IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 4, 2020, accepted February 10, 2020, date of publication February 19, 2020, date of current version February 28, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2975078

Parameters Identification of Photovoltaic Models Using a Multi-Strategy Success-History-Based Adaptive Differential Evolution

QINZHI HAO[1], ZHONGLIANG ZHOU[1], ZHENGLEI WEI[2], AND GUANGHUI CHEN[3]

[1]Air Traffic Control and Navigation College, Air Force Engineering University, Xi'an 710100, China
[2]Institute of Aeronautics Engineering, Air Force Engineering University, Xi'an 710038, China
[3]School of Pharmacy, Fourth Military Medical University, Xi'an 710038, China

Corresponding authors: Qinzhi Hao (991550782@qq.com) and Zhenglei Wei (zhenglei_wei@126.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 51579209 and Grant 61601505, in part by the Aeronautical Science Foundation of China under Grant 20175196019, and in part by the Natural Science Foundation of Shanxi Province under Grant 2017JM6078.

ABSTRACT Parameters identification of photovoltaic (PV) models based on measured current-voltage characteristics curves is significant for the simulation, evaluation and control of PV systems. To accurately and reliably identify the parameters of different PV models, a novel optimization algorithm, multi-strategy success-history based adaptive differential evolution with linear population size reduction (MLSHADE), is proposed. MLSHADE mainly divides evolutionary process into two phases during every generation. According to the definition of class probability variable, the population individuals of first phase are assigned to different two populations for exploration and exploitation, respectively. The novelty of MLSHADE algorithm lies primarily in three improvements: (i) a new weighted mutation strategy is used to enrich the population diversity of later iterations for differential evolution population in the first phase; (ii) inferior solutions search (ISS) technique is presented to avoid falling into local optimum for covariance matrix adaptation evolution strategy population in the first phase; and (iii) Eigen Gaussian random walk strategy is proposed to help maintain effectively the balance between the global exploration and local exploitation abilities in the second phase. The experiments on CEC 2018 test suite illustrate that the proposed MLSHADE exerts the better performances against the stat-of-the-art algorithms in terms of accuracy, reliability and time consumption. The proposed MLSHADE is used to solve the parameters identification problems of different PV models including single diode, double diode, and PV module. Comprehensive experiment results and analyses indicate that MLSHADE can obtain a highly competitive performance compared with other state-of-the-art algorithms, especially in terms of accuracy and reliability.

INDEX TERMS Differential evolution operator, multi-strategy LSHADE, numerical optimization, parameters identification of photovoltaic.

I. INTRODUCTION

Due to richness, cleanliness and pollution-free of the solar energy, it is considered as one of the most promising renewable energy sources [1]. Through photovoltaic (PV) systems such as solar cell, solar energy is transformed into electrical energy. To develop the photovoltaic power generation and apply the photovoltaic power generation to comprehensive field, photovoltaic model is one of the most important parts in photovoltaic power generation system. In order to control and optimize the parameters identification of photovoltaic models, it is vital to evaluate the actual behavior of PV arrays in operation using accurate model based on measured current-voltage data. In addition to accurate parameters identification, automatic fault detection and diagnosis techniques for photovoltaic arrays are crucial to promote the efficiency and reliability of photovoltaic models. In recent years, many conventional artificial intelligence approaches have been successfully applied to automatically establish fault detection and diagnosis model using fault data samples [2]. In [3], based on the output I-V characteristic of the PV arrays, kernel based extreme learning machine (KELM), is explored for the first

The associate editor coordinating the review of this manuscript and approving it for publication was Sudhakar Babu Thanikanti.

time to automatically identify four common PV array faults. In [4], a novel intelligent fault detection and diagnosis method for photovoltaic arrays based on a newly designed deep residual network model is proposed. In [5], the random forest (RF) ensemble learning algorithm is explored for the detection and diagnosis of PV arrays early faults. Meanwhile, to build the efficient PV models and obtain the accurate parameters of PV models, the functions in practice are proposed, which can track the maximal power point and forecast the PV power [6].

In recent years, there are several mathematical models which can successfully describe the performance and nonlinear behavior of PV systems. The most common and widely adopted models are single diode model and double diode model [7]. These parameters can be employed for describing the current-voltage characteristics including photo-generated current, reverse saturation current, series resistors, parallel resistance and diode ideal factor. Through parameters identification of photovoltaic models, the relationship curve of current-voltage can be obtained [8]. Hence, the accurate current-voltage curve is meaningful for forecasting power of PV system and simulating the PV models [9]. In reality, the PV module consists of several solar cells models in series and in parallel in this system [10]. In the simulating system of PV power generation and system, the parameters of solar cells models can be obtained firstly and the relationship of current-voltage can be described then. Hence, the accurate identification for parameters is indispensable to the simulation, evaluation, and control of PV model system.

At present, some attempts have been devoted to using a variety of techniques for parameter identification. These techniques can be divided into three classes including mathematical analysis approximate solution [11], big data fitting [12] and optimization algorithms [13]. Due to neglecting a few inessential solutions or searching some key points which can be open-circuit voltage, short-circuit current and maximum power point, the solution can be estimated roughly by mathematical analysis approximate. The parameter identification based on data fitting needs spending a lot of time. The parameter identification based on optimization algorithm can get more accurate solutions through less experiment data. The approaches can include deterministic techniques and heuristic methods. The deterministic techniques could lead to high probability of falling in local optimal for PV models parameter identification [14]. The heuristic approaches can impose no restrictions on the problem characteristic, which are different from the deterministic techniques. Hence, many heuristic methods inspired by various natural phenomena have been widely employed to the parameter identification problem of PV models.

At present, there are a lot of heuristic approaches, which can be divided into four types. Recent literatures present the algorithm classification, including evolutionary algorithms (EAs), swarm intelligence (SI) algorithms, human behavior-based (HB) algorithms and physics-based algorithm (PA). There are some powerful algorithms applied to the parameter identification problem of PV models. In [14], an improved JAYA optimization algorithm with a self-adaptive weight and experience-based learning strategy is employed for parameters identification of PV models. In [15], the variant of differential evolution (DE) with a penalty strategy has been proposed for parameters identification of solar PV models. In [16], artificial bee swarm optimization (ABSO) is employed to identify the solar cell parameters. In [17], the model parameter estimation model of solar cell based on biogeography-based optimization with mutation strategies (BBO-M) is proposed. In [18] and [19], the original and the variant of teaching-learning-based optimization are employed for identifying the parameters of proton exchange membrane fuel and solar. In [20], an efficient approach based on slap swarm algorithm (SSA) for extracting the parameters of PV models is proposed. In [21], a multiple learning backtracking search algorithm (MLBSA) is employed for parameter identification of PV model. In addition to stochastic optimizations, some deterministic optimization algorithms are employed to solve parameter extraction problems of solar PV models. In [22], crow search algorithm (CSA) is accurately applied to identify the parameters based on the single diode and double diode PV models. In [23], a combination of the trust-region reflective (TRR) deterministic algorithm with the artificial bee colony (ABC) meta-heuristic algorithm is proposed to improve the parameter extraction of PV models. Besides, Nelder-Mead simplex [24], Levenberg-Marquardt [25], and Pattern Search [26] are employed to solve parameter extraction problems. In contrast to the white-box PV modeling techniques, data-driven black-box PV modeling techniques directly build PV models from measured data by regression. The data-driven black-box PV modeling techniques include artificial neural networks (ANN) [27], radial basis function neural network (RBFNN) [28], deep residual network (ResNet) [29] and so on. CEC 2019 competition of smart grid and sustainable energy systems, a novel algorithm inspired by nuclear reaction process, called nuclear reaction optimization (NRO), was proposed and won the first rank in the parameter identification problem of PV models [30]. Although NRO can obtain good solutions, it is difficult to get out of local optimal. Meanwhile, NRO performs worse than MLBSA. Hence, it is necessary to propose a novel optimization algorithm.

During the last decade, numerous meta-heuristic optimization algorithms have been proposed [31]. In [30], as a novel meta-heuristic optimization algorithm, nuclear reaction optimization (NRO) is proposed to solve the parameter identification problem of PV models. In [32], a new optimization algorithm, spherical search (SS) algorithm, which employs the calculation of spherical boundary to search optimal solution, is proposed to solve the bound-constrained non-linear global optimization problems. In [33], according to different knowledge rate during junior phase and senior phase, gaining-sharing knowledge based algorithm (GSK) is proposed to solve various optimization test suites. According to the source of inspiration, the meta-heuristic optimization algorithms can be classified into four groups,

including evolutionary techniques, swarm intelligence techniques, physics-based techniques and human-related techniques [33]. Evolutionary algorithm is one of meta-heuristic approaches, which includes genetic algorithm (GA) [34], differential evolution (DE) [35], estimation of distribution algorithm (EDA) [36] and so on. DE has the superior exploration ability, which depends on three intrinsic control parameters including population size *NP*, scaling factor *F* and the crossover rate *Cr*. At present, the variants of DE algorithm have been proposed to solve a variety of optimization problems. An enhancement to DE, called success-history based adaptive DE with linear population size reduction (LSHADE), is presented to solve IEEE CEC 2014 suite and wins the top [37]. In addition to LSHADE, the variants of DE include iLSHADE [38], LSHADE-SPACMA [39], LSHADE-EpSin [39] and jSO [41]. At present, a novel mutation strategy, *DE/current-to-ord_pbest/1*, is combined with LSHADE. The combination, EBLSHADE, exerts the superior performance in [42]. This paper mainly focuses on Gaussian EDA (GEDA) with a Gaussian probability model in which the variables follow a Gaussian distribution [41]. In GEDA, both the distribution scope and evolutionary direction determine the performance of GEDA. The covariance matrix adaptation evolution strategy (CMA-ES) with rank-1 and rank-μ updating is a classic GEDA, which has the superiority on exploitation ability [43]. In order to improve the efficiency of DE and CMA-ES, this paper proposes a novel optimization algorithm, which is a combination of above two algorithms.

This research paper aims to introduce an improved LSHADE algorithm to solve the parameter identification problem of PV models. The main contributions of this study are as follows: (1) A multi-strategy LSHADE, called MLSHADE, which is a combination of LSAHDE and CMA-ES, is proposed. (2) Based on enhanced LSHADE with semi-parameter adaption hybrid with CMA-ES (ELSHADE-SPACMA) [30], MLSHADE mainly employs weighted mutation strategy, inferior solution search strategy and Eigen Gaussian random walk strategy to improve the diversity of population and enhance the ability of balancing between exploration and exploitation. (3) The proposed MLSHADE algorithm is applied to solve parameters identification problems of PV models. (4) The effectiveness of TSO is discussed through comparison experiments on CEC 2018 test suite and parameters identification problems of different PV models.

The rest of this paper is organized as follows: the PV models formulation is reviewed in Section II. The MLSHADE algorithm is presented in Section III. The experimental results on CEC 2018 test suite and different PV models are shown and analyzed in Section IV. Finally, the conclusions are given in Section V.

II. PHOTOVOLTAIC MODELS FORMULATION

In the literature, there are three PV models: single diode model, double diode model and PV module model. These models are presented as follows in this section. Meanwhile, the objective functions of these PV problems are given in this section.

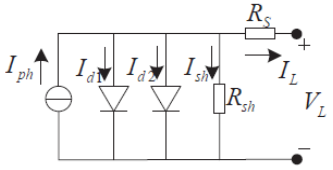


FIGURE 1. The equivalent circuit of double diode model.

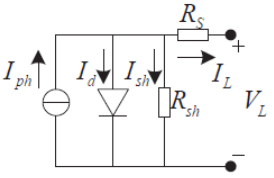


FIGURE 2. The equivalent circuit of single diode model.

A. DOUBLE DIODE MODEL

A solar cell can ideally be considered as a current source in parallel with a diode. The current source is connected in parallel with a diode that simulates the space charge current. Meanwhile, The current source can be connected in parallel with a resistor. In addition, the resistances of the metal contact and semiconductor material body of solar cell are represented by series resistances. Therefore, the equivalent circuit of the solar cell double diode model is shown in Fig. 1.

In the double diode model, I_L is the solar cell output current, I_{ph} is the total current generated by solar cell, I_{sh} is the reverse saturation current of diode. R_S and R_{sh} are the series and shunt resistances, respectively. I_{d1} and I_{d2} are the diffusion and saturation currents, respectively. q is the electron charge $\left(1.60217646 \times 10^{-19} C\right)$, and k is the Boltzmann constant $\left(1.3806503 \times 10^{-23} J/K\right)$. T is the cell absolute temperature in Kelvin. n_1 and n_2 are the diode ideal factors. Hence, according to equivalent circuit of double diode model, the output current is written as follows:

$$\begin{aligned} I_L &= I_{ph} - I_{d1} - I_{d2} - I_{sh} \\ &= I_{ph} - I_{sd1} \cdot \left[\exp\left(\frac{q \cdot (V_L + R_S \cdot I_L)}{n_1 \cdot k \cdot T}\right) - 1\right] \\ &\quad - I_{sd1} \cdot \left[\exp\left(\frac{q \cdot (V_L + R_S \cdot I_L)}{n_2 \cdot k \cdot T}\right) - 1\right] - \frac{V_L + R_S \cdot L_L}{R_{sh}} \end{aligned} \tag{1}$$

where seven unknown parameters ($I_{ph}, I_{sd1}, I_{sd2}, R_S, R_{sh}, n_1, n_2$) are needed to be identified to obtain the actual performance of solar cell double diode model.

B. SINGLE DIODE MODEL

Based on the foundation of the double diode model, although the diffusion current and the compound current are linearly independent, these two current can usually combined when the diode ideal factor n is introduced. The equivalent circuit of the single diode model is shown in Fig. 2. According to

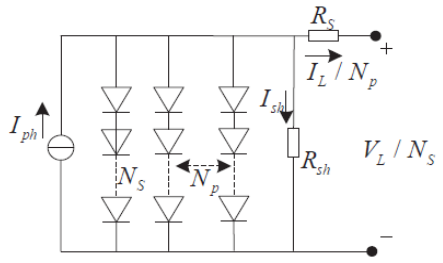


FIGURE 3. The PV module model.

the equivalent circuit, the five-parameter model of the single diode is shown as follows:

$$\begin{aligned} I_L &= I_{ph} - I_d - I_{sh} \\ &= I_{ph} - I_{sd} \cdot \left[\exp\left(\frac{q \cdot (V_L + R_S \cdot I_L)}{n \cdot k \cdot T} \right) - 1 \right] \\ &\quad - \frac{V_L + R_S \cdot L_L}{R_{sh}} \end{aligned} \tag{2}$$

where I_d is the diode current, I_{sh} is the shunt current, I_{sd} is the reverse saturation current of diode and n is the diode ideal factor. Meanwhile, five unknown parameters $(I_{ph}, I_{sd}, R_S, R_{sh}, n)$ are needed to be identified to obtain the actual performance of solar cell single diode model.

C. PV MODULE MODEL

The photovoltaic module is composed of a number of solar cell units in series and parallel, and its equivalent circuit is shown in Fig. 3.

The solar cell branch is connected with a diode in series to prevent current backflow from burning the branch due to different output current of the different branches. Meanwhile, in order to shunt the photovoltaic current in the shaded area and prevent the battery from being damaged due to excessive negative pressure under the maximum short-circuit current, each solar cell unit is connected with a diode in parallel. The PV module model is shown in formula (3):

$$\begin{aligned} I_L / N_p = I_{ph} - I_{sd} \cdot \left[\exp\left(\frac{q \cdot (V_L / N_S + R_S \cdot I_L / N_p)}{n \cdot k \cdot T} \right) - 1 \right] \\ - \frac{V_L / N_S + R_S \cdot I_L / N_p}{R_{sh}} \end{aligned} \tag{3}$$

where N_p represents the number of solar cells in parallel, N_S represents the number of solar cells in series. Meanwhile, five unknown parameters $(I_{ph}, I_{sd}, R_S, R_{sh}, n)$ are needed to be identified to obtain the actual performance of the PV module model.

D. OBJECTIVE FUNCTION

To minimize the difference between the experimental data and simulated data obtained by estimated parameters, the parameters identification problem of PV models is converted into as an optimization problem. For single diode model, the error function for each pair of experimental and simulated data point is defined by (4). The error function of double diode model is defined by (5). According to previous literatures, in order to further quantify the overall difference between the experimental and simulated current data, the root mean square error (RMSE) and the individual absolute error (IEA) are presented. In this paper, the RMSE defined by (6) and the IAE defined by (7) between the experimental data and simulated data are used as evaluation indexes.

$$\begin{aligned} f_k(V_L, I_L, x) &= I_{ph} - I_{sd} \cdot \left[\exp\left(\frac{q \cdot (V_L + R_S \cdot I_L)}{n \cdot k \cdot T} \right) - 1 \right] \\ &\quad - \frac{V_L + R_S \cdot L_L}{R_{sh}} - I_L \end{aligned} \tag{4}$$

$$\begin{aligned} f_k(V_L, I_L, x) &= I_{ph} - I_{sd1} \cdot \left[\exp\left(\frac{q \cdot (V_L + R_S \cdot I_L)}{n_1 \cdot k \cdot T} \right) - 1 \right] \\ &\quad - I_{sd1} \cdot \left[\exp\left(\frac{q \cdot (V_L + R_S \cdot I_L)}{n_2 \cdot k \cdot T} \right) - 1 \right] \\ &\quad - \frac{V_L + R_S \cdot L_L}{R_{sh}} - I_L \end{aligned} \tag{5}$$

$$RMSE(x) = \sqrt{\frac{1}{N} \sum_{k=1}^{N} f_k(V_L, I_L, x)^2} \tag{6}$$

$$IAE = \left| I_t - I_t' \right| \tag{7}$$

where $f_k(V_L, I_L, x)$ represents the error function, x is the solution vector consists of unknown parameters, N is the number of experimental data, I_t is measured current and I_t' is simulated current.

III. MLSHADE

In this section, we present the reviews of success-history based adaptive DE with linear population size reduction (LSHADE) and CMA-ES. Through combining improved LSHADE and enhanced CMA-ES, the novel algorithm, called multi-strategy LSHADE, is described.

A. REVIEWS OF THE BASIC SHADE AND CMA-ES

1) BASIC SHADE

LSHADE is an improved differential evolutionary algorithm, which employs success-history based parameter adaption and linear population size reduction mechanism [37]. The basic steps of LSHADE are given as follows:

Step 1. An initial population P^0 is created as follows:

$$x_{i,j}^0 = lb_j + rand \cdot \left(ub_j - lb_j \right) \tag{8}$$

where *rand* represents a uniformly distributed random number in [0, 1] and is same as the following *rand*, $x_{j,i}^0$ presents the j^{th} component $(j = 1, 2, \cdots, D)$ of the i^{th} individual $(i = 1, 2, \cdots, N)$ in the initial population P^0, lb_j and ub_j denote the lower and upper bounds of the j^{th} variable in the search space, respectively.

Step 2. Set the algorithm parameters crossover rate Cr and scaling factor F.

Step 3. According to current-to-pbest/1 mutation strategy, a mutant vector v_i^G is created as follows:

$$v_i^G = x_i^G + F_i^G \left(x_{pbest}^G - x_i^G \right) + F_i^G \left(x_{r1}^G - x_{r2}^G \right) \tag{9}$$

where x_i^G represents the i^{th} target vector of the G^{th} generation. F_i^G is the scaling factor of the i^{th} target vector at generation G. x_{pbest}^G is random one of p target vectors with the best fitness value. $r1$ and $r2$ are random indexes selected from current population and combination of current population and an external archive, respectively.

Step 4. According to the crossover rate, the trial vector u_i^G is rooted in the mixture of target vectors and mutant vectors.

$$u_{i,j}^G = \begin{cases} v_{i,j}^G & if \ \left(rand_{i,j} \le Cr_i \ or \ j = j_{rand}\right) \\ x_{i,j}^G & otherwise \end{cases} \tag{10}$$

where $rand_{i,j}$ is a uniformly distributed random number in [0, 1]. The crossover rate Cr_i can determine that some variables are inherited from mutant vector. j_{rand} which is a uniformly distributed random integer in [1, D] can ensure that at least one variable of trial vector belongs to the mutant vector.

Step 5. According to comparing the fitness function value between x_i^G and u_i^G, the old vectors with better fitness function value are selected.

Step 6. According to linear population size reducing (LPSR) [24], the population size is updated by evaluation number.

Step 7. Repeat **Steps 2** to **6** until a stopping criterion is met.

2) BASIC CMA-ES

Compared with other evolutionary algorithms, CMA-ES can efficiently solve diverse types of optimization problems [43]. The main steps of CMA-ES are presented as the following:

Step 1. Generate an initial population by uniformly distributed random strategy.

Step 2. Generate new solutions by sampling Gaussian distribution as follows:

$$x_i = m + \sigma y_i \tag{11}$$

where m is the mean of the selected superior solutions in set A of high-quality solutions. y_i is a search direction. In general, according to eigenvalue decomposition $C = BD^2B^T$, y_i can be obtained below formula:

$$y_i = BDz_i, z_i \sim N\,(0, I) \tag{12}$$

where B is the eigenvector matrix of covariance matrix. D is the diagonal matrix based on eigenvalue [41].

Step 3. Evaluate the solutions obtained by calculating objective function and sort the fitness values.

Step 4. Update the mean m, covariance matrix C and step size σ.

Step 5. Repeat **Steps 2** to **4** until a stopping criterion is met.

B. DESCRIPTION OF MLSHADE

Based on the ELSHADE-SPACMA [44], a novel algorithm, called multi-strategy LSHADE, is proposed. The main framework is divided into two phases, including LSHADE with semi-parameter adaptation hybrid with CMA-ES (LSHADE-SPACMA) [39] phase and adaptive guided differential evolution (AGDE) [45] phase. The main framework is described in [44]. In order to improve the performance of MLSHADE, three strategies are proposed as follows:

1) WEIGHTED MUTATION STRATEGY

For the variant of LSHADE, due to lack of the diversity of population, the process of search optimal falls into local optimum trap. Hence, we propose a weighted mutation strategy, called current-to-pbest-w/1.

$$v_i^G = x_i^G + Fw_i^G\left(x_{pbest}^G - x_i^G\right) + F_i^G\left(x_{r1}^G - x_{r2}^G\right) \tag{13}$$

where Fw is calculated:

$$Fw = \begin{cases} 0.7F & nfes < 0.2\max\_nfes \\ 0.8F & nfes < 0.4\max\_nfes \\ 1.2F & otherwise \end{cases} \tag{14}$$

where Fw is the first scaling factor which is employed to enhance the diversity of population in Eq. (6). In addition to this function, Fw has a significant impact on x_{pbest}. F is the second scaling factor. $nfes$ and $\max\_nfes$ represent the current number of fitness evaluations and the maximum number of fitness evaluations, respectively.

2) INFERIOR SOLUTION SEARCH STRATEGY

To enhance the diversity of population and overcome above weakness, the inferior solutions are selected to improve the performance of optimization process according to a criterion judging the stagnation of population. Regardless of the first phase and the second phase, a criterion judging the stagnation of the population is presented to help jump out of local optimum. The identifier of stagnation β is calculated as follows:

$$\beta = \begin{cases} 1 & \frac{1}{\left|S^{G+1}\right|}\sum_{i}^{\left|S^{G+1}\right|} f\left(x_i^{G+1}\right) < \frac{1}{\left|S^{G}\right|}\sum_{i}^{\left|S^{G}\right|} f\left(x_i^{G}\right) \\ 0 & other \end{cases} \tag{15}$$

where β is the identifier of stagnation. $\left|S^G\right|$ is the number of superior solutions at generation G. If the mean fitness value remains unchanged, $\beta = 0$ represents that algorithm falls into local optimum; otherwise, $\beta = 1$ denotes that condition of stagnation doesn't occur. For stagnation condition, the ISS strategy would be used. In ISS technique, parts of population are updated with inferior solutions, which do not participate in calculating the weighted mean value.

According to fitness values of solutions, the performance rank P_{rank} can distinguish the superior or inferior solutions.

$$P_{rank}\,(i) = (N - i + 1)\big/N \tag{16}$$

For the i^{th} solution, if $P_{rank}\,(i) < 0.5$, it is regarded as the superior solution and use the original technique of CMA-ES to generate a new vector. If $P_{rank}\,(i) > 0.5$, the individual represents the inferior solution, which can be employed to enhance the exploration ability of CMA-ES method. The last $N - \mu$ inferior solutions can be applied to two different update states. These states can be selected randomly. More details are presented as follows:

a). In state 1, new candidate is generated by superior and inferior solutions in Eigen coordinate. As you see in Fig.1(a), the mechanism can make algorithm have better exploration ability in Eigen coordinate system than in normal coordinate system. The detail of this state is described as

$$x_i^{eig,G+1} = r \cdot x_i^{eig} + (1-r)\, x_k^{eig} + N(0, \zeta) \tag{17}$$

where

$$x_i^{eig} = B^T x_i \tag{18}$$

$$\zeta = rand \cdot \left\| x_{best}^{eig} - x_i^{eig} \right\| \tag{19}$$

$$r = rand \cdot \left(1 - rand^{(1-nfes/\max\_nfes)}\right) \tag{20}$$

where $x_i^{eig,G+1}$ is the i^{th} solution of $G+1$ generation under Eigen coordinate. x_k^{eig} is randomly selected from set A. r value can gradually shift the addition of two first solutions in Eq. (10) from x_i to x_k to enhance offspring diversity. x_{best}^{eig} represents the superior solution **under Eigen coordinate**.

b). Unlike state 1, state 2 updates the shifted mean value by utilizing the difference between the inferior and superior solutions. As shown in Fig.1(b), the search strategy can't only enhance the convergence speed for unimodal problems, but also can enrich the diversity of population and improve the exploration ability for multimodal problems. The state can be modeled as

$$x_i^{G+1} = m^G + rand \cdot \left(x_k^G - x_i^G\right) + N\left(0, C^G\right) \tag{21}$$

where x_i^{G+1} is the i^{th} solution of $G+1$ generation. m^G, x_k^G, x_i^G and C^G are the mean of the selected superior solutions in A, the selected superior solution in set A, the i^{th} inferior solution and covariance matrix during G generation.

3) EIGEN GAUSSIAN RANDOM WALK STRATEGY

In the second phase, a Gaussian random walk and the Eigen coordinate system are presented to improve the exploitation performance of the AGDE. Under the Eigen coordinate system, the strategy makes full use of the superior solutions to access to promising candidates further. Meanwhile, the differential guidance directions between the randomly weighted superior solutions and middle solutions are introduced to improve the diversity of population.

$$x_i^{eig,G+1} = Gaussian\left(x_{pbest}^{eig,G}, \sigma\right) + rand_1 \cdot x_{pbest}^{eig,G} - rand_2 \cdot x_r^{eig,G},\ rand_1, rand_2 \sim U(0,1) \tag{22}$$

$$\sigma = \left| x_{pbest}^{eig,G} - x_{pworst}^{eig,G} \right| \tag{23}$$

where $x_r^{eig,G}$ is selected randomly from the middle NP-2*(100p%) at generation G under the eigen coordinate. $x_{pbest}^{eig,G}$ and $x_{pworst}^{eig,G}$ are chosen randomly from the best and worst 100p% under eigen coordinate.

Based on improved strategies and main framework, the pseudocode of MLSHADE is presented below.

Algorithm 1 MLSHADE

1. Settings: $N = N^{init}$, $nfes = 0$, max_nfes;
2. Initialize population according to Eq. (1) and evaluate the fitness function;
3. Settings: $p = p^{init}$, $M_{FCP} = 0.5$, $M_F = 0.5$, $M_{Cr} = 0.5$,
   Archive $A = \emptyset$; Initialize CMA-ES parameters;
4. **While (*nfes* < *max_nfes*) do**
   // First phase (LSHADE-SPACMA)
5. Calculate F, F_w, Cr and FCP; Calculate P_{rank}; Calculate the stagnation flag β;
6. Split the population into N_1 and N_2 according to the class probability variable;
   /* N_1 population updating for improved LSHADE*/
7. **for** $i = 1$ to N_1 **do**
8. Update the LSHADE population by using weighted mutation strategy described as Eq. (6);
9. **end for**
   /* N_2 population updating for improved CMA-ES*/
10. **for** $i = 1$ to N_2 **do**
11. **if** $P_{rank}(i) > 0.5$, update current individual by Eq. (4) and (5);
12. **else**
13. **if** $\beta \neq 1$
14. **if** $rand < 0.5$, update individual by state 1 of inferior solution search strategy in Eq. (10);
15. **else**, update individual by state 2 of inferior solution search strategy in Eq. (21); **end if**
16. **else,** update current individual by Eq. (4) and (5); **end if**
17. **end if**
18. **end for**
19. Concatenate the populations after mutation and generate the trial vectors by Eq. (3);
20. Check the boundary, evaluate the fitness function and update the individuals by selection operators;
21. Update the archive A, memory M_F, M_{Cr} and M_{FCP}; Calculate current population size N;
22. $nfes = nfes + 1$;
    // Second phase (GADE)
23. Calculate F, F_w, Cr and the stagnation flag β;
24. **for** $i = 1$ to N **do**
25. **if** $\beta \neq 1$, adopt the Eigen Gaussian random walk strategy in Eq. (22) and (23);
26. **else**, update current individual by GADE mutation strategy; **end if**
27. **end for**
28. Check the boundary, evaluate the fitness function and update the individuals by selection operators;
29. Update the archive A, memory M_F, M_{Cr} and M_{FCP}; Calculate current population size N;
30. $nfes = nfes + 1$;
31. **end while**
32. Output the best solution.

TABLE 1. Summary of the CEC 2018 test functions.

Type	No.	Functions	$f(x^*)$
Unimodal Functions	1	Shifted and Rotated Bent Cigar Function	100
	3	Shifted and Rotated Zakharov Function	300
Simple Multimodal Functions	4	Shifted and Rotated Rosenbrock's Function	400
	5	Shifted and Rotated Rastrigin's Function	500
	6	Shifted and Rotated Expanded Scaffer's F6 Function	600
	7	Shifted and Rotated Lunacek Bi_Rastrigin Function	700
	8	Shifted and Rotated Non-Continuous Rastrigin's Function	800
	9	Shifted and Rotated Levy Function	900
	10	Shifted and Rotated Schwefel's Function	1000
Hybrid Functions	11	Hybrid Function 1 (N=3)	1100
	12	Hybrid Function 2 (N=3)	1200
	13	Hybrid Function 3 (N=3)	1300
	14	Hybrid Function 4 (N=4)	1400
	15	Hybrid Function 5 (N=4)	1500
	16	Hybrid Function 6 (N=4)	1600
	17	Hybrid Function 6 (N=5)	1700
	18	Hybrid Function 6 (N=5)	1800
	19	Hybrid Function 6 (N=5)	1900
	20	Hybrid Function 6 (N=6)	2000
Composition Functions	21	Composition Function 1 (N=3)	2100
	22	Composition Function 2 (N=3)	2200
	23	Composition Function 3 (N=4)	2300
	24	Composition Function 4 (N=4)	2400
	25	Composition Function 5 (N=5)	2500
	26	Composition Function 6 (N=5)	2600
	27	Composition Function 7 (N=6)	2700
	28	Composition Function 8 (N=6)	2800
	29	Composition Function 9 (N=3)	2900
	30	Composition Function 10 (N=3)	3000

IV. EXPERIMENTAL STUDY

This section presents the experiments on modern CEC 2018 test suite [31] and parameters identification of photovoltaic models to integrally evaluate the MLSHADE proposed in this paper. All experiments are carried out on a Core(TM) i7-7700HQ GPU 2.80 GHz with 8 GB RAM, and the simulations are executed in MATLAB 2017a software.

A. EXPERIMENTAL STUDY USING CEC 2018

The challenging CEC 2018 50D test bed is employed to assess the performance of MLSHADE. The CEC 2018 test includes 29 benchmarks (without f_{CEC2}) which can be categorized into four groups. The detailed descriptions of this test suite are given in Table 1. As recommended by its proposers, this experimental study carries out 50D test with $D = 50$, where each benchmark should be run 51 times independently with same maximum function evaluations ($max\_nfes = D \times 10000$). Additionally, the fitness error is utilized to record the performance results. The fitness error can be defined as $f(x) - f(x^*)$, where $f(x)$ is the best fitness value obtained by current algorithms and $f(x^*)$ is the real global optimal of corresponding test function.

1) COMPARISONS OF MLSHADE WITH THE STATE-OF-THE-ART ALGORITHMS

The parameter settings of MLSHADE and compared algorithms are presented as follows:

(1) ELSHADE-SPACMA: Memory size $H = 5$, archive rate $Ar = 1.4$, initial class probability variable $M_{FCP}^{init} = 0.5$ and learning rate of class probability variable $c_{FCP} = 0.8$; Other parameter settings are denoted in previous sections [44];

(2) iLSHADE: Historical memory size $H = 6$, $M_F = 0.5$, $p^{init} = 0.2$, $p^{\min} = 0.1$ and $N^{init} = 12D$ as in [38];

(3) jSO: The *current-to-pBest-w/1* mutation strategy is adopted as in [40]; historical memory size $H = 5$, $M_F = 0.3$, $p^{init} = 0.25$, $p^{\min} = p^{init}/2$ and $N^{init} = 25\log(D)\sqrt{(D)}$;

(4) EBLSHADE: The *current-to-ord_pbest/1* mutation strategy is adopted as in [42]; the other parameter settings are similar as LSHADE in [37];

(5) GEDGWO: The parameter settings can be described in [46];

(6) NRO: Fission probability $P_{Fi} = 0.75$, β decay probability $P_\beta = 0.1$ and sinusoidal function frequency $freq = 0.05$; the parameter settings of Levy distribution strategy are set in [31];

(7) MLSHADE: The *current-to-pBest-w/1* mutation strategy is adopted as in [40]; the other parameter settings are similar as ELSHADE-SPACMA in [44].

To adequately highlight the performance of our work, a comparison with advanced modern algorithms is carried out. The statistical data, including the mean and SD values, are presented in Table 2. The **bold data** are the best solution in Table 2. For two unimodal functions, MLSHADE exhibits a well matched performance compared with ELSHADE-SPACMA, iLSHADE, jSO and EBLSHADE. For the seven multimodal functions f_{04} to f_{10}, MLSHADE achieves the best performance with the best results on four benchmarks, which include f_{05}, f_{07}, f_{08} and f_{10}. However, ELSHADE-SPACMA obtains better results in functions f_{06} and f_{09}. EBLSHADE has the best results in f_{04} and f_{09}. NRO and jSO obtain the best results in f_{09}. Additionally, MLSHADE also exhibits superiority in addressing hybrid functions except for f_{11}, f_{16} and f_{17}. In contrast to MLSHADE, NRO obtains the best results in f_{11}, f_{16} and f_{17}. For the last group of composite benchmarks, MLSHADE shows the superior performance in f_{21} and f_{26}. However, NRO can exert the advantage over other algorithms in most composite benchmarks. NRO obtains the best results in f_{23}-f_{25}, f_{27}-f_{28} and f_{30}. Meanwhile, ELSHADE-SPACMA obtains the best results in f_{22} and f_{29}. In Addition to statistical results in terms of mean and SD, the spider chart for mean values among all compared algorithms over CEC 2018 50D test is shown in Fig.S-1 (see in supplementary material). As you can see in Fig.S-1, the proposed MLSHADE wins the first ranks in $f_{01}, f_{05}, f_{03}, f_{07}, f_{08}, f_{10}, f_{12}$-$f_{15}, f_{18}$-$f_{22}$ and f_{26}. In conclusion, according to mean and SD, MLSHADE ranks the first. To further explore convergence performance, the convergence curves of the compared algorithms and MLSHADE over CEC 2018 50D test are given in Fig.S-2 (see in supplementary material). Although the proposed MLSHADE obtains better results in f_{01}, f_{03}, f_{07}, f_{08}, f_{10}, f_{12}-f_{15}, f_{18}-f_{22} and f_{26}, MLSHADE can rapidly converge to best solutions

TABLE 2. Experimental results evaluated by all compared algorithms in CEC 2018 50D test.

No.	ELSAHDE-SPACMA	iLSHADE	jSO	EBLSHADE	GEDGWO	NRO	MLSHADE
	Mean SD	Mean SD	Mean SD	Mean SD	Mean SD	Mean SD	Mean SD
01	**0.00e+00**0.00e+00	**0.00e+00**0.00e+00	**0.00e+00**0.00e+00	**0.00e+00**0.00e+00	1.72e+03 4.44e+03	4.71e-03 2.94e-03	**0.00e+00**0.00e+00
03	**0.00e+00**0.00e+00	**0.00e+00**0.00e+00	**0.00e+00**0.00e+00	**0.00e+00**0.00e+00	2.83e-03 9.57e-03	4.44e-03 2.66e-03	**0.00e+00**0.00e+00
04	4.27e+01 3.33e+01	7.04e+01 5.41e+01	4.86e+01 3.63e+01	**3.85e+01** 2.92e+01	6.96e+01 4.89e+01	5.89e+01 1.32e+00	4.81e+01 4.65e+01
05	1.46e+01 5.23e+00	1.33e+01 4.89e+00	2.47e+01 4.34e+00	2.83e+01 6.52e+00	8.44e+01 1.82e+01	3.89e+01 1.02e+01	**3.08e+00** 1.91e+00
06	**0.00e+00**0.00e+00	1.06e-04 5.98e-04	7.98e-06 7.63e-06	4.12e-07 4.74e-07	5.86e-01 3.15e-01	3.16e-08 2.78e-08	6.29e-07 1.00e-06
07	6.16e+01 3.43+00	6.43e+01 4.77e+00	6.74e+01 4.99e+00	7.67e+01 6.79e+00	1.23e+02 1.85e+01	7.81e+01 1.09e+01	**5.57e+01** 6.41e-01
08	1.54e+01 5.86e+00	1.38e+01 5.46e+00	2.41e+01 5.51e+00	2.74e+01 6.65e+00	8.20e+01 1.91e+01	3.88e+01 1.03e+01	**4.29e+00** 2.43e+01
09	**0.00e+00**0.00e+00	1.76e-03 1.25e-02	**0.00e+00**0.00e+00	**0.00e+00**0.00e+00	1.49e+01 1.22e+01	**0.00e+00**0.00e+00	8.91e-03 6.36e-02
10	3.66e+03 5.75e+02	3.99e+03 7.43e+02	4.69e+03 6.78e+02	3.03e+03 2.61e+02	5.52e+03 8.36e+02	2.52e+03 5.45e+02	**7.77e+02** 4.97e+02
11	2.58e+01 3.55e+00	4.14e+01 5.26e+00	3.62e+01 6.41e+00	2.70e+01 1.96e+00	4.57e+01 1.09e+01	**2.07e+01** 1.93e+01	2.33e+01 2.14e+00
12	1.33e+03 3.18e+02	2.05e+03 5.12e+02	1.79e+03 3.62e+02	1.40e+03 3.55e+02	5.41e+03 2.64e+03	1.68e+03 4.90e+02	**2.52e+02** 1.92e+02
13	4.02e+01 2.48e+01	4.59e+01 2.16e+01	4.23e+01 2.81e+01	6.22e+01 4.08e+01	2.31e+02 6.47e+01	7.09e+01 2.63e+01	**3.78e+01** 2.96e+01
14	3.09e+01 3.57e+00	2.84e+01 3.43e+00	2.90e+01 2.78e+00	2.62e+01 1.98e+00	6.03e+01 3.21e+01	3.46e+01 6.24e+00	**2.11e+01** 1.11e+00
15	2.31e+01 2.56e+00	3.67e+01 8.64e+00	3.22e+01 7.83e+00	2.39e+01 2.37e+00	1.04e+02 3.84e+01	2.30e+01 1.60e+01	**1.68e+01** 3.86e+00
16	4.40e+02 1.95e+02	2.56e+02 1.47e+02	4.62e+02 1.81e+02	3.28e+02 1.33e+02	9.54e+02 3.97e+02	**2.06e+02** 1.09e+02	5.32e+02 1.74e+02
17	2.32e+02 1.19e+02	2.14e+02 8.84e+01	2.94e+02 1.16e+02	2.44e+02 7.01e+01	7.91e+02 2.94e+02	**6.86e+01** 1.60e+01	3.32e+02 1.95e+02
18	2.38e+01 1.27e+00	3.44e+01 8.95e+00	2.81e+01 3.03e+00	2.45e+01 2.74e+00	5.08e+01 2.24e+01	3.29e+01 2.62e+00	**2.05e+01** 1.39e-01
19	1.48e+01 2.56e+00	2.22e+01 4.38e+00	1.84e+01 3.17e+00	1.72e+01 3.10e+00	5.48e+01 2.22e+01	1.60e+01 2.93e+00	**9.63e+00** 8.57e+00
20	9.74e+01 7.88e+01	8.88e+01 5.38e+01	1.91e+02 1.24e+02	2.10e+02 5.35e+01	5.16e+02 2.17e+02	8.73e+01 1.03e+02	**3.86e+01** 4.32e+01
21	2.39e+02 1.11e+01	2.17e+02 4.67e+00	2.25e+02 6.21e+00	2.28e+02 7.75e+00	2.88e+02 2.19e+01	2.39e+02 2.18e+01	**2.06e+02** 1.78e+00
22	**3.77e+01** 1.36e+03	1.11e+03 1.61e+03	1.16e+03 1.98e+03	1.52e+03 1.62e+03	4.83e+03 2.51e+03	1.14e+03 1.45e+03	1.00e+02 5.33e-09
23	4.58e+02 1.19e+01	4.46e+02 6.46e+00	4.44e+02 1.08e+01	4.40e+02 6.35e+00	5.11e+02 3.08e+01	**3.89e+02** 1.02e+01	4.30e+02 9.18e+00
24	5.31e+02 9.84e+00	5.18e+02 4.86e+00	5.12e+02 6.63e+00	5.08e+02 6.94e+00	5.71e+02 2.66e+01	**4.70e+02** 1.17e+01	4.90e+02 2.41e+00
25	4.81e+02 3.48e+00	4.81e+02 2.27e+00	4.99e+02 2.32e+01	4.82e+02 4.04e+00	5.14e+02 3.57e+01	**3.87e+02** 2.47e-02	5.45e+02 2.57e+01
26	1.37e+03 1.68e+02	1.25e+03 7.29e+01	1.26e+03 8.23e+01	1.23e+03 8.72e+01	1.96e+03 2.99e+02	1.41e+03 1.08e+02	**6.63e+02** 1.59e+02
27	5.09e+02 8.24e+00	5.40e+02 1.92e+01	5.24e+02 1.95e+01	5.22e+02 8.25e+00	5.15e+02 2.05e+01	**4.95e+02** 8.50e+00	5.87e+02 1.45e+01
28	4.59e+02 3.94e-13	4.78e+02 2.35e+01	4.63e+02 1.33e+01	4.67e+02 1.88e+01	4.70e+02 2.31e+01	**3.15e+02** 3.90e+01	5.02e+02 1.16e+01
29	**3.58e+02** 1.79e+01	3.67e+02 2.02e+01	4.40e+02 3.60e+01/	3.49e+02 8.11e+00	7.29e+02 2.38e+02	4.95e+02 6.91e+01	4.60e+02 1.52e+02
30	5.99e+05 2.91e+04	6.49e+05 7.21e+04	6.01e+05 3.93e+04	6.34e+05 5.69e+04	9.70e+05 1.43e+05	**2.04e+03** 6.76e+01	5.99e+05 1.76e+04
Average rank	2.90/3	4.03/3.69	4.03/3.83	3.41/**2.86**	6.69/6.86	3.41/4	**2.62/2.86**
Overall rank	2/3	5/4	5/5	3/**1**	7/7	3/6	**1/1**

only $f_{01}, f_{05}, f_{08}, f_{22}$ and f_{26}. Hence, the proposed MLSHADE can exert better performance at later stage.

In order to display the time consuming of MLSHADE and other competitors, the time consuming and time ranks for CEC 2018 50D are given in Table S-1 (see in supplementary material) and Fig.S-3 (see in supplementary material). Table S-1 gives the mean time required in seconds by MLSHADE and compared algorithms for 51 independent runs for 29 benchmark functions. As you can see in Table S-1 and Fig.S-3, iLSHADE requires the least time in the mass. The proposed MLSHADE has the second rank, which follows the iLSHADE algorithm. Although the MLSHADE does not rank the best (MLSHADE ranks the second) in term of time consuming, MLSHADE algorithm ranks the first in terms of mean values and SD. Based on the accuracy and reliability of solution for CEC 2018 50D test, the time consuming of MLSHADE is acceptable.

To highlight the efficacy of our proposed algorithm, the Wilcoxon signed rank test and the Friedman test are carried out through statistical methods [47]. The pairwise comparison results according to the Wilcoxon signed rank test with $\alpha = 0.05$ are presented in Table 3. Meanwhile, "R +" and "R-" denote the sums of ranks in the cases that MLSHADE performs better and worse respect to the corresponding competitor, respectively. In addition, "+", "−" and "≈" mean that MLSHADE outperforms better efficacy over the competitor, the competitor outperforms significantly better than MLSHADE and the performances of compared

TABLE 3. Comparison results of the Wilcoxon signed rank test ($\alpha = 0.05$).

MLSHADE	p-value	R+	R-	+/-/=
vs. ELSHADE-SPACMA	3.16e-01	247	159	17/11/1
vs. iLSHADE	7.20e-02	282	124	20/8/1
vs. jSO	5.58e-02	287	119	20/8/1
vs. EBLSHADE	1.11e-01	273	133	18/10/1
vs. GEDGWO	3.80e-05	408	27	26/3/0
vs. NRO	3.93e-01	257	178	18/11/0

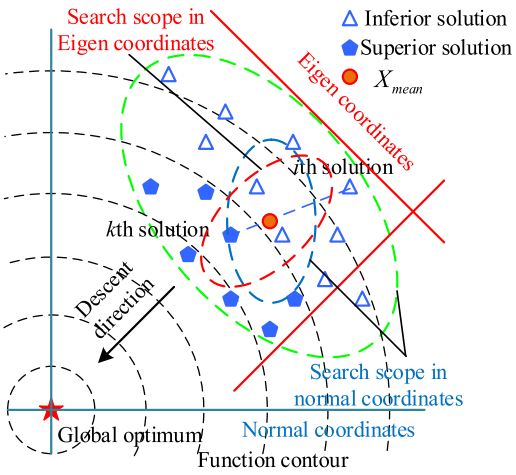
algorithms are similar, respectively. As shown in Table 3, although MLSHADE is not significantly better than the competitor in several test functions, MLSHADE obtains more "+"s than the competitor. Hence, MLSHADE exerts the better performance compared to other algorithms in Wilcoxon signed ranks test with the significance level $\alpha = 0.05$.

The Friedman test is employed to further determine the difference in multiple algorithms [47]. As a non-parametric multiple comparison technique, the Friedman test is executed according to the mean values, standard deviations (SD) and computation time costs. In Friedman test, a lower rank means a greater outperformance of the corresponding algorithm. Table 4 provides the average ranks evaluated by the Friedman test ($\alpha = 0.05$), where the chi-squares with the p-values 7.9152e-12, 9.2881e-13 and 2.0874e-21 is 63.71, 68.26 and 109.95 in terms of the mean value, SD and time, respectively. According to analysis, MLSHADE whose scores are 2.8276, 3.0345 and 2.3103 ranks the first in Mean, the first in SD and the second in Time, respectively. The differences between the different algorithms based on Iman Davenport

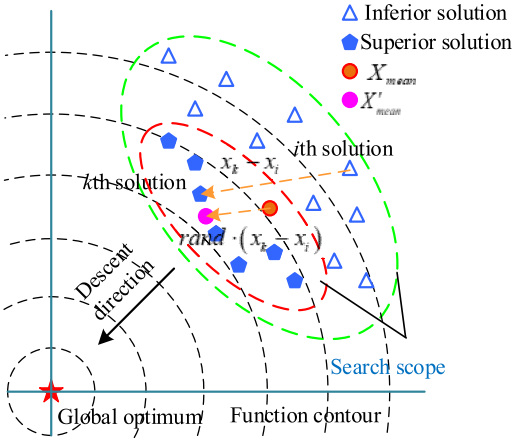
TABLE 4. The average ranks of terms of Mean for six algorithms according to the Friedman test ($\alpha = 0.05$).

Algorithms	ELSAHDE-SPACMA	iLSHADE	jSO	EBLSHADE	GEDGWO	NRO	MLSHADE
Mean ranks[a]	3.0690	4.1552	4.2069	3.5862	6.6897	3.4655	**2.8276**
SD ranks[b]	3.1724	3.8103	4	3.0690	6.8621	4.0517	**3.0345**
Time ranks[c]	6.8276	**1.9655**	3.7586	5.5517	3.5172	4.0690	2.3103

a. 1) p-value computed by the Friedman test: 7.9152e-12. Chi-square is 63.71.
2) F-distribution with 6 and 168 degrees of freedom is 16.1738.
3) p-value computed by the Iman-Davenport test: 5.9851e-08.
b. 1) p-value computed by the Friedman test: 9.2881e-13. Chi-square is 68.26.
2) F-distribution with 6 and 168 degrees of freedom is 18.0759.
3) p-value computed by the Iman-Davenport test: 1.8407e-08.
c. 1) p-value computed by the Friedman test: 2.0874e-21. Chi-square is 109.95.
2) F-distribution with 6 and 168 degrees of freedom is 48.0665.
3) p-value computed by the Iman-Davenport test: 1.8252e-13.



(a) State 1 can generate new candidate by superior and inferior solutions in Eigen coordinate.



(b) State 2 can update shift mean with difference vector between superior and inferior solutions.

FIGURE 4. Inferior solution search strategies.

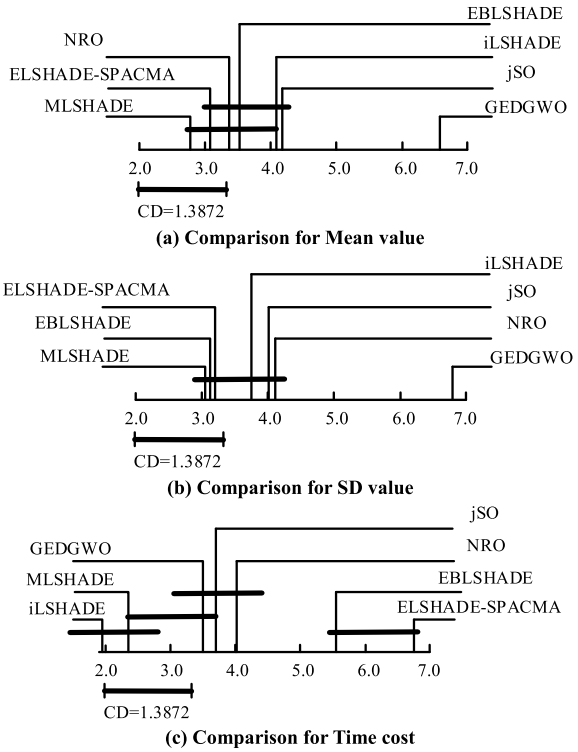


FIGURE 5. Multiple comparisons according to the post hoc Iman Davenport test.

test are illustrated in Fig.5, where the p-values are 5.9851e-08 for the mean value, 1.8407e-08 for standard deviation and 1.8252e-13 for time cost and the critical value (CD) value is 1.3872. According to CD value, Fig.5(a) shows little differences among ELSHADE-SPACMA, iLSHADE, NRO, EBLSHADE and MLSHADE in the mean value, and Fig.5(b) shows little differences among ELSHADE-SPACMA, iLSHADE, jSO, EBLSHADE, NRO and MLSHADE in the standard deviation. As you can see in Fig.5(c), there is no significant difference between iLSHADE and MLSHADE in the time cost.

2) MAIN COMPONENTS ANALYSIS OF MLSHADE

This subsection describes an experimental analysis of the effectiveness of three main components which are added to the original algorithm [48]. The three main components include weighted mutation strategy, inferior solution search strategy and Eigen Gaussian random walk strategy. Therefore, three different versions of MLSHADE are tested and compared against the proposed MLSHADE. The three different versions of MLSHADE are described as follows:

(1) MLSHADE-1, which is MLSHADE algorithm without weighted mutation strategy;

(2) MLSHADE-2, which is MLSHADE algorithm without inferior solution search strategy;

(3) MLSHADE-3, which is MLSHADE algorithm without Eigen Gaussian random walk strategy.

Meanwhile, the above three versions must be applied to solve the same problems with the same other parameters and two components seen in section 4.1.1. To test the performances of MLSHADE and its different versions, the CEC 2018 50D test suite is selected. Each function is tested independently 51 times by MLSHADE and its three versions. The statistical results obtained from the experiments are presented in Table S-2 (see in supplementary material). The statistical results are provided including the minimum, mean and standard deviation. As you can see in Table S-2, MLSHADE and its three versions obtain the best results in terms of the best value, mean value and standard deviation for unimodal functions. For simple multimodal functions, MLSHADE obtains the best ranks of all functions in term of Min, the best ranks of $f_{04}, f_{05}, f_{07}, f_{08}$ and f_{10} in term of Mean, and the best ranks of f_{05}, f_{07} and f_{08} in term of SD. However, MLSHADE-1 obtains the best results of f_{06} and f_{09} in Min, f_{09} in Mean and f_{09} in SD, respectively; MLSHADE-2 obtains the best results of f_{04}, f_{06} and f_{09} in Min, f_{09} in Mean and f_{09} in SD, respectively; MLSHADE-3 obtains the best results of f_{06} and f_{09} in Min, f_{06} and f_{09} in Mean and f_{04}, f_{06} and f_{09} in SD, respectively. For hybrid functions, MLSHADE gets the best results of f_{11}-f_{15} and f_{19}-f_{20} in Min, f_{11}-f_{12}, f_{14}-f_{15} and f_{18}-f_{20} in Mean, and f_{11}-f_{12}, f_{14}, f_{18} and f_{20} in SD, respectively. However, MLSHADE-1 gets the best results of f_{17} in Min, f_{13} in Mean and f_{13} and f_{19} in SD, respectively; MLSHADE-2 gets the best results of f_{16} and f_{18} in Min, f_{17} in Mean and f_{15}-f_{17} in SD, respectively; MLSHADE-3 gets the best result of f_{16} in Mean. For composition functions, MLSHADE gets the best results of f_{21}-f_{25} and f_{29} in Min, f_{21}-f_{24} in Mean and f_{21}-f_{22} and f_{24} in SD, respectively. However, MLSHADE-1 obtains the best results of f_{22}, f_{26}-f_{28} and f_{30} in Min, f_{22}, f_{25}, f_{27}-f_{28} and f_{30} in Mean and SD, respectively; MLSHADE-2 obtains the best results of f_{22} and f_{28} in Min, f_{22} in Mean and f_{22} and f_{29} in SD, respectively; MLSHADE-3 obtains the best results of f_{22} and f_{26} in Min and SD, f_{22}, f_{26} and f_{29} in Mean, respectively. According to overall ranks of Min, Mean and SD, MLSHADE obtains the first ranks. In term of Min, MLSHADE is followed by MLSHADE-2, MLSHADE-1 and MLSHADE-3. In term of Mean, MLSHADE-1, MLSHADE-2 and MLSHADE-3 rank the third, second and fourth, respectively. In term of SD, MLSHADE-1, MLSHADE-2 and MLSHADE-3 obtain the third, third and second ranks, respectively. As a whole, MLSHADE exerts the better performances in Min, Mean and SD.

To highlight the efficacy of our proposed algorithm and compare MLSHADE and its three versions, the Wilcoxon signed rank test is carried out. The pairwise comparison results according to the Wilcoxon signed rank test with $\alpha = 0.05$ are presented in Table S-3. As shown in Table S-3, although MLSHADE is not significantly better than its versions in several test functions, MLSHADE obtains more "+"s than the competitors. Hence, MLSHADE exerts the better performance compared to its component versions in Wilcoxon signed ranks test with the level that is not smaller than 0.05.

In general, several significant conclusions are tersely drawn as follows: First, MLSHADE outperforms MLSHADE-1, MLSHADE-2 and MLSHADE-3 in terms of Min, Mean and SD, which illustrates that the three components added to the original algorithm are very vital. Second, the performance of MLSHADE-3 is worst in terms of minimum and mean values, which means that the Eigen Gaussian random walk strategy has impact on accuracy of MLSHADE. Third, MLSHADE-1 and MLSHADE-2 perform the worst rank in term of standard deviation, which means that the weighted mutation strategy and inferior solution search strategy have an evident impact on stability of MLSHADE. Finally, MLSHAD-2 and MLSHADE-3 perform worse in composition functions, which means that the inferior solution search strategy and Eigen Gaussian random walk strategy have an impact on the ability of exploration. Because the addition of three components impacts the performance of MLSHADE, the algorithm can solve better the above optimization problems. In brief, three components are very important for MLSHADE.

TABLE 5. Algorithm complexity of MLSHADE.

Dimension	T_0	T_1	$\hat{T}_2$	$(\hat{T}_2 - T_1)/T_0$
D=10	0.1401s	0.6627s	2.9329s	16.2082
D=30		1.1320s	8.4809s	52.4670
D=50		**1.7979s**	9.9355s	58.0982
D=100		**6.1075s**	14.7987s	62.0504

3) ALGORITHM COMPLEXITY

In order to further investigate MLSHADE complexity, according to the guidelines of CEC competition [31], the computational complexity of MLSHADE is presented in Table 5. Firstly, T_0 denotes the running time of test code below (**Algorithm 2**).

Algorithm 2 Reference Code

1. **For** $i = 1{:}1000000$
2. $x = 0.55 + (\text{double})i$; $x = x + x$; $x = x/2$; $x = x * x$;
3. $x = sqrt(x)$; $x = log(x)$; $x = exp(x)$; $x = x/(x + 2)$;
4. **end for**

Additionally, T_1 denotes the computing time of 200000 evaluations just for f_{CEC18}. T_2 denotes the complete computing time for corresponding algorithm with 200000 evaluations of current dimension D in f_{CEC18}. $\hat{T}_2$ is obtained by the mean for T_2, which is evaluated five times. Finally, T_1, T_2 and $\hat{T}_2$ is listed as algorithm complexity in Table 5.

TABLE 6. Comparison among different algorithms on single diode model.

Algorithm	I_{ph}(A)	I_{sd}(μA)	R_s(Ω)	R_{sh}(Ω)	n	RMSE
MLSHADE	0.7608	0.3230	0.0364	53.7158	1.4812	**9.8602e-04**
NRO	0.7608	0.3230	0.0364	53.7185	1.4812	**9.8602e-04**
IJAYA	0.7608	0.3224	0.0364	53.4499	1.1481	9.8613e-04
TLBO	0.7607	0.03255	0.0363	53.9868	1.4819	9.8625e-04
CSA	0.7608	0.3230	0.0364	53.7186	1.4812	**9.8602e-04**
MLBSA	0.7608	0.3230	0.0364	53.7185	1.4812	**9.8602e-04**

B. EXPERIMENTAL STUDY ON PARAMETERS IDENTIFICATION OF PV MODELS

In this section, the experiments performed on parameters identification of different PV models including single diode, double diode and PV module are presented to prove the effectiveness of the proposed MLSHADE. In order to make a fair comparison, the benchmark experimental current-voltage data of a solar cell and a solar module are employed, which can be obtained in [48], where a 57 mm diameter commercial RTC France silicon solar cell and a solar module named Photowatt-PWP201 are included [48]. Meanwhile, the lower and upper bounds for each parameter are shown in [48].

To validate the superior performance of the proposed MLSHADE algorithms, the comparisons are carried out with other state-of-the-art algorithms including NRO [30], improved JAYA [14], teaching learning based optimization (TLBO) [19], crow search algorithm (CSA) [22] and multiple learning backtracking search algorithm (MLBSA) [21]. For fair comparison, the maximum number of function evaluations of all algorithms is set to 50000 in each run for each problem. Meanwhile, each algorithm is tested 30 times independently for each problem. The population sizes of all compared algorithms are set to be 50. The settings for the parameters for compared algorithms are presented as follows:

(1) NRO: Fission probability $P_{Fi} = 0.75$, β decay probability $P_\beta = 0.1$ and sinusoidal function frequency $freq = 0.05$; the parameter settings of Levy distribution strategy are set in [30];

(2) Improved JAYA: No parameter setting [14];

(3) TLBO: No parameter setting [19];

(4) CSA: Flight length $fl = 2$ and awareness probability $AP = 0.1$ [22];

(5) MLBSA: No parameter setting [21];

(6) MLSHADE: The *current-to-pBest-w/1* mutation strategy is adopted as in [40]; Memory size H and archive rate Arc_rate are set to 5 and 1.4; In first phase, the class probability variable FCP is set to 0.5, and learning rate c_{FCP}; In second phase, p value is set to 0.1; Finally, p^{init} and $p^{\min}$ are set to 0.3 and 0.15, respectively.

1) EXPERIMENTS ON THE SINGLE DIODE MODEL

For the single diode model, the comparison results involving the estimated parameters and RMSE are shown in Table 6. The overall best RMSE values among all compared algorithms are highlighted in **boldface**. As you can see in Table 6, MLSHADE obtains the best RMSE value same as NRO, CSA and MLBSA. The best RMSE is 9.8602e-04. To further prove the quality of results obtained, the best parameters of MLSHADE are used to reconstruct the current-voltage curve shown in Fig. 6. As seen in Fig. 6, the calculated data obtained by MLSHADE are highly in coincidence with the measured data over the whole voltage range. In addition, Table 7 presents the individual absolute error (IAE) between the measured data and calculated data. The 26 items represent data points which have been measured by experiments. In Table 7 where *Ic-s* represents the experimental data and IAE-s is the individual absolute error for single diode model, the IAE values are smaller than 2.5074e-03, which can validate the accuracy of estimated parameters.

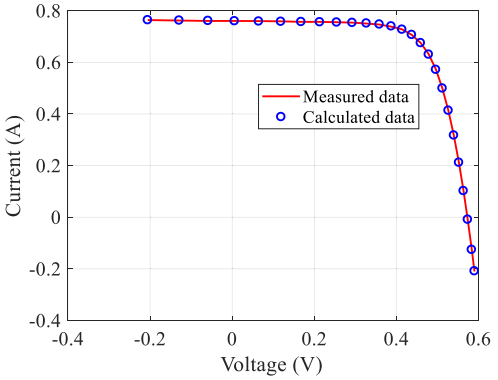


FIGURE 6. Comparisons between measured data and calculated data obtained by TSO for single diode model in current-voltage characteristics.

TABLE 7. Absolute error of NRO for each measurement on single and double diode model.

Item	*Vm(V)*	*Im(A)*	*Ic-s(A)*	*Ic-d(A)*	IAE-s	IAE-d
1	-0.2057	0.7640	0.7641	0.7640	8.7704e-05	1.6588e-05
2	-0.1291	0.7620	0.7627	0.7626	6.6309e-04	6.0410e-04
3	-0.0588	0.7605	0.7614	0.7613	8.5531e-04	8.3770e-04
4	0.0057	0.7605	0.7602	0.7602	3.4601e-04	3.2621e-03
5	0.0646	0.7600	0.7591	0.7591	9.4479e-04	8.9232e-04
6	0.1185	0.7590	0.7580	0.7581	9.5765e-04	8.7858e-04
7	0.1678	0.7570	0.7571	0.7572	9.1654e-05	1.8861e-04
8	0.2132	0.7570	0.7561	0.7562	8.5864e-04	7.5639e-04
9	0.2545	0.7555	0.7551	0.7552	4.1313e-04	3.2270e-04
10	0.2924	0.7540	0.7537	0.7537	3.3612e-04	2.7765e-04
11	0.3269	0.7505	0.7514	0.7514	8.9097e-04	8.9913e-04
12	0.3585	0.7465	0.7474	0.7473	8.5385e-04	8.0144e-04
13	0.3873	0.7385	0.7401	0.7400	1.6172e-03	1.5107e-03
14	0.4137	0.7280	0.7274	0.7272	6.1777e-04	7.5305e-04
15	0.4373	0.7065	0.7070	0.7069	4.7265e-04	3.5030e-04
16	0.4590	0.6755	0.6753	0.6752	2.1985e-04	2.8946e-04
17	0.4784	0.6320	0.6308	0.6308	1.2417e-03	1.2392e-03
18	0.4960	0.5730	0.5719	0.5720	1.0716e-03	1.0052e-03
19	0.5119	0.4990	0.4996	0.4997	6.0702e-04	7.0614e-04
20	0.5265	0.4130	0.4136	0.4137	6.4879e-04	7.3367e-04
21	0.5398	0.3165	0.3175	0.3175	1.0101e-03	1.0462e-03
22	0.5521	0.2120	0.2122	0.2121	1.5494e-04	1.2300e-04
23	0.5633	0.1035	0.1023	0.1022	1.2487e-03	1.3367e-03
24	0.5736	-0.0100	-0.0087	-0.0088	1.2824e-03	1.2082e-03
25	0.5833	-0.1230	-0.1255	-0.1255	2.5074e-03	2.5434e-03
26	0.5900	-0.2100	-0.2085	-0.2084	1.5277e-03	1.6284e-03

2) EXPERIMENTS ON THE DOUBLE DIODE MODEL

For the double diode model, the estimated parameters and the RMSE of all compared algorithms are presented in Table 8.

TABLE 8. Comparison among different algorithms on double diode model.

Algorithm	I_{ph}(A)	I_{sd1}(μA)	R_s(Ω)	R_{sh}(Ω)	n_1	I_{sd2}(μA)	n_2	RMSE
MLSHADE	0.7608	0.7494	0.0367	55.4854	2	0.2260	1.4510	**9.8248e-04**
NRO	0.7608	0.0433	0.0364	53.8266	1.9615	0.0317	1.4797	9.8571e-04
IJAYA	0.7608	0.4739	0.0366	54.6863	1.9948	0.2584	1.4623	9.8354e-04
TLBO	0.7608	0.3174	0.0364	54.1838	1.4797	0.0389	1.9987	9.8615e-04
CSA	0.7608	0.8281	0.0369	55.6744	1.9178	0.1827	1.4347	9.8467e-04
MLBSA	0.7608	0.7631	0.0367	55.5335	2	0.0044	1.4505	9.8249e-04

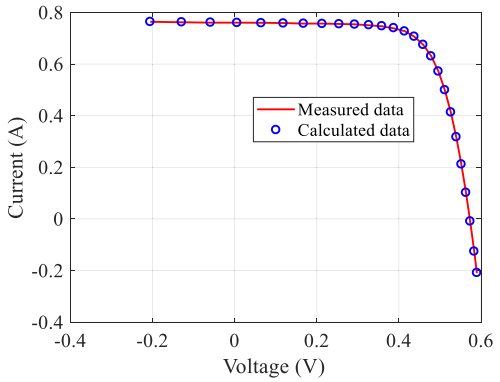


FIGURE 7. Comparisons between measured data and calculated data obtained by TSO for single diode model in current-voltage characteristics.

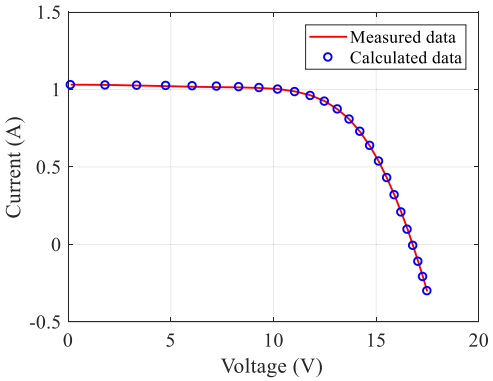


FIGURE 8. Comparisons between measured data and calculated data obtained by TSO for single diode model in current-voltage characteristics.

TABLE 9. Comparison among different algorithms on PV module model.

Algorithm	I_{ph}(A)	Isd(μA)	R_s(Ω)	R_{sh}(Ω)	n	RMSE
MLSHADE	1.0305	3.4823	1.2013	981.9823	48.6428	**2.4251e-03**
NRO	1.0305	3.4823	1.2013	981.9823	48.6428	**2.4251e-03**
IJAYA	1.0305	3.4676	1.2016	976.9201	48.6268	**2.4251e-03**
TLBO	1.0304	3.5421	1.1991	997.4387	48.7086	2.4259e-03
CSA	1.0305	3.4916	1.2011	989.6143	48.6529	**2.4251e-03**
MLBSA	1.0305	3.4823	1.2013	981.9822	48.6428	**2.4251e-03**

It is clear that MLSHADE gives the best RMSE value that is not obtained by other algorithms. The corresponding RMSE rank of MLSHADE is the first. The best RMSE of MLSHADE is 9.8248e-04. To further prove the quality of results obtained, the best parameters of MLSHADE are used to reconstruct the current-voltage curve shown in Fig. 7. As seen in Fig. 7, the calculated data obtained by MLSHADE are highly in coincidence with the measured data over the whole voltage range. Meanwhile, the IAE values of the double diode model (IAE-d) and the experimental data (*Ic-d*) are shown in Table 7. From Table 7, all the IAE values are smaller than 3.2621e-03.

3) EXPERIMENTS ON THE PV MODULE MODEL

For the PV module model, the estimated parameters and the RMSE values are shown in Table 9. It is clear that the RMSE value is 2.4251e-03, which can be obtained MLSHADE, NRO, IJAYA, CSA and MLBSA. To further prove the quality of results obtained, the best parameters of MLSHADE are used to reconstruct the current-voltage curve shown in Fig. 8. As seen in Fig. 8, the calculated data obtained by MLSHADE are highly in coincidence with the measured data over the whole voltage range. From Table 10, the IAE values obtained by MLSHADE are smaller than 4.8328e-03.

TABLE 10. Absolute error of NRO for each measurement on PV module model.

Item	*Vm(V)*	*Im(A)*	*Ic-PV(A)*	IAE-PV
1	0.1248	1.0315	1.0291	2.3808e-03
2	1.8093	1.0300	1.0274	2.6189e-03
3	3.3511	1.0260	1.0257	2.5820e-04
4	4.7622	1.0220	1.0241	2.1072e-03
5	6.0538	1.0180	1.0223	4.2918e-03
6	7.2364	1.0155	1.0199	4.4307e-03
7	8.3189	1.0140	1.0164	2.3631e-03
8	9.3097	1.0100	1.0105	4.9615e-04
9	10.2163	1.0035	1.0006	2.8710e-03
10	11.0449	0.9880	0.9845	3.4516e-03
11	11.8018	0.9630	0.9595	3.4783e-03
12	12.4929	0.9255	0.9228	2.6612e-03
13	13.1231	0.8725	0.8726	9.9663e-05
14	13.6983	0.8075	0.8073	2.2574e-04
15	14.2221	0.7265	0.7283	1.8365e-03
16	14.6995	0.6345	0.6371	2.6380e-03
17	15.1346	0.5345	0.5362	1.7131e-03
18	15.5311	0.4275	0.4295	2.0113e-03
19	15.8929	0.3185	0.3188	2.7448e-04
20	16.2229	0.2085	0.2074	1.1105e-03
21	16.5241	0.1010	0.0962	4.8328e-03
22	16.7987	-0.0080	-0.0083	3.2539e-04
23	17.0499	-0.1110	-0.1109	6.3517e-05
24	17.2793	-0.2090	-0.2092	2.4727e-04
25	17.4885	-0.3030	-0.3009	2.1364e-03

4) COMPARISONS OF MLSHADE WITH ITS COMPETITIONS

According to above analysis, the superior performance of MLSHADE in terms of accuracy is proved. In this subsection, the statistical results and computational time for all

TABLE 11. Statistical results of RMSE of all compared algorithms for three models.

Model	RMSE	MLSHADE	NRO	IJAYA	TLBO	CSA	MLBSA
Single diode model	Min	**9.8602e-04**	**9.8602e-04**	9.8613e-04	9.8625e-04	**9.8602e-04**	**9.8602e-04**
	Mean	**9.8602e-04**	**9.8602e-04**	1.0030e-03	1.0194e-03	1.0362e-03	**9.8602e-04**
	SD	**3.6556e-17**	2.8664e-14	6.1010e-05	8.1798e-05	1.0178e-04	3.0980e-14
Double diode model	Min	**9.8248e-04**	9.8571e-04	9.8354e-04	9.8615e-04	9.8467e-04	9.8249e-04
	Mean	**9.8336e-04**	1.0628e-03	1.0128e-03	1.0569e-03	1.3258e-03	9.8498e-04
	SD	**1.4528e-06**	9.0606e-06	6.7236e-05	7.2152e-05	5.1016e-04	1.3471e-05
PV module model	Min	**2.4251e-03**	**2.4251e-03**	**2.4251e-03**	2.4259e-03	**2.4251e-03**	**2.4251e-03**
	Mean	2.4373e-03	**2.4251e-03**	2.4377e-03	2.4468e-03	2.6458e-03	1.1487e-02
	SD	4.6443e-05	**5.0064e-17**	2.9308e-05	1.5596e-05	6.2070e-04	4.9628e-02
Average ranks of Min/Mean/SD		**1/1.33**/2	2.67/2.33/**1.67**	3/3.33/3.67	6.33/4.33/4	2/5.67/5.67	1.33/3/4
Overall ranks of Min/Mean/SD		1/1/2	4/2/**1**	5/4/3	6/5/4	3/6/6	2/3/4

TABLE 12. Computational time of all compared algorithms for three models.

Model	MLSHADE	NRO	IJAYA	TLBO	CSA	MLBSA
Single diode model	26.3561	28.7057	29.8712	**21.0288**	21.4116	32.0103
Double diode model	26.5465	25.4406	28.0638	22.3505	**21.2191**	37.1204
PV module model	25.8301	**13.7628**	28.3686	22.3412	22.0715	18.4379
Average/Overall ranks	4/4	2.67/3	5.33/6	2.33/2	**2/1**	4.67/5

compared algorithms over 30 independent runs are shown in Table 11 and Table 12. As you can see in Table 11, for single diode model, MLSHADE has same accuracy as NRO and MLBSA. Compared with NRO and MLBSA, the MLSHADE outperforms in term of standard deviation. For double diode model, the mean and minimum RMSE of MLSHADE ranks the first. In term of standard deviation, the MLSHADE ranks the first. For PV module model, although MLSHADE obtains the same minimum RMSE as all compared algorithms except for TLBO, the mean of TSO ranks the second, which follows NRO. In the mass, TSO performs the better results in term of mean value for single and double diode models. Meanwhile, MLSHADE exerts the better performance in term of the minimum RMSE over all models. Overall ranks of MLSHADE are the first, the first and the second in terms of the Min, the Mean and the SD. To further verify and compare the convergence of all compared algorithms, the convergence curves among all competitors for three PV models are shown in Fig.9. As you can see in Fig.9(a), for single diode model, MLSHADE, MLBSA and NRO obtain the best results. In Fig.9(b), for double diode model, MLSHADE obtains the best results. In Fig.9(c), for PV module model, NRO converges rapidly to best result. However, MLSHADE reach the best result at 12000 numbers of function evaluations. Besides, the computational time of all compared algorithms for three models is shown in Table 12. As can be seen in Table 12, TLBO, CSA and NRO have the first ranks for single diode model, double diode model and PV module model, respectively. In all, the MLSHADE ranks the forth in term of time consuming. Hence, MLSHADE can obtain a competitive performance in terms of robustness and accuracy.

To highlight the efficacy of our proposed algorithm, the Wilcoxon signed rank test is employed [47]. A summary of the performance comparisons of all the algorithms is presented in Table 13. In Table 13, "R +" and "R−" denote the sums of the ranks in the cases in which the

TABLE 13. The result of the Wilcoxon signed ranks test of compared algorithms and NRO for three models with 30 runs.

Model	MLSHADE vs.	R+	R-	p-value	Win
Single diode model	NRO	465	0	1.73e-06	+
	IJAYA	465	0	1.73e-06	+
	TLBO	465	0	1.73e-06	+
	CSA	465	0	1.73e-06	+
	MLBSA	465	0	1.73e-06	+
Double diode model	NRO	465	0	1.73e-06	+
	IJAYA	465	0	1.92e-06	+
	TLBO	465	0	1.73e-06	+
	CSA	396	69	1.73e-06	+
	MLBSA	432	33	7.71e-04	+
PV module model	NRO	402	63	4.86e-04	+
	IJAYA	406	59	3.59e-04	+
	TLBO	406	59	3.59e-04	+
	CSA	443	22	1.49e-05	+
	MLBSA	407	58	3.31e-04	+

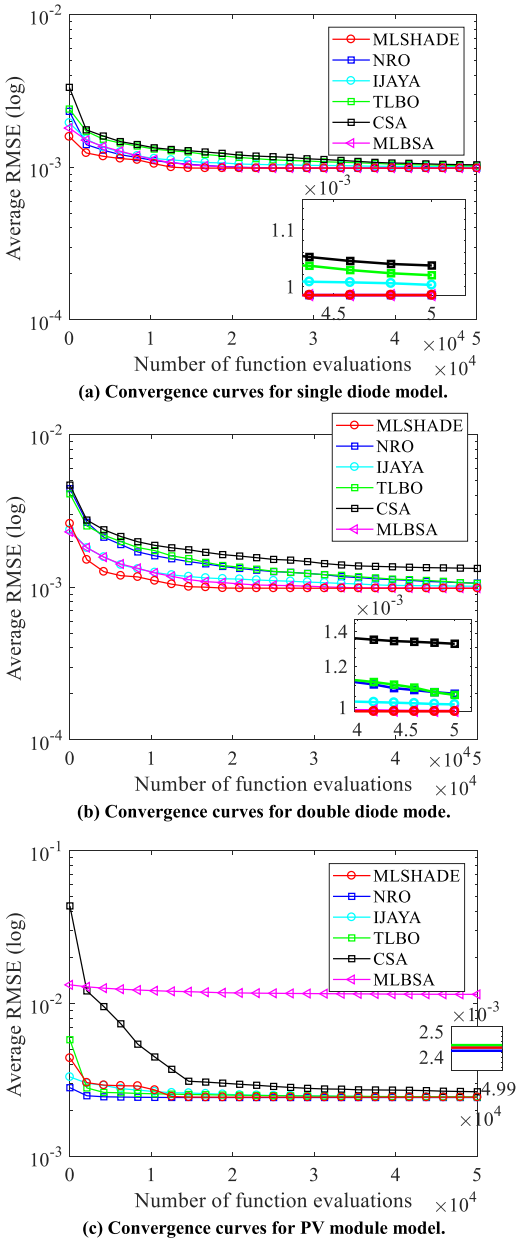
MLSHADE performs better and worse than its competitors, respectively [47]. '+' of Win item means the NRO is superior to the other algorithms. Over all models, MLSHADE obtains all "+"s. Hence, the MLSHADE algorithm significantly outperforms the other algorithms in the Wilcoxon signed ranks test with the significance level $\alpha = 0.05$.

To further determine the difference in multiple algorithms, the Friedman test is employed [47]. In Friedman test, a lower rank means a greater outperformance of the corresponding algorithm. Table 14 provides the average ranks evaluated by the Friedman test ($\alpha = 0.05$), where the chi-squares with the p-values 0.0193, 0.1129, 0.34 and 0.0477 is 13.48, 8.9, 5.67 and 11.19 in terms of the minimum value, mean value, SD and time, respectively. Because the p-values of Mean and SD are not smaller than 0.05, the differences among all competitors are not significant. According to analysis, MLSHADE whose scores are 1, 1.3333, 2.3333 and 4 ranks the first in Min, the first in Mean, the first in SD and the second in Time, respectively. The p-values based on Iman-Davenport test are 0.0544 for the minimum value,

TABLE 14. The average ranks of terms of Mean for six algorithms according to the Friedman test ($\alpha = 0.05$).

Algorithms	MLSHADE	NRO	IJAYA	TLBO	CSA	MLBSA
Min ranks[a]	**1**	3.6667	4	6	4.3333	2
Mean ranks[b]	**1.3333**	3	3.3333	4.3333	5.6667	3.3333
SD ranks[c]	**2.3333**	3	3.3333	3.6667	5.6667	3
Time ranks[d]	4	3	5.3333	**1**	2.6667	5

a. 1) p-value computed by the Friedman test: 0.0193. Chi-square is 13.48.
2) F-distribution with 6 and 168 degrees of freedom is 17.6875.
3) p-value computed by the Iman-Davenport test: 0.0544.
b. 1) p-value computed by the Friedman test: 0.1129. Chi-square is 8.9.
2) F-distribution with 6 and 168 degrees of freedom is 2.9219.
3) p-value computed by the Iman-Davenport test: 0.2744.
c. 1) p-value computed by the Friedman test: 0.34. Chi-square is 5.67.
2) F-distribution with 6 and 168 degrees of freedom is 1.2143.
3) p-value computed by the Iman-Davenport test: 0.5093.
d. 1) p-value computed by the Friedman test: 0.0477. Chi-square is 11.19.
2) F-distribution with 6 and 168 degrees of freedom is 5.8750.
3) p-value computed by the Iman-Davenport test: 0.1518.



(a) Convergence curves for single diode model.

(b) Convergence curves for double diode mode.

(c) Convergence curves for PV module model.

FIGURE 9. Convergence curves among all competitors.

0.2744 for the mean value, 0.5093 for standard deviation and 0.1518 for time cost and the critical value (CD) value is 29.4758. According to CD value and p-value based on Iman-Davenport test, the differences among all competitors are not evident.

V. CONCLUSION

This paper introduces a powerful optimization technique, multi-strategy success-history based adaptive DE with linear population size reduction, which is a variant of LSHADE, to accurately and steadily estimate the parameters of different PV models. This algorithm possesses a two-phase framework to balance exploration and exploitation. The weighted mutation strategy is employed to improve the diversity of improved LSAHDE subpopulation in first phase. The inferior solution search strategy is employed to improve the exploration ability of CMA-ES subpopulation in first phase. In the second phase, Eigen Gaussian random walk strategy is employed to improve the exploitation performance of the second phase. The MLSHADE algorithm is evaluated through parameters identification problems of single diode, double diode, and PV module models. Experiment results illustrate that MLSHADE has better performance in terms of accuracy and reliability against other well-established algorithms.

Future works may focus on improvement of MLSHADE. Although MLSHADE has better accuracy, the MLSHADE algorithm exerts the inferior reliability against NRO in PV module model. Hence, the improvement of MLSHADE should be further addressed and discussed in the future.

REFERENCES

[1] D. H. Muhsen, A. B. Ghazali, T. Khatib, and I. A. Abed, "Parameters extraction of double diode photovoltaic module's model based on hybrid evolutionary algorithm," *Energy Convers. Manag.*, vol. 105, pp. 552–561, Nov. 2015, doi: 10.1016/j.enconman.2015.08.023.

[2] A. Youssef, M. El-Telbany, and A. Zekry, "The role of artificial intelligence in photo-voltaic systems design and control: A review," *Renew. Sustain. Energy Rev.*, vol. 78, pp. 72–79, Oct. 2017, doi: 10.1016/j.rser.2017.04.046.

[3] Z. Chen, L. Wu, S. Cheng, P. Lin, Y. Wu, and W. Lin, "Intelligent fault diagnosis of photovoltaic arrays based on optimized kernel extreme learning machine and I-V characteristics," *Appl. Energy*, vol. 204, pp. 912–931, Oct. 2017, doi: 10.1016/j.apenergy.2017.05.034.

[4] Z. Chen, Y. Chen, L. Wu, S. Cheng, and P. Lin, "Deep residual network based fault detection and diagnosis of photovoltaic arrays using current-voltage curves and ambient conditions," *Energy Convers. Manage.*, vol. 198, Oct. 2019, Art. no. 111793, doi: 10.1016/j.enconman.2019.111793.

[5] Z. Chen, F. Han, L. Wu, J. Yu, S. Cheng, P. Lin, and H. Chen, "Random forest based intelligent fault diagnosis for PV arrays using array voltage and string currents," *Energy Convers. Manage.*, vol. 178, pp. 250–264, Dec. 2018, doi: 10.1016/j.enconman.2018.10.040.

[6] Z. Chen, L. Wu, P. Lin, Y. Wu, and S. Cheng, "Parameters identification of photovoltaic models using hybrid adaptive Nelder-mead simplex algorithm based on eagle strategy," *Appl. Energy*, vol. 182, pp. 47–57, Nov. 2016, doi: 10.1016/j.apenergy.2016.08.083.

[7] D. F. Alam, D. A. Yousri, and M. B. Eteiba, "Flower pollination algorithm based solar PV parameter estimation," *Energy Convers. Manage.*, vol. 101, pp. 410–422, Sep. 2015, doi: 10.1016/j.enconman.2015.05.074.

[8] M. R. AlRashidi, M. F. AlHajri, K. M. El-Naggar, and A. K. Al-Othman, "A new estimation approach for determining the I-V characteristics of solar cells," *Sol. Energy*, vol. 85, no. 7, pp. 1543–1550, Jul. 2011, doi: 10.1016/j.solener.2011.04.013.

[9] A. Abbassi, R. Gammoudi, M. Ali Dami, O. Hasnaoui, and M. Jemli, "An improved single-diode model parameters extraction at different operating conditions with a view to modeling a photovoltaic generator: A comparative study," *Sol. Energy*, vol. 155, pp. 478–489, Oct. 2017, doi: 10.1016/j.solener.2017.06.057.

[10] K. Ishaque, Z. Salam, S. Mekhilef, and A. Shamsudin, "Parameter extraction of solar photovoltaic modules using penalty-based differential evolution," *Appl. Energy*, vol. 99, pp. 297–308, Nov. 2012, doi: 10.1016/j.apenergy.2012.05.017.

[11] S. Pindado and J. Cubas, "Simple mathematical approach to solar cell/panel behavior based on datasheet information," *Renew. Energy*, vol. 103, pp. 729–738, Apr. 2017, doi: 10.1016/j.renene.2016.11.007.

[12] J. Meng, J. Feng, Q. Sun, Z. Pan, and T. Liu, "Degradation model of the orbiting current for GaInP/GaAs/Ge triple-junction solar cells used on satellite," *Sol. Energy*, vol. 122, pp. 464–471, Dec. 2015, doi: 10.1016/j.solener.2015.09.028.

[13] J. A. Jervase, H. Bourdoucen, and A. Al-Lawati, "Solar cell parameter extraction using genetic algorithms," *Meas. Sci. Technol.*, vol. 12, no. 11, pp. 1922–1925, Oct. 2001, doi: 10.1088/0957-0233/12/11/322.

[14] K. Yu, J. J. Liang, B. Y. Qu, X. Chen, and H. Wang, "Parameters identification of photovoltaic models using an improved JAYA optimization algorithm," *Energy Convers. Manage.*, vol. 150, pp. 742–753, Oct. 2017, doi: 10.1016/j.enconman.2017.08.063.

[15] L. L. Jiang, D. L. Maskell, and J. C. Patra, "Parameter estimation of solar cells and modules using an improved adaptive differential evolution algorithm," *Appl. Energy*, vol. 112, pp. 185–193, Dec. 2013, doi: 10.1016/j.apenergy.2013.06.004.

[16] A. Askarzadeh and A. Rezazadeh, "Artificial bee swarm optimization algorithm for parameters identification of solar cell models," *Appl. Energy*, vol. 102, pp. 943–949, Feb. 2013, doi: 10.1016/j.apenergy.2012.09.052.

[17] Q. Niu, L. Zhang, and K. Li, "A biogeography-based optimization algorithm with mutation strategies for model parameter estimation of solar and fuel cells," *Energy Convers. Manag.*, vol. 86, pp. 1173–1185, Oct. 2014, doi: 10.1016/j.enconman.2014.06.026.

[18] Q. Niu, H. Zhang, and K. Li, "An improved TLBO with elite strategy for parameters identification of PEM fuel cell and solar cell models," *Int. J. Hydrogen Energy*, vol. 39, no. 8, pp. 3837–3854, Mar. 2014, doi: 10.1016/j.ijhydene.2013.12.110.

[19] S. J. Patel, A. K. Panchal, and V. Kheraj, "Extraction of solar cell parameters from a single current-voltage characteristic using teaching learning based optimization algorithm," *Appl. Energy*, vol. 119, pp. 384–393, Apr. 2014, doi: 10.1016/j.apenergy.2014.01.027.

[20] R. Abbassi, A. Abbassi, A. A. Heidari, and S. Mirjalili, "An efficient salp swarm-inspired algorithm for parameters identification of photovoltaic cell models," *Energy Convers. Manage.*, vol. 179, pp. 362–372, Jan. 2019, doi: 10.1016/j.enconman.2018.10.069.

[21] K. Yu, J. J. Liang, B. Y. Qu, Z. Cheng, and H. Wang, "Multiple learning backtracking search algorithm for estimating parameters of photovoltaic models," *Appl. Energy*, vol. 226, pp. 408–422, Sep. 2018, doi: 10.1016/j.apenergy.2018.06.010.

[22] A. Omar, H. M. Hasanien, M. A. Elgendy, and M. A. L. Badr, "Identification of the photovoltaic model parameters using the crow search algorithm," *J. Eng.*, vol. 2017, no. 13, pp. 1570–1575, 2017, doi: 10.1049/joe.2017.0595.

[23] L. Wu, Z. Chen, C. Long, S. Cheng, P. Lin, Y. Chen, and H. Chen, "Parameter extraction of photovoltaic models from measured I-V characteristics curves using a hybrid trust-region reflective algorithm," *Appl. Energy*, vol. 232, pp. 36–53, Dec. 2018, doi: 10.1016/j.apenergy.2018.09.161.

[24] X. Gao, Y. Cui, J. Hu, G. Xu, and Y. Yu, "Lambert W-function based exact representation for double diode model of solar cells: Comparison on fitness and parameter extraction," *Energy Convers. Manage.*, vol. 127, pp. 443–460, Nov. 2016, doi: 10.1016/j.enconman.2016.09.005.

[25] J. Appelbaum and A. Peled, "Parameters extraction of solar cells—A comparative examination of three methods," *Sol. Energy Mater. Sol. Cells*, vol. 122, pp. 164–173, Mar. 2014, doi: 10.1016/j.solmat.2013.11.011.

[26] M. F. AlHajri, K. M. El-Naggar, M. R. AlRashidi, and A. K. Al-Othman, "Optimal extraction of solar cell parameters using pattern search," *Renew. Energy*, vol. 44, pp. 238–245, Aug. 2012, doi: 10.1016/j.renene.2012.01.082.

[27] F. Almonacid, E. F. Fernández, T. K. Mallick, and P. J. Pérez-Higueras, "High concentrator photovoltaic module simulation by neuronal networks using spectrally corrected direct normal irradiance and cell temperature," *Energy*, vol. 84, pp. 336–343, May 2015, doi: 10.1016/j.energy.2015.02.105.

[28] F. Bonanno, G. Capizzi, G. Graditi, C. Napoli, and G. M. Tina, "A radial basis function neural network based approach for the electrical characteristics estimation of a photovoltaic module," *Appl. Energy*, vol. 97, pp. 956–961, Sep. 2012, doi: 10.1016/j.apenergy.2011.12.085.

[29] Z. Chen, Y. Chen, L. Wu, S. Cheng, P. Lin, and L. You, "Accurate modeling of photovoltaic modules using a 1-D deep residual network based on I-V characteristics," *Energy Convers. Manage.*, vol. 186, pp. 168–187, Apr. 2019, doi: 10.1016/j.enconman.2019.02.032.

[30] Z. Wei, C. Huang, X. Wang, and H. Zhang, "Parameters identification of photovoltaic models using a novel algorithm inspired from nuclear reaction," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jun. 2019, doi: 10.1109/CEC.2019.8790223.

[31] Z. Wei, C. Huang, X. Wang, T. Han, and Y. Li, "Nuclear reaction optimization: A novel and powerful physics-based algorithm for global optimization," *IEEE Access*, vol. 7, pp. 66084–66109, 2019, doi: 10.1109/ACCESS.2019.2918406.

[32] A. Kumar, R. K. Misra, D. Singh, S. Mishra, and S. Das, "The spherical search algorithm for bound-constrained global optimization problems," *Appl. Soft Comput.*, vol. 85, Dec. 2019, Art. no. 105734, doi: 10.1016/j.asoc.2019.105734.

[33] A. W. Mohamed, A. A. Hadi, and A. K. Mohamed, "Gaining-sharing knowledge based algorithm for solving optimization problems: A novel nature-inspired algorithm," *Int. J. Mach. Learn. Cybern.*, pp. 1–29, Dec. 2019, doi: 10.1007/s13042-019-01053-x.

[34] R. Storn and K. Price, "Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces," *J. Global Optim.*, vol. 11, no. 4, pp. 341–359, 1997, doi: 10.1023/A:1008202821328.

[35] P. Larrañaga and J. A. Lozano, *Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation*. Alphen aan den Rijn, The Netherlands: Kluwer, 2002.

[36] R. Tanabe and A. S. Fukunaga, "Improving the search performance of SHADE using linear population size reduction," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jul. 2014, pp. 1658–1665, doi: 10.1109/CEC.2014.6900380.

[37] J. Brest, M. S. Maucec, and B. Boskovic, "IL-SHADE: Improved L-SHADE algorithm for single objective real-parameter optimization," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jul. 2016, pp. 1188–1195, doi: 10.1109/CEC.2016.7743922.

[38] A. W. Mohamed, A. A. Hadi, A. M. Fattouh, and K. M. Jambi, "LSHADE with semi-parameter adaptation hybrid with CMA-ES for solving CEC 2017 benchmark problems," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jun. 2017, pp. 145–152, doi: 10.1109/CEC.2017.7969307.

[39] N. H. Awad, M. Z. Ali, P. N. Suganthan, and R. G. Reynolds, "An ensemble sinusoidal parameter adaptation incorporated with L-SHADE for solving CEC2014 benchmark problems," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jul. 2016, pp. 2958–2965, doi: 10.1109/CEC.2016.7744163.

[40] J. Brest, M. S. Maučec, and B. Bošković, "Single objective real-parameter optimization: Algorithm jSO," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jun. 2017, pp. 1311–1318, doi: 10.1109/CEC.2017.7969456.

[41] X. Wang, H. Zhao, T. Han, Z. Wei, Y. Liang, and Y. Li, "A Gaussian estimation of distribution algorithm with random walk strategies and its application in optimal missile guidance handover for multi-UCAV in over-the-horizon air combat," *IEEE Access*, vol. 7, pp. 43298–43317, 2019, doi: 10.1109/ACCESS.2019.2908262.

[42] A. W. Mohamed, A. A. Hadi, and K. M. Jambi, "Novel mutation strategy for enhancing SHADE and LSHADE algorithms for global numerical optimization," *Swarm Evol. Comput.*, vol. 50, Nov. 2019, Art. no. 100455, doi: 10.1016/j.swevo.2018.10.006.

[43] N. Hansen and A. Ostermeier, "Completely derandomized self-adaptation in evolution strategies," *Evol. Comput.*, vol. 9, no. 2, pp. 159–195, Jun. 2001, doi: 10.1162/106365601750190398.

[44] A. A. Hadi, A. Wagdy, and K. Jambi, "Single-objective real-parameter optimization: Enhanced LSHADE-SPACMA algorithm," IEEE CEC 2018, Rio de Janeiro, Brazil, Tech. Rep. 2018, 2018, doi: 10.13140/RG.2.2.33283.20005.

[45] A. W. Mohamed and A. K. Mohamed, "Adaptive guided differential evolution algorithm with novel mutation for numerical optimization," *Int. J. Mach. Learn. Cybern.*, vol. 10, no. 2, pp. 253–277, Aug. 2017, doi: 10.1007/s13042-017-0711-7.

[46] X. Wang, H. Zhao, T. Han, H. Zhou, and C. Li, "A grey Wolf optimizer using Gaussian estimation of distribution and its application in the multi-UAV multi-target urban tracking problem," *Appl. Soft Comput.*, vol. 78, pp. 240–260, May 2019, doi: 10.1016/j.asoc.2019.02.037.

[47] J. Derrac, S. García, D. Molina, and F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm Evol. Comput.*, vol. 1, no. 1, pp. 3–18, Mar. 2011, doi: 10.1016/j.swevo.2011.02.002.

[48] A. W. Mohamed and A. S. Almazyad, "Differential evolution with novel mutation and adaptive crossover strategies for solving large scale global optimization problems," *Appl. Comput. Intell. Soft Comput.*, vol. 2017, pp. 1–18, Mar. 2017, doi: 10.1155/2017/7974218.

[49] T. Easwarakhanthan, J. Bottin, I. Bouhouch, and C. Boutrit, "Nonlinear minimization algorithm for determining the solar cell parameters with microcomputers," *Int. J. Sol. Energy*, vol. 4, no. 1, pp. 1–12, Jan. 1986, doi: 10.1080/01425918608909835.