$F3$	$2.33E+04$	$6.94E+04$	$4.30E+04$	$1.15E+04$	$1.83E+03$	$2.48E+04$	$7.18E+03$	$5.54E+03$	+
$F4$	$3.87E-01$	$1.15E+02$	$7.14E+01$	$2.82E+01$	$2.52E-01$	$1.13E+02$	$7.22E+01$	$2.45E+01$	≈
$F5$	$1.79E+01$	$2.02E+02$	$1.30E+02$	$6.73E+01$	$1.19E+01$	$1.70E+02$	$4.10E+01$	$4.01E+01$	+
$F6$	$1.65E-03$	$4.88E-02$	$1.55E-02$	$1.35E-02$	$2.05E-04$	$2.59E-02$	$3.56E-03$	$4.91E-03$	+
$F7$	$7.32E+01$	$2.49E+02$	$2.05E+02$	$3.95E+01$	$4.82E+01$	$2.31E+02$	$8.21E+01$	$4.79E+01$	+
$F8$	$2.09E+01$	$1.98E+02$	$1.25E+02$	$6.91E+01$	$1.40E+01$	$4.41E+01$	$2.71E+01$	$7.98E+00$	+
$F9$	$1.36E+00$	$5.63E+01$	$1.26E+01$	$1.17E+01$	$8.95E-02$	$2.20E+01$	$4.96E+00$	$4.97E+00$	+
$F10$	$6.69E+03$	$7.89E+03$	$7.41E+03$	$3.38E+02$	$1.35E+03$	$7.72E+03$	$6.00E+03$	$1.93E+03$	+
$F11$	$4.52E+00$	$4.28E+02$	$1.17E+02$	$9.65E+01$	$6.60E+00$	$1.22E+02$	$3.66E+01$	$3.94E+01$	+
$F12$	$1.04E+04$	$2.30E+05$	$1.13E+05$	$5.48E+04$	$2.45E+04$	$3.39E+05$	$1.04E+05$	$7.61E+04$	≈
$F13$	$1.11E+02$	$3.12E+04$	$9.79E+03$	$9.09E+03$	$2.53E+02$	$3.24E+04$	$8.88E+03$	$9.16E+03$	≈
$F14$	$1.03E+03$	$4.80E+04$	$1.64E+04$	$1.27E+04$	$1.12E+03$	$5.77E+04$	$1.79E+04$	$1.75E+04$	≈
$F15$	$2.50E+01$	$2.43E+04$	$5.29E+03$	$6.31E+03$	$1.93E+02$	$3.31E+04$	$6.98E+03$	$8.02E+03$	≈
$F16$	$1.38E+01$	$1.99E+03$	$1.38E+03$	$4.13E+02$	$9.26E+00$	$1.60E+03$	$6.05E+02$	$5.33E+02$	+
$F17$	$3.86E+01$	$6.44E+02$	$1.90E+02$	$1.60E+02$	$4.62E+00$	$3.87E+02$	$7.34E+01$	$9.68E+01$	+
$F18$	$1.10E+05$	$3.02E+06$	$8.89E+05$	$8.46E+05$	$6.42E+04$	$2.08E+06$	$4.65E+05$	$4.46E+05$	+
$F19$	$2.38E+01$	$4.21E+04$	$9.55E+03$	$1.27E+04$	$1.51E+01$	$4.41E+04$	$8.01E+03$	$9.92E+03$	≈
$F20$	$4.53E+00$	$7.87E+02$	$1.97E+02$	$1.84E+02$	$6.62E+00$	$2.89E+02$	$8.55E+01$	$7.50E+01$	+
$F21$	$2.16E+02$	$4.04E+02$	$3.63E+02$	$5.04E+01$	$2.16E+02$	$3.90E+02$	$2.49E+02$	$4.83E+01$	+
$F22$	$1.00E+02$	$7.37E+03$	$1.06E+03$	$2.49E+03$	$1.00E+02$	$7.21E+03$	$3.37E+02$	$1.30E+03$	≈
$F23$	$3.60E+02$	$5.42E+02$	$3.95E+02$	$4.74E+01$	$3.54E+02$	$3.92E+02$	$3.75E+02$	$9.80E+00$	+
$F24$	$4.49E+02$	$6.31E+02$	$5.94E+02$	$4.03E+01$	$4.29E+02$	$6.26E+02$	$5.22E+02$	$7.55E+01$	+
$F25$	$3.84E+02$	$4.24E+02$	$3.89E+02$	$6.79E+00$	$3.84E+02$	$3.90E+02$	$3.87E+02$	$1.25E+00$	≈
$F26$	$1.20E+03$	$2.58E+03$	$1.42E+03$	$2.46E+02$	$1.08E+03$	$1.44E+03$	$1.27E+03$	$9.24E+01$	+
$F27$	$5.14E+02$	$5.57E+02$	$5.30E+02$	$1.19E+01$	$5.08E+02$	$5.55E+02$	$5.31E+02$	$1.18E+01$	≈
$F28$	$3.00E+02$	$4.60E+02$	$3.78E+02$	$5.77E+01$	$3.00E+02$	$4.54E+02$	$3.61E+02$	$5.51E+01$	≈
$F29$	$4.03E+02$	$7.77E+02$	$5.36E+02$	$1.09E+02$	$4.22E+02$	$7.80E+02$	$5.00E+02$	$9.69E+01$	≈
$F30$	$2.63E+03$	$1.38E+04$	$6.57E+03$	$3.16E+03$	$2.93E+03$	$1.28E+04$	$4.59E+03$	$2.49E+03$	+

and (14), CurrentVal depicts the current value of termination criterion, and TerminationVal represents the maximum value of termination criterion.

$$\text{Ps\_rate} = \text{Ps\_rate\_min} + \frac{\text{CurrentVal} \times (\text{Ps\_rate\_max} - \text{Ps\_rate\_min})}{\text{TerminationVal}} \tag{13}$$

$$R_{\text{rate}} = R_{\text{rate}_{\text{min}}} - \frac{\text{CurrentVal} \times \left(R_{\text{rate}_{\text{max}}} - R_{\text{rate}_{\text{min}}}\right)}{\text{TerminationVal}} \tag{14}$$

Second, the boundary control and randomization procedures of the traditional EFO are modified. In the AEFO, electromagnets that become higher or lower than the limits are set back to corresponding limits, instead of the random generation within the search limits. Additionally, in the random search step of the proposed AEFO, a randomly selected electromagnet is regenerated within limits instead of a sequence-based approach.

Algorithm 3 summarizes the main steps of the AEFO, which also presents the main change of the proposed algorithm as adaptive control mechanism of Ps_rate and R_rate (line: 35) and the modified boundary check and randomization procedures (line: 26 and line: 28) are shown in Algorithm 3.

Since the SCPIP is a global optimization problem, the proposed AEFO is easily adapted to solve the SD and DD models by integrating equation (10) to the algorithm as the cost function. The decision variables of both the SD and DD models are defined as the electromagnets of the EMPs. On the other hand, the lower and upper limits of the solar cell parameters are set as the boundary values of the electromagnets. With regard to this solution representation, the RMSE value of any solution vector represents the fitness value of

Table 3: Comparisons of the AEFO with the EFO on CEC 2017 benchmark functions ($D = 50$).

Func.	EFO				AEFO				t-test
	Best	Worst	Mean	StdDev	Best	Worst	Mean	StdDev	
$F1$	$1.09E-02$	$1.23E+04$	$2.24E+03$	$2.89E+03$	$8.40E+00$	$1.96E+04$	$4.15E+03$	$5.29E+03$	≈
$F3$	$8.98E+04$	$1.78E+05$	$1.29E+05$	$2.43E+04$	$3.82E+04$	$1.06E+05$	$5.94E+04$	$1.56E+04$	+
$F4$	$2.85E+01$	$1.50E+02$	$8.48E+01$	$4.32E+01$	$2.26E+01$	$1.80E+02$	$8.52E+01$	$4.80E+01$	≈
$F5$	$4.28E+01$	$4.19E+02$	$2.08E+02$	$1.57E+02$	$4.11E+01$	$1.20E+02$	$6.63E+01$	$1.79E+01$	+
$F6$	$1.04E-02$	$1.40E-01$	$6.65E-02$	$3.55E-02$	$5.40E-03$	$6.00E-02$	$2.70E-02$	$1.44E-02$	+
$F7$	$1.18E+02$	$4.84E+02$	$3.96E+02$	$9.87E+01$	$9.78E+01$	$4.58E+02$	$1.74E+02$	$9.17E+01$	+
$F8$	$4.28E+01$	$3.81E+02$	$1.97E+02$	$1.47E+02$	$4.78E+01$	$1.20E+02$	$7.02E+01$	$2.03E+01$	+
$F9$	$3.35E+01$	$7.02E+02$	$1.44E+02$	$1.38E+02$	$9.84E+00$	$2.74E+02$	$5.51E+01$	$5.47E+01$	+
$F10$	$1.28E+04$	$1.44E+04$	$1.36E+04$	$3.79E+02$	$1.24E+04$	$1.42E+04$	$1.34E+04$	$4.32E+02$	≈
$F11$	$1.93E+01$	$3.35E+03$	$4.72E+02$	$6.48E+02$	$2.30E+01$	$1.13E+03$	$1.57E+02$	$2.43E+02$	+
$F12$	$3.43E+05$	$2.07E+06$	$1.11E+06$	$4.00E+05$	$2.04E+05$	$2.38E+06$	$1.09E+06$	$5.97E+05$	≈
$F13$	$1.53E+02$	$2.47E+04$	$5.09E+03$	$6.03E+03$	$7.25E+01$	$9.82E+03$	$1.75E+03$	$2.18E+03$	+
$F14$	$6.00E+03$	$2.05E+05$	$6.03E+04$	$3.98E+04$	$3.01E+03$	$1.94E+05$	$5.24E+04$	$4.03E+04$	≈
$F15$	$5.76E+01$	$1.16E+04$	$4.31E+03$	$3.81E+03$	$2.74E+01$	$1.53E+04$	$4.23E+03$	$4.51E+03$	≈
$F16$	$2.99E+02$	$3.36E+03$	$2.05E+03$	$1.14E+03$	$2.47E+02$	$3.23E+03$	$1.83E+03$	$1.04E+03$	≈
$F17$	$3.82E+02$	$2.27E+03$	$1.89E+03$	$3.49E+02$	$1.96E+02$	$2.04E+03$	$1.22E+03$	$5.75E+02$	+
$F18$	$3.61E+05$	$6.24E+06$	$2.31E+06$	$1.30E+06$	$1.03E+05$	$2.85E+06$	$9.86E+05$	$7.10E+05$	+
$F19$	$7.93E+02$	$2.87E+04$	$1.34E+04$	$7.75E+03$	$5.06E+01$	$3.98E+04$	$1.51E+04$	$1.01E+04$	≈
$F20$	$1.17E+03$	$2.06E+03$	$1.74E+03$	$2.28E+02$	$9.18E+01$	$2.04E+03$	$1.37E+03$	$5.60E+02$	+
$F21$	$2.45E+02$	$5.98E+02$	$4.53E+02$	$1.42E+02$	$2.38E+02$	$5.72E+02$	$3.10E+02$	$9.87E+01$	+
$F22$	$1.29E+04$	$1.47E+04$	$1.39E+04$	$4.41E+02$	$1.20E+04$	$1.42E+04$	$1.34E+04$	$5.69E+02$	+
$F23$	$4.79E+02$	$5.74E+02$	$5.18E+02$	$2.34E+01$	$4.52E+02$	$5.20E+02$	$4.86E+02$	$1.79E+01$	+
$F24$	$5.49E+02$	$9.15E+02$	$8.27E+02$	$1.16E+02$	$5.50E+02$	$9.03E+02$	$7.60E+02$	$1.43E+02$	≈
$F25$	$4.66E+02$	$6.03E+02$	$5.50E+02$	$3.29E+01$	$4.60E+02$	$6.00E+02$	$5.36E+02$	$4.04E+01$	≈
$F26$	$1.79E+03$	$2.63E+03$	$2.18E+03$	$2.42E+02$	$1.58E+03$	$2.63E+03$	$1.95E+03$	$2.52E+02$	+
$F27$	$6.43E+02$	$8.81E+02$	$7.29E+02$	$5.65E+01$	$5.68E+02$	$7.81E+02$	$6.69E+02$	$5.22E+01$	+
$F28$	$4.59E+02$	$6.13E+02$	$5.04E+02$	$2.55E+01$	$4.78E+02$	$5.15E+02$	$5.04E+02$	$8.57E+00$	≈
$F29$	$3.88E+02$	$9.69E+02$	$5.71E+02$	$1.52E+02$	$3.38E+02$	$9.41E+02$	$5.61E+02$	$1.65E+02$	+
$F30$	$7.09E+05$	$1.30E+06$	$9.19E+05$	$1.66E+05$	$6.12E+05$	$1.31E+06$	$9.02E+05$	$1.62E+05$	≈

the EMP. Figure 3 represents the flowchart of the parameter estimation process for the SD and DD models by the proposed AEFO.

5. Computational Results

In order to validate the performance of the proposed AEFO, computational experiments are performed into two main parts. First, the AEFO is tested on a recently introduced global optimization benchmark functions and compared to the EFO. In the second part of the computational studies, a well-known SCPIP benchmark problem set is used. This part initially presents the effects of the parameter values on the algorithm performance. Then, the AEFO is compared to the EFO and also three well-known metaheuristic algorithms: ABC, PSO, and DE. Finally, the results of the proposed AEFO are compared to ten state-of-the-art SCPIP optimizers. For the experimental studies, the AEFO and other competitor algorithms (EFO, ABC, PSO, and DE) are implemented using Matlab 8.1 and executed on the same computer with Intel Xeon CPU (2.67 GHz) and 16 GB of memory.

5.1. Results on Benchmark Functions. In the first part of the computational studies, the proposed AEFO is tested on recently introduced CEC 2017 benchmark functions [48]. CEC 2017 benchmark problem set consists of 30 minimization functions ($F1, F2, \cdots, F30$) including 2 unimodal, 7 multimodal, 10 hybrid, and 10 composition test functions. However, $F2$ is excluded because it shows unstable behavior. The test suite includes four versions based on the problem size (i.e., $D = 10, 30, 50, 100$), where the search space is $[-100,100]^D$ in all test functions. In this study, 30, 50, and 100 dimensional versions are taken into account for the experiments.

Table 4: Comparisons of the AEFO with the EFO on CEC 2017 benchmark functions ($D = 100$).

Func.	EFO				AEFO				t-test
	Best	Worst	Mean	StdDev	Best	Worst	Mean	StdDev	
$F1$	$4.44E+01$	$6.13E+04$	$1.26E+04$	$1.61E+04$	$5.02E+00$	$5.51E+04$	$6.17E+03$	$1.10E+04$	≈
$F3$	$3.17E+05$	$4.93E+05$	$3.99E+05$	$4.12E+04$	$2.31E+05$	$4.50E+05$	$3.14E+05$	$5.39E+04$	+
$F4$	$1.60E+02$	$3.05E+02$	$2.36E+02$	$4.03E+01$	$1.54E+02$	$3.47E+02$	$2.52E+02$	$4.30E+01$	≈
$F5$	$1.70E+02$	$8.58E+02$	$2.41E+02$	$1.21E+02$	$1.49E+02$	$9.20E+02$	$2.36E+02$	$1.43E+02$	≈
$F6$	$1.64E-01$	$4.95E-01$	$2.92E-01$	$8.79E-02$	$8.27E-02$	$3.32E-01$	$1.86E-01$	$5.66E-02$	+
$F7$	$3.43E+02$	$5.81E+02$	$4.42E+02$	$4.98E+01$	$2.74E+02$	$1.12E+03$	$4.20E+02$	$1.50E+02$	≈
$F8$	$1.69E+02$	$3.19E+02$	$2.24E+02$	$4.08E+01$	$1.41E+02$	$2.36E+02$	$1.85E+02$	$2.59E+01$	+
$F9$	$5.56E+02$	$1.48E+04$	$8.00E+03$	$3.71E+03$	$3.70E+02$	$6.60E+03$	$1.79E+03$	$1.56E+03$	+
$F10$	$2.96E+04$	$3.16E+04$	$3.05E+04$	$5.29E+02$	$2.93E+04$	$3.13E+04$	$3.02E+04$	$4.72E+02$	+
$F11$	$2.29E+02$	$6.62E+03$	$2.00E+03$	$1.73E+03$	$3.62E+02$	$2.10E+03$	$6.84E+02$	$3.89E+02$	+
$F12$	$1.05E+06$	$2.88E+06$	$1.90E+06$	$5.04E+05$	$9.83E+05$	$5.12E+06$	$2.13E+06$	$1.04E+06$	≈
$F13$	$9.72E+01$	$2.78E+04$	$5.25E+03$	$7.01E+03$	$1.16E+02$	$2.37E+04$	$5.86E+03$	$6.15E+03$	≈
$F14$	$9.07E+04$	$2.88E+05$	$1.90E+05$	$5.31E+04$	$1.07E+05$	$6.98E+05$	$2.79E+05$	$1.30E+05$	−
$F15$	$1.57E+02$	$1.24E+04$	$3.17E+03$	$3.47E+03$	$1.34E+02$	$1.17E+04$	$2.16E+03$	$2.66E+03$	≈
$F16$	$1.18E+03$	$8.21E+03$	$5.11E+03$	$2.83E+03$	$1.49E+03$	$8.34E+03$	$4.41E+03$	$2.67E+03$	≈
$F17$	$1.46E+03$	$5.51E+03$	$4.76E+03$	$9.15E+02$	$1.17E+03$	$5.33E+03$	$4.40E+03$	$1.24E+03$	≈
$F18$	$8.49E+05$	$7.05E+06$	$2.75E+06$	$1.77E+06$	$2.27E+05$	$3.02E+06$	$1.18E+06$	$6.55E+05$	+
$F19$	$7.97E+01$	$2.00E+04$	$4.32E+03$	$4.93E+03$	$5.51E+01$	$1.28E+04$	$3.15E+03$	$3.40E+03$	≈
$F20$	$4.62E+03$	$5.42E+03$	$5.10E+03$	$1.92E+02$	$4.41E+03$	$5.38E+03$	$5.06E+03$	$2.31E+02$	≈
$F21$	$4.03E+02$	$8.03E+02$	$4.83E+02$	$9.39E+01$	$3.67E+02$	$1.13E+03$	$4.73E+02$	$1.82E+02$	≈
$F22$	$3.03E+04$	$3.25E+04$	$3.14E+04$	$5.01E+02$	$3.00E+04$	$3.26E+04$	$3.12E+04$	$6.00E+02$	≈
$F23$	$7.28E+02$	$8.70E+02$	$7.83E+02$	$4.02E+01$	$6.56E+02$	$8.24E+02$	$7.22E+02$	$4.22E+01$	+
$F24$	$1.13E+03$	$1.37E+03$	$1.26E+03$	$5.84E+01$	$1.09E+03$	$1.29E+03$	$1.18E+03$	$5.41E+01$	+
$F25$	$6.09E+02$	$8.95E+02$	$7.61E+02$	$7.62E+01$	$6.33E+02$	$9.13E+02$	$7.87E+02$	$6.15E+01$	≈
$F26$	$5.69E+03$	$1.02E+04$	$7.03E+03$	$8.31E+02$	$4.85E+03$	$7.66E+03$	$6.06E+03$	$7.32E+02$	+
$F27$	$8.36E+02$	$1.06E+03$	$9.16E+02$	$6.83E+01$	$7.42E+02$	$9.48E+02$	$8.46E+02$	$4.74E+01$	+
$F28$	$5.46E+02$	$6.71E+02$	$5.81E+02$	$3.15E+01$	$5.52E+02$	$6.60E+02$	$5.95E+02$	$2.64E+01$	≈
$F29$	$1.33E+03$	$3.14E+03$	$2.20E+03$	$5.29E+02$	$1.39E+03$	$2.84E+03$	$2.02E+03$	$3.83E+02$	≈
$F30$	$3.20E+03$	$1.84E+04$	$7.31E+03$	$4.31E+03$	$3.39E+03$	$1.70E+04$	$7.92E+03$	$3.91E+03$	≈

Table 5: Effect of P_field and N_field for the SD model.

Algorithm	P_field	N_field	Mean	StdDev	Significance
EFO	0.10	0.45	$9.862039E-04$	$1.327244E-07$	+
AEFO_Standard	0.10	0.45	$9.873994E-04$	$4.897306E-06$	≈
AEFO_0.05P_0.45N	**0.05**	**0.45**	$\mathbf{9.860219E-04}$	$\mathbf{1.747734E-11}$	**NA**
AEFO_0.10P_0.35N	0.10	0.35	$9.860500E-04$	$1.926723E-08$	+
AEFO_0.10P_0.55N	0.10	0.55	$9.861423E-04$	$9.289916E-08$	+
AEFO_0.20P_0.45N	0.20	0.45	$1.326226E-03$	$1.210199E-04$	+

To show the efficiency of the adaptive control mechanism of the AEFO, the proposed algorithm is compared to the original version of the EFO. In order to make a fair comparison between the EFO and the AEFO, the control parameters of the EFO is set as P_field = 0.10, N_field = 0.45, Ps_rate = 0.20, and R_rate = 0.30 [45]. For the AEFO, the parameter values of P_field and N_field are the same as in the EFO, while Ps_rate_min = 0.1, Ps_rate_max = 0.4, R_rate_min =

Table 6: Effect of P_field and N_field for the DD model.

Algorithm	P_field	N_field	Mean	StdDev	Significance
EFO	0.10	0.45	$9.924904E-04$	$3.158046E-06$	+
AEFO_Standard	0.10	0.45	$9.916958E-03$	$1.896457E-05$	+
AEFO_0.05P_0.45N	**0.05**	**0.45**	$\mathbf{9.872174E-04}$	$\mathbf{1.966037E-06}$	**NA**
AEFO_0.10P_0.35N	0.10	0.35	$9.912627E-04$	$4.157936E-06$	+
AEFO_0.10P_0.55N	0.10	0.55	$9.897241E-04$	$3.903512E-06$	+
AEFO_0.20P_0.45N	0.20	0.45	$1.032144E-03$	$1.482347E-05$	+

Table 7: Computational results for the SD model.

Algorithm	Mean	StdDev	Significance
AEFO	$\mathbf{9.860219E-04}$	$\mathbf{1.747734E-11}$	**NA**
EFO	$9.862039E-04$	$1.327244E-07$	+
ABC	$1.064912E-03$	$2.665352E-05$	+
PSO	$2.544277E-03$	$7.974587E-04$	+
DE	$2.092645E-03$	$4.638956E-04$	+

Table 8: Computational results for the DD model.

Algorithm	Mean	StdDev	Significance
AEFO	$\mathbf{9.872174E-04}$	$\mathbf{1.966037E-06}$	**NA**
EFO	$9.924904E-04$	$3.158046E-06$	+
ABC	$1.030773E-03$	$1.867134E-05$	+
PSO	$2.090188E-03$	$4.464503E-04$	+
DE	$2.308334E-03$	$5.233565E-04$	+

0.1, and Ps_rate_max = 0.4. Since these values are the lower and upper limits of Ps_rate and R_rate in the EFO, the AEFO with these parameter values are called hereafter as "AEFO_Standard." Additionally, the population sizes and termination criteria of the algorithms are set to 50 and $D \times 10^3$ maximum function evaluation number, respectively. With regard to these parameter settings, 30 independent runs are performed for each test function.

Tables 2–4 show the results of algorithms for 30, 50, and 100 dimensional versions of the benchmark functions, respectively. To analyze the obtained results, best, worst, mean, and standard deviation (StdDev) of the runs are presented of the algorithms. Furthermore, the last column of the tables shows the statistical significance of the differences between the AEFO and the EFO, which are determined using t-tests with a significance level of 0.05, where the symbols $(+, -, \approx)$ indicate that the AEFO is significantly better than the EFO $(+)$, significantly worse than the EFO $(-)$, and no significant difference between the AEFO and the EFO $(\approx)$. Results of the statistical comparisons show that the proposed AEFO obtains significantly better results than the EFO for most of the benchmark functions over all dimensions. In particular, 17 better results are obtained by the AEFO for both 30 and 50 dimensional versions. On the problems with $D = 100$, the EFO outperforms the AEFO for only one $F14$ while the AEFO shows significantly better performance than the EFO for 11 benchmark functions. On the other hand, it should be noted that the AEFO reaches better average results with smaller deviations compared to the EFO, which demonstrates the robustness of the proposed AEFO.

5.2. Results on the Solar Cell Parameter Identification Problem. In this section, two sets of tests are carried on different PV models (i.e., SD and DD models) to exhibit the performance of the proposed AEFO on the SCPIP. First, the effect of different positive and negative field ratios on the performance of the AEFO is analyzed. Second, the AEFO is compared to the original EFO and other well-known metaheuristic algorithm (i.e., ABC, PSO, and DE) in detail. Furthermore, in this subsection, results of the AEFO are compared with a number of state-of-the-art algorithm results. For these experiments, the population sizes are set to 40, and 5-second time limit is employed as the termination criteria for all algorithms.

For the experiments, the $I - V$ characteristics of a 57 mm diameter commercial (R.T.C. France) silicon solar cell (under 1000 W/m^2 at 33°C) are used as the benchmark data [7]. This benchmark set has been widely used to evaluate the performance of different optimization algorithms [27, 32, 34, 37–39]. The lower and upper values of the solar cell parameters of both models are given in Table 1.

5.2.1. Effect of Positive and Negative Field Ratios. Similar to most of the metaheuristic algorithms, the performance of the EFO is affected by the main control parameters, i.e., P_field and N_field. Therefore, to evaluate the effect of different positive and negative field ratios, four different AEFO variants are additionally generated based on parameter values. For these variants, P_field and N_field vary between 0.05 to 0.20 and 0.35 to 0.55, respectively. Further, Ps_rate_min, Ps_rate_max, R_rate_max, and R_rate_min are set as the lowest and highest values given in [45], where Ps_rate_min = 0.1, Ps_rate_max = 0.4, Ps_rate_max = 0.4, and R_rate_min = 0.1 for the AEFO variants.

The computational results reached by the EFO, AEFO_Standard, and AEFO variants on the SD and DD models are given in Tables 5 and 6. In these tables, results are given in terms of mean and standard deviations of the RMSE values, achieved from 30 independent runs. For each model, the best mean RMSE values are highlighted in boldface. As can be seen from Tables 5 and 6, AEFO having P_field =

Table 9: Best solar cell parameters estimated for the SD model.

Algorithm	$R_s(\Omega)$	$R_{sh}(\Omega)$	$I_{ph}(A)$	$I_{sd}(\mu A)$	n	RMSE
AEFO	0.036377	53.718780	0.760776	0.323024	1.481185	**9.860219*E* − 04**
EFO	0.036343	53.985341	0.760766	0.325679	1.482009	9.861529*E* − 04
ABC	0.036764	52.392307	0.760847	0.297008	1.472745	1.004136*E* − 03
PSO	0.036849	47.298544	0.761006	0.281247	1.467460	1.040425*E* − 03
DE	0.035667	52.086190	0.761849	0.388961	1.500131	1.223675*E* − 03

Table 10: Best solar cell parameter estimated for the DD model.

Algorithm	$R_s(\Omega)$	$R_{sh}(\Omega)$	$I_{ph}(A)$	$I_{sd1}(\mu A)$	$I_{sd2}(\mu A)$	n_1	n_2	RMSE
AEFO	0.036725	55.350717	0.760780	0.224652	0.318297	1.450886	1.959721	**9.829448*E* − 04**
EFO	0.030634	54.035734	0.760763	0.052122	0.276559	1.515336	1.477839	9.862828*E* − 04
ABC	0.036426	52.480446	0.760710	0.133564	0.270400	1.749054	1.467930	1.000899*E* − 03
PSO	0.035545	66.627653	0.760260	0.015350	0.390903	1.590451	1.502377	1.101223*E* − 03
DE	0.036308	80.219564	0.759208	0.359122	0.045338	1.491640	2.435739	1.423969*E* − 03

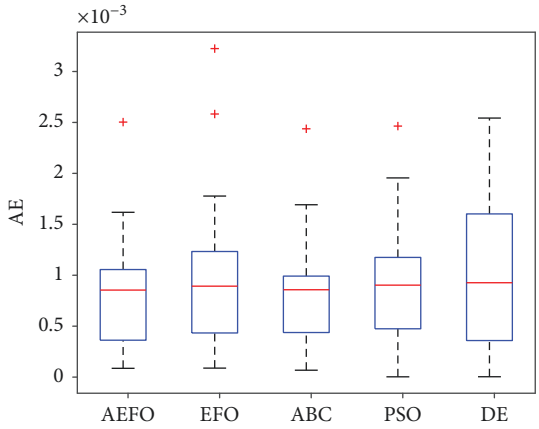

Figure 4: Box plots of AE for the SD model.

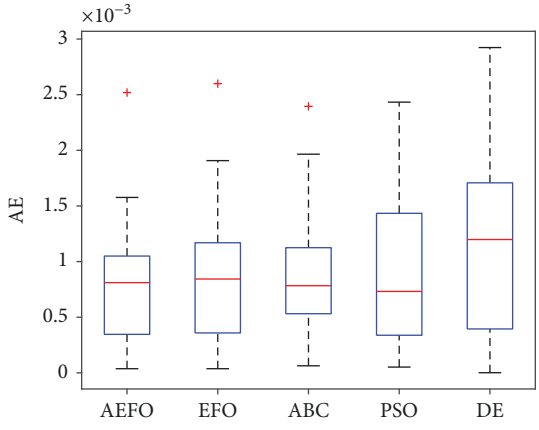

Figure 5: Box plots of AE for the DD model.

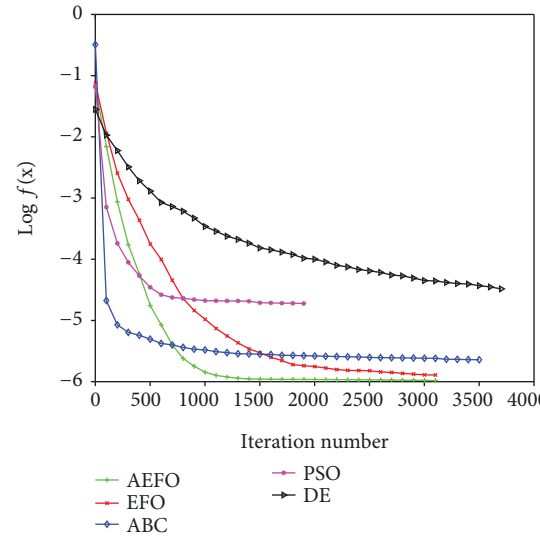

Figure 6: Convergence curves for the SD model.

0.05 and $N\_field = 0.45$ (AEFO_0.05P_0.45N) outperforms other algorithms based on the mean RMSE values. For a precise and pairwise comparison, statistical significances of the differences between the mean RMSE values of the AEFO_0.05P_0.45N and other AEFO variants are analyzed using t-tests where the significance level is set to 0.05. As in the previous subsection, " + " in the last column shows that AEFO_0.05P_0.45N performs statistically better than the competitor algorithm, " ≈ " indicates that the difference between the AEFO_0.05P_0.45N and the competitor algorithm is not statistically significant, and " − " depicts that the competitor algorithm outperforms the AEFO_0.05P_0.45N at a level of 0.05 significance.

The computation results for the SD model are shown in Table 5, which reveal that the standard deviation of the AEFO is less and the mean is better than the peer algorithms. As can be seen from Table 5, the mean RMSE values achieved by the AEFO_0.05P_0.45N is $9.860219E-04$ with $1.747734E-11$ standard deviation after 30 runs. The

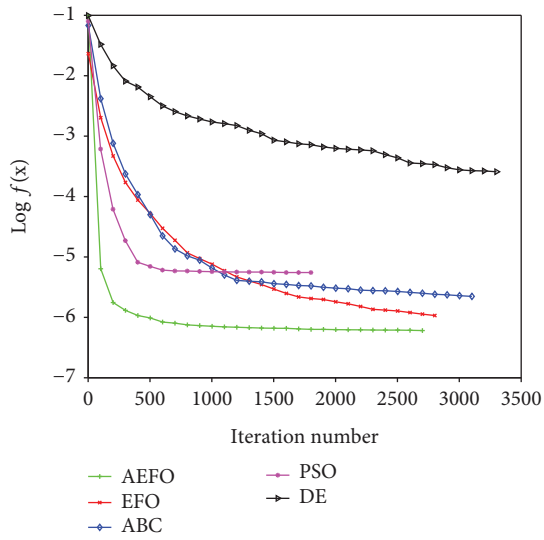

Figure 7: Convergence curves for the DD model.

AEFO_0.05P_0.45N is apparently superior to all competitor algorithms except AEFO_Standard, where there is no statistically significant difference between the two algorithms. For the DD model, which is presented in Table 6, the AEFO_0.05P_0.45N reaches the mean RMSE value of $9.872174E-04$ with $1.966037E-06$ standard deviation after 30 runs. It can be clearly concluded from Table 6 that the AEFO produces promising results and outperforms all algorithms. As a result of the experiments, the P_field and N_field values are set as 0.05 and 0.45, respectively. Further, the AEFO_0.05P_0.45N is depicted as the AEFO for the upcoming comparison studies.

5.2.2. Results of Comparisons. In this subsection, the performance of the AEFO is analyzed in detail and compared with the EFO and also with well-known metaheuristic algorithms. First, the AEFO is compared with the EFO, ABC, PSO, and DE, where all algorithms are coded and executed in the same environment. The parameters of the competitor algorithms are set by the original parameter values, which were given in corresponding papers, except the termination value and population sizes, which are set the same for all algorithms. For the DE, the classical DE (rand/1/bin) [49] was used the same parameter settings as in [50] were followed, where $F = 0.9$ and $\mathrm{CR} = 0.9$. For the PSO, the parameter values for w, $c1$, and $c2$ are set as 1.193, 1.193, and 0.721, respectively [51]. For the ABC, the only control parameter is the limit value, which is set to $\mathrm{Pop} \times D$, where Pop is the population size and D is problem size, as given in [52].

The computational results are tabulated in Tables 7 and 8 in terms of mean and standard deviations of the RMSE values from 30 independent runs. The last columns in Tables 7 and 8 show the significance values similarly to the pervious subsection. It can be seen from Tables 7 and 8 that the AEFO performs significantly better than the EFO, PSO, ABC, and DE, which validates the promising performance of the AEFO over the SCPIP. In other words, this further indicates the effectiveness of the proposed adaptive approach in the AEFO.

Additionally, the estimated solar cell parameter values and corresponding RMSE values, which are achieved from the best run, are listed in Tables 9 and 10 for the SD and DD models, respectively. Figures 4 and 5 present the box plots of the absolute error (AE) values for each model. The AE can be described as follows:

$$\mathrm{AE}_m = \left| I_e - I_{\exp} \right| \tag{15}$$

where $m = 1, \cdots, M$ are the data points; I_e and $I_{\exp}$ are the estimated and experimentally determined currents, respectively. It can be seen from Figures 4 and 5 that the AEFO achieves smaller AE values when compared to the EFO, ABC, PSO, and DE. Lastly, Figures 6 and 7 demonstrate the convergence histories of the test algorithms on both benchmark problems. To show the convergence processes clearly, here, the y axes adopt the RMSE logarithm values. From Figures 6 and 7, it is obvious that the AEFO has a better convergence performance than the EFO, PSO, ABC, and DE, which also shows the outperforming convergence ability of the AEFO.

Second, to further show the superiority of the AEFO, the performance of the AEFO is compared with the ten state-of-the-art SCPIP optimizers, such as the ABC [22], artificial bee swarm optimization (ABCO) [11], biogeography-based optimization algorithm with mutation strategies (BBO-M) [28], cat swarm optimization (CSO) [33], generalized oppositional teaching-learning-based optimization (GOTLBO) [27], harmony search-based algorithm (HSA) [4], improved JAYA algorithm (IJAYA) [34], mutative-scale parallel chaos optimization algorithm (MPCOA) [29], and teaching-learning-based ABC (TLABC) [53]. Tables 11 and 12 present the best found RMSE values, corresponding the parameters, mean, and standard deviations of RMSE over the runs obtained by the AEFO and other ten algorithms on SD and DD. It should be noted that the results of competitor algorithms are directly taken from the corresponding studies. Among the results listed in Tables 11 and 12, the best RMSE, mean, and deviations on each problem among all algorithms are shown in bold. As per the results in Table 11, the AEFO reaches better solutions than all algorithms expect the CSO, where they show similar performance. On the other hand, the AEFO achieves smaller standard deviations than the CSO, which indicated the robustness of the AEFO. Inspecting the results in Table 12, it is evident that the AEFO outperforms all algorithms in terms of mean and standard deviation. As summary, results in Tables 11 and 12 reveal that the AEFO has the potential to be very effective in solving the SCPIP.

6. Conclusion

In this paper, an adaptive version of the EFO is introduced to solve the SCPIP for single- and double-diode mathematical models. The proposed AEFO efficiently searches the solution space with regard to adaptive changes on the importance of negative and positive electromagnetic particles. To test the performance of the proposed algorithm, computational studies are carried out into two parts. In the first part, the

Table 11: Comparisons of 10 recent algorithms with the best found SD model solution by the AEFO.

Algorithm	$R_s(\Omega)$	$R_{sh}(\Omega)$	I_{ph}(A)	$I_{sd}(\mu$A)	n	RMSE	Mean	StdDev
AEFO	0.036377	53.718780	0.760776	0.323024	1.481185	**9.860219*E* − 04**	**9.860219*E* − 04**	**1.747734*E* − 11**
ABC	0.036400	53.643300	0.760800	0.325100	1.481700	9.862*E* − 04	1.0*E* − 03	1.497*E* − 05
ABSO	0.036590	52.290300	0.760800	0.306230	1.475830	9.9124*E* − 04	NA	NA
BBO_M	0.036420	53.362270	0.760780	0.318740	1.479840	9.8634*E* − 04	NA	NA
CSO	0.036380	53.718500	0.760780	0.323000	1.481180	**9.8602*E* − 04**	**9.8602*E* − 04**	5.4941*E* − 09
GOTLBO	0.036265	54.115426	0.760780	0.331552	1.483820	9.87442*E* − 04	1.33488*E* − 03	2.99407*E* − 04
HSA	0.036630	53.594600	0.760700	0.304950	1.475380	9.9510*E* − 04	NA	NA
IJAYA	0.036400	53.759500	0.760800	0.322800	1.481100	9.8603*E* − 04	9.9204*E* − 04	1.4033*E* − 05
MPCOA	0.036350	54.632800	0.760730	0.326550	1.481680	9.4457*E* − 04	NA	NA
STLBO	0.036380	53.718700	0.760780	0.323020	1.481140	**9.8602*E* − 04**	NA	NA
TLABC	0.036380	53.716360	0.760780	0.323020	1.481180	**9.86022*E* − 04**	9.98523*E* − 04	1.86022*E* − 06

NA: not available in the literature.

Table 12: Comparisons of 10 recent algorithms with the best found DD model solution by the AEFO.

Algorithm	$R_s(\Omega)$	$R_{sh}(\Omega)$	I_{ph}(A)	$I_{sd1}(\mu$A)	$I_{sd2}(\mu$A)	n_1	n_2	RMSE	Mean	StdDev
AEFO	0.036725	55.350717	0.760780	0.224652	0.318297	1.450886	1.959721	9.829448*E* − 04	**9.872174*E* − 04**	**1.966037*E* − 06**
ABC	0.036400	53.780400	0.760800	0.040700	0.287400	1.449500	1.488500	9.861*E* − 04	1.0*E* − 03	3.285*E* − 05
ABSO	0.036570	54.621900	0.760780	0.267130	0.381910	1.465120	1.981520	9.8344*E* − 04	NA	NA
BBO_M	0.036640	55.049400	0.760830	0.591150	0.245230	2.000000	1.457980	9.8272*E* − 04	NA	NA
CSO	0.036737	55.381300	0.760780	0.227320	0.727850	1.451510	1.997690	9.8252*E* − 04	9.9619*E* − 04	3.4671*E* − 05
GOTLBO	0.036783	56.075304	0.760752	0.800195	0.220462	1.999973	1.448974	9.83177*E* − 04	1.24360*E* − 03	2.09115*E* − 04
HSA	0.035450	46.826960	0.761760	0.125450	0.254700	1.494390	1.499890	1.26*E* − 03	NA	NA
IJAYA	0.037600	77.851900	0.760100	0.005045	0.750940	1.218600	1.624700	9.8293*E* − 04	1.0269*E* − 03	9.8625*E* − 05
MPCOA	0.036350	54.253100	0.760780	0.312590	0.045280	1.478440	1.784590	**9.2163*E* − 04**	**NA**	**NA**
STLBO	0.036740	55.492000	0.760780	0.225660	0.752170	1.450850	2.000000	9.8248*E* − 04	NA	NA
TLABC	0.036670	54.667970	0.760810	0.423940	0.240110	1.907500	1.456710	9.84145*E* − 04	1.05553*E* − 03	1.55034*E* − 04

NA: not available in the literature.

proposed AEFO is tested on a recently introduced global optimization benchmark problem set and compared to the standard EFO. Results of these experiments show that the AEFO exhibits better performance and outperforms the EFO by finding better results for most of the functions. In the second part of the computational studies, the AEFO is tested on a well-known SCPIP benchmark problem set, which is generated from 57 mm diameter commercial silicon solar cell. First, the performance of the AEFO and AEFO variants is pointed out by comparing it with the EFO. After that, the results of the AEFO are compared with the results of the EFO, ABC, PSO, and DE in detail. Since the $I - V$ characteristics of the solar cells are determined with very small values, the RMSE values are directly affected by the rounding procedures while computing the errors. In addition to the comparisons based on the RMSE values, t-tests are carried out to reveal whether there exists a significant difference between the AEFO and the other metaheuristic approaches. Finally, the performance of the AEFO is compared with ten state-of-the-art algorithms. As a result of the computational studies, it should be specified that the proposed AEFO is capable of finding effective results for the SCPIP concerning other metaheuristic algorithms considered in this study.

As a future work, this study may be extended by considering other mathematical models for the solar cells. Furthermore, the proposed AEFO can be performed on new benchmark problems generated by different commercial solar cells to further validate the performance of the algorithm. Finally, the proposed algorithm may be adapted for other continuous optimization problems existing in the literature.

Data Availability

Previously reported solar cell data were used to support this study and are available at doi.org/10.1080/01425918608909835. This prior study (and datasets) is cited at relevant places within the text as reference [7].

Conflicts of Interest

The author declares that there is no conflict of interests regarding the publication of this article.

References

[1] D. H. Muhsen, A. B. Ghazali, T. Khatib, and I. A. Abed, "Parameters extraction of double diode photovoltaic module's model based on hybrid evolutionary algorithm," *Energy Conversion and Management*, vol. 105, pp. 552–561, 2015.

[2] G. K. Singh, "Solar power generation by PV (photovoltaic) technology: a review," *Energy*, vol. 53, pp. 1–13, 2013.

[3] Z. Chen, L. Wu, P. Lin, Y. Wu, and S. Cheng, "Parameters identification of photovoltaic models using hybrid adaptive Nelder-Mead simplex algorithm based on eagle strategy," *Applied Energy*, vol. 182, pp. 47–57, 2016.

[4] A. Askarzadeh and A. Rezazadeh, "Parameter identification for solar cell models using harmony search-based algorithms," *Solar Energy*, vol. 86, no. 11, pp. 3241–3249, 2012.

[5] D. Allam, D. A. Yousri, and M. B. Eteiba, "Parameters extraction of the three diode model for the multi-crystalline solar cell/module using moth-flame optimization algorithm," *Energy Conversion and Management*, vol. 123, pp. 535–548, 2016.

[6] M. Diantoro, T. Suprayogi, A. Hidayat, A. Taufiq, A. Fuad, and R. Suryana, "Shockley's Equation Fit Analyses for Solar Cell Parameters from I-V Curves," *International Journal of Photoenergy*, vol. 2018, 7 pages, 2018.

[7] T. Easwarakhanthan, J. Bottin, I. Bouhouch, and C. Boutrit, "Nonlinear minimization algorithm for determining the solar cell parameters with microcomputers," *International Journal of Solar Energy*, vol. 4, no. 1, pp. 1–12, 1986.

[8] D. S. H. Chan, J. R. Phillips, and J. C. H. Phang, "A comparative study of extraction methods for solar cell model parameters," *Solid-State Electronics*, vol. 29, no. 3, pp. 329–337, 1986.

[9] A. Ortizconde, F. Garciasanchez, and J. Muci, "New method to extract the model parameters of solar cells from the explicit analytic solutions of their illuminated *I–V* characteristics," *Solar Energy Materials and Solar Cells*, vol. 90, no. 3, pp. 352–361, 2006.

[10] H. Saleem and S. Karmalkar, "An analytical method to extract the physical parameters of a solar cell from four points on the illuminated *J-V* curve," *IEEE Electron Device Letters*, vol. 30, no. 4, pp. 349–352, 2009.

[11] A. Askarzadeh and A. Rezazadeh, "Artificial bee swarm optimization algorithm for parameters identification of solar cell models," *Applied Energy*, vol. 102, pp. 943–949, 2013.

[12] K. M. El-Naggar, M. R. AlRashidi, M. F. AlHajri, and A. K. Al-Othman, "Simulated annealing algorithm for photovoltaic parameters identification," *Solar Energy*, vol. 86, no. 1, pp. 266–274, 2012.

[13] J. A. Jervase, H. Bourdoucen, and A. Al-Lawati, "Solar cell parameter extraction using genetic algorithms," *Measurement Science and Technology*, vol. 12, no. 11, pp. 1922–1925, 2001.

[14] M. Zagrouba, A. Sellami, M. Bouaïcha, and M. Ksouri, "Identification of PV solar cells and modules parameters using the genetic algorithms: application to maximum power extraction," *Solar Energy*, vol. 84, no. 5, pp. 860–866, 2010.

[15] M. Ye, X. Wang, and Y. Xu, "Parameter extraction of solar cells using particle swarm optimization," *Journal of Applied Physics*, vol. 105, no. 9, article 094502, 2009.

[16] A. R. Jordehi, "Time varying acceleration coefficients particle swarm optimisation (TVACPSO): a new optimisation algorithm for estimating parameters of PV cells and modules," *Energy Conversion and Management*, vol. 129, pp. 262–274, 2016.

[17] K. Ishaque and Z. Salam, "An improved modeling method to determine the model parameters of photovoltaic (PV) modules using differential evolution (DE)," *Solar Energy*, vol. 85, no. 9, pp. 2349–2359, 2011.

[18] L. L. Jiang, D. L. Maskell, and J. C. Patra, "Parameter estimation of solar cells and modules using an improved adaptive differential evolution algorithm," *Applied Energy*, vol. 112, pp. 185–193, 2013.

[19] W. Gong and Z. Cai, "Parameter extraction of solar cell models using repaired adaptive differential evolution," *Solar Energy*, vol. 94, pp. 209–220, 2013.

[20] K. Ishaque, Z. Salam, S. Mekhilef, and A. Shamsudin, "Parameter extraction of solar photovoltaic modules using penalty-based differential evolution," *Applied Energy*, vol. 99, pp. 297–308, 2012.

[21] M. F. AlHajri, K. M. El-Naggar, M. R. AlRashidi, and A. K. Al-Othman, "Optimal extraction of solar cell parameters using pattern search," *Renewable Energy*, vol. 44, pp. 238–245, 2012.

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

[23] N. Rajasekar, N. Krishna Kumar, and R. Venugopalan, "Bacterial foraging algorithm based solar PV parameter estimation," *Solar Energy*, vol. 97, pp. 255–265, 2013.

[24] M. A. Awadallah, "Variations of the bacterial foraging algorithm for the extraction of PV module parameters from nameplate data," *Energy Conversion and Management*, vol. 113, pp. 312–320, 2016.

[25] Q. Niu, H. Zhang, and K. Li, "An improved TLBO with elite strategy for parameters identification of PEM fuel cell and solar cell models," *International Journal of Hydrogen Energy*, vol. 39, no. 8, pp. 3837–3854, 2014.

[26] S. J. Patel, A. K. Panchal, and V. Kheraj, "Extraction of solar cell parameters from a single current–voltage characteristic using teaching learning based optimization algorithm," *Applied Energy*, vol. 119, pp. 384–393, 2014.

[27] X. Chen, K. Yu, W. Du, W. Zhao, and G. Liu, "Parameters identification of solar cell models using generalized oppositional teaching learning based optimization," *Energy*, vol. 99, pp. 170–180, 2016.

[28] Q. Niu, L. Zhang, and K. Li, "A biogeography-based optimization algorithm with mutation strategies for model parameter estimation of solar and fuel cells," *Energy Conversion and Management*, vol. 86, pp. 1173–1185, 2014.

[29] X. Yuan, Y. Xiang, and Y. He, "Parameter extraction of solar cell models using mutative-scale parallel chaos optimization algorithm," *Solar Energy*, vol. 108, pp. 238–251, 2014.

[30] W. Han, H.-H. Wang, and L. Chen, "Parameters identification for photovoltaic module based on an improved artificial fish swarm algorithm," *The Scientific World Journal*, vol. 2014, 12 pages, 2014.

[31] A. Askarzadeh and L. dos Santos Coelho, "Determination of photovoltaic modules parameters at different operating conditions using a novel bird mating optimizer approach," *Energy Conversion and Management*, vol. 89, pp. 608–614, 2015.

[32] B. Jacob, K. Balasubramanian, S. Babu T, S. M. Azharuddin, and N. Rajasekar, "Solar PV modelling and parameter extraction using artificial immune system," *Energy Procedia*, vol. 75, pp. 331–336, 2015.

[33] L. Guo, Z. Meng, Y. Sun, and L. Wang, "Parameter identification and sensitivity analysis of solar cell models with cat swarm optimization algorithm," *Energy Conversion and Management*, vol. 108, pp. 520–528, 2016.

[34] K. Yu, J. J. Liang, B. Y. Qu, X. Chen, and H. Wang, "Parameters identification of photovoltaic models using an improved JAYA optimization algorithm," *Energy Conversion and Management*, vol. 150, pp. 742–753, 2017.

[35] K. Yu, B. Qu, C. Yue, S. Ge, X. Chen, and J. Liang, "A performance-guided JAYA algorithm for parameters identification of photovoltaic cell and module," *Applied Energy*, vol. 237, pp. 241–257, 2019.

[36] D. Oliva, M. Abd el Aziz, and A. Ella Hassanien, "Parameter estimation of photovoltaic cells using an improved chaotic whale optimization algorithm," *Applied Energy*, vol. 200, pp. 141–154, 2017.

[37] A. Fathy and H. Rezk, "Parameter estimation of photovoltaic system using imperialist competitive algorithm," *Renewable Energy*, vol. 111, pp. 307–320, 2017.

[38] J. P. Ram, T. S. Babu, T. Dragicevic, and N. Rajasekar, "A new hybrid bee pollinator flower pollination algorithm for solar PV parameter estimation," *Energy Conversion and Management*, vol. 135, pp. 463–476, 2017.

[39] X. Gao, Y. Cui, J. Hu et al., "Parameter extraction of solar cell models using improved shuffled complex evolution algorithm," *Energy Conversion and Management*, vol. 157, pp. 460–479, 2018.

[40] Y. Yoon and Z. W. Geem, "Parameter optimization of single-diode model of photovoltaic cell using memetic algorithm," *International Journal of Photoenergy*, vol. 2015, 7 pages, 2015.

[41] D. Kler, Y. Goswami, K. P. S. Rana, and V. Kumar, "A novel approach to parameter estimation of photovoltaic systems using hybridized optimizer," *Energy Conversion and Management*, vol. 187, pp. 486–511, 2019.

[42] H. G. G. Nunes, J. A. N. Pombo, P. M. R. Bento, S. J. P. S. Mariano, and M. R. A. Calado, "Collaborative swarm intelligence to estimate PV parameters," *Energy Conversion and Management*, vol. 185, pp. 866–890, 2019.

[43] X. Chen and K. Yu, "Hybridizing cuckoo search algorithm with biogeography-based optimization for estimating photovoltaic model parameters," *Solar Energy*, vol. 180, pp. 192–206, 2019.

[44] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 67–82, 1997.

[45] H. Abedinpourshotorban, S. Mariyam Shamsuddin, Z. Beheshti, and D. N. A. Jawawi, "Electromagnetic field optimization: a physics-inspired metaheuristic optimization algorithm," *Swarm and Evolutionary Computation*, vol. 26, pp. 8–22, 2016.

[46] M. Şahin and T. Kellegöz, "Balancing multi-manned assembly lines with walking workers: problem definition, mathematical formulation, and an electromagnetic field optimisation algorithm," *International Journal of Production Research*, pp. 1–19, 2019, In press.

[47] H. R. E. H. Bouchekara, M. Zellagui, and M. A. Abido, "Optimal coordination of directional overcurrent relays using a modified electromagnetic field optimization algorithm," *Applied Soft Computing*, vol. 54, pp. 267–283, 2017.

[48] N. H. Awad, M. Z. Ali, P. N. Suganthan, J. J. Liang, and B. Y. Qu, *Problem definitions and evaluation criteria for the CEC 2017 special session and competition on single objective real-parameter numerical optimization*, Technical Report, Nanyang Technological University, Singapore, 2016.

[49] K. Price, R. M. Storn, and J. A. Lampinen, *Differential evolution: a practical approach to global optimization*, Springer Science & Business Media, 2006.

[50] J. Rönkkönen, S. Kukkonen, and K. V. Price, "Real-parameter optimization with differential evolution," in *2005 IEEE Congress on Evolutionary Computation*, pp. 506–513, Edinburgh, UK, September 2005.

[51] Particle Swarm Central, "Standard PSO 2007 code," 2010, http://www.particleswarm.info.

[52] D. Karaboga and B. Akay, "A comparative study of artificial bee colony algorithm," *Applied Mathematics and Computation*, vol. 214, no. 1, pp. 108–132, 2009.

[53] X. Chen, B. Xu, C. Mei, Y. Ding, and K. Li, "Teaching–learning–based artificial bee colony for solar photovoltaic parameter estimation," *Applied Energy*, vol. 212, pp. 1578–1588, 2018.