mathematics

MDPI

Article

Photovoltaic Models' Parameter Extraction Using New Artificial Parameterless Optimization Algorithm

Mohana Alanazi [1], Abdulaziz Alanazi [2], Ahmad Almadhor [3] and Hafiz Tayyab Rauf [4,*]

1 Department of Electrical Engineering, College of Engineering, Jouf University, Sakaka 72388, Saudi Arabia
2 Department of Electrical Engineering, College of Engineering, Northern Border University, Ar'Ar 73222, Saudi Arabia
3 Department of Computer Engineering and Networks, College of Computer and Information Sciences, Jouf University, Sakaka 72388, Saudi Arabia
4 Centre for Smart Systems, AI and Cybersecurity, Staffordshire University, Stoke-on-Trent ST4 2DE, UK
* Correspondence: hafiztayyabrauf093@gmail.com

Citation: Alanazi, M.; Alanazi, A.; Almadhor, A.; Rauf, H.T. Photovoltaic Models' Parameter Extraction Using New Artificial Parameterless Optimization Algorithm. *Mathematics* **2022**, *10*, 4617. https://doi.org/10.3390/math10234617

Academic Editors: Yuan Dong and Yuchao Hua

Received: 11 November 2022
Accepted: 1 December 2022
Published: 6 December 2022

Abstract: Identifying parameters in photovoltaic (PV) cell and module models is one of the primary challenges of the simulation and design of photovoltaic systems. Metaheuristic algorithms can find near-optimal solutions within a reasonable time for such challenging real-world optimization problems. Control parameters must be adjusted with many existing algorithms, making them difficult to use. In real-world problems, many of these algorithms must be combined or hybridized, which results in more complex and time-consuming algorithms. This paper presents a new artificial parameter-less optimization algorithm (APLO) for parameter estimation of PV models. New mutation operators are designed in the proposed algorithm. APLO's exploitation phase is enhanced by each individual searching for the best solution in this updating operator. Moreover, the current best, the old best, and the individual's current position are utilized in the differential term of the mutation operator to assist the exploration phase and control the convergence speed. The algorithm uses a random step length based on a normal distribution to ensure population diversity. We present the results of a comparative study using APLO and well-known existing parameter-less meta-heuristic algorithms such as grey wolf optimization, the salp swarm algorithm, JAYA, teaching-learning based optimization, colliding body optimization, as well as three major parameter-based algorithms such as differential evolution, genetic algorithm, and particle swarm optimization to estimate the parameters of PV the modules. The results revealed that the proposed algorithm could provide excellent exploration–exploitation balance and consistency during the iterations. Furthermore, the APLO algorithm shows high reliability and accuracy in identifying the parameters of PV cell models.

Keywords: solar cells; photovoltaic modeling; metaheuristic algorithm; global optimization; power system management; renewable energy

MSC: 68T20; 90C26

1. Introduction

1.1. Background

Because of solar energy's outstanding environmental, technical, and economic properties, increasing the integration of solar photovoltaic systems with electric utilities is inevitable [1]. Solar radiation is abundant in most areas of the world. These systems can be used for energy generation, enabling customers to invest quickly in their electrical systems. Photovoltaic (PV) systems convert solar energy into electricity. Solar energy's potential for generating electricity depends on various factors, including temperature and solar radiation [2,3]. Therefore, it is vital to assess how PV systems perform in operation to be modeled, managed, and optimized for future operations [4].

The inherent characteristics of PV cells give rise to a nonlinear power–voltage (P–V) curve, which is highly influenced by the environment. Ideally, a PV panel should operate at maximum efficiency at the peak of the curve. A proper maximum power point tracking (MPPT) technique enables PV systems to be used more efficiently in different environmental conditions, requiring accurate and reliable data regarding PV parameters [5,6]. Unknown parameters are the most significant contributors to PV models. It is, therefore, crucial to identify the model's unknown parameters early using a feasible optimization algorithm, regardless of the model used. Accordingly, a robust optimization algorithm to accurately estimate the parameters of PV models and track the maximum power point under different conditions is urgent.

1.2. Related Works

Generally, solar PV systems can be optimized using deterministic and metaheuristic algorithms. Due to their reliance on gradient information and sensitivity to initial points, deterministic algorithms are unreliable. Moreover, because of their nonlinearity, these classical algorithms also have trouble capturing local optima in the nonconvex space the equivalent PV circuits created. The result may be an inaccurate estimation of parameters and, consequently, a failure to track the maximum power point [7–9].

Metaheuristic optimization techniques are considered modern and straightforward alternatives to deterministic algorithms. In general, metaheuristic algorithms fall into two main categories: single-solution-based and multiple-solution-based algorithms [10]. It is understood that the former algorithm uses an iterative process to achieve a superior solution by starting with a randomly selected candidate solution and moving and improving it in a promising search space iteratively. A common single-solution metaheuristic algorithm is simulation annealing (SA) [11]. Multiple-solution algorithms employ several random solutions to enhance their performance.

In classifications of population-based metaheuristic algorithms, evolutionary algorithms, physics algorithms, chemistry algorithms, and swarm-based algorithms are all included [10,12]. Based on evolution in nature, evolutionary-based metaheuristic algorithms move populations based on improvements and movements. The physics-based metaheuristic algorithm enhances the initial population through search space by using principles established based on the physics' lows, such as mechanics, relativity, gravity, electrodynamics, electromagnetism, and optics. A chemical reaction and molecule characteristics are utilized to develop chemistry-based algorithms. Living organisms, including birds, ants, swarms, schools, and so on, are modeled using swarm-based algorithms. Some known population-based metaheuristic algorithms are introduced in Table 1.

Table 1. The classification of population-based metaheuristic algorithms.

Category	Most Popular Algorithms and Abbreviations
Evolutionary-based	Genetic algorithm (GA) [13] Evolutionary programming (EP) [14] Genetic programming (GP) [15] Biogeography-based optimizer (BBO) [16] Differential evolution (DE) [17] Evolutionary strategy (ES)
Physics-based	Gravitational Ssearch Algorithm (GSA) [18] Charged system search (CSS) [19] River formation dynamics algorithm (RFDA) [20] Big bang–big crunch (BB-BC) [21] Extremal optimization (EO) [22] Galaxy-based search algorithm (GBSA) [23] Central force optimization (CFO) [24] Ray optimization (RO) [25] Water cycle algorithm (WCA) [26] Intelligent water drops (IWD) [27] Chaos optimization algorithm (COA) [28] Electromagnetism-like mechanism (EM) [29]

Table 1. *Cont.*

Category	Most Popular Algorithms and Abbreviations
Chemistry-based	Artificial chemical reaction optimization algorithm (ACROA) [30] Artificial chemical process (ACP) [31] Gases Brownian motion optimization (GBMO) [32]
Swarm-based	Particle swarm optimization (PSO) [33] Cuckoo search (CS) [34] Ant lion optimizer (ALO) [35] Bees algorithm (BA) [36] Shuffled frog-leaping algorithm (SFLA) [37] Bat algorithm (BA) [38] Moth–flame optimization (MFO) [39] Bacterial foraging algorithm (BFA) [40] Krill herd (KH) [41] Whale optimization algorithm (WOA) [42] Ant colony algorithms (ACO) [43] Grey wolf optimizer (GWO) [44] Firefly algorithm (FA) [45] Artificial bee colony (ABC) [46] Fruit fly optimization algorithm (FOA) [47] Glowworm swarm optimization (GSO) [48]

Unlike deterministic algorithms, metaheuristic algorithms find near-optimal solutions within reasonable time for challenging real-world optimization problems. As a result, many optimization problems in science and engineering have been solved thanks to easy implementation and efficiency (see sample [49–54]). Hence, researchers have been motivated to develop successful algorithms inspired by natural and artificial processes, to solve complex optimization problems.

In recent years, metaheuristic algorithms have been used to optimize PV systems more accurately and flexibly. The development of parameterless metaheuristic algorithms for optimal parameter identification of solar PV cells has been the subject of considerable research. For example, for the extraction of parameters from PV cell-based single and double diode models [55], the salp swarm algorithms (SSA) were used. The JAYA algorithm was developed by Rao [56] as a powerful heuristic for solving optimization problems. It has been demonstrated by [57] that JAYA can be used to estimate PV cell and module parameters based on performance-guided criteria. Through a chaotic learning process, ref. [58] proposed an improved JAYA algorithm (IJAYA) to find PV model parameters reliably and accurately. In Ref. [59], TLBO was numerically simulated interactively and applied to various solar cells. TLBO was improved and simplified by [60] using an elite strategy and a local search to identify the parameters of the solar PV cells. GOTLBO (generalized oppositional TLBO) was derived from generalized opposition-based learning and was employed to identify solar cell models' parameters [61]. The modified salp swarm algorithm (MSSA) has been used as an efficient metaheuristic for identifying PV model parameters [62]. However, according to the results of this modification, sufficient robust solutions haven't been achieved for all PV models.

Examples of metaheuristic algorithms with control parameters for optimal parameter estimation of PV cells include DE, GA, and PSO. An estimation method for solar PV module parameters based on penalized differential evolution (P-DE) was proposed by [63]. Adding new scaling factors and crossover rates to adaptive DE improved the parameter estimation [64]. The genetic algorithm was utilized by [65] to determine solar cells' I-V characteristics. PSOs have been successfully applied in some modified and hybrid forms to identify the parameters of solar cells. For instance, the chaotic heterogeneous comprehensive learning particle swarm optimizer was proposed by [66] for dynamic and static PV models' parameter identification. In Ref. [67], the flexible particle swarm optimizer was presented for parameter extraction of different PV models. A classified perturbation mutation-based particle swarm optimization was introduced by [68]. In Ref. [69], a niche particle swarm optimization in parallel computing was proposed and applied to identify the unknown parameters of PV cells. A fractional chaotic ensemble particle swarm optimizer was utilized for estimating the parameters of three models of PV cells [70]. Exploitation and exploration phases were balanced effectively to mitigate the premature convergence associated with PSO [71]. A variation of opposition-based GOA (OBGOA) has been proposed by [72] to identify the electrical parameters of various PV models. A modified spotted hyena optimization algorithm MSHOA was proposed by [73] to boost the optimal solution's performance through an accelerated function. In Ref. [74], a hybrid optimization algorithm called hARS-PS was presented that uses adaptive rat swarm optimization (ARSO) and pattern search (PS) to extract PV parameters.

1.3. Motivation

Estimating the parameters of PV systems has been widely done using metaheuristic algorithms. Nonetheless, researchers must develop algorithms that efficiently take advantage of these two factors. The first issue with most metaheuristic algorithms in PV system optimization is determining their particular control parameters. For instance, DE needs the crossover probability and scaling factor, and GA needs crossover and mutation rates. In PSO, the inertia weight and cognitive parameters should be defined. Choosing an incorrect parameter set can distract the user from the main problem and disrupt the algorithms. A parameter-free algorithm, such as TLBO, JAYA, SSA, colliding bodies optimization (CBO), and GWO, makes optimization more effortless and efficient by avoiding adjusting parameters. Note that a parameter-less algorithm in this paper must only determine standard parameters, including the number of iterations (or function evaluations) and initial population size.

Secondly, no single algorithm can solve all optimization problems, according to Wolpert and Macready's "No Free Lunch" theorem [75]. Since the theorem was published 30 years ago, researchers have significantly improved metaheuristic algorithms and developed new ones. Nevertheless, many articles related to improving the performance of these original algorithms show their weakness in effectively and reliably detecting the parameters of different photovoltaic models. Consequently, developing new ideas that result in simple and efficient metaheuristic algorithms without requiring additional parameter settings and modifications for practical optimization problems is necessary.

1.4. Contribution

For the reasons outlined above, this paper proposes a metaheuristic algorithm called APLO for the parameter estimation of PV models. The algorithm is efficient and straightforward. In each iteration of the algorithm, each individual is updated using the current best, last best, and individual's current solutions. The individual moves to a new position if a better solution is obtained or remains unchanged. The process is iterated until all individuals converge on the best solution or a specified criterion is reached. The performance of the proposed APLO for the problems of optimal parameters estimation of PV models is evaluated and compared with five well-known parameterless metaheuristic algorithms, i.e., TLBO, JAYA, SSA, CBO, GWO, and three conventional algorithms such as DE, GA, and PSO. Moreover, in some cases, the related results are also reported from the literature and are compared.

The main contributions of this paper are:

- A novel, simple, and efficient parameterless optimization algorithm is proposed;
- For parameter estimation of PV models, APLO is tested in a series of experiments;
- High accuracy and reliability in finding the PV models' unknown parameters;
- Reasonable performance of the proposed algorithm compared with other original, improved, and hybrid metaheuristic algorithms.

1.5. Paper Structure

The remainder of the paper is organized as follows. The mathematical formulation of the proposed APLO algorithm is presented in Section 2. Detailed descriptions of the single-diode, double-diode, and PV module models are presented in Section 3. Section 4 simulates and evaluates the results of the experiment. Lastly, Section 5 summarizes some concluding remarks.

2. APLO Algorithm

2.1. Mathematical Model

The main steps of the proposed APLO algorithm are mathematically described in this section.

2.1.1. Initialization

In most metaheuristic algorithms, an initial population is randomly generated. Then, this population is improved in the problem's solution space during the iterative process using a proper evolutionary mechanism. APLO is an artificial population-based metaheuristic algorithm that follows the general procedure of evolutionary algorithms such as the DE algorithm. The population consists of *npop* individuals, which *i*th individual is represented by $\boldsymbol{P}_i(t) = [P_{i,1}(t), \ldots, P_{i,D}(t)]|i = 1, 2, \ldots, npop$ at iteration *t*. The initial population can be created as follows:

$$P_{i,j}(t) = p_{\min,j} + rand.(p_{\max,j} - p_{\min,j}); \; \forall t = 1, \forall i = 1, 2, \ldots, npop; \forall j = 1, 2, \ldots, D \quad (1)$$

where $p_{\min,j}$ and $p_{\max}, j$ define the lower and upper bounds of the problem's decision variable j. $rand$ is a uniform distribution value, and D is the dimension of the problem.

2.1.2. Search Operator

After generating the initial population, they are committed to searching around the current best, $P_B(t)$, using the mean knowledge obtained from the last best, $P_L(t)$, to find a better solution. Therefore, for each individual i, the new position of its arbitrary elements, i.e., Υ_i, $P_{i,j}(t+1)$, is updated using the following equation:

$$P_{i,j}(t+1) = P_{B,j}(t) + \frac{r_{1,i}}{r_{2,i}}\left(P_{B,j}(t) - \frac{P_{L,j}(t) + P_{i,j}(t)}{2}\right); \; \forall i = 1, 2, \ldots, npop; \forall j \in \Upsilon_i \quad (2)$$

where r_1, and r_2 are chosen randomly from the normal distribution function. The random value r_1/r_2 is the main factor for balancing the exploration and exploitation phases of the proposed APLO algorithm. An example of this random value over the 1000 iterations is shown in Figure 1. Large values cause $\boldsymbol{P}_i(t+1)$ to leave a feasible space for the problem. Hence, by applying restrictions related to each variable's upper and lower bounds, the violent variables are replaced by new feasible solutions. This preserves diversity in the population during the algorithm process. Thus, it helps the algorithm to search the solutions globally (exploitation phase). In contrast, small random values assist the algorithm in searching locally around the current best solution, helping the exploration phase. It is worth mentioning that an important factor in creating stable behavior in the proposed algorithm is that the updating operation is applied only to a part of the variables of each individual.

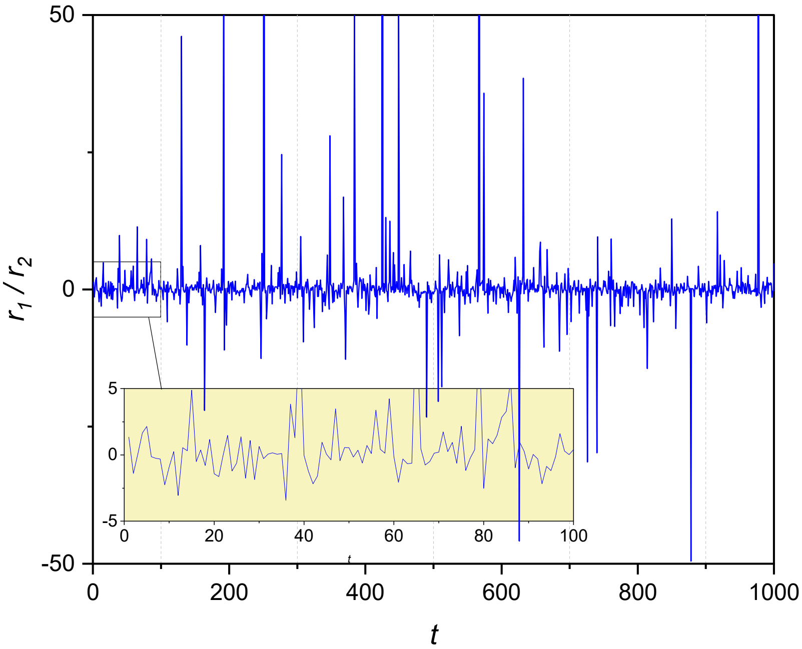

Figure 1. A sample of random value over 1000 iterations.

A representative example of updating the equation in 2D space is shown in Figure 2.

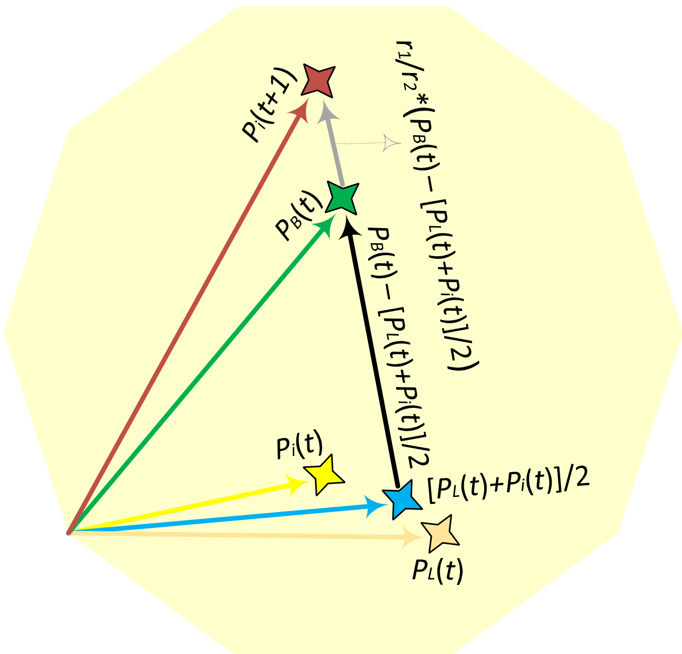

Figure 2. A schematic of position updating in the APLO algorithm.

Some elements of the test vector may exceed their allowed range. Various algorithms may be used to face this challenge. In this article, a simple algorithm is considered for it, such that every element of the new vector that is out of the lower or upper range is replaced with a random value as follows:

$$\begin{array}{l} P_{i,j}(t+1) = p_{\min,j} + rand.(p_{\max,j} - p_{\min,j});\ \text{if } P_{i,j}(t+1) < p_{\min,j} \\ P_{i,j}(t+1) = p_{\min,j} + rand.(p_{\max,j} - p_{\min,j});\ \text{if } P_{i,j}(t+1) < p_{\min,j} \end{array} \quad (3)$$

This algorithm can effectively create diverse solutions during the algorithm process. In addition, random long steps created by r_1/r_2 can contribute to this feature of the algorithm and thus prevent its premature convergence.

2.1.3. Selection

The selection mechanism choices the better solution from the old position $\boldsymbol{P}_i(t)$ and the newly generated position $\boldsymbol{P}_i(t+1)$ based on their fitness values, i.e., $f(.)$. Hence, for instance, the following selection operator is utilized for the minimization problem:

$$\boldsymbol{P}_i(t+1) = \begin{cases} \boldsymbol{P}_i(t+1); & \text{if } f(\boldsymbol{P}_i(t+1)) < f(\boldsymbol{P}_i(t)) \\ \boldsymbol{P}_i(t); & \text{else.} \end{cases} \tag{4}$$

$P_B(t)$ and $P_L(t)$ continually are updated after each function evaluation. The procedure above continues until the termination conditions are met.

The flowchart of the proposed APLO algorithm is outlined in Figure 3, and the pseudocode of the APLO algorithm (Algorithm 1) is defined as follows:

Algorithm 1: The pseudo-code of APLO

Input: Max_{Iter}, $npop$
Output: the best solution
Initialize the $npop$ population randomly using Equation (1)
Calculate the fitness values of all individuals
Determine the current best and last best solutions
$t \leftarrow 1$
While $t < Max\,Iter$ **do**
    for $i = 1 : npop$ **do**
        Update some arbitrary elements of individual i using Equations (2) and (3)
        Calculate the fitness of individual i
        Accept the updated solution if it is better than the old one using Equation (4)
        Update the last best and current best solutions
        $t \leftarrow t+1$
    end
end

2.2. Algorithm Complexity

An algorithm's complexity plays a vital role in assessing its performance. As with all metaheuristic algorithms, APLO requires $\mathcal{O}(n \times npop)$ times to initialize each population time, where n indicates the number of objective functions and $npop$ represents the number of populations. Every algorithm has an $\mathcal{O}(\,Max\,Iter \times fc)$ complexity, where $Max\,Iter$ is the predefined maximum number of iterations. For a given problem, fc represents the complexity of the evaluation function. Simulating the entire process would require $\mathcal{O}(N)$. Accordingly, the algorithm has a computational complexity of $\mathcal{O}(\,N \times Max\,Iter \times fc \times n \times npop)$.

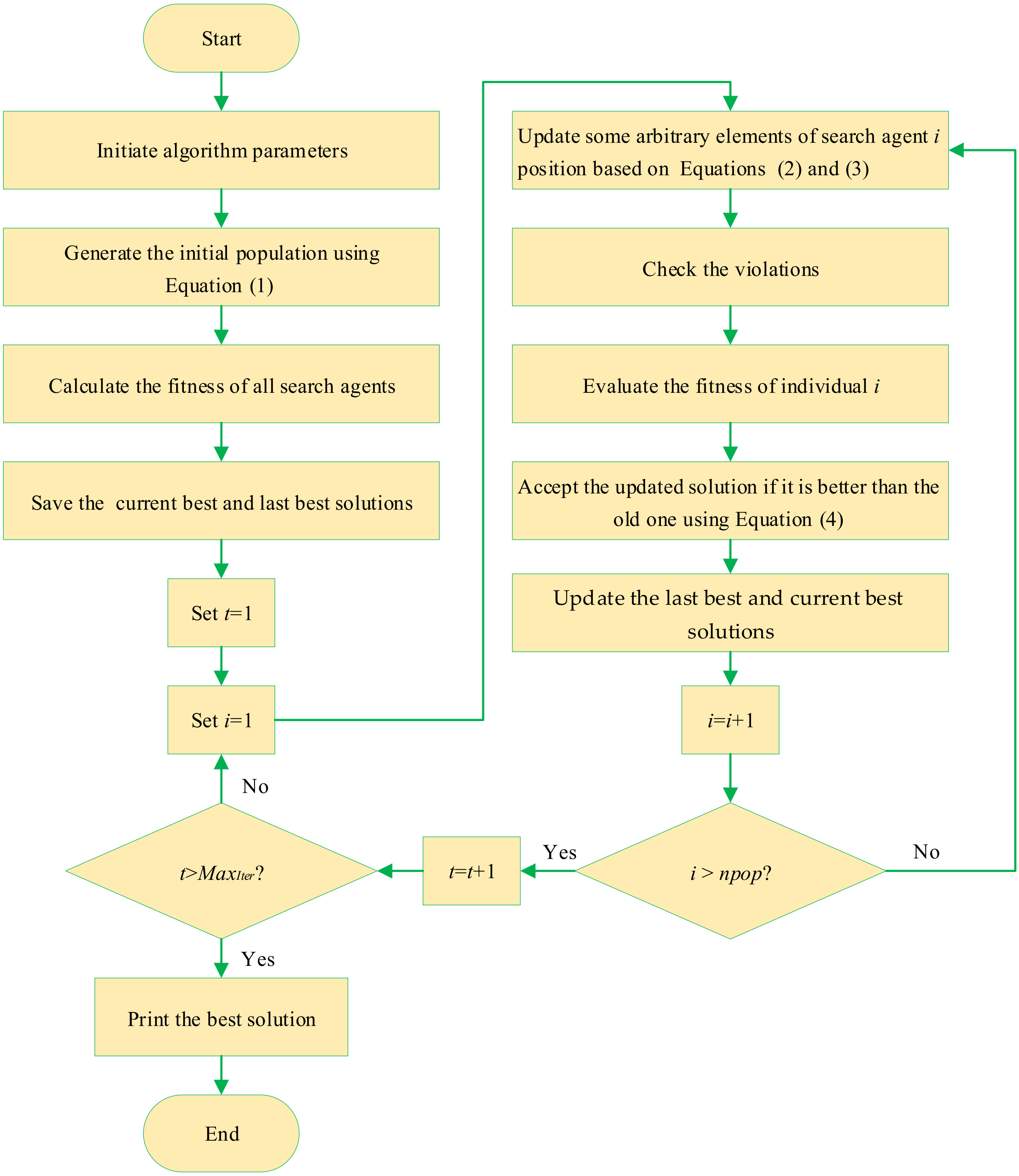

Figure 3. Flowchart of APLO algorithm.

3. The Problem of PV Models' Parameter Extraction

This section introduces three standard models of photovoltaic cells, i.e., SDM, DDM, and PV module models. Then, the mathematical model of the problem of finding the optimal parameters of these three PV models is expressed.

3.1. Single-Diode Model (SDM)

PV systems must be mathematically modeled, considering their practicalities to show their real-time characteristics. PV arrays can be modeled from their basic unit, which is the cell. Due to its simplicity and ease of implementation, the SDM is popular. A parallel resistor, a series resistor, a diode, and a current source make up the equivalent circuit for the single diode model, as shown in Figure 4. To determine the output current, I_{PV}, the following formula can be used [65]:

$$I_{PV} = I_{ph} - (I_{sh} + I_d) \tag{5}$$

where the photogenerated, the diode, and the shunt resistor currents, respectively, are represented by I_{ph}, I_d, and I_{sh}. Shockley's equation and Kirchhoff's voltage law (KVL) can be used to calculate I_d and I_{sh} as follows:

$$I_d = I_{sd}\left[\exp\left(\frac{V_{PV} + R_s I_{PV}}{mv}\right) - 1\right] \tag{6}$$

$$I_{sh} = \frac{V_{PV} + R_s I_{PV}}{R_{sh}} \tag{7}$$

where I_{sd} indicates the diode reverse saturation current; I_{PV} denotes the cell output voltage; R_{sh} and R_s represent the shunt and series resistances, respectively; moreover, the non-physical diode ideality factor is defined by m. v in Equation (6) is the junction thermal voltage that can be expressed as follows:

$$v = \frac{kT}{q} \tag{8}$$

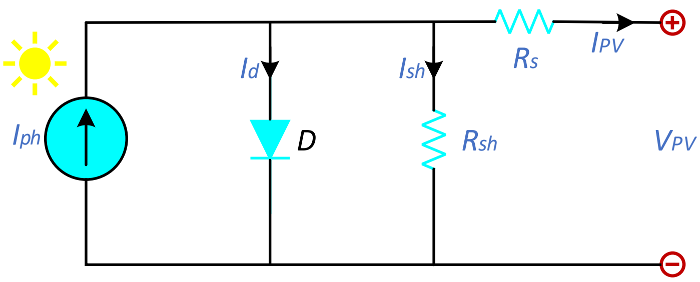

Figure 4. Equivalent circuit of SDM.

Boltzmann's constant, k, is $1.8865033 \times 10^{-23}$ J/K, T is junction temperature, and q is electron charge ($1.60217646 \times 10^{-19}$ C). The output current I_{PV} can be expressed in the following manner by combining Equations (5)–(8):

$$I_{PV} = I_{ph} - \frac{V_{PV} + R_s I_{PV}}{R_{sh}} - I_{sd}\left[\exp\left(\frac{V_{PV} + R_s I_{PV}}{mv}\right) - 1\right] \tag{9}$$

The single-diode model requires the identification of five unknown parameters, that includes I_{ph}, I_{sd}, R_{sh}, R_s, and m.

3.2. Double Diode Model (DDM)

In SDM, recombination current is ignored in the depletion region despite being widely used to simulate PV cells [37]. DDM solves this problem by having three components: a photo-generated current source, a shunt resistance, two rectifying diodes, and a series resistance, as shown in Figure 5.

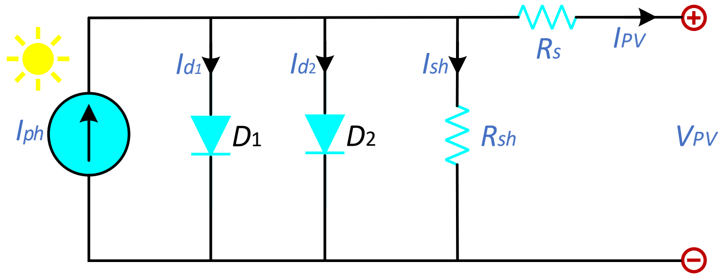

Figure 5. Equivalent circuit of DDM.

As a result of KCL, the output current I_{PV} in DDM can be calculated as follows:

$$I_{PV} = I_{ph} - (I_{sh} + I_{d2} + I_{d1}) \tag{10}$$

As shown in Figure 4, I_{d1} is the current flowing through the first diode (D_1), and I_{d2} is the current flowing through the second diode (D_2). It is also possible to express the magnitude of these currents in terms of the Shockley diode equation as follows:

$$I_{d1} = I_{sd,1}\left[\exp\left(\frac{V_{PV} + R_s I_{PV}}{m_1 v}\right) - 1\right] \tag{11}$$

$$I_{d2} = I_{sd,2}\left[\exp\left(\frac{V_{PV} + R_s I_{PV}}{m_2 v}\right) - 1\right] \tag{12}$$

There are two ideality factors for diodes, m_1 and m_2, as well as diffusion and saturation currents, $I_{sd,1}$ and $I_{sd,2}$, respectively. Hence, Equation (10) can be rewritten as follows by substituting Equations (7), (11) and (12):

$$I_{PV} = I_{ph} - \frac{V_{PV} + R_s I_{PV}}{R_{sh}} - I_{sd,2}\left[\exp\left(\frac{V_{PV} + R_s I_{PV}}{m_2 v}\right) - 1\right] - I_{sd,1}\left[\exp\left(\frac{V_{PV} + R_s I_{PV}}{m_1 v}\right) - 1\right] \tag{13}$$

Compared to SDM, DDM requires more parameters to be identified, i.e., I_{ph}, $I_{sd,2}$, $I_{sd,1}$, R_{sh}, R_s, m_1, and m_2.

3.3. PV Module Model

Multiple PV cells are arranged parallel or in series in PV modules to increase current and voltage. The equivalent circuit model of the PV module is shown in Figure 6. The PV module model output current can be calculated as follows:

$$I_{PV} = N_p I_{ph} - \frac{V_{PV} + [N_s/N_p] R_s I_{PV}}{[N_s/N_p] R_{sh}} - N_p I_{sd}\left[\exp\left(\frac{V_{PV} + [N_s/N_p] R_s I_{PV}}{mv}\right) - 1\right] \tag{14}$$

where N_p represents the number of solar cells in parallel and N_s represents the number of solar cells in series. The PV module has five unknown parameters, similar to the SDM (I_{ph}, I_{sd}, R_{sh}, R_s, and m).

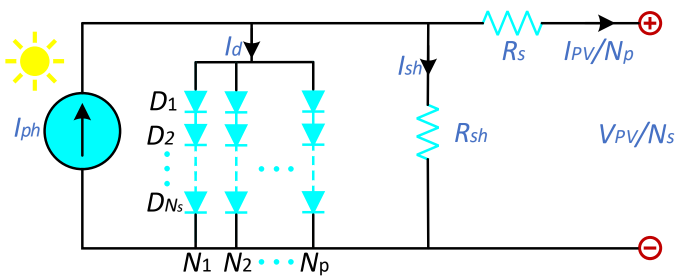

Figure 6. Equivalent circuit of PV module model.

3.4. Problem Formulation of PV Models' Parameters Extraction

Based on measurements of I-V from actual PV cells and PV modules, mathematical models for PV models aim to estimate unknown parameters with remarkable accuracy. Experimental and built I-V data differences are commonly minimized with an optimization technique. It is, therefore, common to consider the minimization of root mean square error (RMSE) between the estimated current ($I_{PV,n}$) and experiment current ($\hat{I}_{PV,n}$) as an objective function:

$$\text{Minimize RMSE} = \left(\frac{1}{N} \sum_{n=1}^{N} \left(\hat{I}_{PV,n} - I_{PV,n} \right)^2 \right)^{1/2} \tag{15}$$

S.t.

$$\underline{I}_{ph} \leq I_{ph} \leq \overline{I}_{ph} \tag{16}$$

$$\underline{R}_s \leq R_s \leq \overline{R}_s \tag{17}$$

$$\underline{R}_{sh} \leq R_{sh} \leq \overline{R}_{sh} \tag{18}$$

$$\underline{I}_{sd} \leq I_{sd} \leq \overline{I}_{sd} \tag{19}$$

$$\underline{m} \leq m \leq \overline{m} \tag{20}$$

$$\underline{I}_{sd} \leq I_{sd,i} \leq \overline{I}_{sd} \tag{21}$$

$$\underline{m} \leq m_i \leq \overline{m} \tag{22}$$

where N is the number of experimental data. $\hat{I}_{PV,n}$ and $I_{PV,n}$ are the nth measured samples and the calculated value of PV output current based on each model. Constraints (16)–(20) indicate the upper and lower limits on the decision variables (unknown parameters) for the SDM and PV module model. Constraints (16)–(18), (21) and (22) indicate the upper and lower limits on the decision variables for the DDM.

In Equation (15), the estimated PV output current at each measured sample n can be calculated using Equations (23)–(25) for SDM, DDM, and PV module models, respectively.

$$I_{PV,n} = I_{ph} - \frac{\hat{V}_{PV,n} + R_s \hat{I}_{PV,n}}{R_{sh}} - I_{sd} \left[\exp\left(\frac{\hat{V}_{PV,n} + R_s \hat{I}_{PV,n}}{mv} \right) - 1 \right] \tag{23}$$

$$I_{PV,n} = I_{ph} - \frac{\hat{V}_{PV,n} + R_s \hat{I}_{PV,n}}{R_{sh}} - I_{sd,2} \left[\exp\left(\frac{\hat{V}_{PV,n} + R_s \hat{I}_{PV,n}}{m_1 v} \right) - 1 \right] - I_{sd,1} \left[\exp\left(\frac{\hat{V}_{PV,n} + R_s \hat{I}_{PV,n}}{m_2 v} \right) - 1 \right] \tag{24}$$

$$I_{PV,n} = N_p I_{ph} - \frac{\hat{V}_{PV,n} + [N_s/N_p] R_s \hat{I}_{PV,n}}{[N_s/N_p] R_{sh}} - N_p I_{sd} \left[\exp\left(\frac{\hat{V}_{PV,n} + [N_s/N_p] R_s \hat{I}_{PV,n}}{mv} \right) - 1 \right] \tag{25}$$

4. Experimental Results

In this section, we evaluate APLO's effectiveness for parameter estimation with three types of PV models: SDM ($N_s = N_p = 1$), DDM ($N_s = 1,\ N_p = 2$), and the PV module ($N_s = 36,\ N_p = 1$). As a standard experiment for SDM and DDM, current–voltage data were collected on silicon solar cells with a diameter of 57 mm (R.T.C. France) [76]. The PV cell characteristics are as follows: $V_{oc} = 0.5728$ V, $I_{sc} = 0.7603$ A, $V_m = 0.4507$ V, and $I_m = 0.6894$ A. In addition, a PV module (Photo Watt-PWP 201) with 36 polycrystalline PV cells is used under 1000 W/m^2 irradiance [76]. This PV module's characteristics are as follows: $V_{oc} = 16.778$ V, $I_{sc} = 1.030$ A, $V_m = 12.649$ V, and $I_m = 0.912$ A. A wide range of algorithms has been developed to estimate the parameters of PV models based on experimental data. Table 2 provides the minimum and maximum limits for PV model parameters [58].

Table 2. Parameters' upper and lower ranges.

Model	Parameters' Limits									
	$\underline{I}_{ph}$ (A)	$\overline{I}_{ph}$ (A)	$\underline{I}_{sd}$ (µA)	$\overline{I}_{sd}$ (µA)	$\underline{m}$	$\overline{m}$	$\underline{R}_s$ (Ω)	$\overline{R}_s$ (Ω)	$\underline{R}_{sh}$ (Ω)	$\overline{R}_{sh}$ (Ω)
SDM	0	1	0	1	1	2	0	0.5	0	100
DDM	0	1	0	1	1	2	0	0.5	0	100
PV module	0	2	0	50	1	50	0	2	0	2000

Moreover, eight well-known algorithms, including DE [17], GA [13], PSO [33], the original parameterless algorithms such as GWO [44], TLBO [77], JAYA [56], SSA [78], CBO [79], are selected to validate and verify the effectiveness of APLO. It is assumed that the population size and the maximum number of iterations are set to 50 and 1000 (i.e., 50,000 evaluations of each function), respectively. The other parameters of the algorithm are maintained as they were in the original literature. To perform the statistical analysis, each algorithm is run 30 times independently in MATLAB 2021b.

4.1. Exploration and Exploitation Analysis

One of the effective factors in creating a balance between exploration and exploitation is ensuring sufficient diversity among individuals. This can prevent an algorithm from getting trapped in locally optimal solutions and result in a better solution. In this section, some experiments are performed to evaluate the exploration–exploitation and diversity in the nine applied metaheuristics on the SDM problem. The percentage of exploration and exploitation, visualizing the two abilities and population diversity in the individuals of the competitive algorithms through the iterations, is shown in Figures 7 and 8, respectively. These numerical measures are calculated based on the procedure reported in [80].

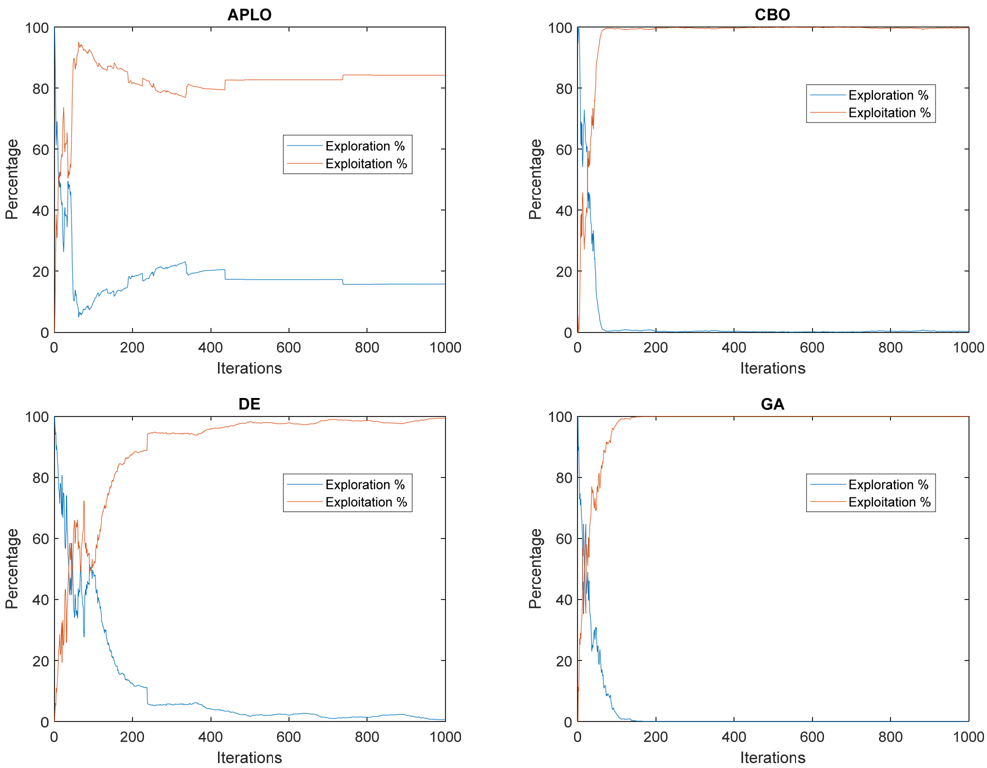

Figure 7. Exploration and exploitation of the competitive algorithms through the iterations on the SDM problem.

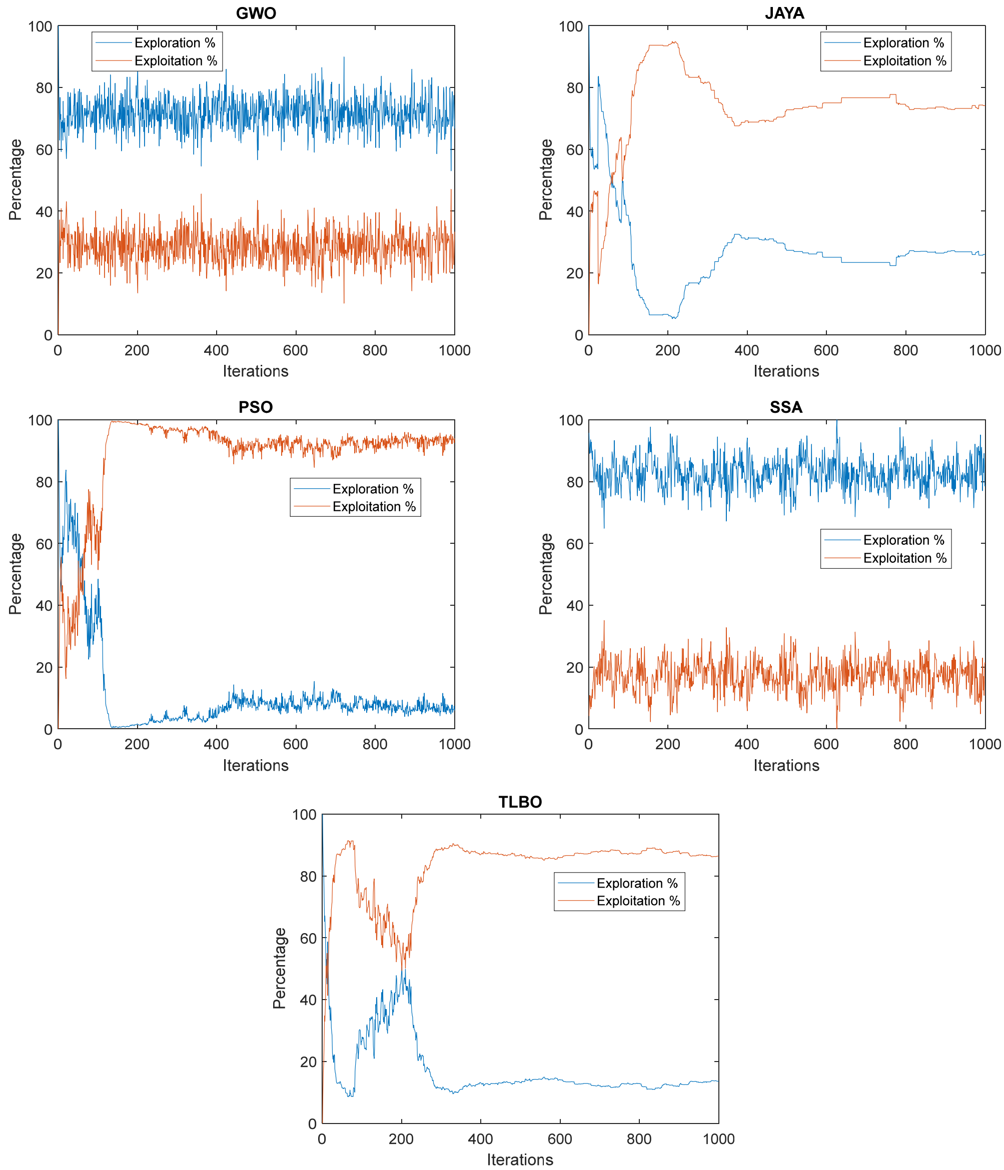

Figure 7. *Cont.*

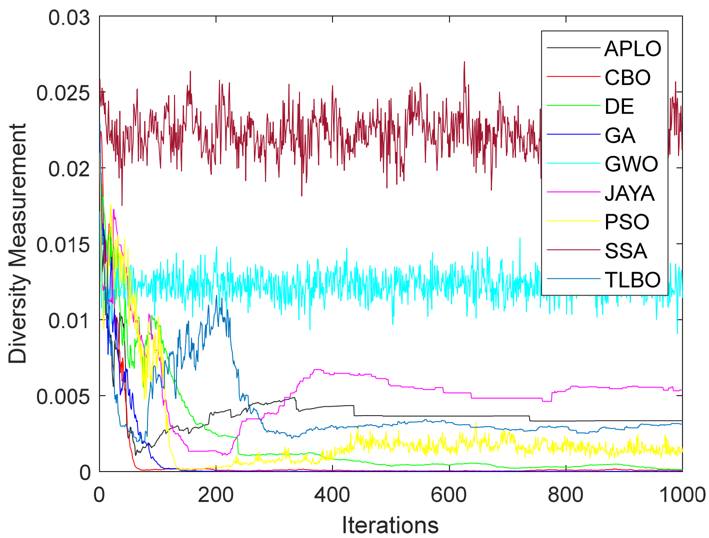

Figure 8. Diversity in individuals of the competitive algorithms on SDM problem.

It can be observed from Figure 7 that, in GWO and SSA algorithms, unlike other algorithms, the percentage of exploration during iterations is much higher than the exploitation. This is evidenced by Figure 9, where the average values of the exploration-exploitation in these two algorithms are 72%:28% and 83%:17%, respectively. As a result, these two algorithms are explorative. Moreover, regarding population diversity, SSA exhibits the highest values during the iterations (see Figure 8). GWO gives the second-highest diversity among the applied algorithms. In contrast, the most exploitation and most minor exploration capabilities are provided by CBO and GA, with mean values of 97% and 3%, respectively. This is further evidenced by the diversity measures illustrated in Figure 8. These two algorithms couldn't provide good diversity throughout the iterations. Hence, premature convergence is the main weakness of these algorithms.

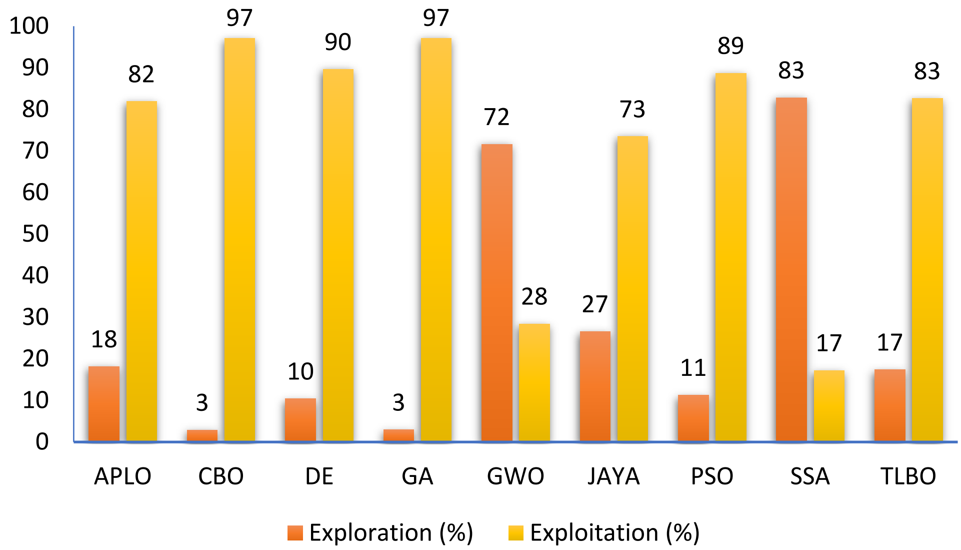

Figure 9. Mean values of exploration and exploitation of competitive algorithms on SDM problem.

Moreover, all algorithms except SSA and GWO, as shown in Figure 7, are explorative at the beginning. After a few iterations, they were threatened as exploitative algorithms. Additionally, the diversity of the population in these algorithms is high initially. After a few iterations, it drops and remains approximately consistent.

The exploration–exploitation and diversity measurements alone cannot show the superiority of an algorithm in solving an optimization problem compared to other algorithms. A better understanding of this goal can be gained by analyzing the characteristic of convergence of these algorithms as shown in Figure 10. The exploration–exploitation balance of 18%:82% in APLO provides the best convergence performance. The final best solution obtained by the proposed algorithm is 9.86×10^{-4}, while the best solution obtained by the TLBO algorithm is 1.20×10^{-3}. TLBO can provide an exploration–exploitation ratio of 17%:83%, which is very close to the proposed algorithm. The explorative algorithms perform the worst performance, i.e., SSA and GWO, which fall into the local optimal after a few iterations.

4.2. Population Size Analysis

The performance of the metaheuristics in solving specific optimization problems can be affected by the population size. Six different populations of 10, 20, 30, 40, 50, and 60 are evaluated for our algorithm while solving the SDM, DDM, and PV module models' parameter identification. The APLO algorithm was run 30 times for each population size independently, and the stop criteria were set at 50,000 function evaluations. The statistical results of this experiment in terms of minimum (Min), average (Mean), maximum (Max), and standard deviation (SD) are summarized in Table 3. Moreover, a Freidman rank test is also applied to compare the algorithm's performance in different cases of the population size. As can be seen form Table 3, the best-suited population size on SDM and DDM problems is *npop* = 40, with mean rank in Freidman of 2.433 and 3.1, respectively. While the population size of 50 exhibits the best when solving the PV module problem with a mean rank of 3.1. Overall, from the last row of Table 3, the sum of Freidman tests over three PV models indicates that the population size of 40 results in the best performance to solve these problems. The population size of 30 and 50 are the second-best and third-best options. However, to perform a fair comparison with other state-of-the-art algorithms, the population size of 50 is adapted in the following sections.

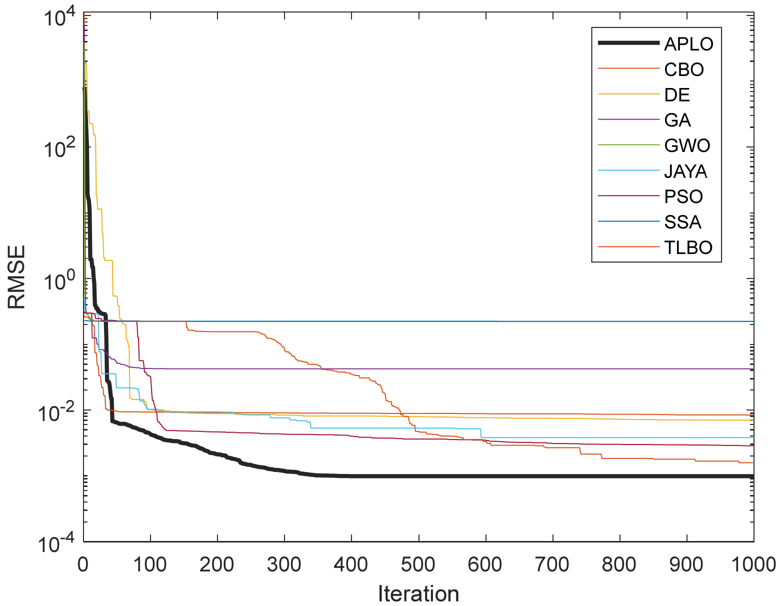

Figure 10. Convergence characteristics of the applied algorithms on SDM problem.

Table 3. Statistical analysis of population size on the performance of APLO for different PV models (the significant values are bolded).

Problem	Measure	*npop* = 10	*npop* = 20	*npop* = 30	*npop* = 40	*npop* = 50	*npop* = 60
SDM	Min	9.860219×10^{-4}	9.860219×10^{-4}	9.860219×10^{-4}	9.860219×10^{-4}	9.860219×10^{-4}	9.860219×10^{-4}
	Mean	9.860219×10^{-4}	9.860219×10^{-4}	9.860219×10^{-4}	9.860219×10^{-4}	9.860219×10^{-4}	9.860219×10^{-4}
	Max	9.860219×10^{-4}	9.860219×10^{-4}	9.860219×10^{-4}	9.860219×10^{-4}	9.860219×10^{-4}	9.860219×10^{-4}
	SD	4.905627×10^{-12}	4.149942×10^{-16}	8.666882×10^{-17}	1.126621×10^{-16}	9.397154×10^{-17}	8.574883×10^{-17}
	Mean rank in Freidman	5.96667	3.61667	2.73333	**2.43333**	3.1	3.15
	Sum rank Freidman	179	108.5	82	**73**	93	94.5
DDM	Min	9.829117×10^{-4}	9.826894×10^{-4}	9.824958×10^{-4}	9.824849×10^{-4}	9.827377×10^{-4}	9.857074×10^{-4}
	Mean	1.008130×10^{-3}	1.006467×10^{-3}	1.003314×10^{-3}	1.011153×10^{-3}	1.013609×10^{-3}	1.022566×10^{-3}
	Max	1.191914×10^{-3}	1.131532×10^{-3}	1.191268×10^{-3}	1.402166×10^{-3}	1.338409×10^{-3}	1.189575×10^{-3}
	SD	4.099853×10^{-5}	3.637408×10^{-5}	4.636394×10^{-5}	7.838701×10^{-5}	7.170668×10^{-5}	5.765975×10^{-5}
	Mean rank in Freidman	3.93333	3.43333	3.16667	**3.1**	3.23333	4.13333
	Sum rank Freidman	118	103	95	**93**	97	124
PV module	Min	9.825837×10^{-4}	9.825044×10^{-4}	9.825115×10^{-4}	9.827521×10^{-4}	9.832545×10^{-4}	9.825992×10^{-4}
	Mean	4.845660×10^{-3}	9.993596×10^{-4}	9.966059×10^{-4}	1.001461×10^{-3}	1.011433×10^{-3}	1.029418×10^{-3}
	Max	1.149226×10^{-1}	1.060000×10^{-3}	1.051849×10^{-3}	1.132161×10^{-3}	1.225400×10^{-3}	1.445858×10^{-3}
	SD	2.079038×10^{-2}	2.217082×10^{-5}	1.878370×10^{-5}	3.269111×10^{-5}	5.876697×10^{-5}	9.490092×10^{-5}
	Mean rank in Freidman	4.66667	3.23333	3.13333	3.3	**3.1**	3.56667
	Sum rank Freidman	140	97	94	99	**93**	107
Sum rank	Mean rank in Freidman	14.56667	10.28333	9.03333	8.83333	9.43333	10.85
	Sum rank Freidman	437	308.5	271	**265**	283	325.5

4.3. Results of Parameter Extraction Based on SDM

In order to investigate the silicon solar cell model developed by RTC France, we analyzed it using SDM. We solved it competitively using nine individual algorithms. Statistics over 30 runs are shown in Figure 11, along with the min, mean, max and SD in Table 4. The bolded values indicate the best results among the applied algorithms.

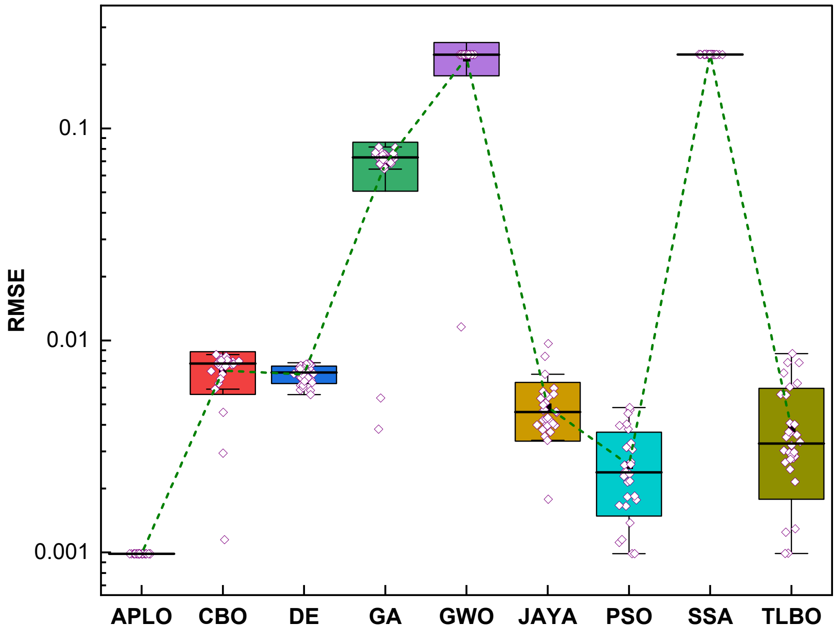

Figure 11. Box chart of different algorithms on SDM parameter extraction over 30 runs.

Table 4. The statistical results of RMSE for SDM (the significant values are bolded).

Algorithm	Min	Mean	Max	SD	Significance
APLO	$\mathbf{9.860218778914 \times 10^{-4}}$	$\mathbf{9.860218778916 \times 10^{-4}}$	$\mathbf{9.860218778922 \times 10^{-4}}$	$\mathbf{1.599419351161 \times 10^{-16}}$	
CBO	$1.148573645374 \times 10^{-3}$	$7.208263932384 \times 10^{-3}$	$8.576997450439 \times 10^{-3}$	$1.645251727229 \times 10^{-3}$	†
DE	$5.548414731899 \times 10^{-3}$	$6.915313985034 \times 10^{-3}$	$7.850952678115 \times 10^{-3}$	$6.541614750785 \times 10^{-4}$	†
GA	$3.817138390879 \times 10^{-3}$	$6.837349221993 \times 10^{-2}$	$8.169353790970 \times 10^{-2}$	$1.780787047086 \times 10^{-2}$	†
GWO	$1.158182213682 \times 10^{-2}$	$2.158244241773 \times 10^{-1}$	$2.228964974584 \times 10^{-1}$	$3.857526966905 \times 10^{-2}$	†
JAYA	$1.782507253178 \times 10^{-3}$	$4.846317911802 \times 10^{-3}$	$9.666422102947 \times 10^{-3}$	$1.496853195820 \times 10^{-3}$	†
PSO	$9.869017992242 \times 10^{-4}$	$2.588954874116 \times 10^{-3}$	$4.826880699904 \times 10^{-3}$	$1.103903013051 \times 10^{-3}$	†
SSA	$2.228658847342 \times 10^{-1}$	$2.230928389927 \times 10^{-1}$	$2.239170090380 \times 10^{-1}$	$2.800981095230 \times 10^{-4}$	†
TLBO	$9.887536713543 \times 10^{-4}$	$3.863042069493 \times 10^{-3}$	$8.677457995748 \times 10^{-3}$	$2.079581847494 \times 10^{-3}$	†

† Indicates APLO has a significant advantage over its competitor when Wilcoxon's rank sum test is performed at 5% confidence.

A glance at the results of Table 4 and the boxplots provided in Figure 11 reveals that the proposed algorithm can achieve better statistical results than other algorithms. For example, the value of the best solution found by the proposed algorithm is 9.860218×10^{-4}, while the best solution of the following algorithm, PSO, is 9.86901×10^{-4}. TLBO algorithm achieves the solutions close to PSO. Also, from the perspective of the robustness of the solutions, as shown in Figure 6 and the standard deviation of Table 4, the superiority of the proposed algorithm over other competitive algorithms is visible. The SSA algorithm is shown to have the worst performance among the applied algorithms. As shown in the last column of Table 4, the proposed algorithm demonstrates superiority over other algorithms when compared using Wilcoxon's rank sum test at a 5% confidence level.

Furthermore, Table 5 contains the best-estimated parameters obtained by APLO and other applied algorithms. As a result of these optimal parameters obtained by APLO, the estimated and measured values of current and power, as well as their individual absolute errors (IAEs), are shown in Figure 7. APLO's simulation provides I-V and P-V characteristics that are highly similar to those of standard data. Figure 12 shows, for instance, that the IAEA of the current range from 2.51×10^{-3} to 8.77×10^{-5}, while the IAEA of the power range from 1.46×10^{-3} to 1.97×10^{-6}, demonstrating that the APLO estimates are highly accurate.

Table 5. The best-identified parameters for SDM using the competitor algorithms (the significant values are bolded).

Algorithm	I_{ph} (A)	I_{sd} (µA)	R_{sh} (Ω)	R_s (Ω)	m	RMSE
APLO	0.7607755	0.3230208	53.71852400	0.0363771	1.4811855	**9.8602188 × 10^{-4}**
CBO	0.7606365	0.4391837	63.84687986	0.0351273	1.5127669	1.1485736 × 10^{-3}
DE	0.7633840	3.5631230	100.0000000	0.0239090	1.7708080	5.5484140 × 10^{-3}
GA	0.7617908	1.8232161	99.99992736	0.0280509	1.6792031	3.8171384 × 10^{-3}
GWO	0.7666861	0.8134494	14.69565125	0.0273766	1.5871288	1.1581822 × 10^{-2}
JAYA	0.7593010	0.5978732	100.0000000	0.0341035	1.5456857	1.7825073 × 10^{-3}
PSO	0.7607664	0.3301579	54.31031901	0.0362896	1.4833891	9.8690180 × 10^{-4}
SSA	0.8361982	0.0000000	1.155093333	0.0000000	2.0000000	2.2286588 × 10^{-1}
TLBO	0.7607049	0.3314945	54.98791923	0.0362715	1.4837785	9.8875367 × 10^{-4}

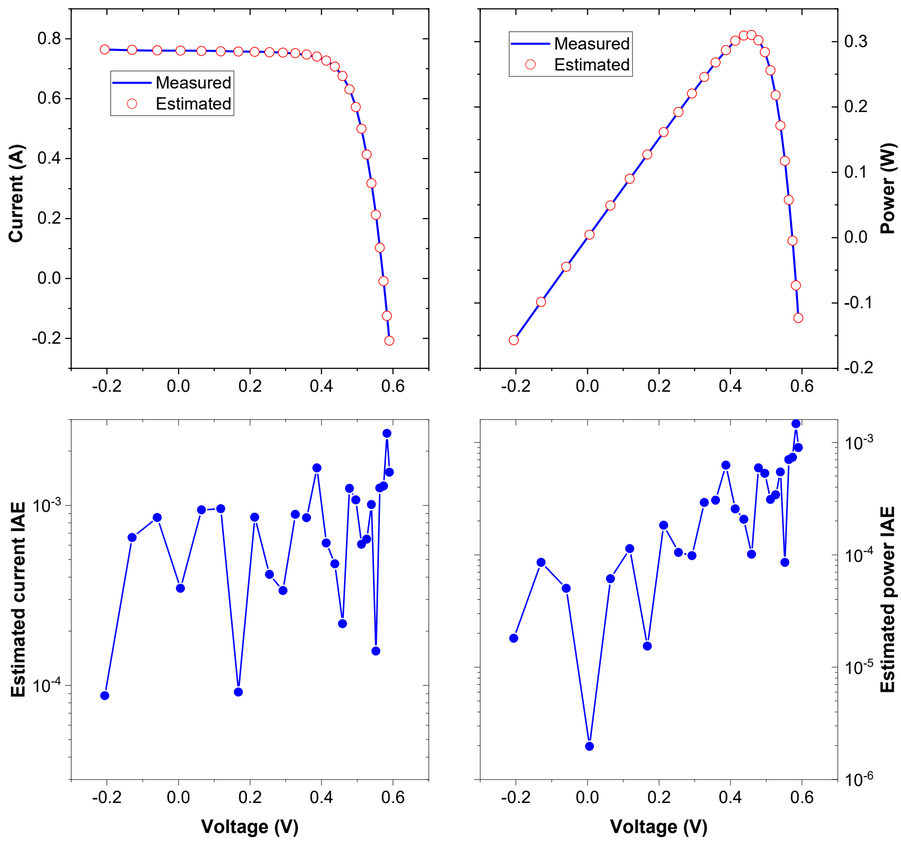

Figure 12. Measured and calculated data of the RTC France silicon solar cell based on SDM by APLO.

4.4. Results of Parameter Extraction Based on DDM

Based on DDM, nine algorithms are used to solve the silicon solar cell model of RTC France. The summary of the statistical results over 30 runs can be found in Figure 13 and Table 6. Among the applied algorithms, the most desirable values are highlighted in bold. As seen in Table 6 and the boxplot provided in Figure 8, the proposed approach outperforms its competitors in terms of statistical results. For example, the value of the best solution found by the proposed algorithm is 9.83065 × 10^{-4}, while the value of the best solution found by the second-best algorithm, i.e., PSO, is 9.84872 × 10^{-4}. As shown in Figure 13 and the fifth column of Table 6, the proposed algorithm has an advantage over other competitive algorithms from a robustness standpoint. Despite this, the DE and SSA show reasonable robustness but fail to find optimal solutions since they fall into the local optimum at every run. A comparison of Table 6 reveals that the SSA has the least effective performance. Moreover, Table 6 demonstrates the superiority of the proposed algorithm over the other algorithms based on a pairwise comparison using Wilcoxon's rank sum test at a 5% confidence level.

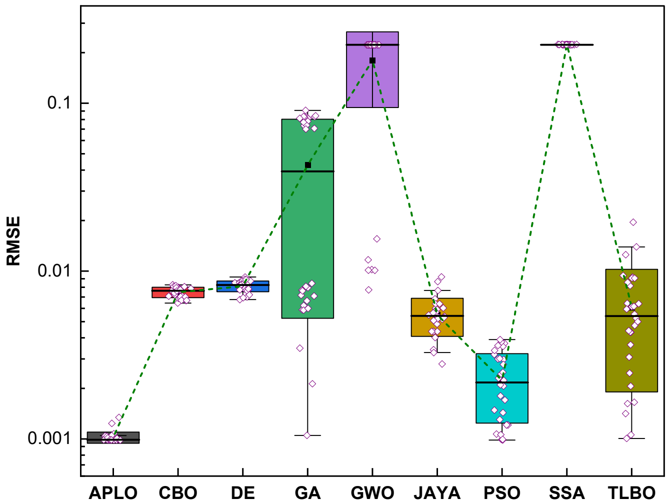

Figure 13. Boxplot comparison of different algorithms for DDM.

Table 6. Statistical results of algorithms for parameter extraction of DDM (the significant values are bolded).

Algorithm	Min	Mean	Max	SD	Significance
APLO	**9.830657938188 × 10^{-4}**	**1.019874828873 × 10^{-3}**	**1.342343716280 × 10^{-3}**	**7.797062995961 × 10^{-5}**	
CBO	6.445682677224 × 10^{-3}	7.474969722257 × 10^{-3}	8.275185158026 × 10^{-3}	5.353607495130 × 10^{-4}	†
DE	6.761178531429 × 10^{-3}	8.130913835822 × 10^{-3}	9.217501072530 × 10^{-3}	6.044793459888 × 10^{-4}	†
GA	1.048914576651 × 10^{-3}	4.277390976996 × 10^{-2}	9.079203590158 × 10^{-2}	3.753851916526 × 10^{-2}	†
GWO	7.723436258756 × 10^{-3}	1.804748010930 × 10^{-1}	2.228793281927 × 10^{-1}	8.624502969932 × 10^{-2}	†
JAYA	2.798207817458 × 10^{-3}	5.479908398673 × 10^{-3}	9.234026428131 × 10^{-3}	1.396786107898 × 10^{-3}	†
PSO	9.848728345415 × 10^{-4}	2.231732852337 × 10^{-3}	3.903661545904 × 10^{-3}	9.893579067548 × 10^{-4}	†
SSA	2.228624210303 × 10^{-1}	2.234264679038 × 10^{-1}	2.243772831227 × 10^{-1}	5.008454232877 × 10^{-4}	†
TLBO	1.006361630316 × 10^{-3}	6.071268460472 × 10^{-3}	1.954518527698 × 10^{-2}	4.166155543866 × 10^{-3}	†

† Indicates APLO has a significant advantage over its competitor when Wilcoxon's rank sum test is performed at 5% confidence.

Furthermore, Table 7 shows the best-estimated parameters for DDM from APLO and other algorithms. Figure 9 shows the calculated and measured values of current and power based on these optimal parameters. According to APLO's simulation, the I-V and P–V characteristics are highly similar to those in the measured data. Figure 14 shows the high accuracy of the estimated parameters by APLO for DDM. It shows the maximum IAE index of current equaling 0.0025 and the maximum IAE of power equaling 0.0015.

Table 7. The best-identified parameters for DDM using the competitor algorithms (the significant values are bolded).

Algorithm	I_{ph} (A)	$I_{sd,1}$ (μA)	$I_{sd,2}$ (μA)	R_s (Ω)	R_{sh} (Ω)	m_1	m_2	RMSE
APLO	0.7607757	0.4697911	0.260965779	0.0365807	54.90462780	1.9999641	1.4631485	**9.8306579382 × 10^{-4}**
CBO	0.7638870	0.2435162	8.040615602	0.0234448	99.98225374	1.5364664	2.0000000	6.4456826772 × 10^{-3}
DE	0.7649050	1.7098070	4.76602770	0.0232610	99.99999000	1.7290370	1.9867510	6.7611785310 × 10^{-3}
GA	0.7605798	0.3884531	0.00000000	0.0356448	60.42384229	1.4999794	1.4754131	1.0489145767 × 10^{-3}
GWO	0.7706357	0.1571963	0.00000000	0.0418987	30.38296958	1.4107497	1.6747318	7.7234362588 × 10^{-3}
JAYA	0.7573180	0.0556882	0.315224634	0.0356605	100.0000000	1.4777275	1.4979513	2.7982078175 × 10^{-3}
PSO	0.7607795	0.2957504	0.145254462	0.0364617	53.99735415	1.4738147	1.9298906	9.8487283454 × 10^{-4}
SSA	0.8377229	0.0000000	0.000000000	0.0000000	1.145114681	1.0000000	2.0000000	2.2286242103 × 10^{-1}
TLBO	0.7606820	0.3543379	0.332865555	0.0360861	56.648982886	1.4906829	1.0000000	1.0063616303 × 10^{-3}

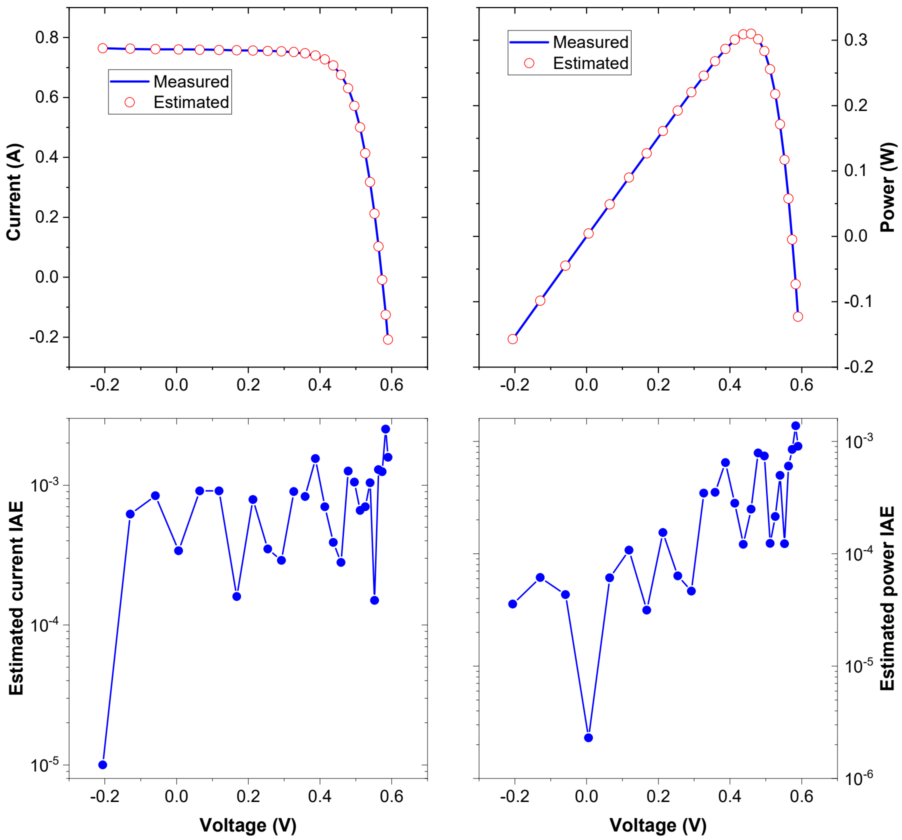

Figure 14. Measured and calculated data of the RTC France silicon solar cell based on DDM by APLO.

4.5. PV Module Model-Based Photo Watt-PWP 201

This section uses the PV module model to model the Photo Watt-PWP 201 solar cell. In contrast, nine algorithms are used to solve the model. Based on 30 runs of this model, Figure 15 and Table 8 illustrate the statistical results of these algorithms. Figure 15 provides box plots showing how the proposed algorithm outperforms other competitor algorithms in terms of statistical performance. In this study, the APLO algorithm provided the best RMSE at $2.42507486809 \times 10^{-3}$, followed by PSO at $2.47057268706 \times 10^{-3}$. Based on the SD results in the fifth column of Table 8, the proposed algorithm appears more reliable than other competitive algorithms. DE obtained the second-best SD value, $7.13076666106 \times 10^{-4}$, while APLO obtained the best SD value, $5.96208271747 \times 10^{-17}$.

Table 8. Statistical results of algorithms for parameter extraction of PV module (the significant values are bolded).

Algorithm	Min	Mean	Max	SD	Significance
APLO	$\mathbf{2.42507486809 \times 10^{-3}}$	$\mathbf{2.42507486810 \times 10^{-3}}$	$\mathbf{2.42507486810 \times 10^{-3}}$	$\mathbf{5.96208271747 \times 10^{-17}}$	
CBO	$2.59323307598 \times 10^{-3}$	$9.58546210390 \times 10^{-3}$	$1.44726212656 \times 10^{-2}$	$1.62949321579 \times 10^{-3}$	†
DE	$6.67237575292 \times 10^{-3}$	$7.74335013025 \times 10^{-3}$	$9.59388984237 \times 10^{-3}$	$7.13076666106 \times 10^{-4}$	†
GA	$5.22983393188 \times 10^{-3}$	$1.72642387255 \times 10^{-1}$	$3.51706994835 \times 10^{-1}$	$1.17731216363 \times 10^{-1}$	†
GWO	$9.30534798242 \times 10^{-3}$	$3.03524536723 \times 10^{-2}$	$1.09058464220 \times 10^{-1}$	$2.40958661843 \times 10^{-2}$	†
JAYA	$3.42058453137 \times 10^{-3}$	$1.63022106280 \times 10^{-2}$	$7.64341081931 \times 10^{-2}$	$2.27085786595 \times 10^{-2}$	†
PSO	$2.47057268706 \times 10^{-3}$	$5.06269373279 \times 10^{-3}$	$6.76745897635 \times 10^{-3}$	$1.11580692510 \times 10^{-3}$	†
SSA	$5.36253332910 \times 10^{-2}$	$1.45064970492 \times 10^{-1}$	$2.75832201568 \times 10^{-1}$	$8.99178958837 \times 10^{-2}$	†
TLBO	$2.81460265874 \times 10^{-3}$	$4.17330355891 \times 10^{-3}$	$8.12934998713 \times 10^{-3}$	$1.16062006801 \times 10^{-3}$	†

† Indicates APLO has a significant advantage over its competitor when Wilcoxon's rank sum test is performed at 5% confidence.

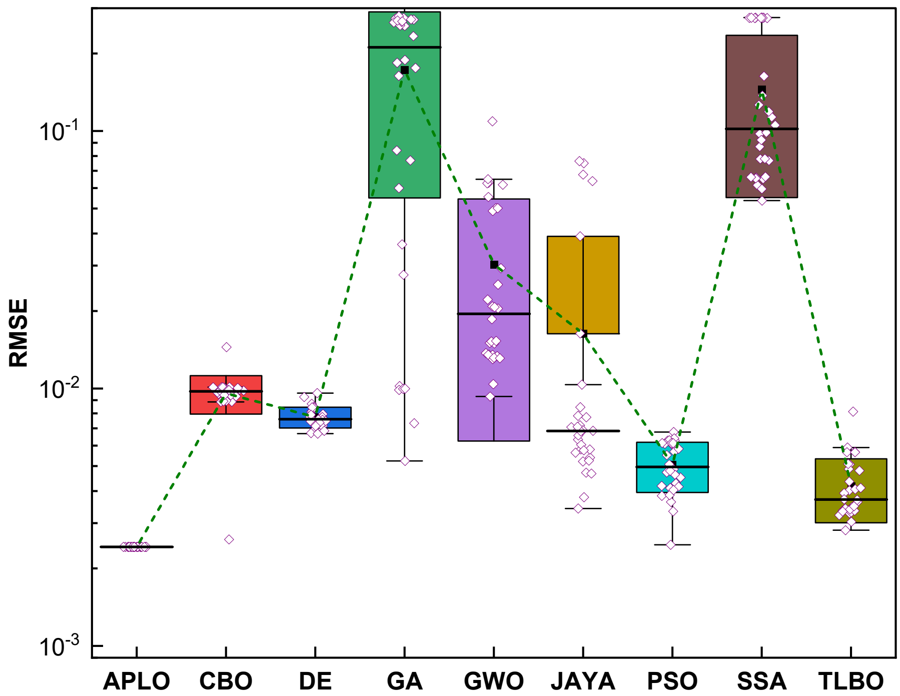

Figure 15. Boxplot comparison of different algorithms for PV module.

In contrast to other applied algorithms, this reasonable difference in SD indicates that the APLO algorithm is significantly more robust than different algorithms. The GA has the worst SD performance of the applied algorithms, and the SSA has the worst Min value performance. Moreover, SSA exhibits the worst performance among all algorithms. Table 8's last column shows that the proposed algorithm is superior to other algorithms when compared pairwise using Wilcoxon's rank sum test at a 5% confidence level.

Further analysis of the best-estimated parameters of the PV module model generated by APLO and other applied algorithms is presented in Table 9. The estimated and measured current and power, along with their IAE, are displayed in Figure 16 using these optimal parameters. Based on the simulation results obtained by APLO, the I-V and P-V characteristics are very similar to those measured. In this case, the IAE of current and power are less than 0.006 and 0.0799, respectively. This demonstrates that the estimated parameters by APLO for the PV module model are highly accurate.

Table 9. The best-identified parameters for PV module using the competitor algorithms (the significant values are bolded).

Algorithm	I_{ph}	I_{sd}	R_{sh}	R_s	m	RMSE
APLO	1.0305143	3.48226289	27.2772845	0.0333686	1.3511916	$\mathbf{2.4250749 \times 10^{-3}}$
CBO	1.0287009	4.84455139	42.7195703	0.0323866	1.3872184	2.5932331×10^{-3}
DE	1.0306156	22.1690238	1999.99959	0.0264998	1.5826458	6.6723758×10^{-3}
GA	1.0284036	15.1965837	1427.98876	0.0281702	1.5290681	5.2298339×10^{-3}
GWO	1.0329527	14.2377500	798.639484	0.0271184	1.5191182	9.3053480×10^{-3}
JAYA	1.0247885	7.90093672	2000.00000	0.0309492	1.4441778	3.4205845×10^{-3}
PSO	1.0294489	4.11606748	33.9829867	0.0328684	1.3691868	2.4705727×10^{-3}
SSA	1.1375146	50.0000000	1.64110086	0.0025254	1.7459220	5.3625333×10^{-2}
TLBO	1.0288704	5.50233338	47.4894427	0.0318549	1.4016305	2.8146027×10^{-3}

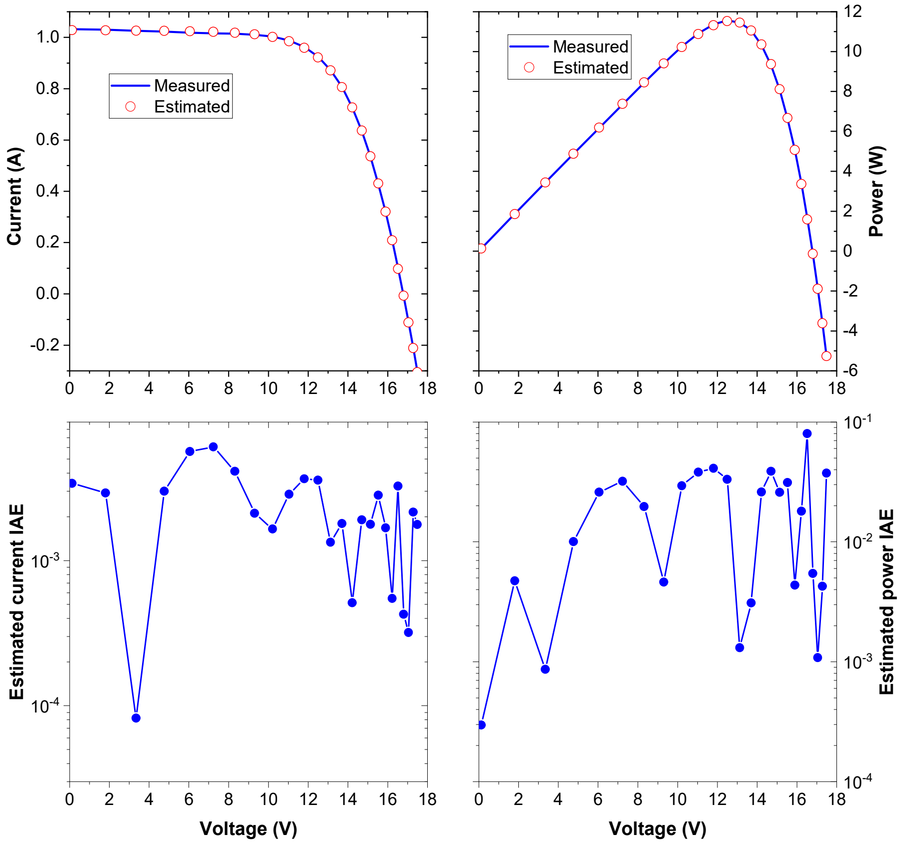

Figure 16. Comparison between the measured and calculated data yielded by APLO for the PV module model based on Photo Watt-PWP 201.

4.6. Comprehensive Comparison

4.6.1. Convergence Characteristics

A comparison of the convergence curves of different algorithms is shown in Figures 17–19 for SDM, DDM, and PV module models under the best run condition (e.g., achieving the “Min” solution). When solving the SDM and PV module models, APLO, CBO, and GA show the fastest convergence rates in the first 50 iterations. Unlike the proposed algorithm (APLO), CBO, and GA come close to obtaining locally optimal solutions after iteration 200. The convergence performance of CBO and GA is insufficient regarding DDM. PSO and APLO, on the other hand, provide better convergence performance than other algorithms used for DDM. However, APLO can converge to a better final solution than PSO in this model after iteration 500. Within the first 500 iterations, the convergence speeds of GA and PSO for the PV module model are faster than APLO. However, APLO can achieve a better final solution than GA and PSO.

It appears that APLO offers an appropriate balance between exploration and exploitation based on the convergence curves for the three models. It should be noted that this comparison is based on the convergence curves presented by the best runs of the applied algorithms. On the other runs, other algorithms are more likely to get stuck in local optimal points and exhibit significantly weaker convergence performance than the proposed algorithm.

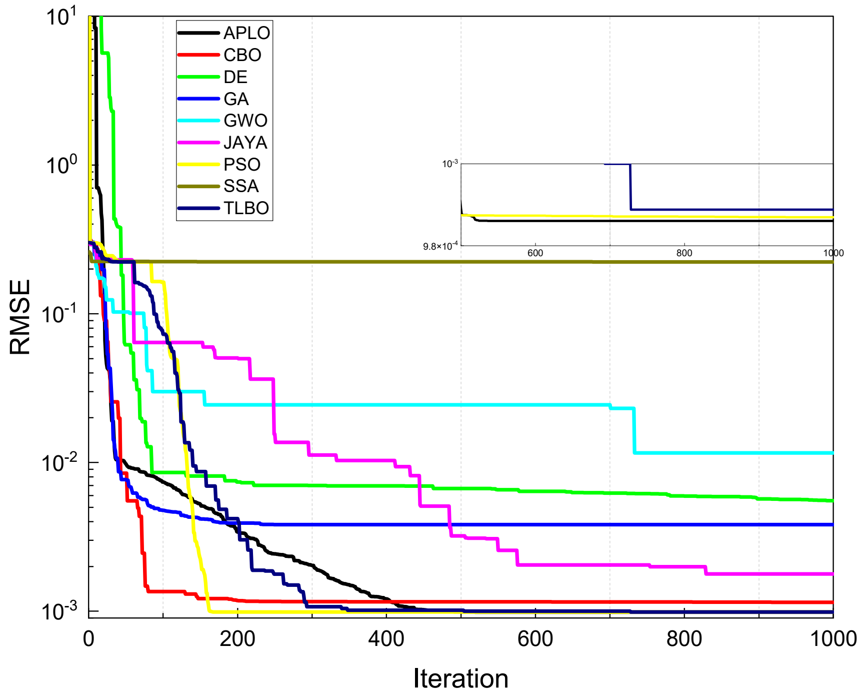

Figure 17. Convergence curves of the competitor algorithms for SDM.

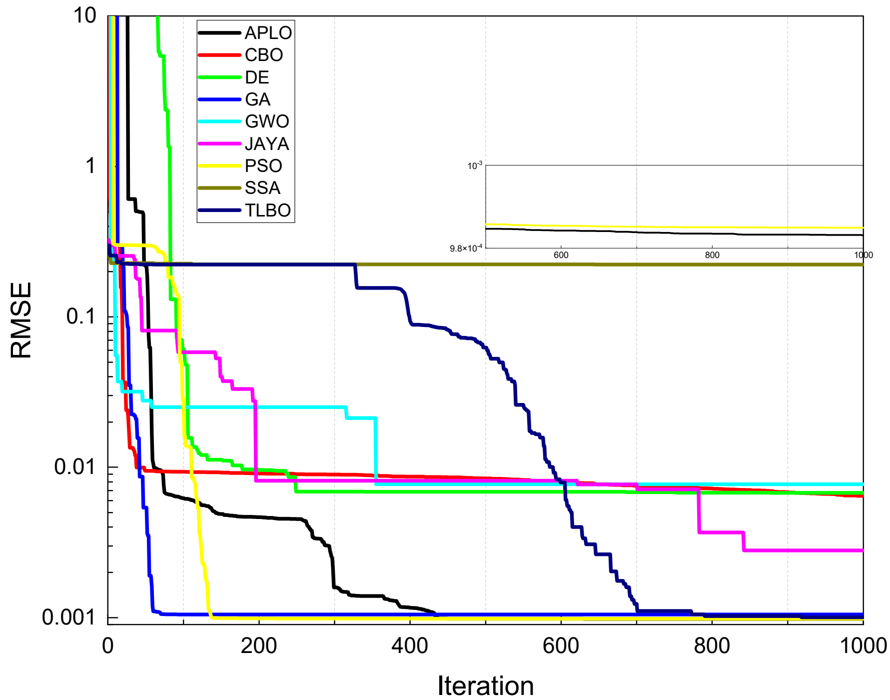

Figure 18. Convergence curves of the competitor algorithms for DDM.

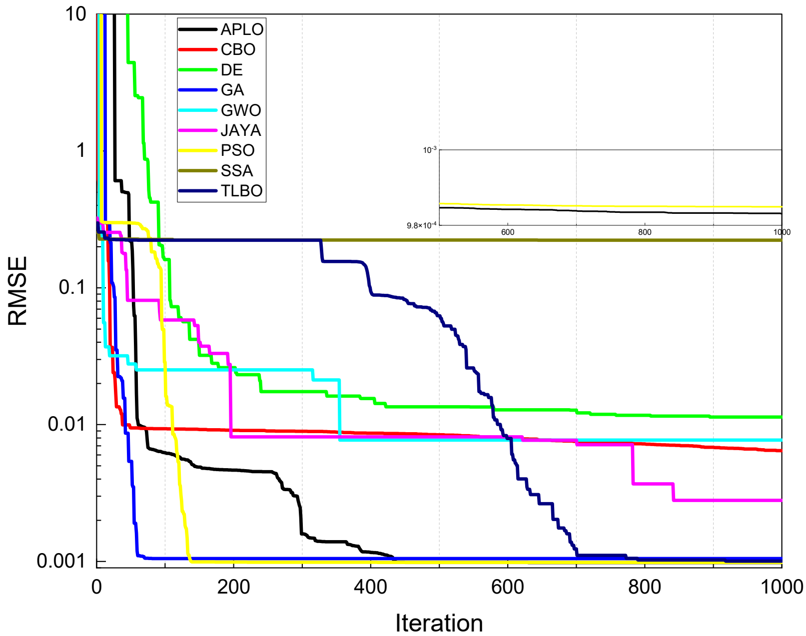

Figure 19. Convergence curves of the competitor algorithms for the PV module model.

4.6.2. Computational Time

There is little difference between APLO, CBO, GWO, JAYA, and SSA regarding their computational time when solving three models, as shown in Figures 20–22. They are also faster than DE, GA, and PSO algorithms. The CBO offers minor computational costs, whereas the TLBO is the most complex parameterless algorithm. It is necessary to teach and learn at different times for each generation of TLBO, which results in two different function evaluations (FEs) for each participant. TLBO is thus more expensive to compute in a single generation than an algorithm with one FE per generation. Compared to parameterless algorithms, GA, DE, and PSO require more computational time. PSO is ranked second in this regard, and GA consumes the most time.

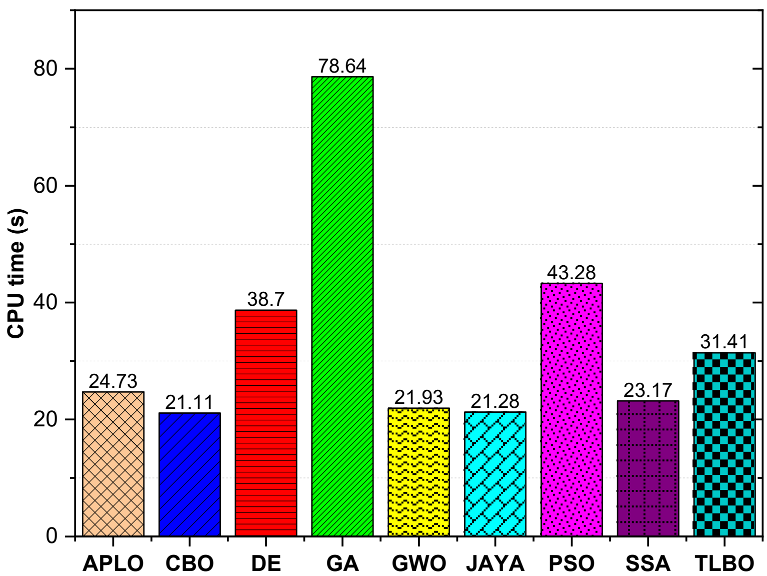

Figure 20. CPU time comparison in solving parameter extraction of SDM by different algorithms over 30 runs.

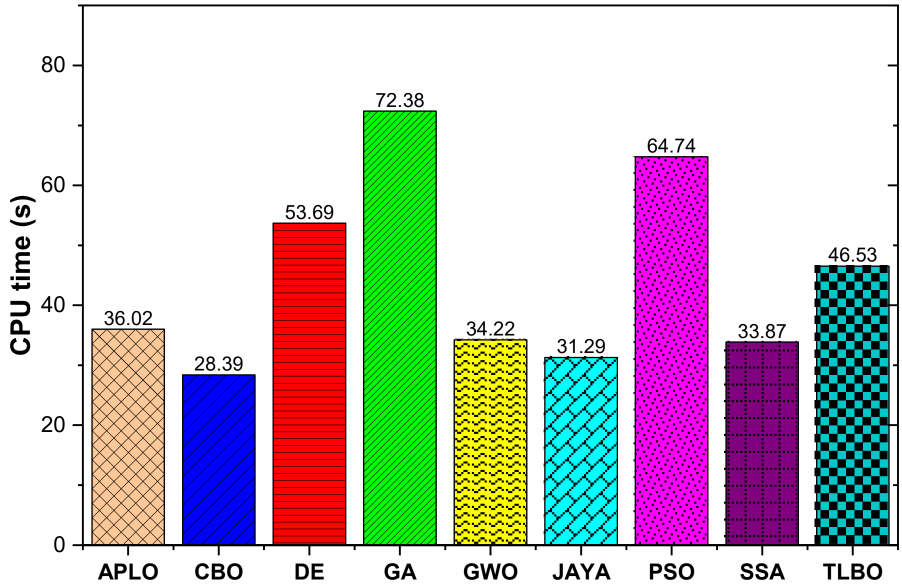

Figure 21. CPU time comparison in solving parameter extraction of DDM by different algorithms over 30 runs.

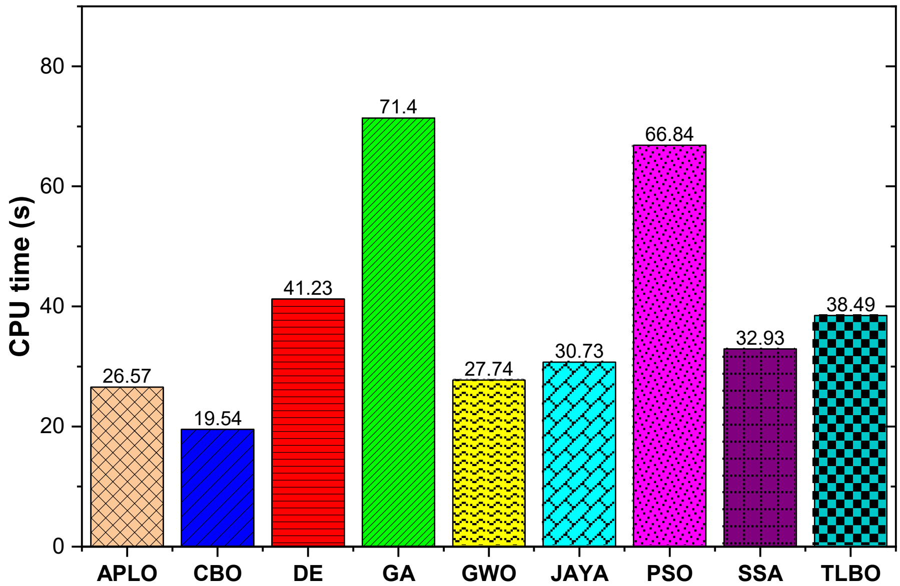

Figure 22. CPU time comparison in solving parameter extraction of PV module model by different algorithms over 30 runs.

4.6.3. Wilcoxon and Friedman Tests

Different algorithms are commonly tested for significant differences in results using the Wilcoxon signed-rank test [13,50,52]. Based on 30 runs of each algorithm, Wilcoxon signed-rank tests were performed. Tables 10–12 summarize the results.

In these tables, "mean_PRs"/"sum_PRs"/"N_PRs" and "mean_NRs"/" sum_NRs"/"N_NRs", respectively, indicate the mean/sum/number of ranks for the problem in which Algorithm I outperformed Algorithm II. The mean/sum/number of ranks for the problem in which Algorithm II outperformed Algorithm I. *p*-values represent the degree of significance, where a smaller value indicates a more significant difference. Based on the comparison of all models considered in Table 13, it is evident that the proposed APLO differs significantly from the compared algorithms.

Based on the Freidman test, APLO takes the top ranking for three models, demonstrating that the proposed algorithm is superior to those compared. PSO and TLBO rank second and third in average rankings, respectively. This algorithm offers a promising approach to extracting accurate parameters for various solar cells. Aside from this, SSA ranks lowest.

Table 10. Wilcoxon ranks test results for SDM.

Algorithm I	Algorithm II	mena_NRs	mean_PRs	sum_NRs	sum_PRs	N_NRs	N_PRs	*p*-Value
APLO	CBO	15.5	NaN	465	0	30	0	1.8627×10^{-9}
APLO	DE	15.5	NaN	465	0	30	0	1.8627×10^{-9}
APLO	GA	15.5	NaN	465	0	30	0	1.8627×10^{-9}
APLO	GWO	15.5	NaN	465	0	30	0	1.8627×10^{-9}
APLO	JAYA	15.5	NaN	465	0	30	0	1.8627×10^{-9}
APLO	PSO	15.5	NaN	465	0	30	0	1.8627×10^{-9}
APLO	SSA	15.5	NaN	465	0	30	0	1.8627×10^{-9}
APLO	TLBO	15.5	NaN	465	0	30	0	1.8627×10^{-9}
CBO	DE	17.1	15	120	345	7	23	1.9660×10^{-2}
CBO	GA	16.5	1.5	462	3	28	2	9.3132×10^{-9}
CBO	GWO	15.5	NaN	465	0	30	0	1.8627×10^{-9}
CBO	JAYA	7.0	17.2	35	430	5	25	7.9945×10^{-6}
CBO	PSO	2.0	16.5	4	461	2	28	1.3039×10^{-8}
CBO	SSA	15.5	NaN	465	0	30	0	1.8627×10^{-9}
CBO	TLBO	9.0	16.0	18	447	2	28	4.7125×10^{-7}
DE	GA	16.5	1.5	462	3	28	2	9.3132×10^{-9}
DE	GWO	15.5	NaN	465	0	30	0	1.8627×10^{-9}
DE	JAYA	13.5	15.6	27	438	2	28	2.3488×10^{-6}
DE	PSO	NaN	15.5	0	465	0	30	1.8627×10^{-9}
DE	SSA	15.5	NaN	465	0	30	0	1.8627×10^{-9}
DE	TLBO	3.8	17.3	15	450	4	26	2.5518×10^{-7}
GA	GWO	16.0	1	464	1	29	1	3.7253×10^{-9}
GA	JAYA	1.0	16.0	1	464	1	29	3.7253×10^{-9}
GA	PSO	NaN	15.5	0	465	0	30	1.8627×10^{-9}
GA	SSA	15.5	NaN	465	0	30	0	1.8627×10^{-9}
GA	TLBO	NaN	15.5	0	465	0	30	1.8627×10^{-9}
GWO	JAYA	NaN	15.5	0	465	0	30	1.8627×10^{-9}
GWO	PSO	NaN	15.5	0	465	0	30	1.8627×10^{-9}
GWO	SSA	15.5	NaN	465	0	30	0	1.8627×10^{-9}
GWO	TLBO	NaN	15.5	0	465	0	30	1.8627×10^{-9}
JAYA	PSO	2.5	16.4	5	460	2	28	1.8627×10^{-8}
JAYA	SSA	15.5	NaN	465	0	30	0	1.8627×10^{-9}
JAYA	TLBO	17.4	14.9	122	343	7	23	2.2100×10^{-2}
PSO	SSA	15.5	NaN	465	0	30	0	1.8627×10^{-9}
PSO	TLBO	17.7	11.2	353	112	20	10	1.2050×10^{-2}
SSA	TLBO	NaN	15.5	0	465	0	30	1.8627×10^{-9}

Table 11. Wilcoxon ranks test results for DDM.

Algorithm I	Algorithm II	mena_NRs	mean_PRs	sum_NRs	sum_PRs	N_NRs	N_PRs	*p*-Value
APLO	CBO	15.5	NaN	465	0	30	0	1.8627×10^{-9}
APLO	DE	15.5	NaN	465	0	30	0	1.8627×10^{-9}
APLO	GA	15.5	NaN	465	0	30	0	1.8627×10^{-9}
APLO	GWO	15.5	NaN	465	0	30	0	1.8627×10^{-9}
APLO	JAYA	15.5	NaN	465	0	30	0	1.8627×10^{-9}
APLO	PSO	17.3	3.75	450	15	26	4	2.5518×10^{-7}
APLO	SSA	15.5	NaN	465	0	30	0	1.8627×10^{-9}
APLO	TLBO	16.0	1	464	1	29	1	3.7253×10^{-9}
CBO	DE	17.0	9.5	408	57	24	6	1.2334×10^{-4}
CBO	GA	18.8	9	375	90	20	10	2.5600×10^{-3}
CBO	GWO	16.0	1	464	1	29	1	3.7253×10^{-9}
CBO	JAYA	10.0	16.1	30	435	3	27	3.7905×10^{-6}
CBO	PSO	NaN	15.5	0	465	0	30	1.8627×10^{-9}
CBO	SSA	15.5	NaN	465	0	30	0	1.8627×10^{-9}
CBO	TLBO	13.2	16.5	119	346	9	21	1.8530×10^{-2}
DE	GA	20.8	8.5	354	111	17	13	1.1300×10^{-2}
DE	GWO	16.0	1	464	1	29	1	3.7253×10^{-9}

Table 11. *Cont.*

Algorithm I	Algorithm II	mena_NRs	mean_PRs	sum_NRs	sum_PRs	N_NRs	N_PRs	*p*-Value
DE	JAYA	4.5	16.3	9	456	2	28	6.1467×10^{-8}
DE	PSO	NaN	15.5	0	465	0	30	1.8627×10^{-9}
DE	SSA	15.5	NaN	465	0	30	0	1.8627×10^{-9}
DE	TLBO	10.9	17.5	98	367	9	21	4.6600×10^{-3}
GA	GWO	16.2	5.5	454	11	28	2	1.0245×10^{-7}
GA	JAYA	11.3	16.0	34	431	3	27	6.9179×10^{-6}
GA	PSO	2.0	16.0	2	463	1	29	5.5879×10^{-9}
GA	SSA	15.5	NaN	465	0	30	0	1.8627×10^{-9}
GA	TLBO	8.7	17.2	52	413	6	24	7.0568×10^{-5}
GWO	JAYA	NaN	15.5	0	465	0	30	1.8627×10^{-9}
GWO	PSO	NaN	15.5	0	465	0	30	1.8627×10^{-9}
GWO	SSA	16.0	1	464	1	29	1	3.7253×10^{-9}
GWO	TLBO	2.0	16.0	2	463	1	29	5.5879×10^{-9}
JAYA	PSO	NaN	15.5	0	465	0	30	1.8627×10^{-9}
JAYA	SSA	15.5	NaN	465	0	30	0	1.8627×10^{-9}
JAYA	TLBO	16.5	14.5	247	218	15	15	7.7657×10^{-1}
PSO	SSA	15.5	NaN	465	0	30	0	1.8627×10^{-9}
PSO	TLBO	16.8	7	437	28	26	4	2.7623×10^{-6}
SSA	TLBO	NaN	15.5	0	465	0	30	1.8627×10^{-9}

Table 12. Wilcoxon ranks test results for the PV module model.

Algorithm I	Algorithm II	mena_NRs	mean_PRs	sum_NRs	sum_PRs	N_NRs	N_PRs	*p*-Value
APLO	CBO	15.5	NaN	465	0	30	0	1.86265×10^{-9}
APLO	DE	15.5	NaN	465	0	30	0	1.86265×10^{-9}
APLO	GA	15.5	NaN	465	0	30	0	1.86265×10^{-9}
APLO	GWO	15.5	NaN	465	0	30	0	1.86265×10^{-9}
APLO	JAYA	15.5	NaN	465	0	30	0	1.86265×10^{-9}
APLO	PSO	15.5	NaN	465	0	30	0	1.86265×10^{-9}
APLO	SSA	15.5	NaN	465	0	30	0	1.86265×10^{-9}
APLO	TLBO	15.5	NaN	465	0	30	0	1.86265×10^{-9}
CBO	DE	15.5	15.5	31	434	2	28	4.42192×10^{-6}
CBO	GA	16.21429	5.5	454	11	28	2	1.02446×10^{-7}
CBO	GWO	15.5	NaN	465	0	30	0	1.86265×10^{-9}
CBO	JAYA	23.71429	13	166	299	7	23	0.17719
CBO	PSO	8	15.75862	8	457	1	29	4.65661×10^{-8}
CBO	SSA	15.5	NaN	465	0	30	0	1.86265×10^{-9}
CBO	TLBO	2	15.96552	2	463	1	29	5.58794×10^{-9}
DE	GA	16.5	1.5	462	3	28	2	9.31323×10^{-9}
DE	GWO	15.5	NaN	465	0	30	0	1.86265×10^{-9}
DE	JAYA	20.2	13.15	202	263	10	20	0.54253
DE	PSO	NaN	15.5	0	465	0	30	1.86265×10^{-9}
DE	SSA	15.5	NaN	465	0	30	0	1.86265×10^{-9}
DE	TLBO	1	16	1	464	1	29	3.72529×10^{-9}
GA	GWO	4.85714	18.73913	34	431	7	23	6.91786×10^{-6}
GA	JAYA	7.5	16.07143	15	450	2	28	2.55182×10^{-7}
GA	PSO	1	16	1	464	1	29	3.72529×10^{-9}
GA	SSA	12	19.5	192	273	16	14	0.41613
GA	TLBO	NaN	15.5	0	465	0	30	1.86265×10^{-9}
GWO	JAYA	20.33333	14.29167	122	343	6	24	0.0221
GWO	PSO	NaN	15.5	0	465	0	30	1.86265×10^{-9}
GWO	SSA	15.93103	3	462	3	29	1	9.31323×10^{-9}
GWO	TLBO	NaN	15.5	0	465	0	30	1.86265×10^{-9}
JAYA	PSO	6.83333	17.66667	41	424	6	24	1.82446×10^{-5}
JAYA	SSA	16.5	1.5	462	3	28	2	9.31323×10^{-9}
JAYA	TLBO	8	16.33333	24	441	3	27	1.41934×10^{-6}
PSO	SSA	15.5	NaN	465	0	30	0	1.86265×10^{-9}
PSO	TLBO	12.25	16.68182	98	367	8	22	0.00466
SSA	TLBO	NaN	15.5	0	465	0	30	1.86265×10^{-9}

Table 13. Friedman ranks test results for three models (the significant values are bolded).

Algorithm	SDM		DDM		PV Module Model	
	Mean Rank	Sum Rank	Mean Rank	Sum Rank	Mean Rank	Sum Rank
APLO	**1.0000**	**30**	**1.1667**	**35**	**1.0000**	**30**
CBO	5.5333	166	5.1667	155	5.7000	171
DE	5.1000	153	5.9000	177	4.7667	143
GA	6.8667	206	5.9667	179	8.0000	240
GWO	7.9667	239	7.8667	236	7.0667	212
JAYA	3.9667	119	3.7667	113	4.6000	138
PSO	2.4667	74	2.0333	61	3.0000	90
SSA	9.0000	270	8.9667	269	8.4333	253
TLBO	3.1000	93	4.1667	125	2.4333	73

4.6.4. Advantages and Disadvantages of Applied Algorithms

Based on the studies conducted so far, the advantages and disadvantages of the used algorithms are compared from the global search ability, convergence speed, local entrapment probability, exploration/exploitation capability, diversity of individuals, balancing in exploration–exploitation, and computational time points of view. The summary of these comparisons is represented in Table 14. As seen from this table, the proposed APLO shows good performance. However, this algorithm is at the beginning of the way. It can still be examined from different aspects and on various problems so that its challenges are well-known and solved.

Table 14. Advantages and disadvantages of applied algorithms on optimal parameter extraction of PV cell and module.

Algorithm	Global Search Ability	Convergence Speed	Local Entrapment Probability	Explorative/ Exploitative	Diversity	Exploration-Exploitation Balance	Computational Complexity
APLO	High	Medium	Low	Exploitative	Adequate	Good	Low
CBO	Low	High	Medium	Exploitative	Inadequate	Medium	Low
DE	Low	High	High	Exploitative	Inadequate	Medium	High
GA	Low	High	High	Exploitative	Inadequate	Weak	High
GWO	Low	High	High	Explorative	High	Weak	Low
JAYA	Medium	Low	Medium	Exploitative	Adequate	Medium	Low
PSO	Medium	Medium	Medium	Exploitative	Adequate	Medium	High
SSA	Low	High	High	Explorative	High	Weak	Medium
TLBO	High	Low	Low	Exploitative	Adequate	Good	Medium

4.7. Comparison with the State-of-the-Art Algorithms

Several state-of-the-art improved and hybrid algorithms, including PGJAYA [57], IJAYA [58], STLBO [60], GOTLBO [61], MSSA [62], hARS-PS [74], BLPSO [81], CLPSO [82], TLABC [83], DE/BBO [84], and CMM-DE/BBO [85], are used to validate the performance of the proposed APLO algorithm in identifying PV models' parameters. Table 15 shows the max, min, mean, and SD of the parameters identified by the algorithms on each model corresponding to RMSE. Bold highlights indicate the best results. APLO yields the best results for the single diode and PV module models in terms of max, min, mean, and SD. It should be noted that PGJAYA and TLABC can only obtain the best results in terms of Min for SDM. Moreover, PGJAYA, TLABC, STLBO, MSSA, and CMM-DE/BBO can achieve the best solution in terms of the min value in calculating the parameters of the PV module model. It should be noted that, the results of hARS-PS on SDM model seems incorrectly reported in [74]. For DDM, hARS-PS is the best algorithm for all statistical values. Regarding the Min value for this model, PGJAYA is the second-best, STLBO is the third best, and IJAYA is the fourth-best algorithm. However, in terms of the mean, max and SD, proposed APLO is the third-best algorithm. The results indicate that the performance of the proposed basic algorithm is acceptable compared to other combined and improved algorithms.

Table 15. Statistic results of the APLO and state-of-the-art algorithms for three PV models (the significant values are bolded).

SDM						
Algorithm	PGJAYA	IJAYA	STLBO	GOTLBO	TLABC	MSSA
Min	$\mathbf{9.8602 \times 10^{-4}}$	9.8603×10^{-4}	9.8602×10^{-4}	9.8856×10^{-4}	$\mathbf{9.8602 \times 10^{-4}}$	9.86×10^{-4}
Mean	9.8602×10^{-4}	9.9204×10^{-4}	9.8607×10^{-4}	1.0450×10^{-3}	9.9417×10^{-4}	9.86×10^{-4}
Max	9.8603×10^{-4}	1.0622×10^{-3}	9.8655×10^{-4}	1.2067×10^{-3}	1.0308×10^{-3}	9.87×10^{-4}
SD	1.4485×10^{-9}	1.4033×10^{-5}	1.8602×10^{-5}	5.0218×10^{-5}	1.1896×10^{-5}	3.01×10^{-7}
Algorithm	CLPSO	BLPSO	DE/BBO	CMM-DE/BBO	APLO	hARS-PS
Min	9.9633×10^{-4}	1.0272×10^{-3}	9.9922×10^{-4}	9.8605×10^{-4}	$\mathbf{9.8602 \times 10^{-4}}$	9.84×10^{-4}
Mean	1.0581×10^{-3}	1.3139×10^{-3}	1.2948×10^{-3}	1.0486×10^{-3}	$\mathbf{9.8602 \times 10^{-4}}$	9.85×10^{-4}
Max	1.3196×10^{-3}	1.7928×10^{-3}	2.2258×10^{-3}	1.3475×10^{-3}	$\mathbf{9.8602 \times 10^{-4}}$	9.87×10^{-4}
SD	7.4854×10^{-5}	2.1166×10^{-4}	2.5074×10^{-4}	8.1679×10^{-5}	$\mathbf{1.5994 \times 10^{-16}}$	3.01×10^{-7}
DDM						
Algorithm	PGJAYA	IJAYA	STLBO	GOTLBO	TLABC	MSSA
Min	9.8263×10^{-4}	9.8293×10^{-4}	9.8252×10^{-4}	9.8742×10^{-4}	1.0012×10^{-3}	9.83×10^{-4}
Mean	9.8582×10^{-4}	1.0269×10^{-3}	1.0585×10^{-3}	1.1475×10^{-3}	1.2116×10^{-3}	9.94×10^{-4}
Max	9.9499×10^{-4}	1.4055×10^{-3}	2.4480×10^{-3}	1.3947×10^{-3}	1.9826×10^{-3}	9.99×10^{-4}
SD	2.5375×10^{-6}	9.8325×10^{-5}	2.8978×10^{-4}	1.1330×10^{-4}	2.1100×10^{-4}	1.49×10^{-6}
Algorithm	CLPSO	BLPSO	DE/BBO	CMM-DE/BBO	APLO	hARS-PS
Min	9.9894×10^{-4}	1.0628×10^{-3}	1.0255×10^{-3}	1.0088×10^{-3}	9.8307×10^{-4}	$\mathbf{9.82 \times 10^{-4}}$
Mean	1.1458×10^{-3}	1.4821×10^{-3}	1.5571×10^{-3}	1.5487×10^{-3}	1.0199×10^{-3}	$\mathbf{9.84 \times 10^{-4}}$
Max	1.5494×10^{-3}	1.7411×10^{-3}	2.4042×10^{-3}	2.0589×10^{-3}	1.3423×10^{-3}	$\mathbf{9.87 \times 10^{-4}}$
SD	1.4367×10^{-4}	1.7789×10^{-4}	3.6297×10^{-4}	2.9413×10^{-4}	7.7971×10^{-5}	$\mathbf{1.45 \times 10^{-7}}$
PV Module Model						
Algorithm	PGJAYA	IJAYA	STLBO	GOTLBO	TLABC	MSSA
Min	$\mathbf{2.425075 \times 10^{-3}}$	2.425129×10^{-3}	$\mathbf{2.425075 \times 10^{-3}}$	2.426583×10^{-3}	$\mathbf{2.425075 \times 10^{-3}}$	2.42×10^{-3}
Mean	2.425144×10^{-3}	2.428855×10^{-3}	2.055293×10^{-2}	2.475386×10^{-3}	2.425464×10^{-3}	2.54×10^{-3}
Max	2.426764×10^{-3}	2.439269×10^{-3}	2.742508×10^{-1}	2.563849×10^{-3}	2.428731×10^{-3}	2.78×10^{-3}
SD	3.071420×10^{-7}	3.775523×10^{-6}	6.896273×10^{-2}	2.938836×10^{-5}	8.746462×10^{-7}	1.75×10^{-5}
Algorithm	CLPSO	BLPSO	DE/BBO	CMM-DE/BBO	APLO	hARS-PS
Min	2.428064×10^{-3}	2.425236×10^{-3}	2.428255×10^{-3}	$\mathbf{2.425075 \times 10^{-3}}$	$\mathbf{2.425075 \times 10^{-3}}$	2.42×10^{-3}
Mean	2.454903×10^{-3}	2.437873×10^{-3}	2.461623×10^{-3}	2.425175×10^{-3}	$\mathbf{2.425075 \times 10^{-3}}$	2.43×10^{-3}
Max	2.543269×10^{-3}	2.488348×10^{-3}	2.525560×10^{-3}	2.426796×10^{-3}	$\mathbf{2.425075 \times 10^{-3}}$	2.50×10^{-3}
SD	2.580951×10^{-5}	1.372409×10^{-5}	2.925123×10^{-5}	3.554783×10^{-7}	$\mathbf{5.962083 \times 10^{-17}}$	1.38×10^{-5}

5. Conclusions and Future Directions

This paper proposed a novel parameterless algorithm to estimate parameters without specifying any control parameters, called artificial parameterless optimization (APLO). As part of the proposed APLO, a novel mutation operator was designed. To advance the exploration phase of the APLO, this operator required all participants to move around the best available solution. The current best, the old best, and the individual's current position were incorporated into the differential term of the mutation operator to assist the exploitation phase and maintain convergence speed. Furthermore, a random multiplication term using a normal distribution was proposed to ensure population diversity through the iteration process of the algorithm. Comprehensive studies were established to evaluate the performance of the proposed algorithm to estimate the parameters of the PV cells. The results revealed that the proposed algorithm could provide an excellent exploration–exploitation balance and consistency during the iterations. Two main factors were responsible for this: (a) the endorsement of positive feedback from individuals who had already achieved improvements, and (b) the presence of well-representative individuals ensured that the entire population was consistent in its diversity.

In addition, some comparisons were made in terms of statistical analysis. Based on the experimental parameter estimation results for the SDM, DDM, and PV module models, the APLO algorithm was more accurate and reliable than other original and well-known algorithms. A comparison of the calculated and standard data of the V-I and P-V curves was also conducted to determine the accuracy of the estimated parameters using the proposed algorithm. Moreover, the results indicated the proper performance of the proposed basic algorithm compared to other state-of-the-art combined and improved algorithms. Regarding the four statistical metrics, the proposed algorithm outperforms others for SDM and PV module models. Additionally, in the case of DDM, it showed good performance compared to the comparative algorithms. However, the results can be further improved by improving or combining the proposed algorithm with other algorithms.

Since APLO is simple and efficient, it can also be used to solve more complex engineering optimization problems. A modification of the APLO can also speed up convergence and reduce computational costs. Moreover, we intend to develop binary and multiobjective versions of APLO algorithms. Furthermore, APLO can be used to optimize support vector machines or kernel extreme learning machines.

Author Contributions: All the authors contributed to formulating the research idea, algorithm design, result analysis, writing and reviewing the research. Conceptualization, M.A. and A.A. (Abdulaziz Alanazi); methodology, M.A.; software, A.A. (Ahmad Almadhor); validation, H.T.R. and A.A. (Abdulaziz Alanazi); formal analysis, M.A.; investigation, A.A. (Ahmad Almadhor); resources, A.A. (Abdulaziz Alanazi); data curation, H.T.R.; writing—original draft preparation, M.A.; writing—review and editing, H.T.R.; visualization, A.A. (Ahmad Almadhor); supervision, M.A.; project administration, A.A. (Abdulaziz Alanazi); funding acquisition, M.A. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the Deanship of Scientific Research at Jouf University under Grant Number DSR2022-RG-0113.

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.

References

1. Sheng, R.; Du, J.; Liu, S.; Wang, C.; Wang, Z.; Liu, X. Solar Photovoltaic Investment Changes across China Regions Using a Spatial Shift-Share Analysis. *Energies* **2021**, *14*, 6418. [CrossRef]
2. Leitão, D.; Torres, J.P.N.; Fernandes, J.F.P. Spectral Irradiance Influence on Solar Cells Efficiency. *Energies* **2020**, *13*, 5017. [CrossRef]
3. Wang, P.; Yu, P.; Huang, L.; Zhang, Y. An Integrated Technical, Economic, and Environmental Framework for Evaluating the Rooftop Photovoltaic Potential of Old Residential Buildings. *J. Environ. Manag.* **2022**, *317*, 115296. [CrossRef] [PubMed]
4. Chen, H.; Jiao, S.; Heidari, A.A.; Wang, M.; Chen, X.; Zhao, X. An Opposition-Based Sine Cosine Approach with Local Search for Parameter Estimation of Photovoltaic Models. *Energy Convers. Manag.* **2019**, *195*, 927–942. [CrossRef]
5. Rasheduzzaman, M.; Fajri, P.; Kimball, J.; Deken, B. Modeling, Analysis, and Control Design of a Single-Stage Boost Inverter. *Energies* **2021**, *14*, 4098. [CrossRef]
6. Zhang, W.; Zheng, Z.; Liu, H. A Novel Droop Control Method to Achieve Maximum Power Output of Photovoltaic for Parallel Inverter System. *CSEE J. Power Energy Syst.* **2021**, *8*, 1636–1645. [CrossRef]
7. Mehta, H.K.; Warke, H.; Kukadiya, K.; Panchal, A.K. Accurate Expressions for Single-Diode-Model Solar Cell Parameterization. *IEEE J. Photovoltaics* **2019**, *9*, 803–810. [CrossRef]
8. Hejri, M.; Mokhtari, H.; Azizian, M.R.; Ghandhari, M.; Söder, L. On the Parameter Extraction of a Five-Parameter Double-Diode Model of Photovoltaic Cells and Modules. *IEEE J. Photovoltaics* **2014**, *4*, 915–923. [CrossRef]
9. Chin, V.J.; Salam, Z. A New Three-Point-Based Approach for the Parameter Extraction of Photovoltaic Cells. *Appl. Energy* **2019**, *237*, 519–533. [CrossRef]
10. Akbari, M.A.; Zare, M.; Azizipanah-Abarghooee, R.; Mirjalili, S.; Deriche, M. The Cheetah Optimizer: A Nature-Inspired Metaheuristic Algorithm for Large-Scale Optimization Problems. *Sci. Rep.* **2022**, *12*, 10953. [CrossRef]
11. Kirkpatrick, S.; Gelatt, C.D.; Vecchi, M.P. Optimization by Simulated Annealing. *Science* **1983**, *220*, 671–680. [CrossRef] [PubMed]
12. Ghasemi, M.; Akbari, M.-A.; Jun, C.; Bateni, S.M.; Zare, M.; Zahedi, A.; Pai, H.-T.; Band, S.S.; Moslehpour, M.; Chau, K.-W. Circulatory System Based Optimization (CSBO): An Expert Multilevel Biologically Inspired Meta-Heuristic Algorithm. *Eng. Appl. Comput. Fluid Mech.* **2022**, *16*, 1483–1525. [CrossRef]
13. Holland, J.H. Genetic Algorithms. *Sci. Am.* **1992**, *267*, 66–73. [CrossRef]
14. Yao, X.; Liu, Y.; Lin, G. Evolutionary Programming Made Faster. *IEEE Trans. Evol. Comput.* **1999**, *3*, 82–102.
15. Koza, J.R. Genetic Programming as a Means for Programming Computers by Natural Selection. *Stat. Comput.* **1994**, *4*, 87–112. [CrossRef]
16. Simon, D. Biogeography-Based Optimization. *IEEE Trans. Evol. Comput.* **2008**, *12*, 702–713. [CrossRef]

17. Storn, R.; Price, K. Differential Evolution—A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. *J. Glob. Optim.* **1997**, *11*, 341–359. [CrossRef]
18. Rashedi, E.; Nezamabadi-Pour, H.; Saryazdi, S. GSA: A Gravitational Search Algorithm. *Inf. Sci.* **2009**, *179*, 2232–2248. [CrossRef]
19. Kaveh, A.; Talatahari, S. A Novel Heuristic Optimization Method: Charged System Search. *Acta Mech.* **2010**, *213*, 267–289. [CrossRef]
20. Rabanal, P.; Rodríguez, I.; Rubio, F. Using River Formation Dynamics to Design Heuristic Algorithms. In *International Conference on Unconventional Computation*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 163–177.
21. Erol, O.K.; Eksin, I. A New Optimization Method: Big Bang–Big Crunch. *Adv. Eng. Softw.* **2006**, *37*, 106–111. [CrossRef]
22. Boettcher, S.; Percus, A.G. Optimization with Extremal Dynamics. *Complexity* **2002**, *8*, 57–62. [CrossRef]
23. Shah-Hosseini, H. Principal Components Analysis by the Galaxy-Based Search Algorithm: A Novel Metaheuristic for Continuous Optimisation. *Int. J. Comput. Sci. Eng.* **2011**, *6*, 132–140.
24. Formato, R.A. Central Force Optimization. *Prog. Electromagn. Res.* **2007**, *77*, 425–491. [CrossRef]
25. Zhao, W.; Zhang, Z.; Wang, L. Manta Ray Foraging Optimization: An Effective Bio-Inspired Optimizer for Engineering Applications. *Eng. Appl. Artif. Intell.* **2020**, *87*, 103300. [CrossRef]
26. Eskandar, H.; Sadollah, A.; Bahreininejad, A.; Hamdi, M. Water Cycle Algorithm—A Novel Metaheuristic Optimization Method for Solving Constrained Engineering Optimization Problems. *Comput. Struct.* **2012**, *110*, 151–166. [CrossRef]
27. Hosseini, H.S. Problem Solving by Intelligent Water Drops. In Proceedings of the 2007 IEEE Congress on Evolutionary Computation, Singapore, 25–28 September 2007; pp. 3226–3231.
28. Li, B.; Jiang, W. Chaos Optimization Method and Its Application. *Control Theory Appl.* **1997**, *14*, 613–615.
29. Birbil, Ş.İ.; Fang, S.-C. An Electromagnetism-like Mechanism for Global Optimization. *J. Glob. Optim.* **2003**, *25*, 263–282. [CrossRef]
30. Alatas, B. ACROA: Artificial Chemical Reaction Optimization Algorithm for Global Optimization. *Expert Syst. Appl.* **2011**, *38*, 13170–13180. [CrossRef]
31. Irizarry, R. LARES: An Artificial Chemical Process Approach for Optimization. *Evol. Comput.* **2004**, *12*, 435–459. [CrossRef]
32. Abdechiri, M.; Meybodi, M.R.; Bahrami, H. Gases Brownian Motion Optimization: An Algorithm for Optimization (GBMO). *Appl. Soft Comput.* **2013**, *13*, 2932–2946. [CrossRef]
33. Kennedy, J.; Eberhart, R. Particle Swarm Optimization. In Proceedings of the ICNN'95—International Conference on Neural Networks, Perth, WA, Australia, 27 November–1 December 1995; Volume 4, pp. 1942–1948.
34. Yang, X.-S.; Deb, S. Cuckoo Search via Lévy Flights. In Proceedings of the 2009 World Congress on Nature & Biologically Inspired Computing (NaBIC), Coimbatore, India, 9–11 December 2009; pp. 210–214.
35. Mirjalili, S. The Ant Lion Optimizer. *Adv. Eng. Softw.* **2015**, *83*, 80–98. [CrossRef]
36. Pham, D.T.; Ghanbarzadeh, A.; Koç, E.; Otri, S.; Rahim, S.; Zaidi, M. The Bees Algorithm—A Novel Tool for Complex Optimisation Problems. In *Intelligent Production Machines and Systems*; Elsevier: Amsterdam, The Netherlands, 2006; pp. 454–459.
37. Eusuff, M.M.; Lansey, K.E. Optimization of Water Distribution Network Design Using the Shuffled Frog Leaping Algorithm. *J. Water Resour. Plan. Manag.* **2003**, *129*, 210–225. [CrossRef]
38. Yang, X.; Gandomi, A.H. Bat Algorithm: A Novel Approach for Global Engineering Optimization. *Eng. Comput.* **2012**, *29*, 464–483. [CrossRef]
39. Mirjalili, S. Moth-Flame Optimization Algorithm: A Novel Nature-Inspired Heuristic Paradigm. *Knowl.-Based Syst.* **2015**, *89*, 228–249. [CrossRef]
40. Panigrahi, B.K.; Ravikumar Pandi, V. Bacterial Foraging Optimisation: Nelder–Mead Hybrid Algorithm for Economic Load Dispatch. *IET Gener. Transm. Distrib.* **2008**, *2*, 556. [CrossRef]
41. Gandomi, A.H.; Alavi, A.H. Krill Herd: A New Bio-Inspired Optimization Algorithm. *Commun. Nonlinear Sci. Numer. Simul.* **2012**, *17*, 4831–4845. [CrossRef]
42. Mirjalili, S.; Lewis, A. The Whale Optimization Algorithm. *Adv. Eng. Softw.* **2016**, *95*, 51–67. [CrossRef]
43. Drigo, M. The Ant System: Optimization by a Colony of Cooperating Agents. *IEEE Trans. Syst. Man Cybern. B* **1996**, *26*, 1–13. [CrossRef]
44. Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey Wolf Optimizer. *Adv. Eng. Softw.* **2014**, *69*, 46–61. [CrossRef]
45. Yang, X.-S. Firefly Algorithms for Multimodal Optimization. In *International Symposium on Stochastic Algorithms*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 169–178.
46. Akay, B.; Karaboga, D. Artificial Bee Colony Algorithm for Large-Scale Problems and Engineering Design Optimization. *J. Intell. Manuf.* **2012**, *23*, 1001–1014. [CrossRef]
47. Pan, W.-T. A New Fruit Fly Optimization Algorithm: Taking the Financial Distress Model as an Example. *Knowl.-Based Syst.* **2012**, *26*, 69–74. [CrossRef]
48. Krishnanand, K.N.; Ghose, D. Glowworm Swarm Optimization for Simultaneous Capture of Multiple Local Optima of Multimodal Functions. *Swarm Intell.* **2009**, *3*, 87–124. [CrossRef]
49. Nadimi-Shahraki, M.H.; Taghian, S.; Mirjalili, S.; Abualigah, L.; Abd Elaziz, M.; Oliva, D. EWOA-OPF: Effective Whale Optimization Algorithm to Solve Optimal Power Flow Problem. *Electronics* **2021**, *10*, 2975. [CrossRef]
50. Nadimi-Shahraki, M.H.; Taghian, S.; Mirjalili, S.; Faris, H. MTDE: An Effective Multi-Trial Vector-Based Differential Evolution Algorithm and Its Applications for Engineering Design Problems. *Appl. Soft Comput.* **2020**, *97*, 106761. [CrossRef]

51. Nadimi-Shahraki, M.H.; Taghian, S.; Mirjalili, S. An Improved Grey Wolf Optimizer for Solving Engineering Problems. *Expert Syst. Appl.* **2021**, *166*, 113917. [CrossRef]
52. Eslami, M.; Shareef, H.; Mohamed, A.; Khajehzadeh, M. Optimal Location of PSS Using Improved PSO with Chaotic Sequence. In Proceedings of the International Conference on Electrical, Control and Computer Engineering 2011 (InECCE), Kuantan, Malaysia, 21–22 June 2011; pp. 253–258.
53. Eslami, M.; Shareef, H.; Mohamed, A.; Khajehzadeh, M. Coordinated Design of PSS and SVC Damping Controller Using CPSO. In Proceedings of the 2011 5th International Power Engineering and Optimization Conference, Shah Alam, Malaysia, 6–7 June 2011; pp. 11–16.
54. Khajehzadeh, M.; Taha, M.R.; Eslami, M. Multi-Objective Optimisation of Retaining Walls Using Hybrid Adaptive Gravitational Search Algorithm. *Civ. Eng. Environ. Syst.* **2014**, *31*, 229–242. [CrossRef]
55. Messaoud, R. Ben Extraction of Uncertain Parameters of Single and Double Diode Model of a Photovoltaic Panel Using Salp Swarm Algorithm. *Measurement* **2020**, *154*, 107446. [CrossRef]
56. Rao, R. Jaya: A Simple and New Optimization Algorithm for Solving Constrained and Unconstrained Optimization Problems. *Int. J. Ind. Eng. Comput.* **2016**, *7*, 19–34. [CrossRef]
57. Yu, K.; Qu, B.; Yue, C.; Ge, S.; Chen, X.; Liang, J. A Performance-Guided JAYA Algorithm for Parameters Identification of Photovoltaic Cell and Module. *Appl. Energy* **2019**, *237*, 241–257. [CrossRef]
58. Yu, K.; Liang, J.J.; Qu, B.Y.; Chen, X.; Wang, H. Parameters Identification of Photovoltaic Models Using an Improved JAYA Optimization Algorithm. *Energy Convers. Manag.* **2017**, *150*, 742–753. [CrossRef]
59. Patel, S.J.; Panchal, A.K.; Kheraj, V. Extraction of Solar Cell Parameters from a Single Current–Voltage Characteristic Using Teaching Learning Based Optimization Algorithm. *Appl. Energy* **2014**, *119*, 384–393. [CrossRef]
60. Niu, Q.; Zhang, H.; Li, K. An Improved TLBO with Elite Strategy for Parameters Identification of PEM Fuel Cell and Solar Cell Models. *Int. J. Hydrogen Energy* **2014**, *39*, 3837–3854. [CrossRef]
61. Chen, X.; Yu, K.; Du, W.; Zhao, W.; Liu, G. Parameters Identification of Solar Cell Models Using Generalized Oppositional Teaching Learning Based Optimization. *Energy* **2016**, *99*, 170–180. [CrossRef]
62. Yaghoubi, M.; Eslami, M.; Noroozi, M.; Mohammadi, H.; Kamari, O.; Palani, S. Modified Salp Swarm Optimization for Parameter Estimation of Solar PV Models. *IEEE Access* **2022**, *10*, 110181–110194. [CrossRef]
63. Ishaque, K.; Salam, Z.; Mekhilef, S.; Shamsudin, A. Parameter Extraction of Solar Photovoltaic Modules Using Penalty-Based Differential Evolution. *Appl. Energy* **2012**, *99*, 297–308. [CrossRef]
64. Jiang, L.L.; Maskell, D.L.; Patra, J.C. Parameter Estimation of Solar Cells and Modules Using an Improved Adaptive Differential Evolution Algorithm. *Appl. Energy* **2013**, *112*, 185–193. [CrossRef]
65. AlRashidi, M.R.; AlHajri, M.F.; El-Naggar, K.M.; Al-Othman, A.K. A New Estimation Approach for Determining the I-V Characteristics of Solar Cells. *Sol. Energy* **2011**, *85*, 1543–1550. [CrossRef]
66. Yousri, D.; Allam, D.; Eteiba, M.B.; Suganthan, 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]
67. Ebrahimi, S.M.; Salahshour, E.; Malekzadeh, M.; Gordillo, F. Parameters Identification of PV Solar Cells and Modules Using Flexible Particle Swarm Optimization Algorithm. *Energy* **2019**, *179*, 358–372. [CrossRef]
68. Liang, J.; Ge, S.; Qu, B.; Yu, K.; Liu, F.; Yang, H.; Wei, P.; Li, Z. Classified Perturbation Mutation Based Particle Swarm Optimization Algorithm for Parameters Extraction of Photovoltaic Models. *Energy Convers. Manag.* **2020**, *203*, 112138. [CrossRef]
69. Lin, X.; Wu, Y. Parameters Identification of Photovoltaic Models Using Niche-Based Particle Swarm Optimization in Parallel Computing Architecture. *Energy* **2020**, *196*, 117054. [CrossRef]
70. Yousri, D.; Thanikanti, S.B.; Allam, D.; Ramachandaramurthy, V.K.; Eteiba, M.B. Fractional Chaotic Ensemble Particle Swarm Optimizer for Identifying the Single, Double, and Three Diode Photovoltaic Models' Parameters. *Energy* **2020**, *195*, 116979. [CrossRef]
71. Nunes, H.G.G.; Silva, P.N.C.; Pombo, J.A.N.; Mariano, S.; Calado, M.R.A. Multiswarm Spiral Leader Particle Swarm Optimisation Algorithm for PV Parameter Identification. *Energy Convers. Manag.* **2020**, *225*, 113388. [CrossRef]
72. Premkumar, M.; Jangir, P.; Elavarasan, R.M.; Sowmya, R. Opposition Decided Gradient-Based Optimizer with Balance Analysis and Diversity Maintenance for Parameter Identification of Solar Photovoltaic Models. *J. Ambient Intell. Humaniz. Comput.* **2021**, 1–23. [CrossRef]
73. Gafar, M.; El-Sehiemy, R.A.; Hasanien, H.M.; Abaza, A. Optimal Parameter Estimation of Three Solar Cell Models Using Modified Spotted Hyena Optimization. *J. Ambient Intell. Humaniz. Comput.* **2022**, 1–12. [CrossRef]
74. Eslami, M.; Akbari, E.; Seyed Sadr, S.T.; Ibrahim, B.F. A Novel Hybrid Algorithm Based on Rat Swarm Optimization and Pattern Search for Parameter Extraction of Solar Photovoltaic Models. *Energy Sci. Eng.* **2022**, *10*, 2689–2713. [CrossRef]
75. Wolpert, D.H.; Macready, W.G. No Free Lunch Theorems for Optimization. *IEEE Trans. Evol. Comput.* **1997**, *1*, 67–82. [CrossRef]
76. Easwarakhanthan, T.; Bottin, J.; Bouhouch, I.; Boutrit, C. Nonlinear Minimization Algorithm for Determining the Solar Cell Parameters with Microcomputers. *Int. J. Sol. Energy* **1986**, *4*, 1–12. [CrossRef]
77. Rao, R.V.; Savsani, V.J.; Vakharia, D.P. Teaching–Learning-Based Optimization: A Novel Method for Constrained Mechanical Design Optimization Problems. *Comput. Des.* **2011**, *43*, 303–315. [CrossRef]

78. Mirjalili, S.; Gandomi, A.H.; Mirjalili, S.Z.; Saremi, S.; Faris, H.; Mirjalili, S.M. Salp Swarm Algorithm: A Bio-Inspired Optimizer for Engineering Design Problems. *Adv. Eng. Softw.* **2017**, *114*, 163–191. [CrossRef]
79. Kaveh, A.; Mahdavi, V.R. Colliding Bodies Optimization: A Novel Meta-Heuristic Method. *Comput. Struct.* **2014**, *139*, 18–27. [CrossRef]
80. Hussain, K.; Salleh, M.N.M.; Cheng, S.; Shi, Y. On the Exploration and Exploitation in Popular Swarm-Based Metaheuristic Algorithms. *Neural Comput. Appl.* **2019**, *31*, 7665–7683. [CrossRef]
81. Chen, X.; Tianfield, H.; Mei, C.; Du, W.; Liu, G. Biogeography-Based Learning Particle Swarm Optimization. *Soft Comput.* **2017**, *21*, 7519–7541. [CrossRef]
82. Liang, J.J.; Qin, A.K.; Suganthan, P.N.; Baskar, S. Comprehensive Learning Particle Swarm Optimizer for Global Optimization of Multimodal Functions. *IEEE Trans. Evol. Comput.* **2006**, *10*, 281–295. [CrossRef]
83. Chen, X.; Xu, B.; Mei, C.; Ding, Y.; Li, K. Teaching–Learning–Based Artificial Bee Colony for Solar Photovoltaic Parameter Estimation. *Appl. Energy* **2018**, *212*, 1578–1588. [CrossRef]
84. Gong, W.; Cai, Z.; Ling, C.X. DE/BBO: A Hybrid Differential Evolution with Biogeography-Based Optimization for Global Numerical Optimization. *Soft Comput.* **2010**, *15*, 645–665. [CrossRef]
85. Chen, X.; Tianfield, H.; Du, W.; Liu, G. Biogeography-Based Optimization with Covariance Matrix Based Migration. *Appl. Soft Comput.* **2016**, *45*, 71–85. [CrossRef]