Article

Parameter Estimation of Static/Dynamic Photovoltaic Models Using a Developed Version of Eagle Strategy Gradient-Based Optimizer

Abdelhady Ramadan [1], Salah Kamel [1] , Mohamed H. Hassan [1] , Marcos Tostado-Véliz [2],* and Ali M. Eltamaly [3,4,5,*

[1] Department of Electrical Engineering, Faculty of Engineering, Aswan University, Aswan 81542, Egypt; eng.abdalhady@gmail.com (A.R.); skamel@aswu.edu.eg (S.K.); mohamedhosnymoee@gmail.com (M.H.H.)
[2] Electrical Engineering Department, University of Jaen, 23071 Jaén, Spain
[3] K.A.CARE Energy Research and Innovation Center at Riyadh, Riyadh 11451, Saudi Arabia
[4] Sustainable Energy Technologies Center, King Saud University, Riyadh 11421, Saudi Arabia
[5] Electrical Engineering Department, Mansoura University, Mansoura 35516, Egypt
* Correspondence: mtostado@ujaen.es (M.T.-V.); eltamaly@ksu.edu.sa (A.M.E.)

Abstract: The global trend towards renewable energy sources, especially solar energy, has had a significant impact on the development of scientific research to manufacture high-performance solar cells. The issue of creating a model that simulates a solar module and extracting its parameter is essential in designing an improved and high performance photovoltaic system. However, the nonlinear nature of the photovoltaic cell increases the challenge in creating this model. The application of optimization algorithms to solve this issue is increased and developed rapidly. In this paper, a developed version of eagle strategy GBO with chaotic (ESCGBO) is proposed to enhance the original GBO performance and its search efficiency in solving difficult optimization problems such as this. In the literature, different PV models are presented, including static and dynamic PV models. Firstly, in order to evaluate the effectiveness of the proposed ESCGBO algorithm, it is executed on the 23 benchmark functions and the obtained results using the proposed algorithm are compared with that obtained using three well-known algorithms, including the original GBO algorithm, the equilibrium optimizer (EO) algorithm, and wild horse optimizer (WHO) algorithm. Furthermore, both of original GBO and developed ESCGBO are applied to estimate the parameters of single and double diode as static models, and integral and fractional models as examples for dynamic models. The results in all applications are evaluated and compared with different recent algorithms. The results analysis confirmed the efficiency, accuracy, and robustness of the proposed algorithm compared with the original one or the recent optimization algorithms.

Keywords: solar energy; static PV models; dynamic PV models; optimization; GBO; eagle strategy GBO; chaotic maps

1. Introduction

Recently, since the surge in fossil fuel prices, the world's attention has turned to renewable energy sources, such as wind and solar energy. Solar energy has several advantages beyond resources. These advantages are simple in installation, low maintenance activities, and suitable for different sizes. Although the mentioned advantages of solar energy are still weak regarding high efficiency [1], increased interest in renewable energy sources has led to an increase in researchers' interest in developing these systems and increasing their efficiency. Developing an optimal mathematical model that simulates the natural photovoltaic system is one of the biggest challenges for researchers. The difficulty of developing these models is due to the nonlinear properties of solar cells [2].

Different types of PV models are proposed in the literature. The static model based on the main characteristics of the PV cell as it consists of two semiconductor materials

(p-type and n- type) to achieve simple P-N junction characteristics, is considered the base element of the PV system. The PV module is a connection of series and shunt PV cells [3]. The simplest static model is a single diode model (SDM), which has one diode connected with series and shunt resistance [4–7]. The development in this type of models is based on increasing the accuracy of the models by representing other effects. The double diode model (DDM) is developed to represent the recombination effect in the PV cell. DDM consists of two diodes with one series and one shunt resistance. The total estimated parameters in the DDM are seven parameters, making it more complex than SDM [8,9]. By increasing the number of diodes, other effects can be represented in the model, reflected in increasing the model accuracy; on the other hand, the estimated parameters for the model is also increased, reflected in increasing the model complexity. This idea is illustrated in the three diode model (TDM) [10–12]. TDM has three diodes to represent the effect of leakage current and grain boundaries. TDM has nine estimated parameters, so although the model is more accurate, it is considered more complex. The balance between accuracy and complexity is determined according to the application [13]. Although the static model has a wide range of developments in the literature and is more representative to the PV system, the representation of the load connection, variation, and switching are missed. The dynamic model has been proposed in the literature to overcome this problem by representing the load connection in the model [14–17]. The dynamic models proposed in the literature are the integer and fractional dynamic models. The integer dynamic model is considered the more popular dynamic model and the fractional model has been developed to increase the accuracy of the integer model [18]. All these models have different parameters. These parameters control the model output (dependent variable) based on the model input (independent variable). The well estimation of these parameters has a direct effect in the model accuracy, for which many researchers have proposed to discuss. Parameters estimation using optimization algorithms has been discussed by several researchers. A review about the applied optimization techniques to estimate the parameters of the PV models is proposed in [19]. The numerical/analytical methods are applied for the optimization problems but these methods achieve low accuracy solutions. Population based algorithms are applied widely for these problems as it is simple in application with more accurate results. Population based algorithms are too many to discuss in this paper, but one example includes the recently proposed Whippy Harris Hawks Optimization Algorithm (WHHO) [20]. WHHO is an enhancement for HHO proposed by [21]. HHO is inspired by a group of hawks in catching their prey. WHHO improves the weak in HHO by reducing the local optima problem occurrence besides increasing the algorithm search speed. This meta-heuristic method is a derivative-free optimization method. It has no restrictions on the objective function and has a great advantage in solving multimodal problems. Therefore, it is employed to solve many optimization problems, such as the high performance SPR sensor design [22], solar-based DG allocation [23], optimal reactive power dispatch [24], and parameter estimation of photovoltaic [25–27]. One recent application of meta-heuristic algorithm for PV parameters estimation is the application of wild horse optimizer (WHO) for parameter estimation of modified double-diode and triple-diode photovoltaic models proposed in [28]. Another interesting work is the application of an improved bald eagle search algorithm for parameter estimation of different photovoltaic models proposed in [29]. The improved algorithm has been tested through three static PV models SDM, DDM, and TDM. Moreover, three modified static PV models MSDM, MDDM, and MTDM. The study in [28,29] is interesting due to it using recent algorithms with modified static models, but the dynamic PV models are missed.

　　　The GBO algorithm is considered a metaheuristic optimization algorithm inspired by gradient descent and the Newton method [30]. GBO has advantage of rapid convergence due to gradient search rules and also reducing probability of escaping from local optima. ESCGBO is a modified version of GBO proposed in this paper. The ESCGBO technique is an enhancement applied to the GBO using eagle strategy (ES) with chaotic method to enhance the balance between global search or exploration and local search or

exploitation and then increasing the original algorithm performance. The ES has been used in several studies to improve the performance of the conventional algorithms, such as an improved particle swarm optimization (PSO) algorithm [31,32], adaptive Nelder–Mead simplex algorithm [33], and JAYA algorithm and Nelder–Mead simplex method [34], eagle strategy with flower algorithm [35], while chaotic maps were also used to enhance several optimization algorithms, such as developing chaotic HHO algorithm [36], chaotic artificial ecosystem-based optimization (CAEO) algorithm [37], chaotic JAYA algorithm [38], and chaotic salp swarm algorithm (CSSA) [39]. In this article, the ESCGBO algorithm is proposed and applied to estimate different PV static and dynamic PV models. The results from all applications are analyzed and evaluated.

The principal contributions of this article are summarized as follows:

- Proposing a developed version of GBO algorithm based on the eagle strategy with the chaotic method to enhance the performance and avoiding the local optima.
- Applying the conventional GBO and ESCGBO as well as other well-known optimization algorithms for parameter estimation of different PV models, such as static and dynamic PV models.
- The results confirm that the proposed ESCGBO has the capability to enhance the performance and increase the effectiveness of the conventional GBO and improve the convergence rate

The remainder of this paper is arranged as follow: Section 2 presents the static and dynamic PV models. ESCGBO is proposed in Section 3. The results and analysis is discussed in Section 4. Section 5 presents the conclusion.

2. PV Static and Dynamic Models

PV models should accurately describe the characteristics of the PV systems for different types of PV models proposed in literature in order to achieve this target. In this section, the most popular static and dynamic models are discussed.

2.1. Static SDM and DDM

SDM has one diode connected with series resistance and parallel resistance to the photo generated current, represented with current source connected parallel with the diode. SDM has five parameters; therefore, it is considered the simplest model. Consider x is a vector of model parameters $x = (x_1, x_2, x_3, x_4, x_5)$ equivalent to $(R_s, R_{sh}, I_{ph}, I_s, \eta)$. The SDM equivalent circuit is shown in Figure 1 and represented by Equations (1) and (2). The DDM is developed to represent the effect of recombination in the PV cell, and this is achieved by adding a second diode to the SDM circuit as shown in Figure 2. The DDM has seven parameters $x = (x_1, x_2, x_3, x_4, x_5, x_6, x_7)$ equivalent to $(R_s, R_{sh}, I_{ph}, I_{s1}, I_{s2}, \eta_1, \eta_2)$. The DDM is described by Equations (3) and (4). The objective function for SDM and DDM is described by Equations (5) and (6) respectively.

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

$$I = I_{ph} - I_s \left[\exp(\frac{q(V + R_s * I)}{\eta * K * T}) - 1 \right] - \frac{(V + R_s * I)}{R_{sh}} \tag{2}$$

$$I = I_{ph} - I_{D1} - I_{D2} - I_{sh} \tag{3}$$

$$I = I_{ph} - I_{s1} \left[\exp(\frac{q(V + R_s * I)}{\eta_1 * K * T}) - 1 \right] - I_{s2} \left[\exp(\frac{q(V + R_s * I)}{\eta_2 * K * T}) - 1 \right] - \frac{(V + R_s * I)}{R_{sh}} \tag{4}$$

$$f_{DD}(V, I, X) = I - X_3 + X_4 \left[\exp(\frac{q(V + X_1 * I)}{X_5 * K * T}) - 1 \right] + \frac{(V + X_1 * I)}{X_2} \tag{5}$$

$$f_{DD}(V, I, X) = I - X_3 + X_4 \left[\exp(\frac{q(V + X_1 * I)}{X_6 * K * T}) - 1 \right] + X_5 \left[\exp(\frac{q(V + X_1 * I)}{X_7 * K * T}) - 1 \right] + \frac{(V + X_1 * I)}{X_2} \tag{6}$$
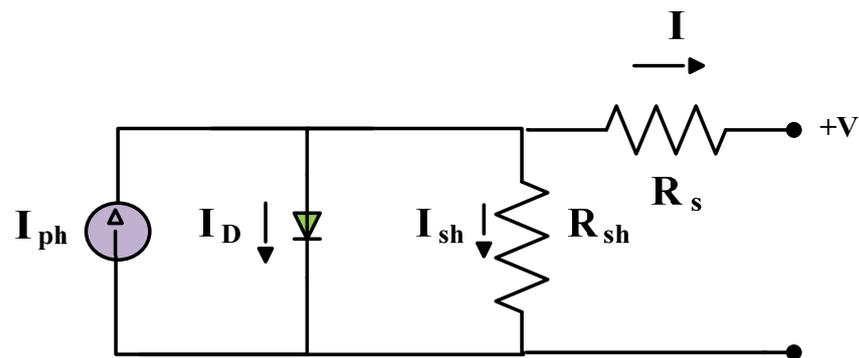
Figure 1. SDM.
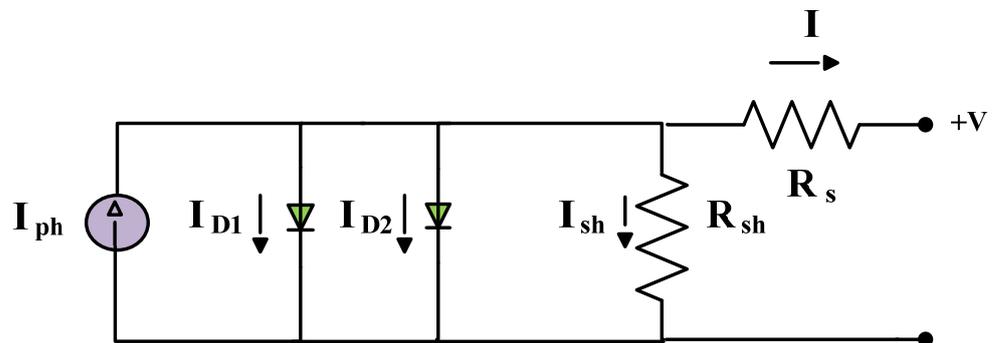
Figure 2. DDM.

2.2. Dynamic PV Model

The integral and fractional dynamic PV models are two popular dynamic models, and are selected to be discussed here.

The integral dynamic model (IOM) is a second order model consists of constant voltage source V_{oc} and series resistance R_S to represent the static model as shown in Figure 3. The dynamic part is represented by capacitor C for junction capacitance and resistance R_c for conductance. The connected cables inductance is represented by the coil inductance. The load is represented by R_L. The total number of unknown parameters are three parameters $(R_c, C$ and $L)$, the IOM is represented by Equations (7) and (8).

$$i_L(s) = \frac{V_{oc}}{s} \frac{a_{11}(s+b_1) + b_2(s-a_{11})}{(s-a_{22})(s-a_{11}) - a_{12}a_{21}} \tag{7}$$

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} = \begin{pmatrix} \frac{-1}{C(R_c+R_s)} & \frac{-R_S}{C(R_c+R_s)} \\ \frac{R_S}{L(R_c+R_s)} & \frac{-[R_L R_c + R_s R_c + R_L R_s]}{L(R_c+R_s)} \end{pmatrix},$$
$$\begin{pmatrix} b_1 \\ b_2 \end{pmatrix} = \begin{pmatrix} \frac{1}{C(R_c+R_s)} \\ \frac{R_c}{L(R_c+R_s)} \end{pmatrix} \tag{8}$$
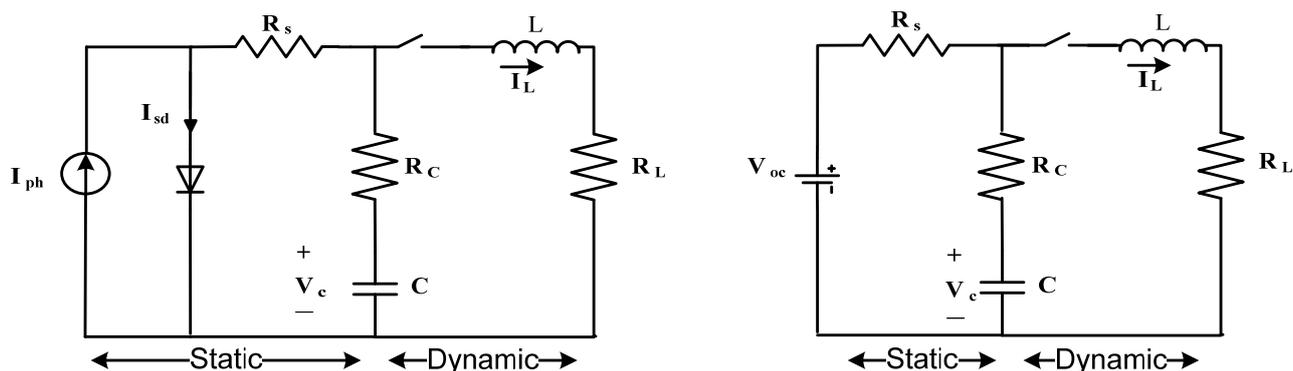
Figure 3. Integral Order Dynamic Model.

The fractional order model (FOM) is developed to represent fractional capacitor in case of low values of R_c due real frequency dependence on fractional capacitance impedance, as shown in Figure 4. The fractional order of capacitance and inductance are represented by α and β, respectively. The total number of FOM is five parameters (R_c, C, L, α and β), the IOM is represented by Equations (9) and (10).

$$i_L(s) = \frac{V_{oc}}{s} \frac{a_{11}(s^\alpha + b_1) + b_2(s^\alpha - a_{11})}{(s^\beta - a_{22})(s^\alpha - a_{11}) - a_{12}a_{21}} \tag{9}$$

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} = \begin{pmatrix} \frac{-1}{C_\alpha(R_c+R_s)} & \frac{-R_S}{C_\alpha(R_c+R_s)} \\ \frac{R_S}{L_\beta(R_c+R_s)} & \frac{-[R_L R_c + R_s R_c + R_L R_s]}{L_\beta(R_c+R_s)} \end{pmatrix},$$

$$\begin{pmatrix} b_1 \\ b_2 \end{pmatrix} = \begin{pmatrix} \frac{1}{C_\alpha(R_c+R_s)} \\ \frac{R_c}{L_\beta(R_c+R_s)} \end{pmatrix} \tag{10}$$
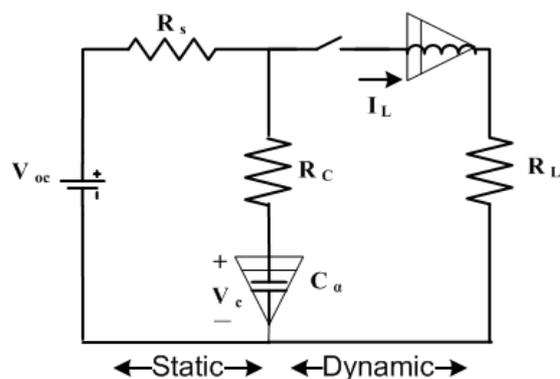
Figure 4. Fractional Order dynamic model.

3. Optimization Methodology

This section describes the basic variant of the gradient-based optimizer (GBO). After, the process of the proposed EGCGBO algorithm is presented.

3.1. Original Gradient-Based Optimizer (GBO) Algorithm

The GBO algorithm is a proposed metaheuristic optimization algorithm, which is presented in reference [30]. This algorithm uses the Newton's gradient-based method to explore the whole search space. This algorithm uses two main machinists, namely gradient search rule (GSR) and local escaping operator (LEO) to achieve the best solution.

3.1.1. GBO Initialization

Like many algorithms, the GBO comprises N vector (members of populations) in the D-dimensional search space. The initial population is randomly generated by the following equation [40]:

$$X_n = X_{min} + rand(0,1) \times (X_{max} - X_{min}) \tag{11}$$

where X_n refers to the nth vector, X_{min}, X_{max} are the limits of the solution space in each problem and $rand(0, 1)$ denotes a random number defined in the range of [0, 1].

3.1.2. Gradient Search Rule (GSR)

In the GBO algorithm, GSR is based on the gradient-based method where the aim of using the GSR is exploration tendency improvement and increasing the convergence rate. Therefore, the new position X_{n+1} is defined as:

$$X_{n+1} = X_n - \frac{2\Delta x \times f(X_n)}{f(X_n + \Delta x) - f(X_n - \Delta x)} \tag{12}$$

The Equation (12) will be adjusted to include the population-based search theory, which is presented by Equation (13).

$$GSR = randn \times \frac{2\Delta x \; X_n}{(x_{worst} - x_{best} + \varepsilon)} \tag{13}$$

where $randn$ is a random number with a normal distribution, x_{worst}, x_{best} denote the worst and best solutions obtained during the process of optimization, ε denotes a small number within the interval [0, 0.1], and Δx refers to the change in position at each iteration. From the previous equations, the GSR can be defined as:

$$GSR = randn \times \rho_1 \times \frac{2\Delta x \; X_n}{(x_{worst} - x_{best} + \varepsilon)} \tag{14}$$

where ρ_1 denotes the randomly generated parameter and it calculated from the following equation:

$$\rho_1 = (2 \times rand \times \alpha) - \alpha \tag{15}$$

$$\alpha = \left| \beta \; \sin\left(\frac{3\pi}{2} + \sin\left(\beta\frac{3\pi}{2}\right)\right) \right| \tag{16}$$

$$\beta = \beta_{min} + (\beta_{min} - \beta_{min})\left(1 - \left(\frac{m}{M}\right)^3\right)^2 \tag{17}$$

where α represents a sine function for the transference from exploration to exploitation, β_{min} and β_{max} are constant values 0.2 and 1.2, respectively, m is the current number of iterations, and M represents the total number of iterations. The change Δx between the best solution x_{best} and a randomly chosen position x_{r1}^m is given by:

$$\Delta x = rand(1 : N) \times |step| \tag{18}$$

$$step = \frac{(x_{best} - x_{r1}^m) + \delta}{2} \tag{19}$$

$$\delta = 2 \times rand \times \left(\left|\frac{x_{r1}^m + x_{r2}^m + x_{r3}^m + x_{r4}^m}{4}\right| - x_n^m\right) \tag{20}$$

where $rand(1 : N)$ denotes a random vector with N dimensions, $r1, r2, r3,$ and $r4$ $(r1 \neq r2 \neq r3 \neq r4 \neq n)$ are different integers randomly chosen from [1, N], $step$ denotes a step size. The new position X_{n+1} represents an updated version based on the GSR as follows:

$$X_{n+1} = X_n - GSR \tag{21}$$

The direction of movement (DM) is added for better exploitation of the nearby area of X_n which can be calculated as:

$$DM = rand \times \rho_2 \times (x_{best} - x_n) \tag{22}$$

$$\rho_2 = (2 \times rand \times \alpha) - \alpha \tag{23}$$

Therefore, the new position $X1_n^m$ is calculated from the following equation after taking the GSR and DM into consideration:

$$X1_n^m = x_n^m - \text{GSR} + \text{DM} \tag{24}$$

$$X1_n^m = x_n^m - randn \times \rho_1 \times \frac{2\Delta x \times x_n^m}{(x_{\text{worst}} - x_{best} + \varepsilon)} + rand \times \rho_2 \times (x_{best} - x_n^m) \tag{25}$$

The GBO used another position to increase the local search by putting the best-so-far solution (x_{best}) rather than the position x_n^m. The new position ($X2_n^m$) is calculated as follows:

$$X2_n^m = x_{best} - randn \times \rho_1 \times \frac{2\Delta x \times x_n^m}{(yp_n^m - yq_n^m + \varepsilon)} + rand \times \rho_2 \times (x_{r1}^m - x_{r2}^m) \tag{26}$$

where

$$yp_n = rand \times (\frac{[z_{n+1} + x_n]}{2} + rand \times \Delta x) \tag{27}$$

$$yq_n = rand \times (\frac{[z_{n+1} + x_n]}{2} - rand \times \Delta x) \tag{28}$$

According to the positions $X1_n^m$, $X2_n^m$, and the current position (X_n^m), the new position at the next iteration (x_n^{m+1}) is defined as:

$$x_n^{m+1} = r_a \times (r_b \times X1_n^m + (1 - r_b) \times X2_n^m) + (1 - r_a) \times X3_n^m \tag{29}$$

$$X3_n^m = X_n^m - \rho_1 \times (X2_n^m - X1_n^m) \tag{30}$$

3.1.3. Local Escaping Operator (LEO)

The LEO is applied to enhance the performance of the GBO algorithm and to escape the local solutions for solving the complicated problems. The LEO generates an appropriate solution (X_{LEO}^m) by using several solutions, which include x_{best}, the solutions $X1_n^m$, and $X2_n^m$, two random solutions x_{r1}^m and x_{r2}^m, and a new randomly generated solution (x_k^m). The solution X_{LEO}^m is formulated as:

$if\ rand < pr$
$if\ rand < 0.5$
$\quad X_{LEO}^m = \ X_n^{m+1} + f_1 \times (u_1 \times x_{\text{best}} - u_2 \times x_k^m) + f_2 \times \rho_1 \times (u_3 \times (X2_n^m - X1_n^m)$
$\qquad\qquad + u_2 \times (x_{r1}^m - x_{r2}^m))/2$
$X_n^{m+1} = X_{LEO}^m$
Else
$\quad X_{LEO}^m = x_{\text{best}} + f_1 \times (u_1 \times x_{\text{best}} - u_2 \times x_k^m) + f_2 \times \rho_1 \times (u_3 \times (X2_n^m - X1_n^m)$ $\tag{31}$
$\qquad\qquad\quad + u_2 \times (x_{r1}^m - x_{r2}^m))/2$
$X_n^{m+1} = X_{LEO}^m$
End
End

where f_1 denotes a uniform distributed random number in the range of $[-1, 1]$, f_2 is a random number from a normal distribution with a mean of 0 and a standard deviation of 1, pr is the probability, and u_1, u_2, and u_3 are random values generated as follows:

$$u_1 = \begin{cases} 2 \times rand & if\ \mu_1 < 0.5 \\ 1 & otherwise \end{cases} \tag{32}$$

$$u_2 = \begin{cases} rand & if \ \mu_1 < 0.5 \\ 1 & otherwise \end{cases} \tag{33}$$

$$u_3 = \begin{cases} rand & if \ \mu_1 < 0.5 \\ 1 & otherwise \end{cases} \tag{34}$$

where *rand* represents a random number in the range of [0, 1], and μ_1 refers to a number in the range of [0, 1]. The above equations are simply explained as follows:

$$u_1 = L_1 \times 2 \times rand + (1 - L_1) \tag{35}$$

$$u_2 = L_1 \times rand + (1 - L_1) \tag{36}$$

$$u_3 = L_1 \times rand + (1 - L_1) \tag{37}$$

where L_1 denotes a binary parameter with a value of 0 or 1. If parameter $\mu_1 < 0.5$, the value of $L_1 = 1$, otherwise, $L_1 = 0$. The solution x_k^m is generated as follows:

$$x_k^m = \begin{cases} x_{rand} & if \ \mu_2 < 0.5 \\ x_p^m & otherwise \end{cases} \tag{38}$$

$$x_{rand} = X_{min} + rand(0,1) \times (X_{max} - X_{min}) \tag{39}$$

where x_{rand} is a random generated solution, x_p^m is a randomly chosen solution of the population ($p \in [1, 2, \ldots, N]$), and μ_2 represents a random number in the range of [0, 1]. Equation (38) is simplified as:

$$x_k^m = L_2 \times x_p^m + (1 - L_2) \times x_{rand} \tag{40}$$

where L_2 denotes a binary parameter with a value of 0 or 1. If $\mu_2 < 0.5$, the value of $L_2 = 1$, otherwise, $L_2 = 0$.

3.2. Eagle Strategy and Chaotic with Gradient-Based Optimizer (ESCGBO) Algorithm

3.2.1. Eagle Strategy

The eagle strategy is proposed for solving real-world optimization problems that is developed by Yang et al. [35]. It is inspired by the foraging behavior of eagles that fly randomly in analogy to the Levy flights. It is the two-stage method: global search randomization stage and an intensive local search [31]. The first stage aims mainly to investigate the search space globally and rapidly obtain a promising solution, while the target of the second stage is to obtain the optimal solution through making an intensive local search based on the achieved solution in the first stage. The benefit of this strategy is that there is no limit to the kinds of techniques or algorithms used in each stage. Any technique that is able to achieve better results in a flexible way could be used in any stage [31].

During the iteration of the proposed technique, the new candidate solution is generated by Levy flight as follows:

$$X_{n+1} = X_n - \gamma(X_n - X_{best}) \bigoplus Levy(\lambda) = X_n + \frac{0.01u}{|v|^{1/\lambda}}(X_n - X_{best}) \tag{41}$$

where γ is the step scaling size, the $\bigoplus$ refers to the process of element-wise multiplications, λ is the Levy flight exponent, while u and v can be expressed as:

$$u \sim N\left(0, \sigma_u^2\right) \ , \ v \sim N\left(0, \sigma_v^2\right) \tag{42}$$

The standard deviations σ_u and σ_v are explained as:

$$\sigma_u = \left[\frac{\sin\left(\frac{\lambda\pi}{2}\right) \cdot \Gamma(1+\lambda)}{2^{(\lambda-1)}\lambda \cdot \Gamma\left(\frac{1+\lambda}{2}\right)} \right]^{1/\lambda}, \ \sigma_v = 1 \tag{43}$$

where Γ is the Gamma function.

3.2.2. Chaotic Maps

Chaotic systems are deterministic systems that present unpredictable conduct, whose action is complex and similar to that of randomness [41]. In [41], chaos-based exploration rate to enhance the performance of three well-known optimization algorithms was proposed. Based on this proposal, the real random number (v) in Equation (41) was replaced by a chaotic number in the eagle strategy. This modification makes the value of v linearly decreased from 2 to 0 throughout the course of iterations. The steps of the proposed ESCGBO are presented in the flowchart illustrated in Figure 5.
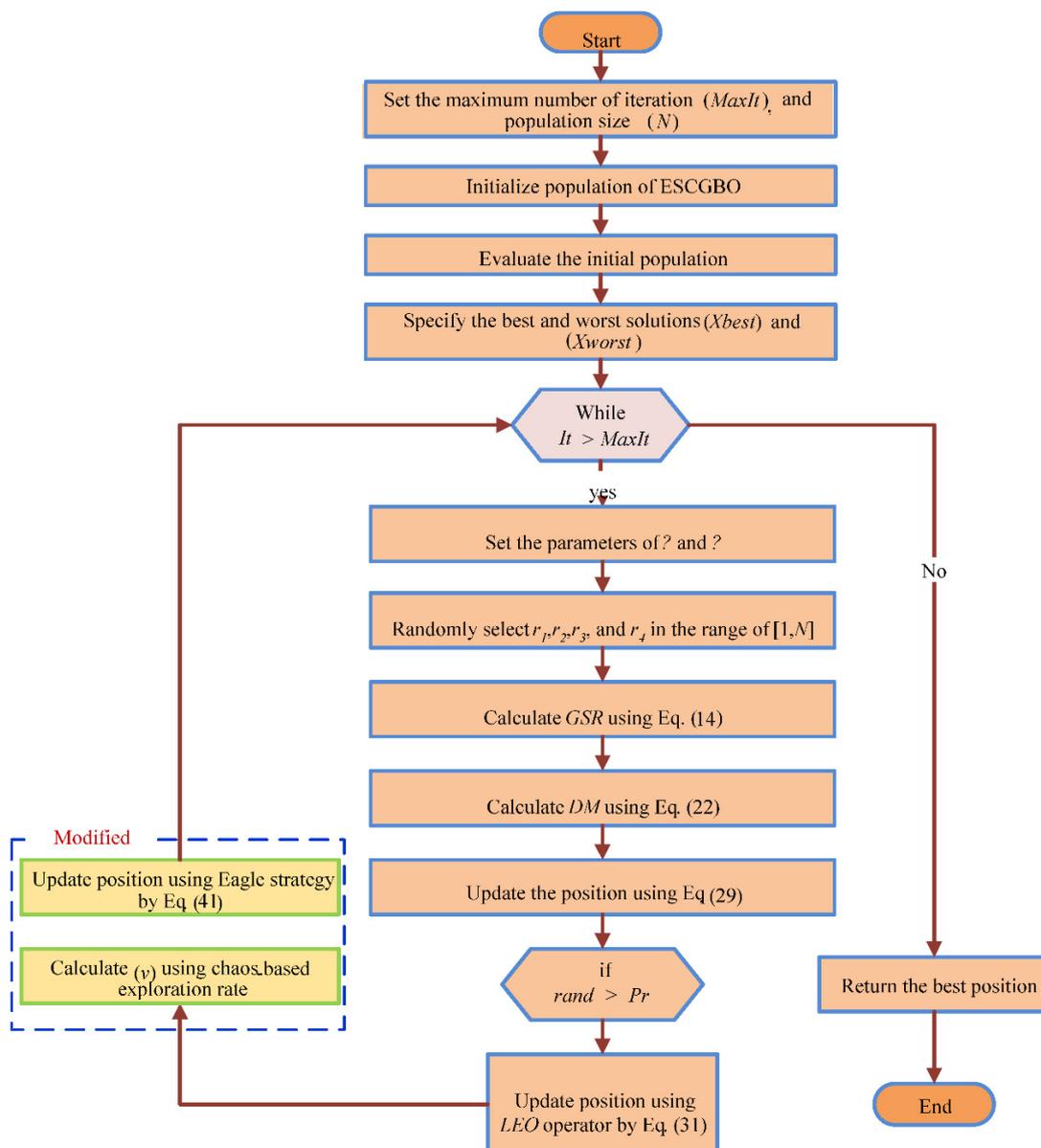
Figure 5. Flowchart of the proposed ESCGBO algorithm.

4. Results

4.1. Performance of the Proposed ESCGBO Algorithm

In this subsection, the effective and performance of the proposed ESCGBO technique were evaluated on several benchmark functions, including statistical measurements, such as best values, mean values, worst values, and standard deviation (STD) for the solutions obtained by the ESCGBO technique, the original GBO algorithm as well as the two recent optimization algorithms, including the equilibrium optimizer (EO) algorithm [42] and wild horse optimizer (WHO) algorithm [43]. The results obtained with the proposed ESCGBO algorithm were compared with these well-known optimization algorithms. All mentioned techniques were executed for the maximum number of the iterations' function of 500 and population size of 50 for 20 independent runs using Matlab R2016a'working on Windows 8.1, 64bit. All computations were performed on a Core i5-4210U CPU@ 2.40 GHz of speed, and 8 GB of RAM. Figure 6 shows the qualitative metrics on F1, F3, F4, F5, F8, F12, F15, and F18, including 2D views of the functions, search history, average fitness history, and convergence characteristics curve using the proposed ESCGBO algorithm.
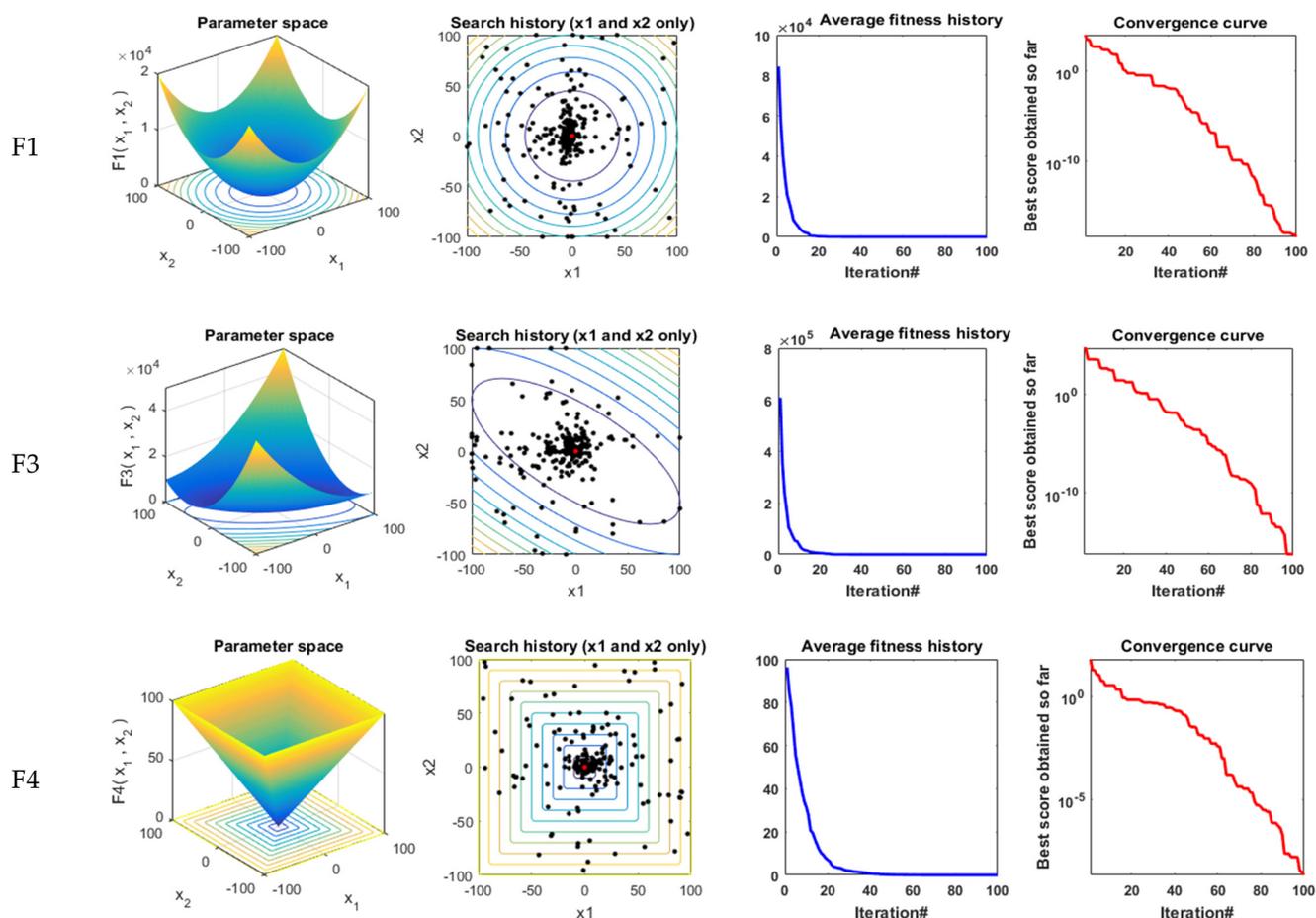
Figure 6. *Cont.*
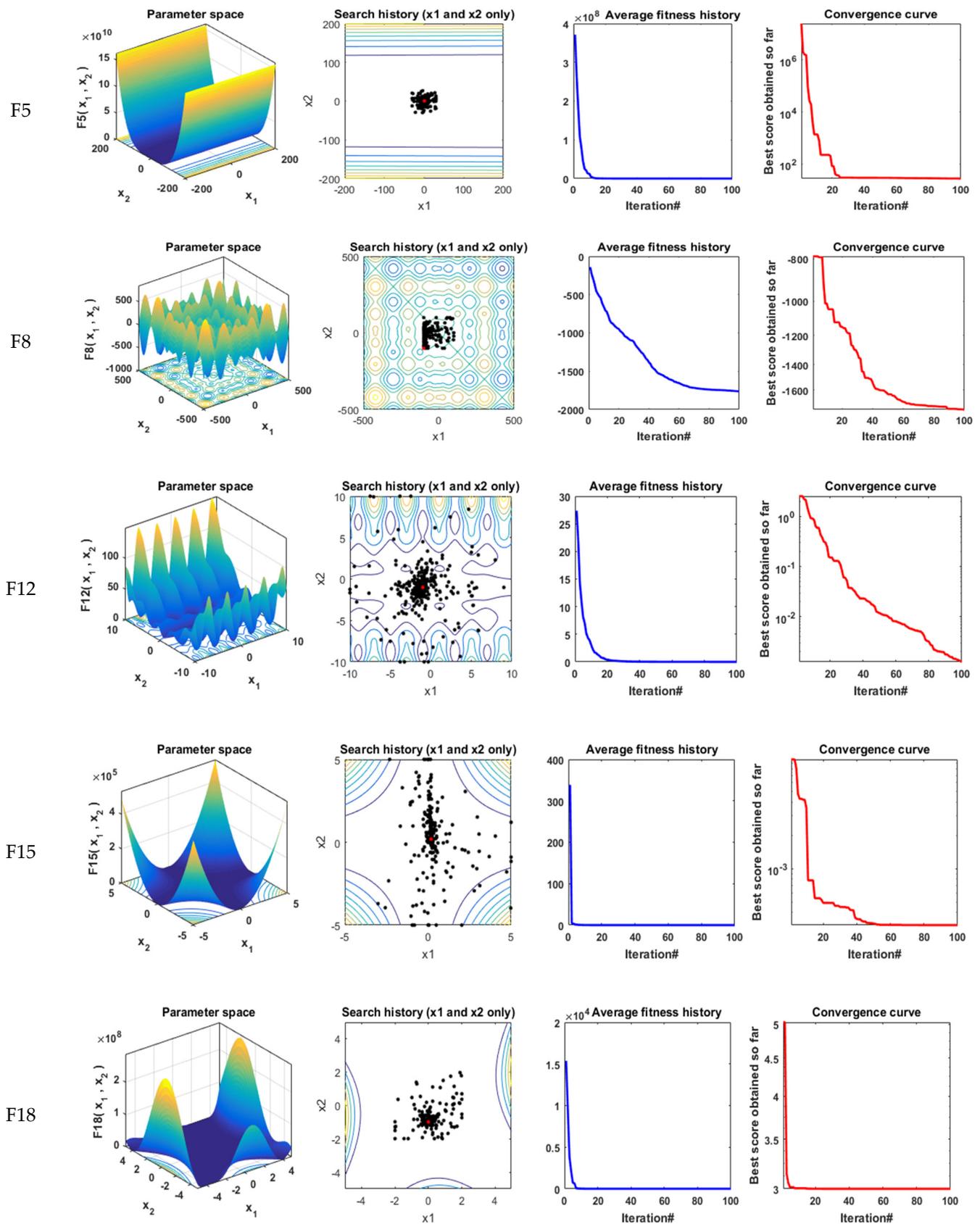
Figure 6. Qualitative metrics on F1, F3, F4, F5, F8, F12, F15, and F18: 2D views of the functions, search history, average fitness history, and convergence curve using ESCGBO algorithm.

Tables [1]–[3] present the statistical results of the proposed ESCGBO algorithm and other recent techniques when applied for unimodal benchmark functions, multimodal benchmark functions, and composite benchmark functions, respectively. The best values were obtained with the proposed ESCGBO, GBO, WHO, and EO algorithms shown in bold. It is clearly seen that the proposed ESCGBO technique achieves the optimal solution for most of the benchmark functions. The convergence curves of these algorithms for the unimodal benchmark functions are shown in Figure [7] and the boxplots for each algorithm for this type of function are presented in Figure [8]. Additionally, Figure [9] displays convergence curves of all algorithms for the multimodal benchmark functions, named F8 to F13 and Figure [10] shows the boxplots for each algorithm for these functions. Furthermore, the convergence curves of these algorithms for the composite benchmark functions are presented in Figure [11] while Figure [12] illustrates the boxplots for each algorithm for this type of benchmark function. From those figures, it is obvious that the proposed algorithm reached a stable point for all functions and the boxplots of the proposed algorithm are narrow for most functions compared to the other algorithms. Table [4] shows the values of the average CPU time of different algorithms on the 23 benchmark functions.

Table 1. The statistical results of unimodal benchmark functions using the proposed technique and other well-known algorithms.

Function		ESCGBO	GBO	WHO	EO
F1	Best	2.20×10^{-136}	$\mathbf{1.44 \times 10^{-136}}$	2.83×10^{-57}	3.89×10^{-51}
	Worst	$\mathbf{1.14 \times 10^{-129}}$	6.32×10^{-128}	3.53×10^{-47}	6.09×10^{-47}
	Mean	$\mathbf{1.33 \times 10^{-130}}$	7.41×10^{-129}	1.78×10^{-48}	4.11×10^{-48}
	std	$\mathbf{3.38 \times 10^{-130}}$	1.86×10^{-128}	7.90×10^{-48}	1.36×10^{-47}
F2	Best	$\mathbf{2.31 \times 10^{-71}}$	2.89×10^{-70}	8.31×10^{-33}	5.47×10^{-29}
	Worst	2.33×10^{-65}	$\mathbf{1.84 \times 10^{-66}}$	9.82×10^{-29}	1.3×10^{-27}
	Mean	1.39×10^{-66}	$\mathbf{2.39 \times 10^{-67}}$	1.39×10^{-29}	3.65×10^{-28}
	std	5.2×10^{-66}	$\mathbf{4.92 \times 10^{-67}}$	2.48×10^{-29}	3.47×10^{-28}
F3	Best	$\mathbf{8.7 \times 10^{-116}}$	1.6×10^{-115}	6.42×10^{-36}	5.41×10^{-17}
	Worst	$\mathbf{6.7 \times 10^{-104}}$	3.9×10^{-101}	6.83×10^{-28}	1.39×10^{-11}
	Mean	$\mathbf{3.4 \times 10^{-105}}$	1.9×10^{-102}	5.22×10^{-29}	2.34×10^{-12}
	std	$\mathbf{1.5 \times 10^{-104}}$	8.7×10^{-102}	1.66×10^{-28}	4.08×10^{-12}
F4	Best	$\mathbf{1.58 \times 10^{-64}}$	8.91×10^{-64}	5.07×10^{-22}	4×10^{-14}
	Worst	3.03×10^{-58}	$\mathbf{1.63 \times 10^{-59}}$	3.71×10^{-19}	6.87×10^{-12}
	Mean	1.72×10^{-59}	$\mathbf{1.6 \times 10^{-60}}$	4.56×10^{-20}	1.2×10^{-12}
	std	6.75×10^{-59}	$\mathbf{3.65 \times 10^{-60}}$	9.74×10^{-20}	1.77×10^{-12}
F5	Best	19.99698	**18.52388**	23.60955	24.44135
	Worst	25.02491	**24.02389**	86.19444	25.0421
	Mean	21.68888	**21.22658**	34.30228	24.81639
	std	1.324568	**1.169705**	20.97713	0.201379
F6	Best	$\mathbf{1.14 \times 10^{-9}}$	5.23×10^{-9}	3.67×10^{-8}	1.39×10^{-8}
	Worst	$\mathbf{1.61 \times 10^{-7}}$	1.07×10^{-6}	2.02×10^{-5}	3.8×10^{-7}
	Mean	$\mathbf{3.35 \times 10^{-8}}$	7.89×10^{-8}	1.42×10^{-6}	9.07×10^{-8}
	std	$\mathbf{3.91 \times 10^{-8}}$	2.35×10^{-7}	4.46×10^{-6}	9.27×10^{-8}
F7	Best	0.000176	0.000111	$\mathbf{4.5 \times 10^{-5}}$	9.73×10^{-5}
	Worst	0.001732	0.001663	0.00184	**0.001432**
	Mean	0.000578	**0.000528**	0.000857	0.00068
	std	0.000427	0.000392	0.000514	**0.000358**

The best values obtained are in bold.

Table 2. The statistical results of multimodal benchmark functions using the proposed technique and other well-known algorithms.

Function		ESCGBO	GBO	WHO	EO
F8	Best	-1881.33	$\mathbf{-1909.05}$	-1789.02	-1798.26
	Worst	-1668.99	-1659.76	-1600.49	$\mathbf{-1715.16}$
	Mean	-1733.03	$\mathbf{-1771.42}$	-1705.23	-1751.04
	Std	54.18392	83.01581	49.62292	$\mathbf{21.99425}$
F9	Best	**0.00**	**0.00**	**0.00**	**0.00**
	Worst	**0.00**	**0.00**	**0.00**	**0.00**
	Mean	**0.00**	**0.00**	**0.00**	**0.00**
	Std	**0.00**	**0.00**	**0.00**	**0.00**
F10	Best	$\mathbf{8.88 \times 10^{-16}}$	$\mathbf{8.88 \times 10^{-16}}$	$\mathbf{8.88 \times 10^{-16}}$	20
	Worst	$\mathbf{8.88 \times 10^{-16}}$	$\mathbf{8.88 \times 10^{-16}}$	20.00111	20
	Mean	$\mathbf{8.88 \times 10^{-16}}$	$\mathbf{8.88 \times 10^{-16}}$	1.005259	20
	Std	**0.00**	**0.00**	4.471221	1.23×10^{-10}
F11	Best	**0.00**	**0.00**	**0.00**	**0.00**
	Worst	**0.00**	**0.00**	**0.00**	**0.00**
	Mean	**0.00**	**0.00**	**0.00**	**0.00**
	Std	**0.00**	**0.00**	**0.00**	**0.00**
F12	Best	$\mathbf{9.51 \times 10^{-12}}$	1.45×10^{-10}	1.18×10^{-11}	1.42×10^{-10}
	Worst	$\mathbf{1.62 \times 10^{-7}}$	0.103669	0.103669	1.98×10^{-7}
	Mean	$\mathbf{9.05 \times 10^{-9}}$	0.005183	0.01555	1.25×10^{-8}
	Std	$\mathbf{3.61 \times 10^{-8}}$	0.023181	0.037979	4.37×10^{-8}
F13	Best	$\mathbf{9.97 \times 10^{-9}}$	2.29×10^{-8}	7.36×10^{-8}	1.63×10^{-8}
	Worst	**0.054779**	**0.054779**	0.397801	0.108359
	Mean	**0.01043**	0.016351	0.035814	0.018527
	Std	**0.014444**	0.019872	0.093881	0.037211

The best values obtained are in bold.

Table 3. The statistical Results of composite benchmark functions using the proposed technique and other well-known algorithms.

Function		ESCGBO	GBO	WHO	EO
F14	Best	**0.998004**	**0.998004**	**0.998004**	**0.998004**
	Worst	**0.998004**	**0.998004**	3.96825	**0.998004**
	Mean	**0.998004**	**0.998004**	1.543534	**0.998004**
	Std	**0.00**	**0.00**	0.936299	1.76×10^{-16}
F15	Best	**0.000307**	**0.000307**	**0.000307**	**0.000307**
	Worst	0.001594	**0.001223**	0.020363	0.020363
	Mean	0.000528	**0.000445**	0.001676	0.002359
	Std	0.00046	**0.000335**	0.004424	0.006161
F16	Best	$\mathbf{-1.03163}$	$\mathbf{-1.03163}$	$\mathbf{-1.03163}$	$\mathbf{-1.03163}$
	Worst	$\mathbf{-1.03163}$	$\mathbf{-1.03163}$	$\mathbf{-1.03163}$	$\mathbf{-1.03163}$
	Mean	$\mathbf{-1.03163}$	$\mathbf{-1.03163}$	$\mathbf{-1.03163}$	$\mathbf{-1.03163}$
	Std	2.28×10^{-16}	2.28×10^{-16}	$\mathbf{1.35 \times 10^{-16}}$	2.1×10^{-16}
F17	Best	**0.397887**	**0.397887**	**0.397887**	**0.397887**
	Worst	**0.397887**	**0.397887**	**0.397887**	**0.397887**
	Mean	**0.397887**	**0.397887**	**0.397887**	**0.397887**
	Std	**0.00**	**0.00**	**0.00**	**0.00**
F18	Best	**3.00**	**3.00**	**3.00**	**3.00**
	Worst	**3.00**	**3.00**	**3.00**	**3.00**
	Mean	**3.00**	**3.00**	**3.00**	**3.00**
	Std	8.7×10^{-16}	$\mathbf{4.2 \times 10^{-16}}$	5.85×10^{-16}	6.76×10^{-16}
F19	Best	$\mathbf{-0.30048}$	$\mathbf{-0.30048}$	$\mathbf{-0.30048}$	$\mathbf{-0.30048}$
	Worst	$\mathbf{-0.30048}$	$\mathbf{-0.30048}$	$\mathbf{-0.30048}$	$\mathbf{-0.30048}$
	Mean	$\mathbf{-0.30048}$	$\mathbf{-0.30048}$	$\mathbf{-0.30048}$	$\mathbf{-0.30048}$
	Std	$\mathbf{1.14 \times 10^{-16}}$	$\mathbf{1.14 \times 10^{-16}}$	$\mathbf{1.14 \times 10^{-16}}$	$\mathbf{1.14 \times 10^{-16}}$

Table 3. *Cont.*

Function		ESCGBO	GBO	WHO	EO
F20	Best	**−3.322**	**−3.322**	**−3.322**	**−3.322**
	Worst	**−3.2031**	**−3.2031**	−3.08668	−1.84092
	Mean	**−3.28633**	**−3.28633**	−3.31023	−3.12852
	Std	0.055899	0.055899	**0.052619**	0.370083
F21	Best	**−10.1532**	**−10.1532**	**−10.1532**	**−10.1532**
	Worst	**−5.0552**	**−5.0552**	−2.68286	**−5.0552**
	Mean	−8.8787	−8.3689	**−9.52706**	−8.88098
	Std	2.264846	2.494761	**1.966723**	2.260817
F22	Best	**−10.4029**	**−10.4029**	**−10.4029**	**−10.4029**
	Worst	−2.7659	**−5.08767**	−2.75193	−2.7659
	Mean	−7.82681	−8.5426	−8.96998	**−9.75532**
	Std	2.973254	2.601082	2.948831	**2.02859**
F23	Best	**−10.5364**	**−10.5364**	**−10.5364**	**−10.5364**
	Worst	**−5.12848**	−2.80663	−2.87114	−2.42173
	Mean	−9.18443	−8.25715	−9.148	**−10.1307**
	Std	2.402536	2.907055	2.855393	**1.814497**

The best values obtained are in bold.

(**a**) F1   (**b**) F2   (**c**) F3   (**d**) F4

(**e**) F5   (**f**) F6   (**g**) F7
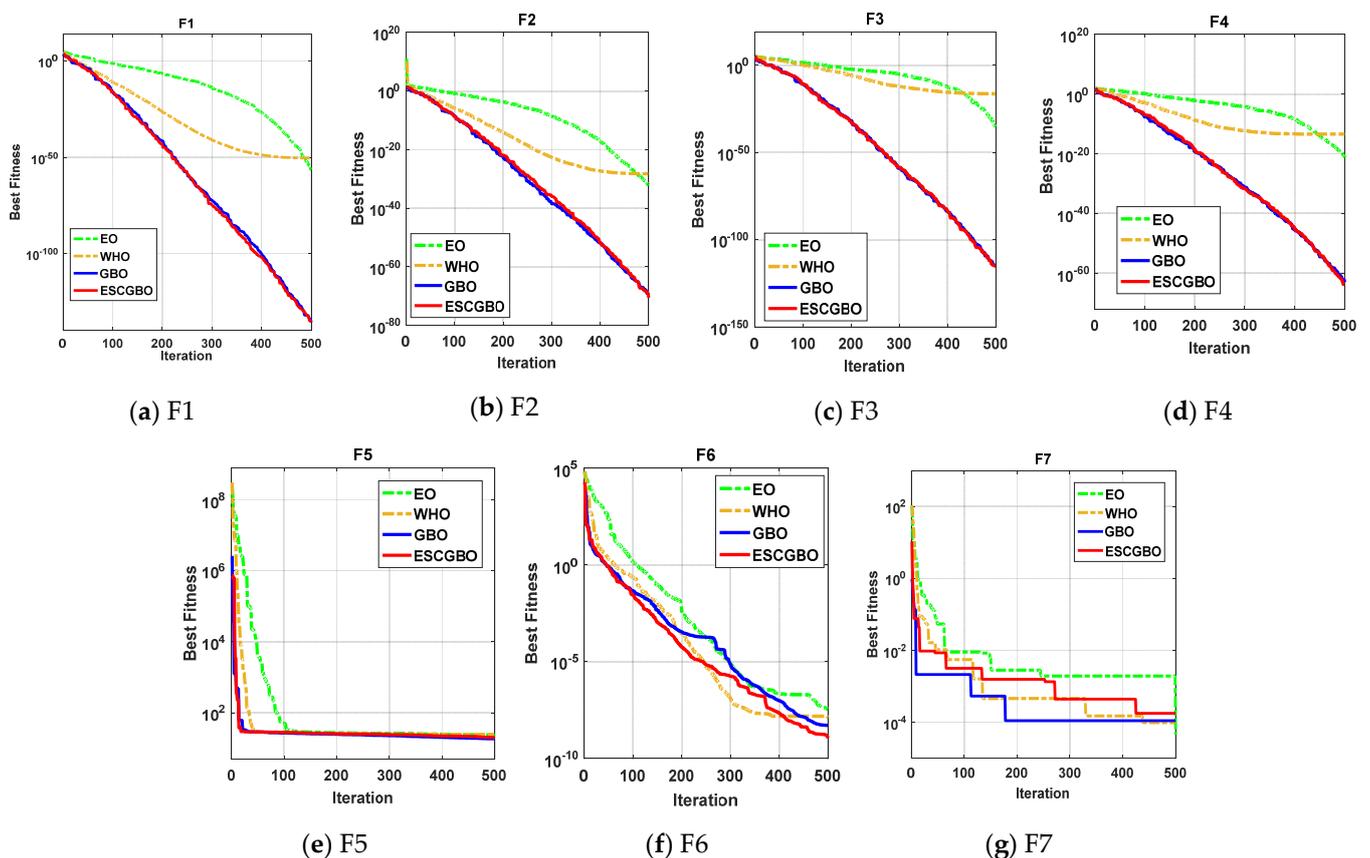
Figure 7. The convergence curves of all algorithms for unimodal benchmark functions (**a**) F1, (**b**) F2, (**c**) F3, (**d**) F4, (**e**) F5, (**f**) F6, and (**g**) F7.

(**a**) F1   (**b**) F2   (**c**) F3   (**d**) F4

(**e**) F5   (**f**) F6   (**g**) F7
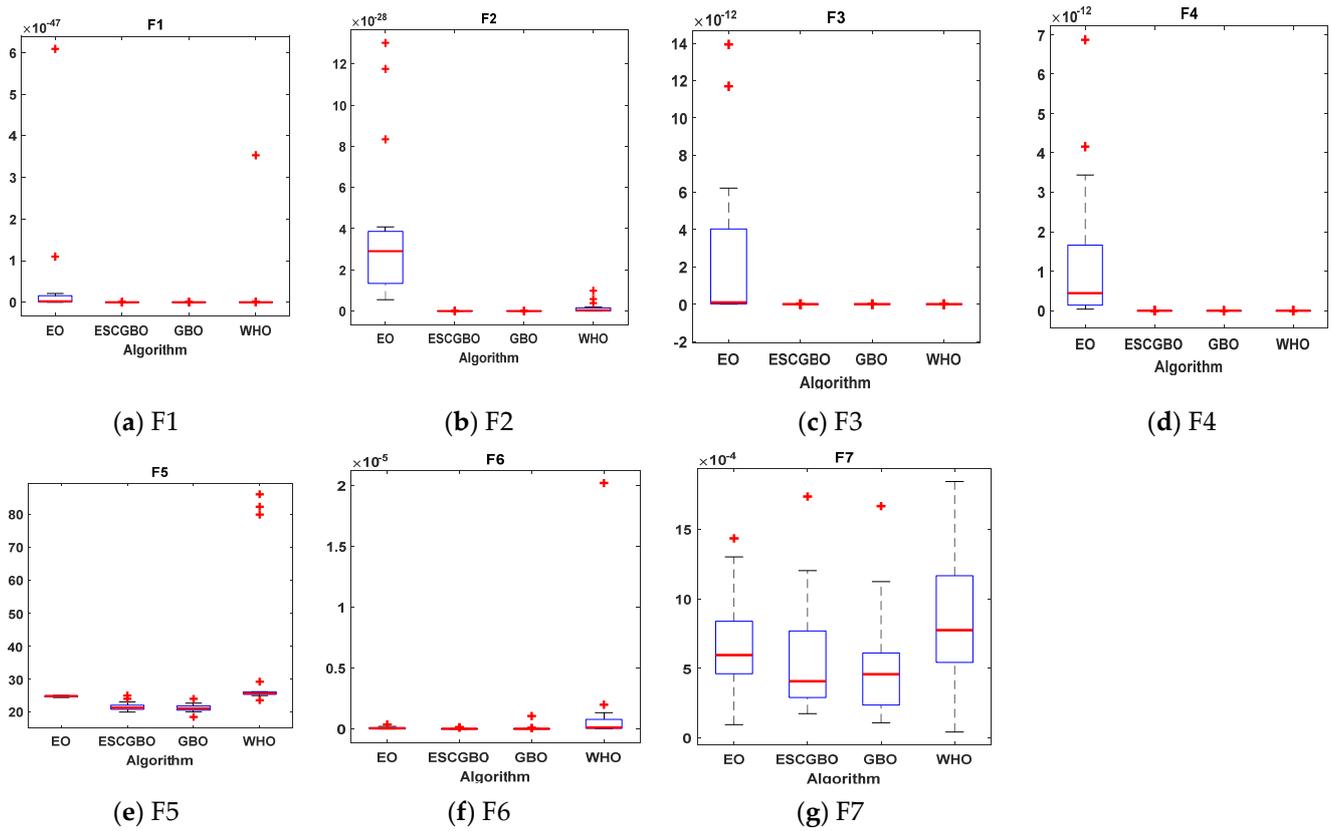
Figure 8. Boxplots for all algorithms for unimodal benchmark functions (**a**) F1, (**b**) F2, (**c**) F3, (**d**) F4, (**e**) F5, (**f**) F6, and (**g**) F7.

(**a**) F8   (**b**) F9   (**c**) F10
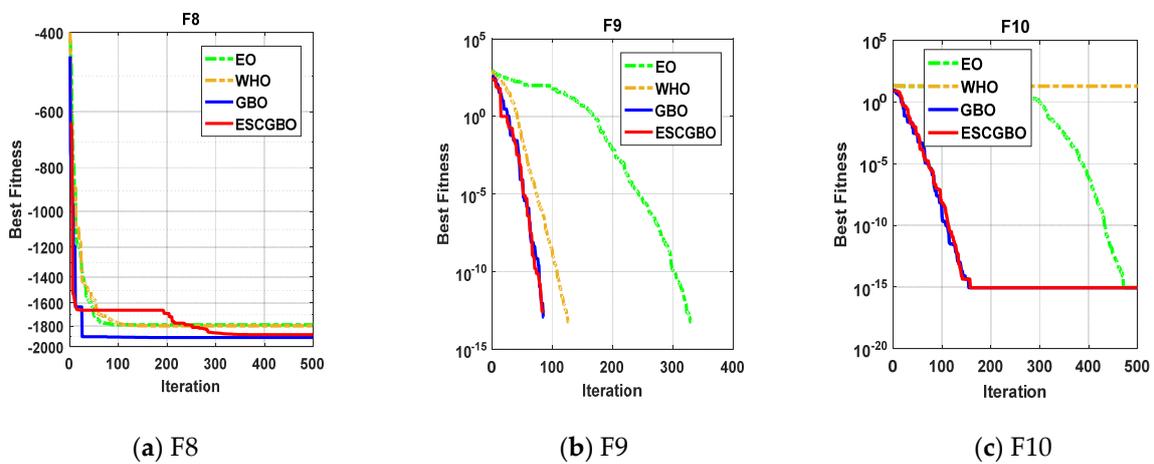
Figure 9. *Cont.*
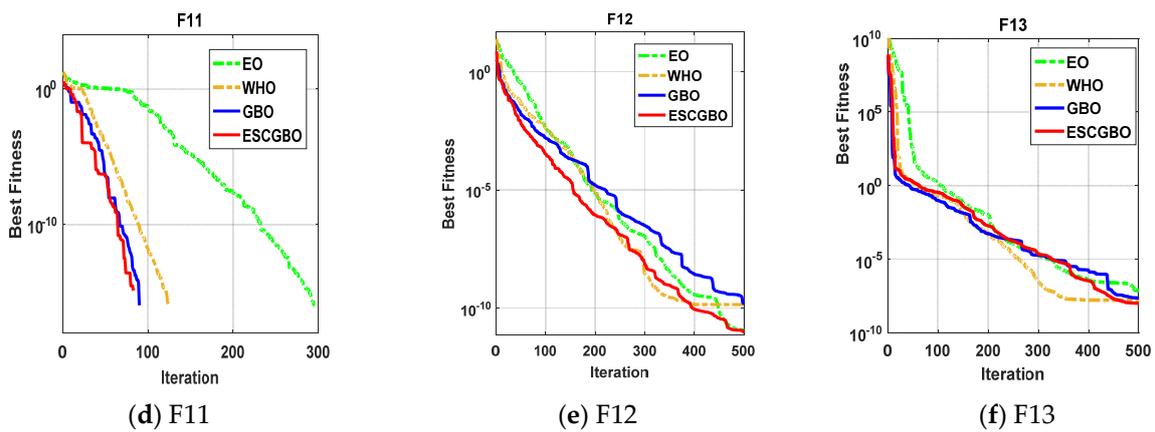
(**d**) F11   (**e**) F12   (**f**) F13

Figure 9. The convergence curves of all algorithms for multi-modal benchmark functions (**a**) F8, (**b**) F9, (**c**) F10, (**d**) F11, (**e**) F12, and (**f**) F13.
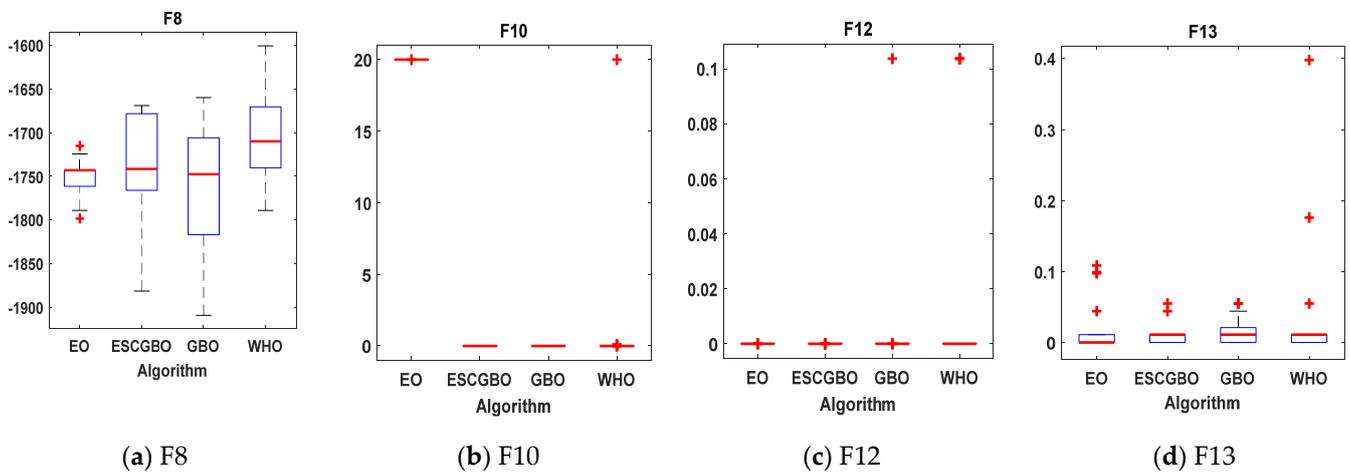
(**a**) F8   (**b**) F10   (**c**) F12   (**d**) F13

Figure 10. Boxplots for all algorithms for some of multi-modal benchmark functions (**a**) F8, (**b**) F10, (**c**) F12, and (**d**) F13.


(**a**) F14   (**b**) F15   (**c**) F16   (**d**) F17

Figure 11. *Cont*.

(**e**) F18　　　　　　(**f**) F19　　　　　　(**g**) F20　　　　　　(**h**) F21

(**i**) F22　　　　　　(**j**) F23
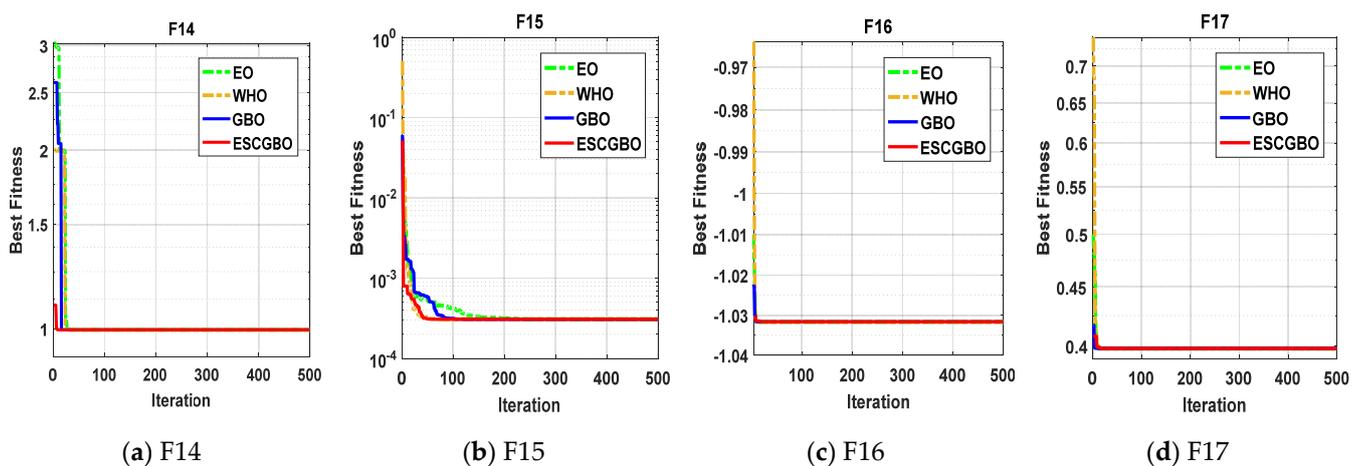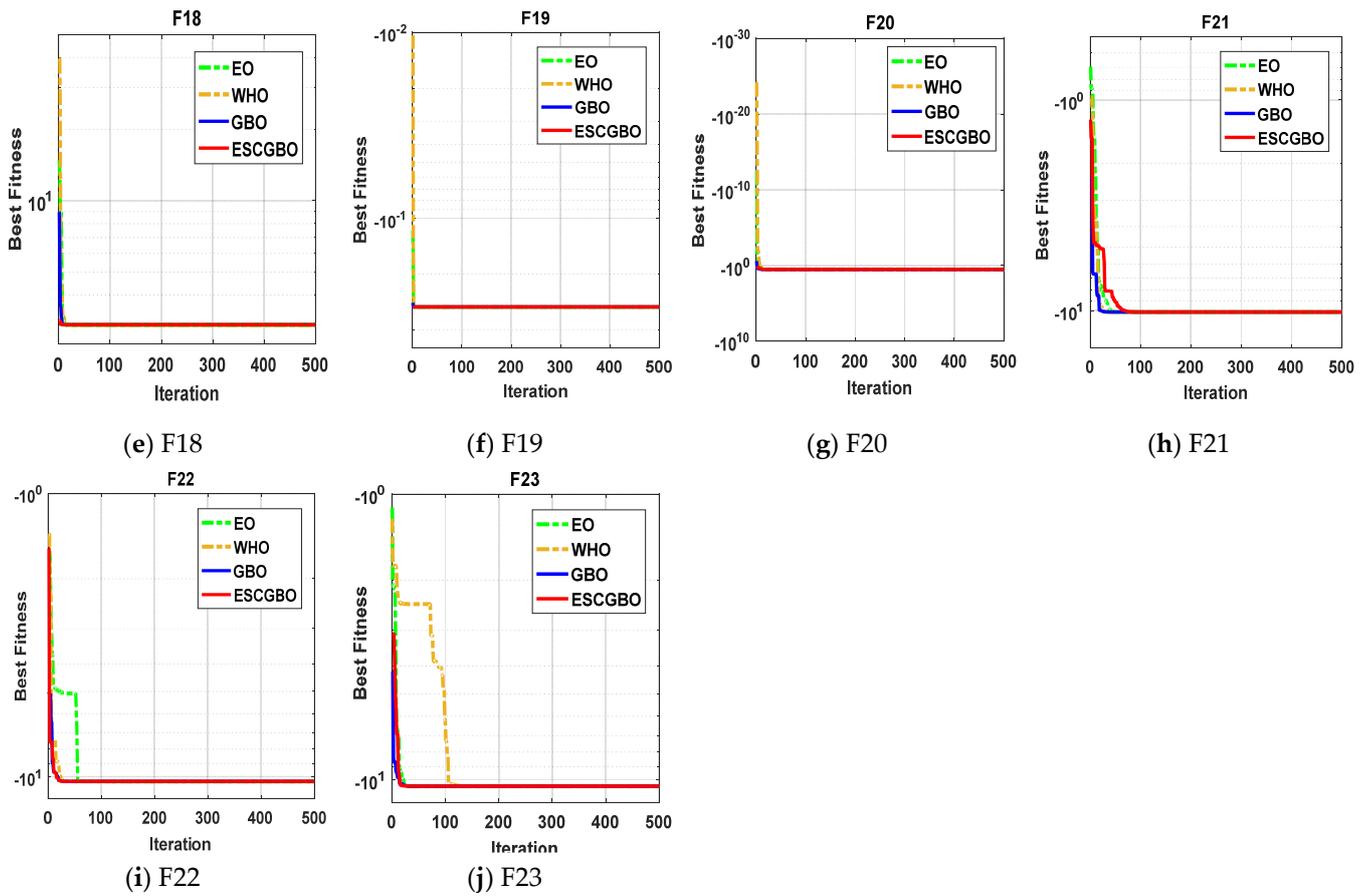
Figure 11. The convergence curves of all algorithms for composite benchmark functions (**a**) F14, (**b**) F15, (**c**) F16, (**d**) F17, (**e**) F18, (**f**) F19, (**g**) F20, (**h**) F21, (**i**) F22, and (**j**) F23.

(**a**) F14　　　　　　(**b**) F15　　　　　　(**c**) F20
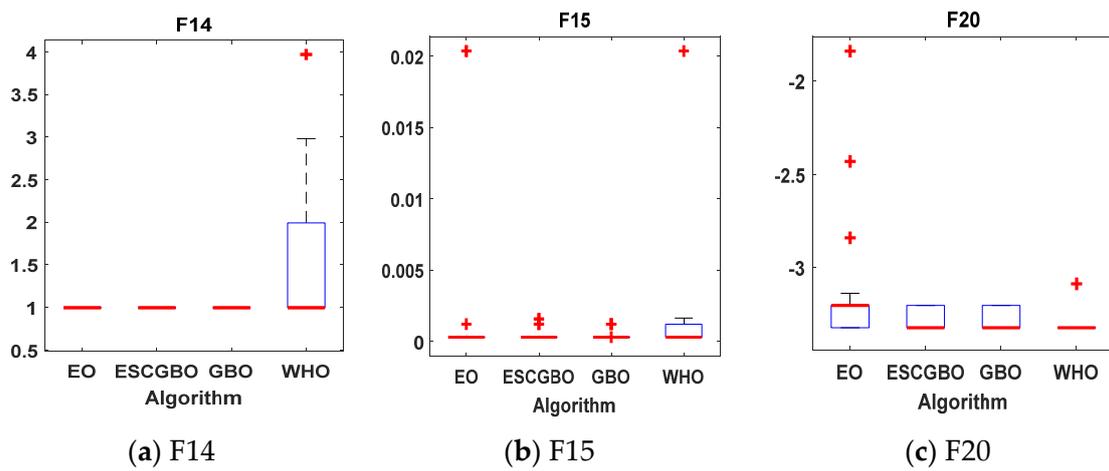
Figure 12. *Cont*.
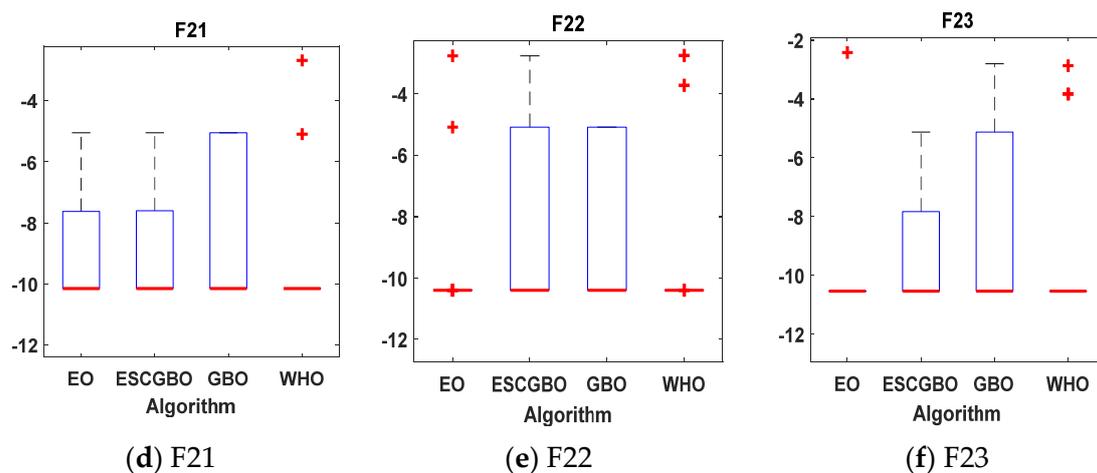
(**d**) F21   (**e**) F22   (**f**) F23

Figure 12. Boxplots for all algorithms for some of composite benchmark functions (**a**) F14, (**b**) F15, (**c**) F20, (**d**) F21, (**e**) F22, and (**f**) F23.

Table 4. CPU time (s) of four algorithms on 23 benchmark functions.

	ESCGBO	**GBO**	**WHO**	**EO**
F1	2.24173	2.045234	1.666734	0.340013
F2	1.831693	1.549915	1.43607	0.3331
F3	2.615374	1.913063	1.671268	0.365052
F4	1.945816	1.773811	2.09404	0.35823
F5	2.032641	2.028365	1.704013	0.414698
F6	1.88023	1.661728	1.679613	0.326943
F7	2.077834	1.637886	1.454167	0.371617
F8	1.795602	1.540629	1.430021	0.341869
F9	1.842514	1.602524	1.657841	0.3381
F10	1.761122	1.503307	1.394121	0.35016
F11	1.780045	1.608481	1.64313	0.416369
F12	1.746382	1.505388	1.366705	0.33033
F13	1.99615	1.835235	1.43058	0.352283
F14	2.417839	1.952414	1.608765	0.561999
F15	1.985862	1.682823	2.037507	0.96656
F16	1.754978	2.052051	1.407174	0.342419
F17	1.72506	1.415434	1.564904	0.375495
F18	2.746123	1.617294	2.4928	0.506535
F19	1.891536	2.200207	1.470604	0.363025
F20	1.792942	1.509464	1.532523	0.360518
F21	2.009832	2.13849	1.56471	0.332705
F22	2.330221	2.10297	2.356706	0.375617
F23	2.127842	2.027598	1.530525	0.360371

4.2. Real-World Application

This results section is concerned with testing the proposed algorithm behavior from different sides and through different scenarios. Scenario 1 presents the results of the parameters estimation process for static SDM and DDM. Scenario 2 presents the results of the parameters estimation process for dynamic IOM and FOM.

4.2.1. Scenario 1

This scenario proposes the results and the analysis of the SDM and DDM parameters estimation process for has 57 mm diameter commercial silicon R.T.C France solar cell. The data captured from the cell at irradiance of 1000 W/m^2 and at temperature 33 °C [29]. The estimation process was carried out by ESCGBO and compared with the original algorithm GBO and some recent algorithms. The compared algorithms were artificial ecosystem-based optimization (AEO) [37] and jellyfish search optimizer (JS) [44]. The upper and lower

constrains for all estimated parameters are presented in Table 5. The control parameters for all the compared algorithms have been presented in Table 6. The estimated parameters by ESCGBO and other algorithms of SDM and DDM are presented in Tables 7 and 8, respectively. The upper and lower limit used in the optimization are listed in Table 1. The best results of the compared algorithms were determined by the best RMSE values of the compared algorithms Equation (44). The obtained RMSE for parameters estimation of the SDM and DDM by all algorithms are listed in Tables 7 and 8 respectively. For SDM the compared algorithms have the same RMSE except JS algorithm. For DDM, the ESCGBO had the best RMSE followed by the AEO algorithm. The convergence curve of all algorithms for SDM are displayed in Figure 13. The convergence curve of all algorithms for DDM are displayed in Figure 14. The best convergence behavior was achieved by ESCGBO for DDM, as can be seen in Figure 14. The robustness of the proposed and the compared algorithms are analyzed using statistical analysis. The statistical analysis of 30 independent runs of all algorithms for SDM and DDM are presented in Tables 9 and 10, respectively, and graphically analyzed using boxplot figures in Figure 15. The best standard deviation (STDEV) is achieved by ESCGBO that refer to its stability and robustness. To check the behavior of the estimated models PV current—voltage characteristics and power—voltage characteristics are presented for SDM and DDM in Figures 16 and 17, respectively. Further details on current absolute error (Equation (45)) and power absolute error (Equation (46)) for SDM and DDM are presented in Figures 18 and 19 respectively.

$$RMSE = \sqrt{\frac{1}{N}\sum_{K=1}^{N} f^2(V_{tm}, I_{tm}, X)} \tag{44}$$

$$Current\ Absolut\ error = \sqrt[2]{(I - I_{estimated})^2} \tag{45}$$

$$Power\ Absolut\ error = \sqrt[2]{(P - P_{estimated})^2} \tag{46}$$

Table 5. Upper and lower constrains for all estimated parameters.

Parameter	Solar Cell	
	Lower Limit	**Upper Limit**
R_s	0	5
R_{sh}	0	100
I_{ph}	0	2
I_{s1}	0	1
I_{s2}	0	1
η_1	1	2
η_2	1	2

Table 6. Parameters setting for all compared algorithms.

Algorithm	Control Prameters		
ESCGBO	nP = 50	pr = 0.5	CM = 4
GBO	nP = 50	pr = 0.5	CM = 4
AEO	PopSize = 50	r1 = rand	
JS	Npop = 50		
WHO	N = 50	PS = 0.2	PC = 0.13
EO	Particles_no = 50	a1 = 2, a2 = 1	GP = 0.5

**Table 7.** Estimated parameters and RMSE of ESCGBO and other algorithms for SDM model.

	ESCGBO	GBO	AEO	JS
R_s (Ω)	0.036377	0.036377	0.036377091	0.035989122
R_{sh} (Ω)	53.71853	53.71852	53.71853164	57.40304457
I_{ph} (A)	0.760776	0.760776	0.76077553	0.760889273
I_s (A)	3.23×10^{-7}	3.23×10^{-7}	3.23×10^{-7}	3.61×10^{-7}
η	1.476894	1.476894	1.476894314	1.48800365
$RMSE$	0.000986022	0.000986022	0.000986022	0.001025784

Table 8. Estimated parameters and RMSE of ESCGBO and other algorithms for DDM model.

	ESCGBO	GBO	AEO	JS
R_s (Ω)	0.036742647	0.036375073	0.036706957	0.035338785
R_{sh} (Ω)	55.50056767	53.68896189	55.32835695	54.95529664
I_{ph} (A)	0.760781168	0.76077892	0.760783208	0.760597151
I_{s1} (A)	7.64×10^{-7}	3.78×10^{-9}	2.34×10^{-7}	3.51×10^{-7}
I_{s2} (A)	2.26×10^{-7}	3.20×10^{-7}	7.00×10^{-7}	2.02×10^{-7}
η_1	1.99999756	1.434376674	1.449582257	1.487619855
η_2	1.446849934	1.477724492	1.999999984	1.916132114
$RMSE$	0.000982418	0.000986029	0.000982451	0.00119615

Table 9. The statistical results of SDM for all other algorithms.

	Minimum	Average	Maximum	STD
ESCGBO	0.000986022	0.000986026	0.000986032	5.507×10^{-9}
GBO	0.000986022	0.000989455	0.000996022	5.688×10^{-6}
AEO	0.000986022	0.000989555	0.000996022	5.608×10^{-6}
JS	0.001025784	0.001525784	0.002025784	0.0005
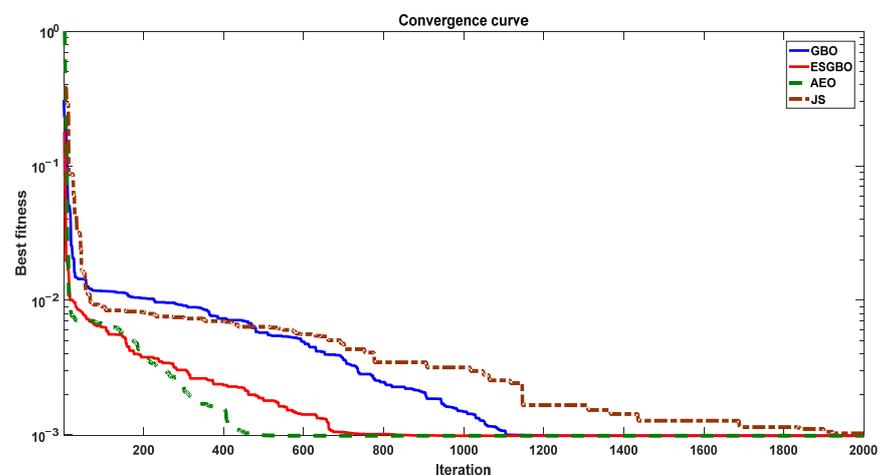
Figure 13. The convergence curve of all algorithms of SDM.
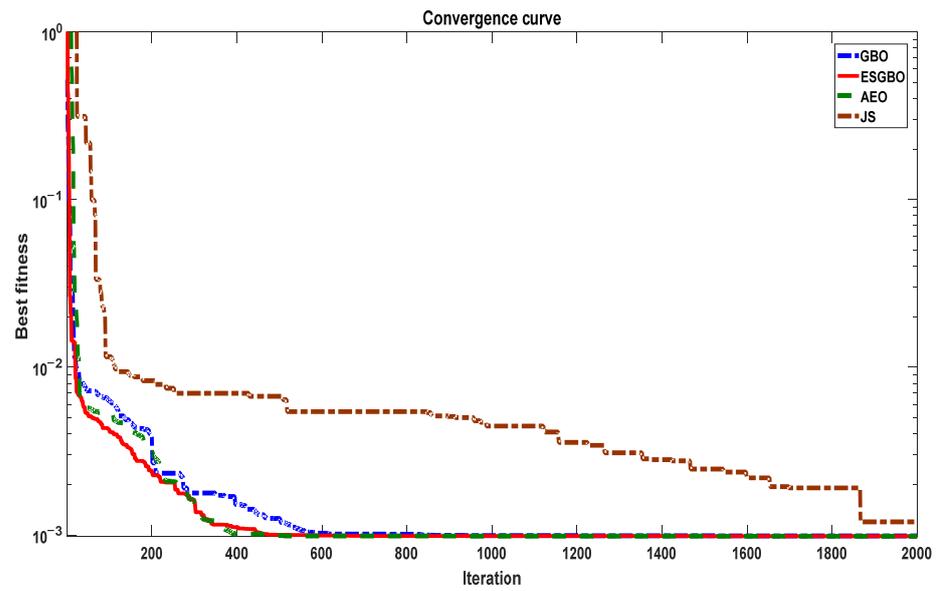
Figure 14. The convergence curve of all algorithms of DDM.

Table 10. The statistical results of DDM for all algorithms.

	Minimum	Average	Maximum	STD
ESCGBO	0.000982418	0.000982454	0.000982518	5.507×10^{-8}
GBO	0.000986029	0.000990032	0.000997039	6.088×10^{-6}
AEO	0.000982451	0.000988451	0.000992451	5.291×10^{-6}
JS	0.00110615	0.001469483	0.00210615	0.00055
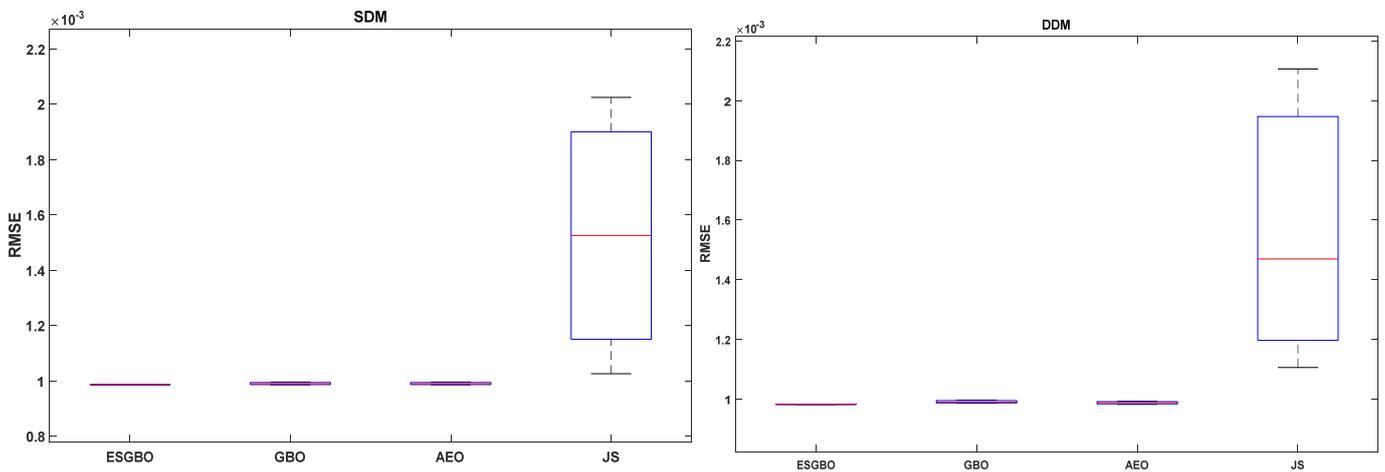
Figure 15. Boxplot figure of all algorithms for 30 independent runs in case of SDM and DDM.
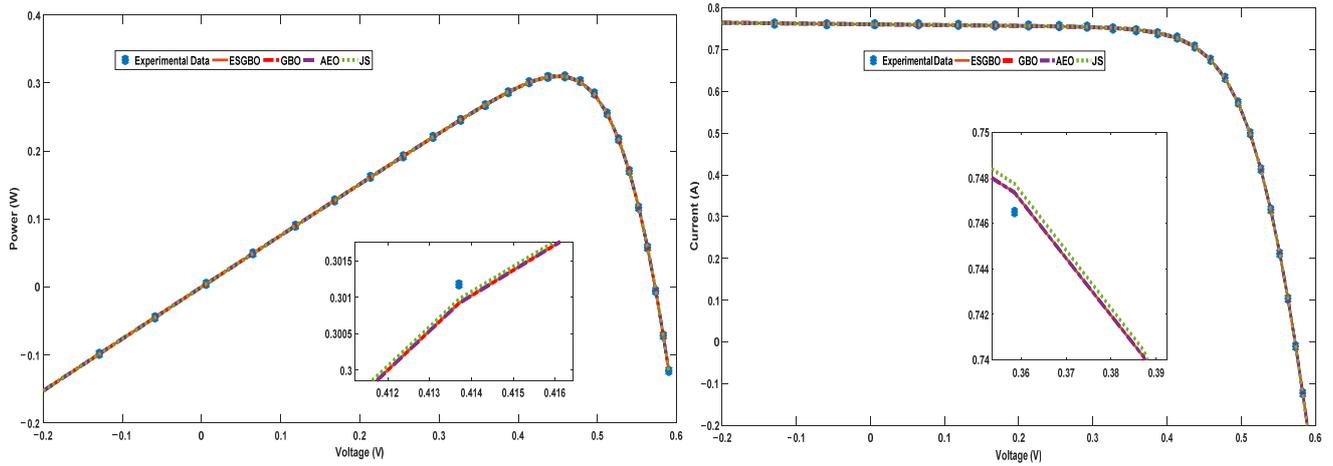
Figure 16. Power and current characteristics for SDM estimated by all algorithms.
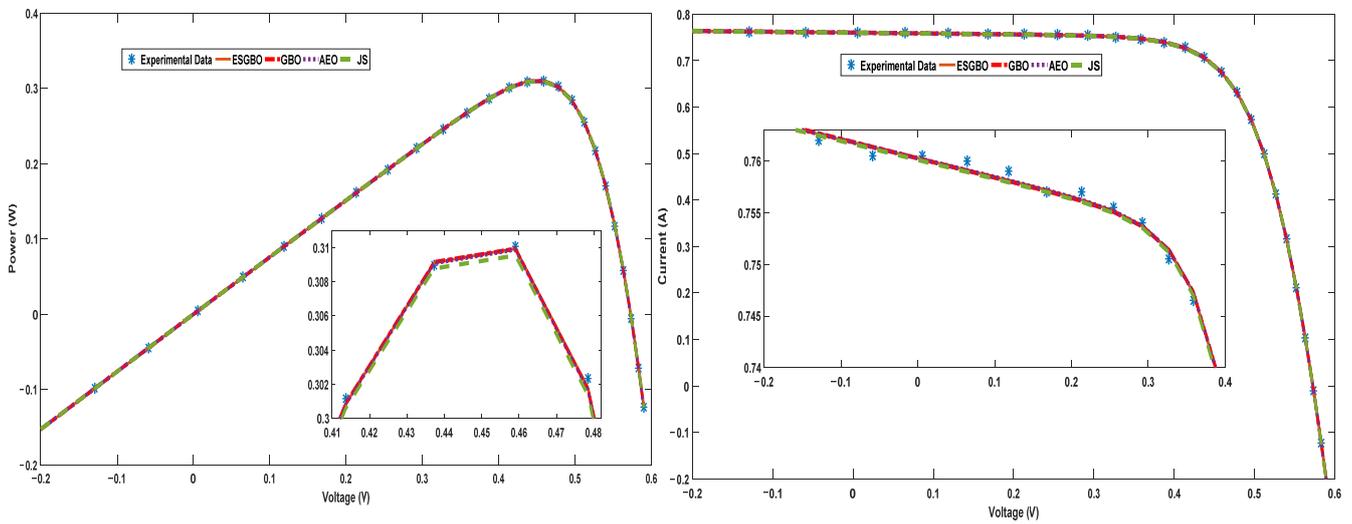
Figure 17. Power and current characteristics for DDM estimated by all algorithms.
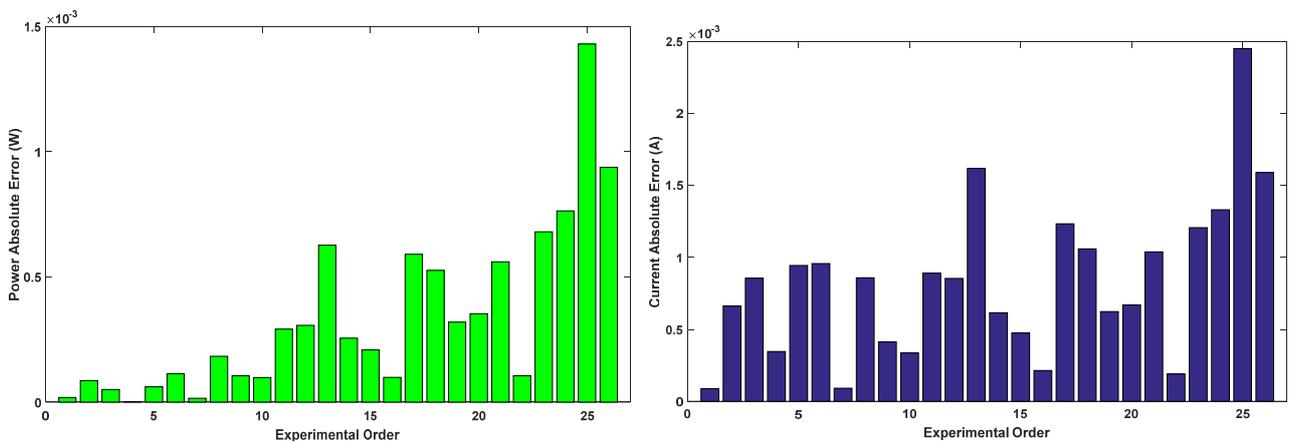
Figure 18. Power and current absolute error for SDM estimated by all algorithms.
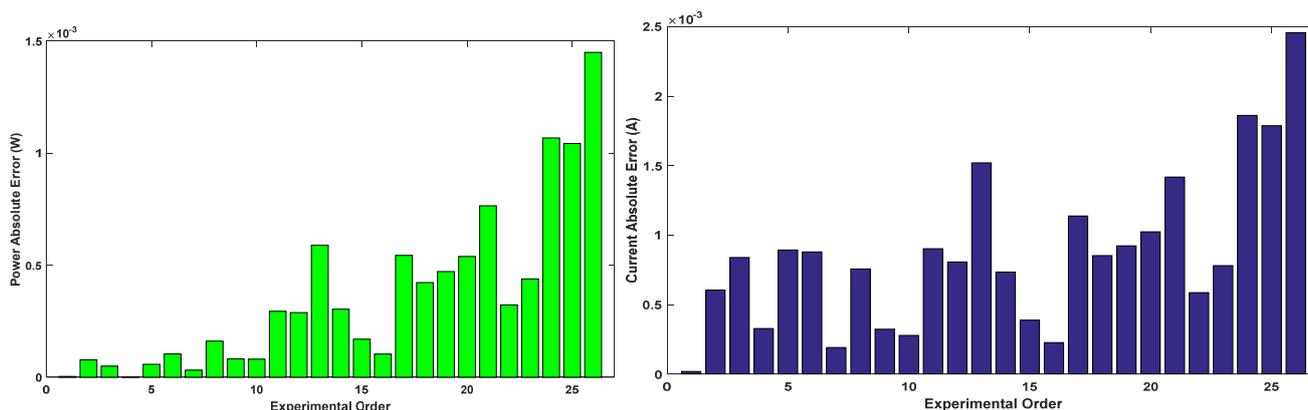
Figure 19. Power and current absolute error for DDM estimated by all algorithms.

4.2.2. Scenario 2

This scenario is concerned with the application of ESCGBO for parameters estimation of dynamic PV models (IOM and FOM). The applied dataset were captured from PV module at a temperature of 25 °C and irradiance level of 655 W/m^2 through connected load of R_l = 23.1. The ranges for all the estimated parameters are presented in Table 11. The obtained results are also compared with other recent algorithms. The three estimated parameters for IOM (R_c, C and L) and *RMSE* obtained by all algorithms are listed in Table 12. The five estimated parameters for FOM (R_c, C, L, α and β) and *RMSE* obtained by all algorithms are listed in Table 13. The best *RMSE* was achieved by ESCGBO in case of IOM and FOM. The convergence curves of all the compared algorithms are presented for IOM and FOM in Figures 20 and 21, respectively. The load current curve for the real experimental data and all algorithms for IOM and FOM are presented in Figures 22 and 23, respectively. The current absolute error curve of all algorithms for IOM and FOM are presented in Figures 24 and 25 respectively. In the previous figures comparison, the results obtained by the ESCGBO are better than other algorithms and the results obtained for FOM are more accurate than IOM.

**Table 11.** Upper and lower constrains for all estimated parameters.

Parameter	Solar Cell	
	Lower Limit	**Upper Limit**
R_c	0	20
C	2×10^{-8}	6×10^{-5}
L	5×10^{-6}	100×10^{-6}
α	0.8	1.1
β	0.8	1.1

Table 12. Estimated parameters of IOM model for all algorithms.

	ESCGBO	**GBO**	**AEO**	**JS**
R_c	5.583588	5.624748753	5.624748647	5.6247490
C	8.30×10^{-6}	8.16×10^{-6}	8.16×10^{-6}	8.15726×10^{-6}
L	7.43×10^{-6}	7.47×10^{-6}	7.47×10^{-6}	7.47323×10^{-6}
RMSE	0.008259444	0.008493067	0.008493067	0.008493067262

Table 13. Estimated parameters of FOM model for all algorithms.

	ESCGBO	**GBO**	**AEO**	**JS**
R_c	4.617127916	5.005984283	4.550201539	4.698924713
C	5.81×10^{-5}	5.04×10^{-6}	1.46×10^{-5}	4.08×10^{-5}
L	1.50×10^{-5}	1.35×10^{-5}	1.73×10^{-5}	1.44×10^{-5}
A	0.8	1.026120535	0.917230623	0.833404373
β	0.950840763	0.957165925	0.940654537	0.953192785
RMSE	0.007951289	0.008236017	0.00819598	0.007995872
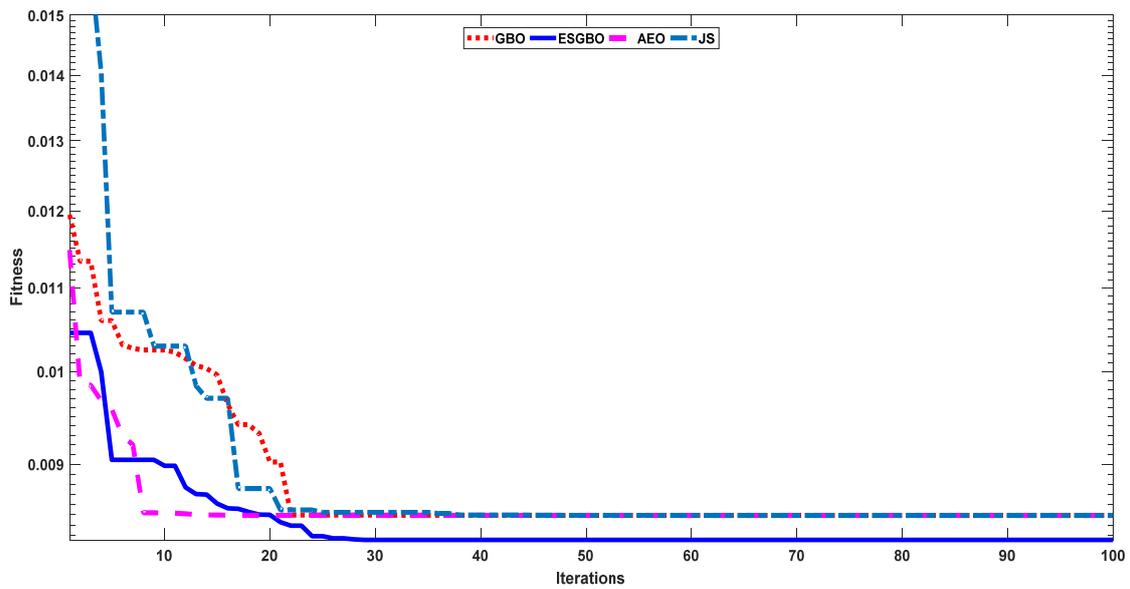
Figure 20. Convergence curves of all algorithms for IOM.
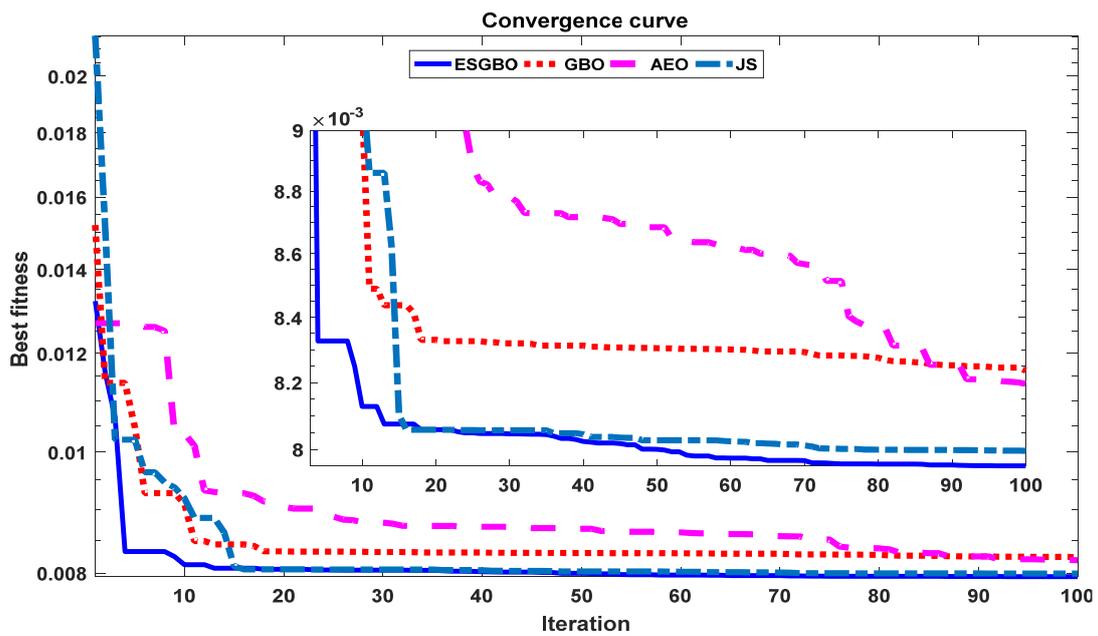
Figure 21. Convergence curves of all algorithms for FOM.
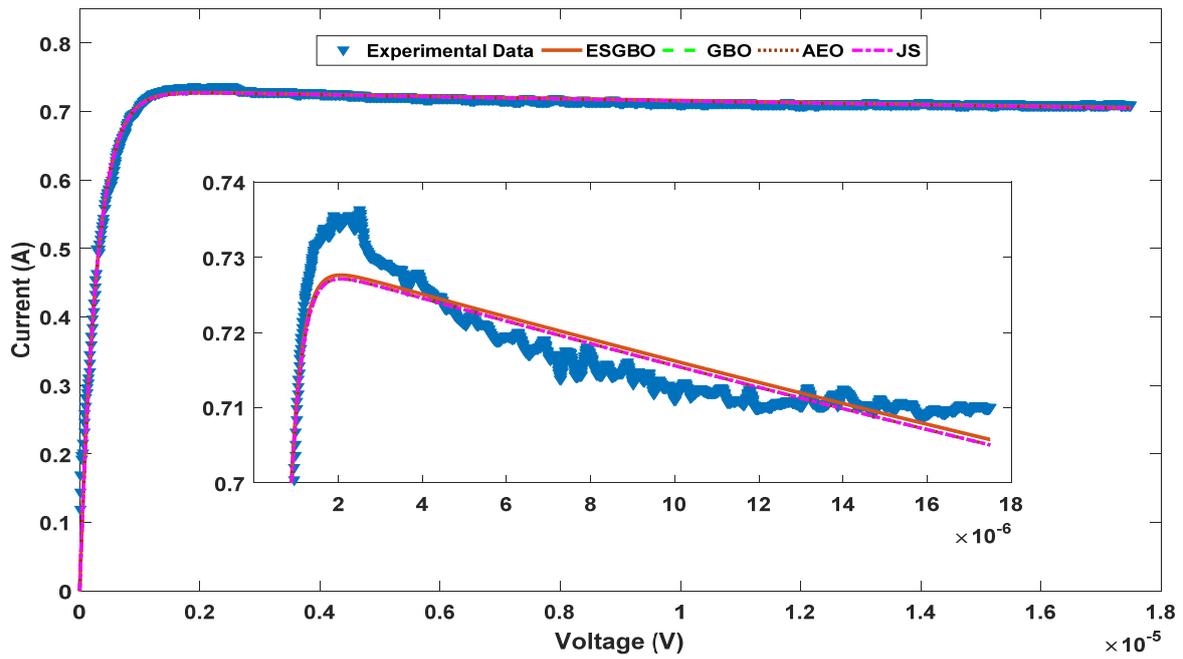
**Figure 22.** Load current curve of real data and the estimated IOM by different algorithms.
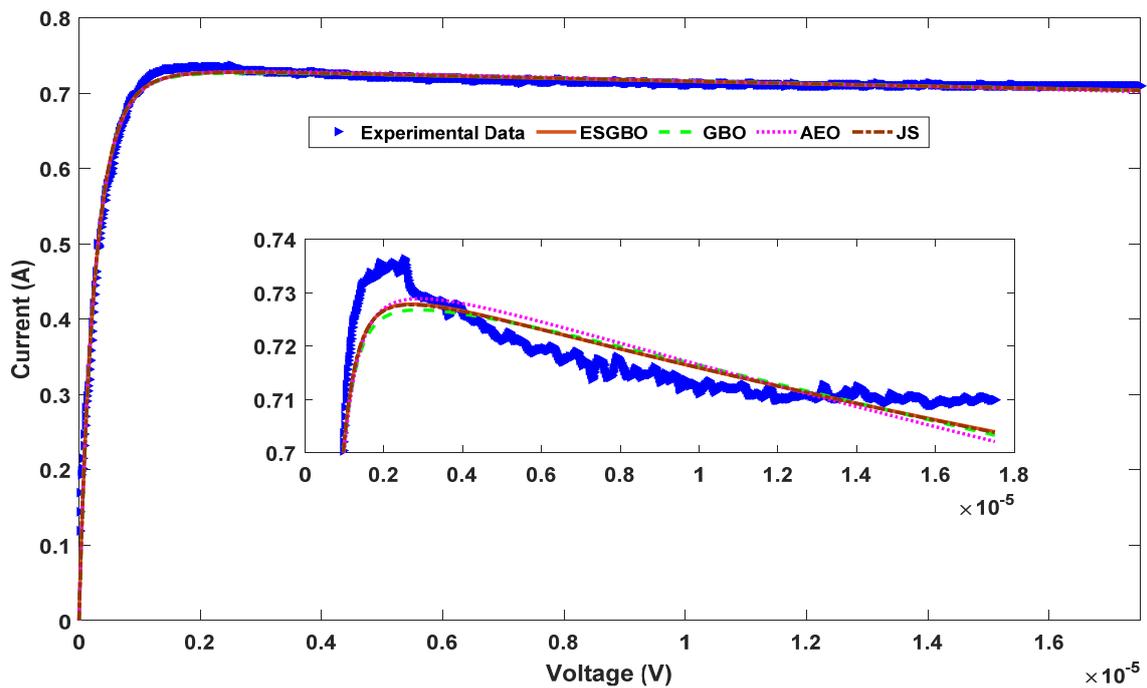
**Figure 23.** Load current curve of real data and the estimated FOM by different algorithms.
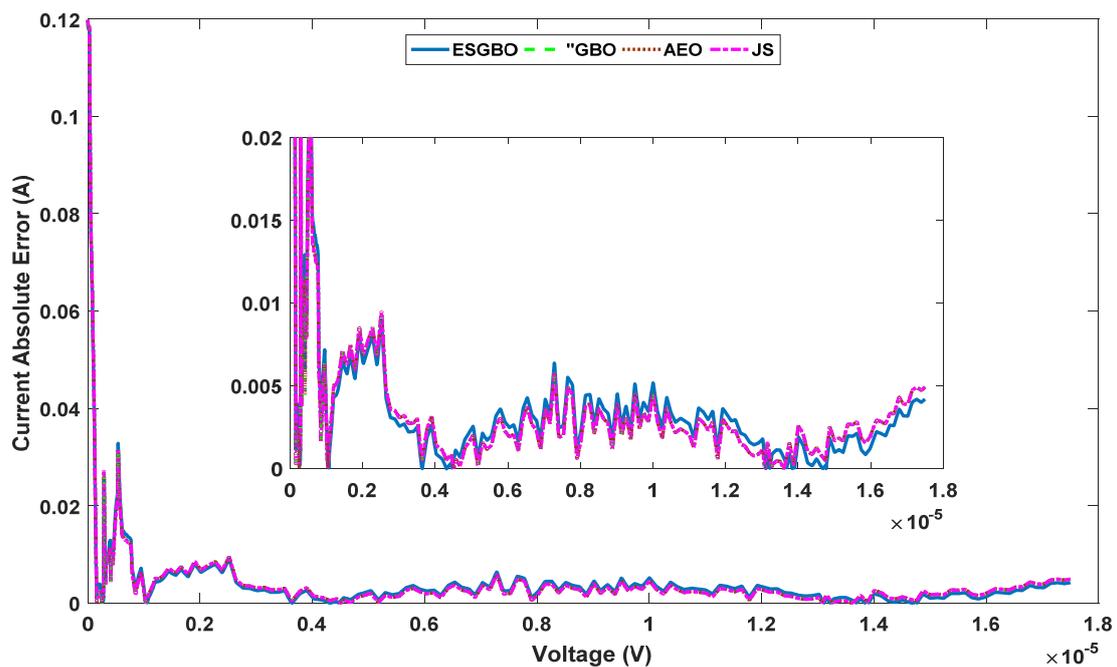
**Figure 24.** Current absolute error of the estimated IOM by different algorithms.
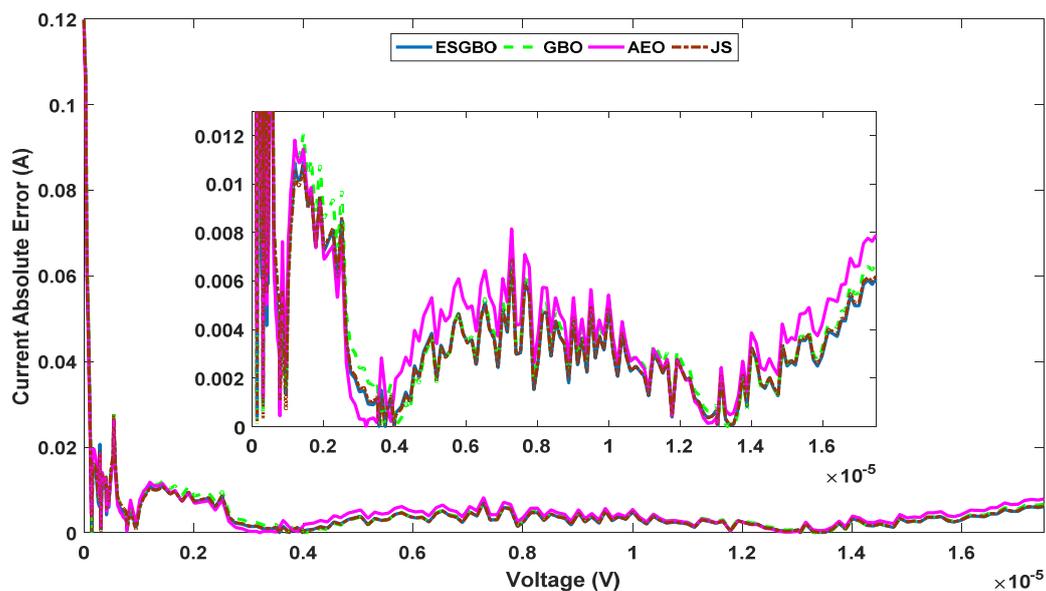
Figure 25. Current absolute error of the estimated FOM by different algorithms.

5. Conclusions

This work proposed a new modified metaheuristic optimization algorithm named ESCGBO. It is considered an enhancement for the original GBO to enhance the balance between exploration and exploitation and completely enhance the algorithm performance. First, the proposed ESCGBO's performance was tested on the 23 benchmark functions. The proposed technique achieved better than three well-known optimization techniques, such as EO and WHO as well as the original GBO. Then, the new algorithm was applied to estimate the parameters of static SDM and DDM models through application 1, which used the real data of 57 mm diameter commercial silicon R.T.C France solar cell, and dynamic IOM and FOM models through application two, using the dataset, were captured from the PV module at temperature 25 °C at an irradiance level of 655 W/m^2 through connected

load of Rl = 23.1. The obtained results were analyzed in different ways to evaluate the performance of the proposed algorithm. The accuracy of the algorithm was tested through calculation of the RMSE and IAE, then by comparing it for all algorithms. The robustness was also checked by running the algorithms with 30 independent runs and analyzing the results through statistical analysis. From all the analysis, the proposed ESCGBO is more accurate and robust when compared with other recent algorithms. The static DDM is more accurate than SDM and dynamic IOM is more accurate than FOM. For future work, this study contributes to research that focuses on studying the applicability of ESCGBO PV parameters' estimation for the large and complex PV system.

Author Contributions: Conceptualization, A.R., S.K. and M.H.H.; data curation, M.T.-V. and A.M.E.; formal analysis, A.R., S.K. and M.H.H.; funding acquisition, M.T.-V., S.K. and A.M.E.; investigation, A.R., S.K. and M.H.H.; methodology, A.R., S.K. and M.H.H.; project administration, M.T.-V. and A.M.E.; resources, M.T.-V. and A.M.E.; software, A.R., M.H.H. and S.K.; supervision, S.K., M.T.-V., and A.M.E.; validation, A.R., S.K. and M.H.H.; visualization, M.T.-V. and A.M.E.; writing—original draft, A.R., S.K. and M.H.H.; writing—review and editing, M.T.-V. and A.M.E. All authors have read and agreed to the published version of the manuscript.

Funding: The authors extend their appreciation to the Deanship of Scientific Research at King Saud University for funding this work through research group No (RG-1441-422).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Nomenclature

Symbol	Description
SDM	Single Diode Model
PV	Photo Voltaic
V	Terminal voltage
I_{ph}	Current source generated from photons
n_1	Ideality factor for the first diode (Diffusion current components)
Rs	Series resistance to represent the total semiconductor material at neutral regions resistance.
$Is1$	Current passing through the first diode
K	constant of = 1.380×10^{-23}(J/Ko)
DDM	Double Diode Model
I	PV module output current
$RMSE$	Root Mean Square Error
$T (Ko)$	Photo cell temperature (Kelvin)
n_2	Ideality Factor for the second Diode (Recombination current components)
R_{sh}	Shunt resistance to represent the total current leakage resistance across the P-N junction of solar c
$Is2$	Current passing through the Second diode
q	1.602×10^{-19} (C) Coulombs.

References

1. Vinothkanna, I.; Senthilkumaran, P.; Tamilselvan, R. Increase the Efficiency of Solar Panel by using Mirrors. *J. Sol. Energy Eng.* **2014**, *1*, 4.
2. Araújoa, N.M.F.T.S.; Sousab, F.J.P.; Costab, F.B. Equivalent Models for Photovoltaic Cell—A Review. *Therm. Eng.* **2020**, *19*, 77–98. [CrossRef]
3. Mayer, M.J.; Gróf, G. Extensive comparison of physical models for photovoltaic power forecasting. *Appl. Energy* **2021**, *283*, 116239. [CrossRef]
4. Lim, H.I.; Ye, Z.; Ye, J.; Yang, D.; Du, H. A linear identification of diode models from single I–V characteristics of PV panels. *IEEE Trans. Ind. Electron.* **2015**, *62*, 4181–4193. [CrossRef]
5. Diab, A.A.Z.; Sultan, H.M.; Do, T.D.; Kamel, O.M.; Mossa, M.A. Coyote Optimization Algorithm for Parameters Estimation of Various Models of Solar Cells and PV Modules. *IEEE Access* **2020**, *8*, 111102–111140. [CrossRef]

6. Sharma, A.; Sharma, A.; Averbukh, M.; Jately, V.; Azzopardi, B. An Effective Method for Parameter Estimation of a Solar Cell. *Electronics* **2021**, *10*, 312. [CrossRef]

7. Rezk, H.; Babu, T.S.; Al-Dhaifallah, M.; Ziedan, H.A. A robust parameter estimation approach based on stochastic fractal search optimization algorithm applied to solar PV parameters. *Energy Rep.* **2021**, *7*, 620–640. [CrossRef]

8. Prakash, S.B.; Singh, G.; Singh, S. Modeling and Performance Analysis of Simplified Two-Diode Model of Photovoltaic Cells. *Front. Phys.* **2021**, *9*, 690588. [CrossRef]

9. Messaoud, R.B. Extraction of uncertain parameters of double-diode model of a photovoltaic panel using Ant Lion Optimization. *Appl. Sci.* **2018**, *2*, 239. [CrossRef]

10. Bayoumi, A.S.; El-Sehiemy, R.A.; Mahmoud, K.; Lehtonen, M.; Darwish, M.M.F. Assessment of an Improved Three-Diode against Modified Two-Diode Patterns of MCS Solar Cells Associated with Soft Parameter Estimation Paradigms. *Appl. Sci.* **2021**, *11*, 1055. [CrossRef]

11. Abdel-Basseta, M.; Mohameda, R.; Mirjalilib, S.; Chakraborttyc, R.K.; Ryanc, M.J. Solar photovoltaic parameter estimation using an improved equilibrium optimizer. *Sol. Energy* **2020**, *209*, 694–708. [CrossRef]

12. Houssein, E.H.; Zaki, G.N.; Diab, A.A.Z.; Younis, E.M.G. An efficient Manta Ray Foraging Optimization algorithm for parameter extraction of three-diode photovoltaic model. *Comput. Electr. Eng.* **2021**, *94*, 107304. [CrossRef]

13. Ibrahim, I.A.; Hossain, M.J.; Duck, B.C.; Nadarajah, M. An improved wind driven optimization algorithm for parameters identification of a triple-diode photovoltaic cell model. *Energy Convers. Manag.* **2020**, *213*, 112872. [CrossRef]

14. Qin, L.; Xie, S.; Yang, C.; Cao, J. Dynamic model and dynamic characteristics of solar cell. In Proceedings of the 2013 IEEE ECCE Asia Downunder, Melbourne, VIC, Australia, 3–6 June 2013.

15. Go, S.I.; Choi, J.H. Design and Dynamic Modelling of PV-Battery Hybrid Systems for Custom Electromagnetic Transient Simulation. *Electronics* **2020**, *9*, 1651. [CrossRef]

16. Batzelis, E.I.; Anagnostou, G.; Cole, I.R.; Betts, T.R.; Pal, B.C. A State-Space Dynamic Model for Photovoltaic Systems with Full Ancillary Services Support. *IEEE Trans. Sustain. Energy* **2018**, *10*, 1399–1409. [CrossRef]

17. Parida, S.M.; Rout, P.K. Differential evolution with dynamic control factors for parameter estimation of photovoltaic models. *J. Comput. Electron.* **2021**, *20*, 330–343. [CrossRef]

18. Yousri, D.; Allam, D.; Eteibaa, M.B.; Suganthanb, P.N. Static and dynamic photovoltaic models' parameters identification using Chaotic Heterogeneous Comprehensive Learning Particle Swarm Optimizer variants. *Energy Convers. Manag.* **2019**, *182*, 546–563. [CrossRef]

19. Maniraj, B.; Fathima, A.P. Parameter extraction of solar photovoltaic modules using various optimization techniques: A review. *J. Phys. Conf. Ser.* **2020**, *1716*, 012001. [CrossRef]

20. Naeijian, M.; Rahimnejad, A.; Ebrahimi, S.M.; Pourmousa, N.; Gadsden, S.A. Parameter estimation of PV solar cells and modules using Whippy Harris Hawks Optimization Algorithm. *Energy Rep.* **2021**, *7*, 4047–4063. [CrossRef]

21. Heidari, A.A.; Mirjalili, S.; Faris, H.; Mafarja, I.A.M.; Chen, H. Harris hawks optimization: Algorithm and applications. *Future Gener. Comput. Syst.* **2019**, *97*, 849–872. [CrossRef]

22. Han, L.; Xu, C.; Huang, T.; Dang, X. Improved particle swarm optimization algorithm for high performance SPR sensor design. *Appl. Opt.* **2021**, *60*, 1753–1760. [CrossRef]

23. Chakraborty, S.; Verma, S.; Salgotra, A.; Elavarasan, R.M.; Elangovan, D.; Mihet-Popa, L. Solar-Based DG Allocation Using Harris Hawks Optimization While Considering Practical Aspects. *Energies* **2021**, *14*, 5206. [CrossRef]

24. Hassan, M.H.; Kamel, S.; El-Dabah, M.A.; Khurshaid, T.; Dominguez-Garcia, J.L. Optimal Reactive Power Dispatch with Time-Varying Demand and Renewable Energy Uncertainty Using Rao-3 Algorithm. *IEEE Access* **2021**, *9*, 23264–23283. [CrossRef]

25. Ridha, H.M.; Gomes, C.; Hizam, H. Estimation of photovoltaic module model's parameters using an improved electromagnetic-like algorithm. *Neural Comput. Appl.* **2020**, *32*, 12627–12642. [CrossRef]

26. Sharma, A.; Dasgotra, A.; Tiwari, S.K.; Sharma, A.; Jately, V.; Azzopardi, B. Parameter Extraction of Photovoltaic Module Using Tunicate Swarm Algorithm. *Electronics* **2021**, *10*, 878. [CrossRef]

27. Xiong, G.; Zhang, J.; Yuan, X.; Shi, D.; He, Y.; Yao, G. Parameter estimation of solar photovoltaic models by means of a hybrid differential evolution with whale optimization algorithm. *Solar. Energy* **2018**, *176*, 742–761. [CrossRef]

28. Ramadan, A.; Kamel, S.; Taha, I.B.M.; Tostado-Véliz, M. Parameter Estimation of Modified Double-Diode and Triple-Diode Photovoltaic Models Based on Wild Horse Optimizer. *Electronics* **2021**, *10*, 2308. [CrossRef]

29. Ramadan, A.; Kamel, S.; Hassan, M.H.; Khurshaid, T.; Rahmann, C. An Improved Bald Eagle Search Algorithm for Parameter Estimation of Different Photovoltaic Models. *Processes* **2021**, *9*, 1127. [CrossRef]

30. Ahmadianfar, I.; Bozorg-Haddad, O.; Chu, X. Gradient-Based Optimizer: A New Metaheuristic Optimization Algorithm. *Inf. Sci.* **2020**, *540*, 131–159. [CrossRef]

31. Yapıcı, H.; Çetinkaya, N. An Improved Particle Swarm Optimization Algorithm Using Eagle Strategy for Power Loss Minimization. *Math. Probl. Eng.* **2017**, *2017*, 1063045. [CrossRef]

32. Santhosh, K.; Neela, R. Optimal Placement of Distribution Generation in Micro-Grid using Eagle Strategy with Particle Swarm Optimizer. *Int. J. Pure Appl. Math.* **2018**, *118*, 3819–3825.

33. Chen, Z.; Wu, L.; Lin, P.; Wu, Y.; Cheng, S. Parameters identification of photovoltaic models using hybrid adaptive Nelder-Mead simplex algorithm based on eagle strategy. *Appl. Energy* **2016**, *182*, 47–57. [CrossRef]

34. Xu, S.; Wang, Y.; Wang, Z. Parameter estimation of proton exchange membrane fuel cells using eagle strategy based on JAYA algorithm and Nelder-Mead simplex method. *Energy* **2019**, *173*, 457–467. [CrossRef]
35. Yang, X.-S.; Deb, S.; He, X. Eagle strategy with flower algorithm. In Proceedings of the 2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Mysore, India, 22–25 August 2013.
36. Menesy, A.S.; Sultan, H.M.; Selim, A.; Ashmawy, M.G.; Kamel, S. Developing and Applying Chaotic Harris Hawks Optimization Technique for Extracting Parameters of Several Proton Exchange Membrane Fuel Cell Stacks. *IEEE Access* **2019**, *8*, 1146–1159. [CrossRef]
37. Hassan, M.H.; Kamel, S.; Salih, S.Q.; Khurshaid, T.; Ebeed, M. Developing Chaotic Artificial Ecosystem-Based Optimization Algorithm for Combined Economic Emission Dispatch. *IEEE Access* **2021**, *9*, 51146–51165. [CrossRef]
38. Premkumar, M.; Jangir, P.; Sowmya, R.; Elavarasan, R.M.; Kumar, B.S. Enhanced chaotic JAYA algorithm for parameter estimation of photovoltaic cell/modules. *ISA Trans.* **2021**, *116*, 139–166. [CrossRef]
39. Ibrahim, A.; Mohammed, S.; Ali, H.; Hussein, S.E. Breast Cancer Segmentation from Thermal Images Based on Chaotic Salp Swarm Algorithm. *IEEE Access* **2020**, *8*, 122121–122134. [CrossRef]
40. Hassan, M.; Kamel, S.; El-Dabah, M.; Rezk, H. A Novel Solution Methodology Based on a Modified Gradient-Based Optimizer for Parameter Estimation of Photovoltaic Models. *Electronics* **2021**, *10*, 472. [CrossRef]
41. Emary, E.; Zawbaa, H.M. Impact of Chaos Functions on Modern Swarm Optimizers. *PLoS ONE* **2016**, *11*, e0158738. [CrossRef]
42. Faramarzi, A.; Heidarinejad, M.; Stephens, B.; Mirjalili, S. Equilibrium optimizer: A novel optimization algorithm. *Knowl.-Based Syst.* **2019**, *191*, 105190. [CrossRef]
43. Naruei, I.; Keynia, F. Wild horse optimizer: A new meta-heuristic algorithm for solving engineering optimization problems. *Eng. Comput.* **2021**, 1–32. [CrossRef]
44. Chou, J.-S.; Truong, D.-N. A novel metaheuristic optimizer inspired by behavior of jellyfish in ocean. *Appl. Math. Comput.* **2020**, *389*, 125535. [CrossRef]