frontiers | Frontiers in Energy Research

ORIGINAL RESEARCH
published: 11 May 2022
doi: 10.3389/fenrg.2022.794732

Parameters Identification of Battery Model Using a Novel Differential Evolution Algorithm Variant

Junfeng Zhou[1,2,3,4], *Yubo Zhang*[1], *Yuanjun Guo*[2,3,4], *Wei Feng*[2,3,4], *Muhammad Ilyas Menhas*[5] *and Yanhui Zhang*[2,3,4]*

[1]School of Electrical Engineering, Zhengzhou University, Zhengzhou, China, [2]Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China, [3]Guangdong Provincial Key Lab of Robotics and Intelligent System, Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China, [4]CAS Key Laboratory of Human-Machine Intelligence-Synergy Systems, Shenzhen Institute of Advanced Technology, Shenzhen, China, [5]Department of Electrical Engineering, Mirpur University of Science and Technology, Mirpur, Pakistan

In order to deal with the fluctuation and intermittency of photovoltaic (PV) cells, the battery energy storage system (BESS) as a supplementary power source has been widely concerned. In BESS, the unknown parameters of the battery can affect its output, and its structure determines these parameters. Therefore, it is essential to establish the battery model and extract the parameters accurately, and the existing methods cannot effectively solve this problem. This study proposes an adaptive differential evolution algorithm with the dynamic opposite learning strategy (DOLADE) to deal with the issue. In DOLADE, the number of elite particles and particles with poor performance is expanded, the population's search area is increased, and the population's exploration capability is improved. The particles' search area is dynamically changed to ensure the population has a good exploitation capability. The dynamic opposite learning (DOL) strategy increases the population's diversity and improves the probability of obtaining the global optimum with a considerable convergence rate. The various discharging experiments are performed, the battery model parameters are identified, and the results are compared with the existing well-established algorithms. The comprehensive results indicate that DOLADE has excellent performance and could deal with similar problems.

Keywords: parameter identification, battery model, dynamic opposite learning, differential evolution, battery energy storage system

OPEN ACCESS

Edited by:
Kang Li,
University of Leeds, United Kingdom

Reviewed by:
Karthik Balasubramanian,
McMaster University, Canada
Yaxing Ren,
University of Warwick, United Kingdom

***Correspondence:**
Yanhui Zhang
zhangyh@siat.ac.cn

Specialty section:
This article was submitted to Electrochemical Energy Conversion and Storage, a section of the journal Frontiers in Energy Research

Received: *14 October 2021*
Accepted: *29 March 2022*
Published: *11 May 2022*

Citation:
Zhou J, Zhang Y, Guo Y, Feng W, Menhas MI and Zhang Y (2022) Parameters Identification of Battery Model Using a Novel Differential Evolution Algorithm Variant. Front. Energy Res. 10:794732. doi: 10.3389/fenrg.2022.794732

1 INTRODUCTION

Nowadays, energy has become a vital driving force in social development and progress, and fossil fuels could bring greenhouse effects and environmental pollution, so the application of renewable energy is becoming more and more widespread (Bullich-Massagué et al., 2020; Maghyereh and Abdoh, 2021). Globally, solar energy is the third largest clean resource in addition to wind energy and tidal energy, whose volatility and intermittent nature will limit its further development (Siecker et al., 2017; Zhou et al., 2021a). The battery energy storage system (BESS) can be used as a supplementary power source to alleviate the volatility and intermittency of photovoltaic (PV) cells, providing a new possibility to make the system stable (Farhadi et al., 2019). The complementary working mode of PV cells and BESS can improve the solar utilization efficiency (Cheng et al., 2021) to deal with the imbalance of power output in different environments such as the

wide range of light intensity change and the large temperature difference between day and night. The application of the system accelerates the pace of clean energy replacing traditional energy and can effectively alleviate the energy crisis (Xia and Xiang, 2020).

Lithium-ion batteries are the best devices for supplementing the power supply due to their sufficient life span and power/energy density (Li et al., 2021). In recent years, with the deepening of the research, in order to make the output power of PV cells stable at the maximum power point, BESS requires frequent charging and discharging to the PV cell system according to the actual situation. Accurate monitoring of BESS's working state is significant, reducing the fault tolerance of the system and improving work efficiency. A critical factor reflecting the working condition of BESS is the battery output, which can be determined by the battery's internal parameters, so obtaining the battery's internal parameters is of decisive significance for detecting the working state of BESS. Extracting the battery parameters in BESS effectively and accurately is an urgent problem, which will help analyze the system and construct the open circuit voltage (*OCV*) state of charge (*SOC*) curve (Yang et al., 2020). The practical application is mainly divided into the deterministic and metaheuristic algorithms, which make the identified parameters different in accuracy, computational efficiency, and resource utilization (Alam et al., 2015). Looking for a more balanced method to solve parameter identification has attracted more and more research and attention in recent years and has become a hot topic.

Through the deterministic algorithm calculated by mathematical regulations, the choice of initial value has remarkable individual aspiration, directly affecting the efficiency of solving problems and the persuasion of results (Chen et al., 2018). Metaheuristic algorithms do not rely on specific mathematical rules but usually combine with defined rules and randomness (Ikeda and Ooka, 2019), which have been employed and utilized to tackle many complex issues as effective solvers due to their simplicity and easy implementation process (Heidari et al., 2019). In recent years, metaheuristic algorithms have been applied to solve parameter identification problems, promoting their development. Tan et al. (2021) pointed out that the real-time *SOC* of the battery is very important in practice as the indexes ensure the battery's stable operation. The *OCV*–*SOC* relationship was conducted on different time scales based on the second-order RC equivalent circuit model. A dynamic identification algorithm was studied, which got more accurate results and stronger robustness than other algorithms. Zhou et al. (2021b) pointed out that a realistic model and precise model parameters are significant for the safety of electric vehicles and the efficiency of batteries. The traditional solution method has a problem of the limited number of search solutions, leading to insufficient accuracy. Therefore, a coupled hybrid adaptive particle swarm optimization–hybrid simulated annealing (HA-PSO) algorithm was adopted, and further exploration was made. The results concluded that the method could be used as one of the tools to solve the parameter identification of the battery model. In El-Sehiemy et al. (2020), a new model identification method was presented: using the state–space equation of the battery in the equivalent circuit. Then, the parameter identification process was transformed into a nonlinear problem for optimization. A new enhanced sunflower optimization algorithm (ESFOA) has an outstanding performance. Choi et al. (2020) established a subset to store the model parameters to be identified, and a steady-state model was established. The genetic algorithm was applied to the parameter estimation of the model. The parameters determined by the information matrix-based method were compared with the experimental results under different conditions, and the average relative error between them is less than 1.9%.

In addition to identifying the battery parameters in BESS, many existing methods have been designed to extract PV cells' parameters and have achieved excellent results (Qais et al., 2019). In Zhang et al. (2020), the differential vectors are added to the intelligent algorithm to form a new backtracking search algorithm (BSARDV), which has attracted attention due to its low requirements for initial parameters. In Liang et al. (2020), a multi-structure optimization algorithm (SGEMTO) was proposed to improve the acquisition of useful information in the population. By enhancing the information exchange among the populations, the information transfer between different tasks was realized, so the quality and efficiency of the solution were improved. Lin and Wu (2020) explained that it has great practical significance to determine the parameters of the PV model, especially in predicting the output of the PV array and tracking the maximum power point. They also proposed a niche-based particle swarm optimization in parallel computing architecture (NPSOPC) algorithm to identify the parameters of the PV models.

Although the aforementioned methods have achieved good results, these methods often need to introduce more computing resources to maintain the accuracy of the products and are sensitive to the changes of the specific parameters, which lead to some methods not being popular. The common shortcoming of most metaheuristic algorithms is that if some parameters are set improperly, the desired results will not be obtained. Another disadvantage is that the metaheuristic algorithm may not always converge to the global optimum. If the parameters are adjusted improperly, it is easy for the particles to reach a local optimum, or the particle ability is insufficient. Hence, in order to solve these shortcomings of metaheuristic algorithms, in addition to the extensive research on the efficiency, performance, and results of the traditional algorithm, in recent years, new optimizers with excellent characteristics and new strategies with balanced performance in many aspects have emerged, providing many possibilities for resolving all kinds of complex problems in a variety of different fields.

In this study, a dynamic opposite learning (DOL) strategy is presented and applied to the differential evolution (DE) (Storn and Price, 1997) algorithm to improve DE's performance in terms of convergence speed and optimization to obtain faster convergence speed and avoid falling into local optimum (Jiang et al., 2013). An adaptive differential evolution algorithm with a dynamic opposite learning strategy (DOLADE) was proposed, which can be used as a potential optimizer for extracting the battery model parameters efficiently and accurately in BESS. In DOLADE, a dynamic opposite learning process

is carried out for the particles in the population to increase the population's diversity and overcome the trap of falling into local optimum and premature convergence. The new population comprises the successfully selected individuals and poorly performing individuals, which is produced by merging two of them into a new set (population). Endowing particles with opposite learning behavior aims to enhance their own exploration (diversification) ability to find those areas with optimal solutions in the early phase and reduce the resource consumption of the particles in the later exploitation phase. By dynamically changing the area to be explored by particles, the particle's exploitation (intensification) ability can be enhanced, and the exploitation can be carried out in the area of the early potential optimal solution to obtain the global optimum quickly. The DOL strategy balances the individuals' exploration capability and exploitation capability in the DE algorithm well, which makes the DE algorithm not only obtain the global optimum but also have a fast convergence speed, improving the disadvantages. The performance of DOLADE and the effect of the DOL strategy have been tested in the parameter identification of BESS, and excellent results have been obtained.

The main contributions of this article are as follows:

1) An adaptive differential evolution algorithm with a dynamic opposite learning strategy is proposed. To the best of our knowledge, the DOL strategy is the first attempt to identify the parameters of the battery model in BESS.
2) The particles' exploration and exploitation capabilities are significantly improved, leading to the speed of acquiring the global optimum considerably, preventing individuals from converging to the local optimal value.
3) According to the large amount of data obtained in the experiment, the parameter identification of the battery model is researched and demonstrated.
4) In order to highlight the effectiveness and competitiveness of DOLADE, the results are compared with several well-established algorithms.

The organization of the rest of this article is arranged as follows: In **Section 2**, the battery mathematical model in BESS is introduced to determine the objective function and extract the parameters. **Section 3** generally presents the basic concepts of an adaptive DE algorithm. In **Section 4**, the DOLADE algorithm is proposed in detail. The experimental results are analyzed and compared with some algorithms in **Section 5**. Finally, **Section 6** is a summary based on the whole article.

2 MODEL AND PROBLEM DESCRIPTION

A suitable model can accurately simulate the operational characteristics of the battery. The battery model is essential for the *SOC* estimation and the collaborative work with PV cells (Wang et al., 2017). The primary purpose of building the battery model is to obtain the mathematical relationship between the internal parameters of the battery and the external characteristics and then establish an effective equation to estimate the output voltage of BESS.

2.1 Battery Model in BESS

Accurate *SOC* estimation requires a precise battery model (Liu and Zhang, 2021). There are mainly three types of battery models: the electrochemical model, the mathematical model, and the equivalent circuit model. Among them, the electrochemical model has high accuracy, but the model is complex. The mathematical model is not affected by the external volt–ampere characteristics of the battery, and the model is uncertain due to significant errors. The equivalent circuit model can better simulate the external characteristics of the battery. The equivalent circuit model is mainly divided into the Thevenin model, the second-order RC equivalent circuit model (SECM), and the PNGV model (Bruch et al., 2021). They are shown in **Figure 1**. For PNGV: In September 1993, USCAR announced the Partnership for a New Generation of Vehicles (PNGVs). The PNGV standard equivalent circuit model was proposed in the PNGV Battery Test Manual published in 2001. A standard method for parameter identification of the model was proposed in the FreedomCAR Battery Test Manual published in 2003. In this research, SECM is employed, which could be used to simulate the external characteristics of the battery. Two RC networks are used to simulate the two dynamic processes of the battery's electrochemical polarization characteristics and concentration polarization characteristics. While considering the influence of internal ohmic resistance, the internal ohmic resistance is connected in series in the circuit. The electromotive force adopts a controlled voltage source, and the electromotive force and *SOC* maintain a relatively stable monotonic relationship under

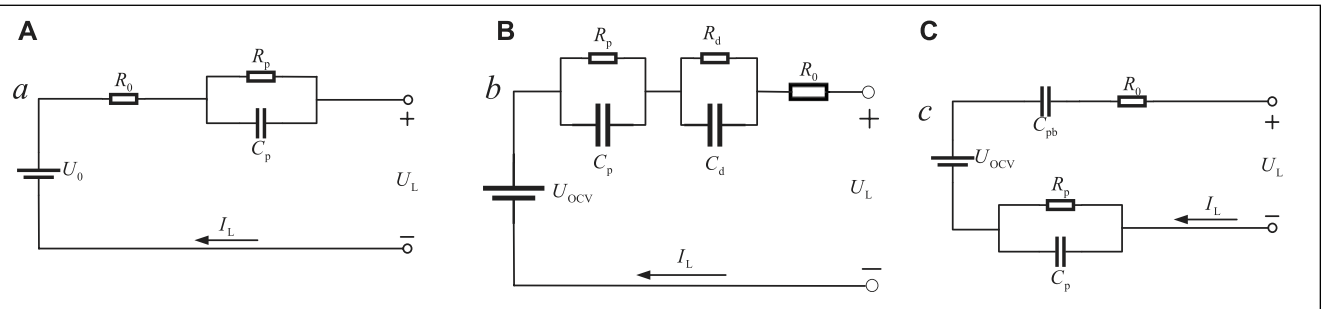

FIGURE 1 | Battery model. **(A)** Thevenin model. **(B)** SECM. **(C)** PNGV.

a specific condition; the electromotive force can be controlled by *SOC* (Wang et al., 2020).

For the SECM, according to Kirchhoff's law of voltage and current, the equation could be known as **Eq. 1** (Xiong et al., 2012; Bruen and Marco, 2016).

$$\begin{cases} U_L = U_{OCV} - U_0 - U_P - U_d \\ I_L = \dfrac{U_P}{R_P} + C_P * \dfrac{dU_P}{dt} \\ I_L = \dfrac{U_d}{R_d} + C_d * \dfrac{dU_d}{dt} \\ I_L = \dfrac{U_0}{R_0} \end{cases} \tag{1}$$

where U_L is the terminal voltage of the BESS output; U_{OCV} is the open circuit voltage; U_0 is R_0's voltage; U_p and U_d express the terminal voltages of the two RC networks, respectively; R_p represents the resistance of electrochemical polarization; C_p denotes the capacitance of electrochemical polarization; R_d and C_d are the resistance and the capacitance of concentration polarization, respectively; R_p, C_p, R_d, and C_d constitute two RC parallel loops used to simulate polarization effects; R_0 is the internal resistance; and I_L is the load current. There are five unknown parameters (R_p, C_p, R_d, C_d, and R_0) that could be calculated to observe the actual behavior of BESS.

2.2 Relation Curve Between *OCV* and *SOC*

The parameter identification of resistance and capacitance in the battery model requires the value of open circuit voltage (*OCV*), so *OCV* is measured first. The identification of *OCV* is actually carried out to calibrate the relationship curve between *OCV* and *SOC* (Wang et al., 2018). The experimental data can depict the *OCV*–*SOC* curve, and the functional relationship between them can be obtained (Kang et al., 2012). According to the experimental principle, the practical steps of the static experiment are as follows: first, use the standard charging method to charge the battery fully, let it stand for 1 h, and measure the *OCV* at *SOC* = 1. Then, discharge the battery with 6A current for 20 min each time. Finally, let the battery stand for 1 h, and approximate the sampled voltage obtained at this time as the *OCV* value. There will be a specific difference in the final curve after the still standing time is different. The discharge data of *SOC* and *OCV* measured by the experiment are shown in **Table 1**. Data fitting is performed on the aforementioned experimental data to obtain the functional relationship between *OCV* and *SOC*. Considering the two factors of accuracy and computational complexity, a fourth-order polynomial function is selected to fit the experimental data. The fitting result is shown in **Eq. 2**.

TABLE 1 | *OCV* – *SOC* experimental data.

SOC	*OCV* (V)	*SOC*	*OCV* (V)
1	3.375	0.4	3.293
0.9	3.334	0.3	3.283
0.8	3.332	0.2	3.257
0.7	3.315	0.1	3.212
0.6	3.303	0	3.116
0.5	3.296		

$$\begin{aligned} U_{OCV} = &-0.925263 * SOC^4 + 2.671602 * SOC^3 - \\ &2.614026 * SOC^2 + 1.118892 * SOC + 3.118363 \end{aligned} \tag{2}$$

2.3 Objective Function

After determining the structure of the battery model, it is necessary to extract the unknown parameters of the battery. The experimental data will be compared with the extracted data. Therefore, choosing an appropriate objective function could minimize the error between the *OCV* and *SOC*. A continuous discharging experiment is used for the battery, and the sampling time $T = 1$s. According to **Eq. 1**, the differential equation of the battery model can be calculated by **Eq. 3**. Since sampling of battery discharge data in the experiment is not continuous but at a fixed time interval, the differential equation of **Eq. 3** can be rewritten as a discrete differential equation, as shown in **Eq. 4**, where $U_p(k)$ and $U_d(k)$ represent the *kth* discrete voltage value of the two RC networks, respectively, $U_L(k)$ denotes the *kth* discrete output voltage value of the battery, and τ_1 and τ_2 are the time constants of the two RC networks, respectively.

In the calculation process of the metaheuristic algorithm, the particles are saved according to their own objective function value, so the choice of the objective function is crucial. For the battery model parameter identification in BESS, the current data are obtained through the identified parameters, that is, the error between the calculated data and the experimental data is as small as possible. **Eq. 5** is used to express the error function between the experimental and measured data for the SECM (Tong et al., 2018), where T is the sampling time; here, $T = 1$.

$$\begin{cases} U_{OCV} = f(SOC) \\ \dot{U}_P = \dfrac{dU_P}{dt} \\ \quad = -\dfrac{1}{R_P * C_P} * U_P + \dfrac{1}{C_P} * I_L \\ \dot{U}_d = \dfrac{dU_d}{dt} \\ \quad = -\dfrac{1}{R_d * C_d} * U_d + \dfrac{1}{C_d} * I_L \\ U_L = U_{OCV} - U_P - U_d - R_0 * I_L \end{cases} \tag{3}$$

$$\begin{cases} U_P(k) = \exp\left(-\dfrac{T}{\tau_1}\right) * U_P(k-1) - \left[1 - \exp\left(-\dfrac{T}{\tau_1}\right)\right] \\ \qquad * R_P * I_L(k-1) \\ U_d(k) = \exp\left(-\dfrac{T}{\tau_2}\right) * U_d(k-1) - \left[1 - \exp\left(-\dfrac{T}{\tau_2}\right)\right] \\ \qquad * R_d * I_L(k-1) \\ U_L(k) = U_{OCV}(SOC(k)) - U_P(k) - U_d(k) \\ \qquad - R_0 * I_L(k-1) \\ \qquad \tau_1 = R_P * C_P \\ \qquad \tau_2 = R_d * C_d \end{cases} \tag{4}$$

$$\begin{cases} f_k(U_L, I_L, SOC, X) = U_{OCV}(SOC(k)) - \\ \exp\left(-\frac{T}{R_p * C_p}\right) * U_p(k-1) - \\ \left[1 - \exp\left(-\frac{T}{R_p * C_p}\right)\right] * R_p * I_L(k-1) - \\ \exp\left(-\frac{T}{R_d * C_d}\right) * U_d(k-1) - \\ \left[1 - \exp\left(-\frac{T}{R_d * C_d}\right)\right] * R_d * I_L(k-1) - \\ R_0 * I_L(k-1) - U_L \\ X = \{R_p, C_p, R_d, C_d, R_0\} \end{cases} \tag{5}$$

In this research, the root mean square error (RMSE) represents the overall difference between the experimental and estimated data. RMSE is defined by **Eq. 6**.

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

where X represents the solution vector consisting of unknown parameters, N denotes the number of experimental data, and k = 1, …., N.

3 JADE ALGORITHM

Differential evolution (DE) algorithm is an efficient intelligent optimization algorithm. It is a metaheuristic search algorithm based on a group, and each individual in the group corresponds to a solution vector (Brest et al., 2020). The evolution process of DE is very similar to the genetic algorithm, but the specific definition of these operations is different from the genetic algorithm (Gao et al., 2021). The three stages of mutation, crossover, and selection are the basic phases of DE. The parameters about these stages need to be set before the start of each phase, and they have the same fixed value during the whole evolutionary process. Adaptive differential evolution with an optional external archive algorithm (JADE) is based on the DE algorithm (Zhang and Sanderson, 2009). JADE adopts a new mutation strategy: "DE/current-to-*p*best/1" (with archive A), as shown in **Eq. 7** (Peng et al., 2009). The greedy strategy ensures better performance of the particles.

$$v_{i,g} = x_{i,g} + F_i * \left(x^p_{best,g} - x_{i,g}\right) + F_i * \left(x_{r1,g} - \tilde{x}_{r2,g}\right) \tag{7}$$

$$F_i = randc_i(\mu F, 0.1) \tag{8}$$

where $x_{i,g}$ and $x_{r1,g}$ are acquired from Np, $x^p_{best,g}$ randomly chosen from the top $100p$ % ($p \in 0,1$). The mutation factor F_i is determined by x_i, and F_i is updated in the process by introducing **Eq. 8**. $\tilde{x}_{r2,g}$ is the individual randomly chosen from $P \cup A$. Here, P represents the current population which saves individuals close to the optimal value, and the external archive A converses poor performance individuals.

The parameter adaption strategy is used for the mutation probability F_i. F_i is independently generated according to the Cauchy distribution with a location parameter μF = 0.5, and the scale parameter is 0.1. If $F_i \geq 1$, it truncates to be 1; if $F_i \leq 0$, it regenerated 1. S_F is the set that saves every successful mutation probability in each iteration. μF is initialized to be 0.5 and then updated in **Eq. 9**.

$$\mu F = \mu F * (1-c) + mean(S_F) * c \tag{9}$$

where **Eq. 10** can calculate mean (S_F).

$$mean(S_F) = \frac{\sum_{F \in S_F} F^2}{\sum_{F \in S_F} F} \tag{10}$$

The parameter adaption strategy is not only used for the mutation probability F_i but also for the crossover factor CR_i. Each individual x_i's crossover factor CR_i is independently generated according to the normal distribution. μCR is the mean value, and the standard deviation is 0.1, as shown in **Eq. 11**.

$$CR_i = randn_i(\mu CR, 0.1) \tag{11}$$

The range of CR_i is limited to [0, 1]. The S_{CR} is a set of CR_i with a good performance. The initial value of μCR is 0.5, which is then regenerated using **Eq. 12** (Peng et al., 2009).

$$\mu CR = \mu CR * (1-c) + mean(S_{CR}) * c \tag{12}$$

where c is a constant in the range of [0,1], and $mean(S_{CR})$ is the average value.

4 A NOVEL DE ALGORITHM—DOLADE

4.1 Dynamic Opposite Learning

There are many successful learning strategies in the algorithm field, and these strategies have provided a significant contribution to the progress of algorithms. The opposition-based learning strategy has a considerable influence on enhancing the searchability of the algorithm itself (Tizhoosh, 2005), but these strategies also have some shortcomings. For example, the QOBL (Rahnamayan et al., 2007) and QRBL (Laquai et al., 2011) approaches are based on deterministic strategies for potential relative points, making the exploitation ability of particles lacking. In the DOL strategy, the current particle works in the new search area. The area is dynamically extended from the current to the opposite, which is explained in **Figure 2**. P is the current point, P^0 denotes the opposite point, P^s represents the existing point in the new area, and a and b are the boundaries of the area. The individuals can obtain a larger area for movement to increase the probability of being close to the optimal value and enhance their exploration capability. However, the search area of individuals inevitably converges to the local optimum so that the particle's exploitation capability is insufficient.

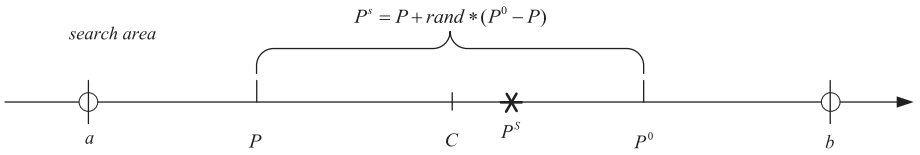

FIGURE 2 | Symmetric area between the current point and the opposite point.

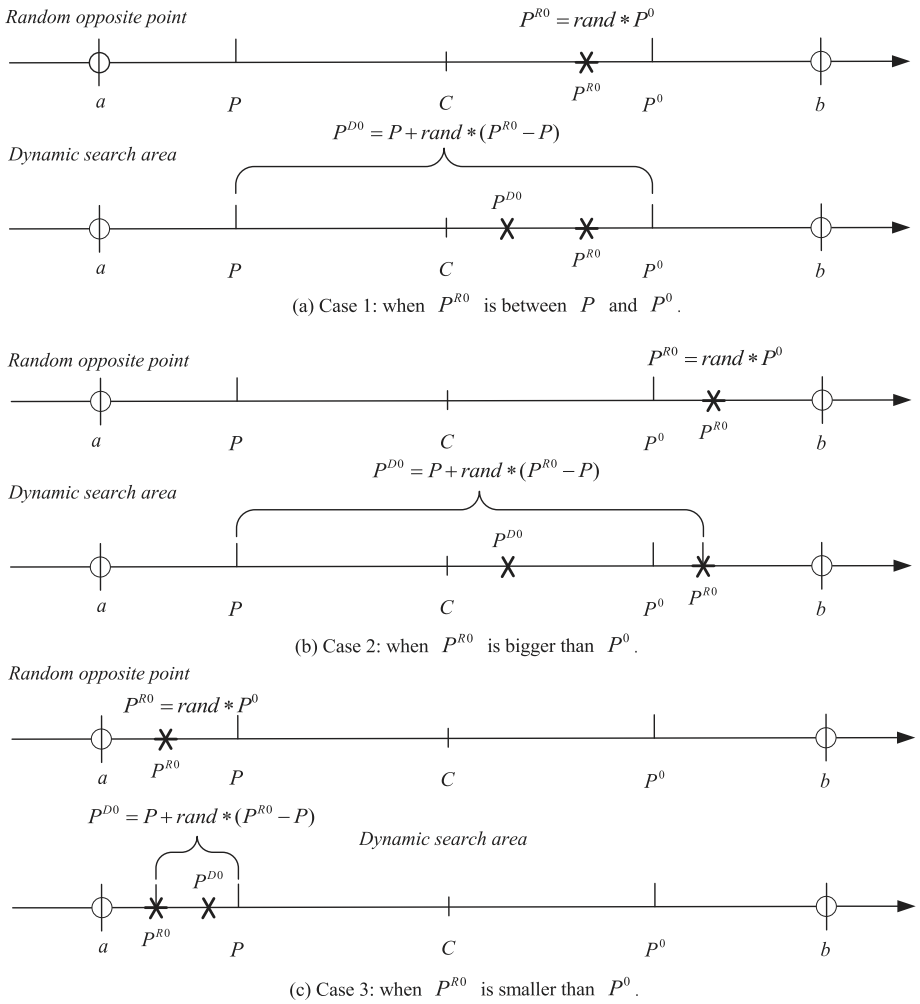

FIGURE 3 | DOL asymmetric dynamic search area when $P^{R0} \in (a,b)$.

A random opposite point (P^{R0}) is introduced to avoid the solution falling into the local optimum, improving the particle's capability; $P^{R0} = rand * P^0$, $rand \in [0,1]$. When P^{R0} replaces P^0, the symmetric search area in **Figure 2** can be transformed into asymmetric search areas, which have a characteristic of dynamic adjusting along with P^{R0}. In **Figures 3**, **4**, the novel search areas are demonstrated. According to the relative position of P^{R0} to P and P^0, three categories of the asymmetric search areas are formed in **Figure 3** when $P^{R0} \in [a,b]$. Case 1 is the scenario of the search area when P^{R0} is between P and P^0, case 2 illustrates the project of the search area when P^0 is smaller than P^{R0}, and when P is larger than P^{R0}, the search area is represented in case 3. The strategy can be formulated by randomly generating a DOL point P^{D0} from P to P^{R0} and $P^{D0} = P + rand(P^{R0} - P)$. In addition, the condition that P^{R0} is out of the area boundaries [a,b] cannot be ignored, such as case 4 and case 5, which is depicted in **Figure 4**. The boundaries need to be limited: if $P^{D0} = P + rand(P^{R0} - P) \in [a,b]$, and P^{D0} is equal to $P^{D0} = P + rand(P^{R0} - P)$. Otherwise,

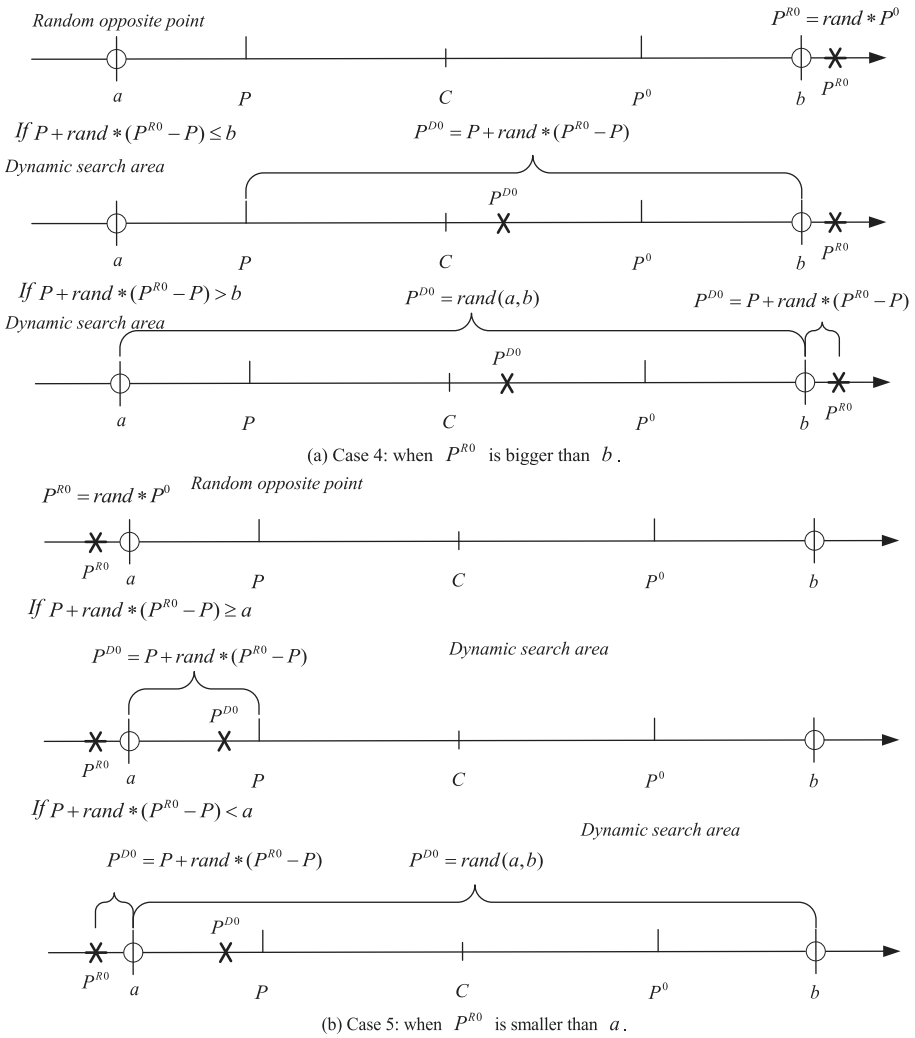

FIGURE 4 | DOL asymmetric dynamic search area when $P^{R0} \notin (a,b)$.

P^{D0} should be regenerated as a random number between a and b.

Although P^{R0} can diversify the search area, the search area may shrink as the iteration progresses so that the particles' exploration and exploitation capabilities cannot be fully utilized to the maximum. In order to prevent particles from being too large or too weak to be unbalanced in their exploration and exploitation capabilities, a weighting factor (ω) is used to improve this problem, which could help the particle arrive at the best region. Therefore, the DOL strategy is redefined as $P^{DOL} = P + \omega * (P^{R0} - P)$, where ω is a constant and $\omega > 0$. Finally, DOL can be calculated by the formula with these foundations. The search area is expanded to a symmetric region so that the probability that is closer to the best solution is enhanced. The dynamic characteristics of the search area make the population become diverse and enhance the particle's exploration and exploitation capabilities.

4.2 Mathematical Expression of DOL

In this subsection, the DOL strategy will be described by mathematical formulas.

- **Dynamic opposite point.** P ($P \in [a,b]$) is defined as a point. The dynamic opposite point P^{D0} is defined in **Eq. 13**, where a is the lower boundary, b is the upper boundary, and $rand \in (0,1)$ is a random number. Moreover, **Eq. 13** applied the weighting factor ω ($\omega > 0$) into DOL. P^0 is the opposite number, as shown by **Eq. 14**.
- **DOL-based optimization.** Assume $P = (P_1, P_2, \ldots, P_D)$ in the $D-$ dimensional area. $P_j, \ldots, P_D \in [a_j, b_j]$, where a_j and b_j are boundaries of the current searching area. P_j^0 is the opposite point defined in **Eq. 15**. $P_j^{D0} = (P_1^{D0}, P_2^{D0}, \ldots, P_D^{D0})$ is the dynamic opposite point, updated by P_j according to **Eq. 16**. P_j^{D0} will replace P_j when the fitness value of P_j^{D0} is better than P_j; otherwise, P_j^{D0} will not be

saved. Note that $P_j^{D0},\ldots,P_D^{D0}$ is limited within $[a_j,b_j]$. If $P_j^{D0}\notin[a_j,b_j]$, P_j^{D0} should be redefined as a random in $[a_j,b_j]$.

$$P^{D0}=P+\omega*rand*\left(rand*P^{0}-P\right) \tag{13}$$

$$P^{0}=a+b-P \tag{14}$$

$$P_j^{0}=a_j+b_j-P_j,\ j=1:D \tag{15}$$

$$P_j^{D0}=P_j+\omega*rand*\left(rand*P_j^{0}-P_j\right),\ j=1:D \tag{16}$$

4.3 DOLADE Algorithm Steps

4.3.1 Population Initialization

oP^{D0} is randomly generated by the initial DOL strategy, which may obtain better knowledge than the original individual to a certain extent. At the beginning of generation, the population size is set as Np (for any individual i, $i=1,2,\ldots,Np$), and D represents dimension of problems (for any problem j, $j=1,2,\ldots,D$). **Eq. 17** can calculate the initialization process.

$$oP_{i,j}^{D0}=oP_{i,j}+r1_i*\left(r2_i*\left(a_j+b_j-oP_{i,j}\right)-oP_{i,j}\right) \tag{17}$$

Boundaries should be checked according to **Eq. 18**, which could improve the DOL effectiveness.

$$oP_{i,j}^{D0}=rand\left(a_j,b_j\right),if:oP_{i,j}^{D0}<a_j\parallel oP_{i,j}^{D0}>b_j \tag{18}$$

where $[a_j,b_j]$ is the range of the search area, $r1_i$ and $r2_i$ are random values among (0,1), and the weight factor ω is set as constant: 10. Then, fittest individuals Np are selected from $\{oP\cup oP^{D0}\}$.

4.3.2 DOL Generation Jumping

The jumping rate Jr is introduced to control DOL to update the population. If $rand<Jr$, the DOL jumping process would be performed by **Eq. 19**.

$$P_{i,j}^{D0}=P_{i,j}+\omega*r3_i*\left(r4_i*\left(a_j+b_j-P_{i,j}\right)-P_{i,j}\right) \tag{19}$$

Meanwhile, the boundaries should be checked by **Eq. 18**. Remarkably, the new candidates are locked within a shrunken search area by dynamically updating the interval boundaries $[a_j,b_j]$ as follows (Wang et al., 2011):

$$\begin{aligned}a_j&=\min\left(P_{i,j}\right)\\ b_j&=\max\left(P_{i,j}\right)\end{aligned} \tag{20}$$

When the DOL jumping step is finished, the best individual Np is selected from $\{oP\cup oP^{D0}\}$.

4.3.3 DOLADE Algorithm Steps

By employing the DOL strategy to JADE, an adaptive differential evolution algorithm with a dynamic opposite learning strategy is generated, namely, DOLADE. In order to make DOLADE convenient and simple to deal with problems, the pseudo-code is described in Algorithm 1, and the flowchart of DOLADE is exhibited in **Figure 5**.

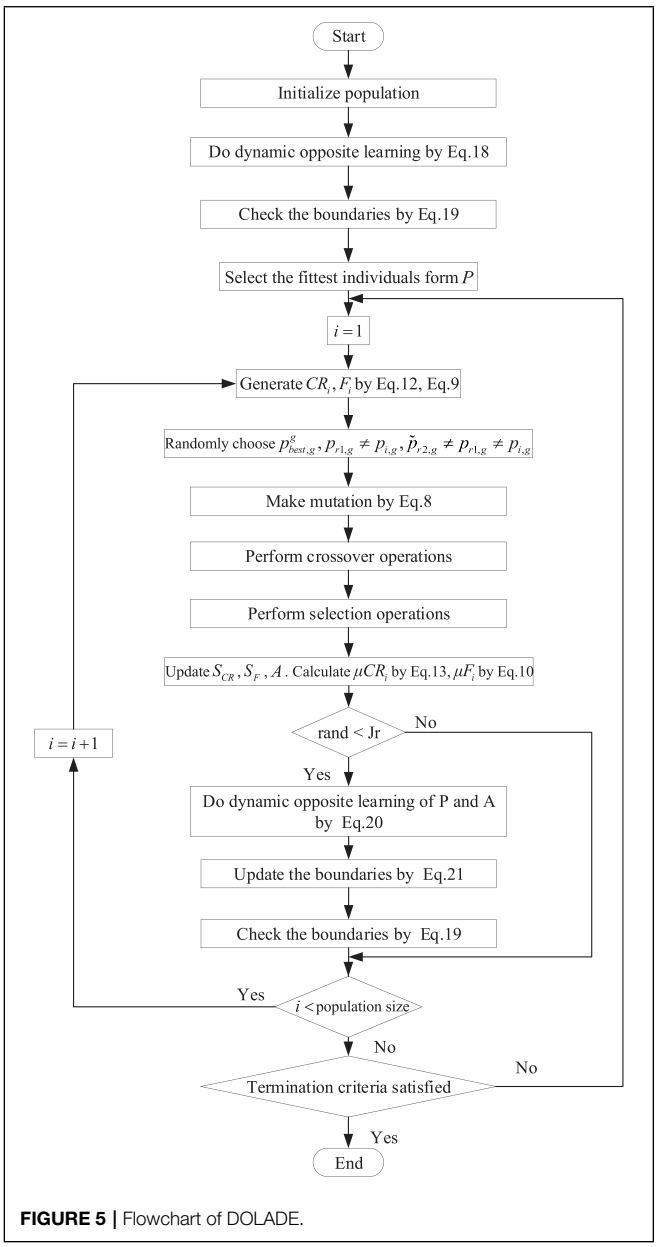

FIGURE 5 | Flowchart of DOLADE.

Algorithm 1: Pseudo-code of DOLADE.

Randomly generate an initial population oP;
for $i = 1$ to Np **do**
  $r1_i = rand; r2_i = rand;$
  for $j = 1$ to D **do**
    $oP_{i,j}^{D0} = oP_{i,j} + r1_i \cdot (r2_i \cdot (a_j + b_j - oP_{i,j}) - oP_{i,j})$
    Check the boundaries;
  end for
end for
Select P number of the fittest individuals from $\{oP \cup oP^{D0}\}$;
Set $G_f = 0; \mu_{CR} = 0.5; \mu_F = 0.5; A = \emptyset;$
for $g = 1$ to G **do**
  $S_F = \emptyset; S_{CR} = \emptyset;$
  for $i = 1$ to P **do**
    Generate CR_i, F_i
    Randomly choose $p_{best,g}^{g}$, $p_{r1,g}$, $\tilde{p}_{r2,g}$
    Mutation, Crossover, Selection;
    Update S_F, S_{CR}, A, μCR, μF;
  end for
  if rand $\leq$ Jr **then**
    for $i = 1$ to P **do**
      $r3_i = rand; r4_i = rand;$
      for $j = 1$ to D **do**
        $a_j = min(P_{i,j}), b_j = max(P_{i,j});$
        $P_{i,j}^{D0} = P_{i,j} + \omega * r3_i * (r4_i * (a_j + b_j - P_{i,j}) - P_{i,j})$
        Check the boundaries;
      end for
    end for
    Select P number of the fittest individuals from $\{oP \cup oP^{D0}\}$;
  end if
  The operation of A is equal P;
  $G_f + +;$
end for

5 EXPERIMENTAL RESULTS AND ANALYSIS

In this section, through the various data of the battery measured experimentally, the DOLADE algorithm is used to carry out the parameter identification experiment of the battery model. The efficiency and performance of the DOLADE algorithm are evaluated in solving such problems through various performance indicators. Continuous current discharging experiments under different current are implemented, the battery's internal parameters are identified and verified using measured experimental data, and the parameter values identified under different conditions are displayed in table. Therefore, all these data are obtained under the same experimental conditions. Among them, the unknown model parameters R_p, C_p, R_d, C_d, and R_0 are shown in **Table 2**, which lists the lower and upper values of these parameters.

TABLE 2 | Parameter range of the SECM.

Parameter	Lower bound	Upper bound
$R_p(\Omega)$	0.00213	0.28017
C_p(F)	6.32838	70.87500
$R_d(\Omega)$	0.00541	0.00865
C_d(F)	27.58706	225.25246
$R_0(\Omega)$	0.01268	0.01414

TABLE 3 | Parameters of the involved algorithms.

Algorithm	Parameters
DOLADE	NP = 30, ω = 10, p = 0.1
JAYA	NP = 30
GWO	NP = 30, a = 2 ~0
MFO	NP = 30
ETLBO	NP = 30
cfPSO	NP = 30

Some well-established algorithms, including JAYA (Venkata Rao, 2016), GWO (Mirjalili et al., 2014), MFO (Mirjalili, 2015), ETLBO (Rao and Patel, 2012), and cfPSO (Pathak and Singh, 2017) are compared to prove the effectiveness of the DOLADE algorithm. All of the data are obtained and compared under the same experimental conditions: the maximum number of function evaluations (MaxFEs) is 50 in each operation under different currents, and each algorithm runs 30 times independently to minimize statistical errors. The optimal values are taken out and listed in table, and the *OCV* and *SOC* are calculated according to the optimal value. The comparison result of each algorithm is to take the optimal value under the aforementioned operating conditions. **Table 3** lists some specific parameter configurations that need to be prepared in advance for each algorithm adopted.

5.1 SECM Model Identification Results

Table 4 lists the values of R_p, C_p, R_d, C_d, and R_0 extracted by the six algorithms under various currents. The RMSE value is also listed in the table, and the best value is highlighted in bold. RMSE represents the difference between the parameters extracted by the algorithm and the experimental data and represents the accuracy of the result. The smaller the value, the more accurate the result. It can be concluded that when the DOLADE algorithm is used, the result is the best, and the RMSE value is the smallest. The parameter values of the battery model identified by the DOLADE, JAYA, GWO, MFO, ETLBO, and cfPSO algorithms are different, and even the parameter values extracted by these algorithms under the same current are also different. This indicates that the internal parameter values of the battery are not constant in actual operation, but changes as a dynamic process in changing, and there must be an optimal value among these values. The evaluation standard of this optimal

TABLE 4 | Comparison among six algorithms on the SECM under different currents.

Current (A)	Algorithm	R_p(Ω)	C_p(F)	R_d(Ω)	C_d(F)	R_0(Ω)	RMSE
4	DOLADE	0.00213017679	69.90178729	0.00768949442	224.2956046	0.01270131803	**9.00277960447401E-03**
	JAYA	0.00213000000	40.70816164	0.00710227925	195.6715474	0.01320692054	9.38146471788928E-03
	GWO	0.00316646278	53.28854845	0.00618805504	48.77001447	0.01292532647	9.40495879771805E-03
	MFO	0.00379783345	39.63209269	0.00555106688	148.4883831	0.01292709912	9.69624084066968E-03
	ETLBO	0.00213093435	25.50123547	0.00602343721	67.76233266	0.01412627898	9.36080986852247E-03
	cfPSO	0.00240572691	29.01658835	0.00739552334	115.5035527	0.01393209178	9.36919323645419E-03
8	DOLADE	0.00217215521	29.00671053	0.00759045499	225.2524600	0.01273007838	**1.00127568954371E-02**
	JAYA	0.00286251443	37.24044335	0.00620165334	100.7567042	0.01314485295	1.08926527570402E-02
	GWO	0.00260864414	48.43932668	0.00608994639	196.6067671	0.01323861824	1.04955842163707E-02
	MFO	0.00314184819	20.22789870	0.00574272277	219.6596206	0.01305792888	1.09525251141804E-02
	ETLBO	0.00385667147	10.38598492	0.00541129247	207.1599653	0.01268810344	1.03028634384998E-02
	cfPSO	0.00234713324	58.83236247	0.00809682112	175.3286103	0.01282388588	1.04204138120596E-02
12	DOLADE	0.00214077987	48.22895464	0.00754835546	225.2495094	0.01277344320	**1.23099262791689E-02**
	JAYA	0.00213000000	51.68677991	0.00753646182	48.31473917	0.01288689266	1.48888363128724E-02
	GWO	0.00233032446	12.51821569	0.00730037899	135.6281089	0.01283946011	1.27334738838063E-02
	MFO	0.00257471495	54.66464669	0.00675589635	151.7628191	0.01313488003	1.30504524516649E-02
	ETLBO	0.00377397122	70.87498203	0.00541000084	106.2625695	0.01327375790	1.33534583734524E-02
	cfPSO	0.00268628932	40.46273046	0.00739490953	107.2711433	0.01333371679	1.53456916921213E-02
16	DOLADE	0.00214888678	68.53428364	0.00771400503	225.2524600	0.01268277122	**1.27107978725237E-02**
	JAYA	0.00213000000	53.46779207	0.00718673028	175.8853941	0.01340384063	1.46004575003135E-02
	GWO	0.00219926472	21.91028041	0.00729014442	108.4375470	0.01303618217	1.35813315682300E-02
	MFO	0.00228451738	63.02478699	0.00700702596	153.0145416	0.01323787499	1.32171333107309E-02
	ETLBO	0.00391104851	70.56217444	0.00541029493	140.1968253	0.01321387794	1.42900697621894E-02
	cfPSO	0.00359569973	53.05733164	0.00662651301	137.3150545	0.01329562068	1.36495909452741E-02

value is determined by the value of RMSE. The algorithm that affects the best in terms of extracted parameters is DOLADE. The parameter values extracted by DOLADE are substituted into the equation to calculate the *OCV* value. The experimentally measured voltage value and the calculated voltage value curve are shown in **Figure 6**. Here, A_E represents the experiment current, and A_S represents the calculated current. The estimated *OCV* curves coincide well with the experimental *OCV* curves. Whether it is a high-current discharging experiment or a low-current discharging experiment, the relationship between the two is in line with expectations, and the error of *OCV* in the figure is within a tiny field, which could satisfy the practical needs.

Furthermore, the relationship of *OCV–SOC* is depicted in **Figure 7**. The figure shows that the estimated *OCV* can replace the experimentally measured *OCV* to calculate *SOC*.

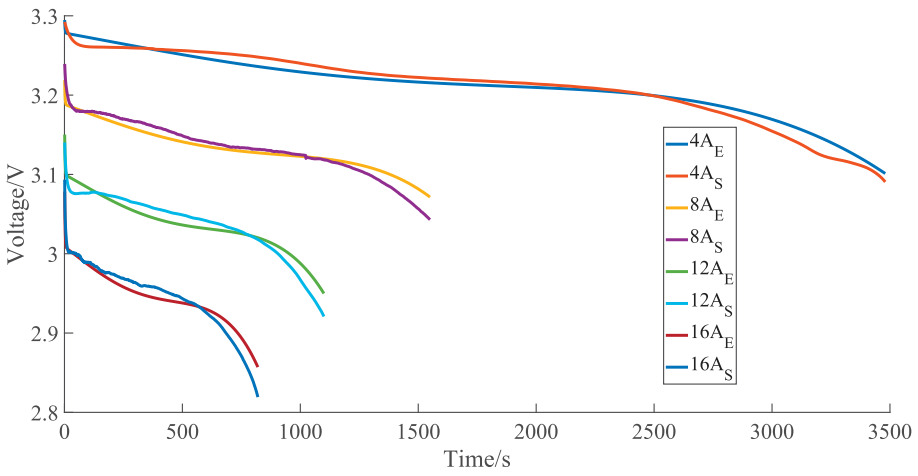

FIGURE 6 | Experimental and estimated *OCV* data comparison under different currents.

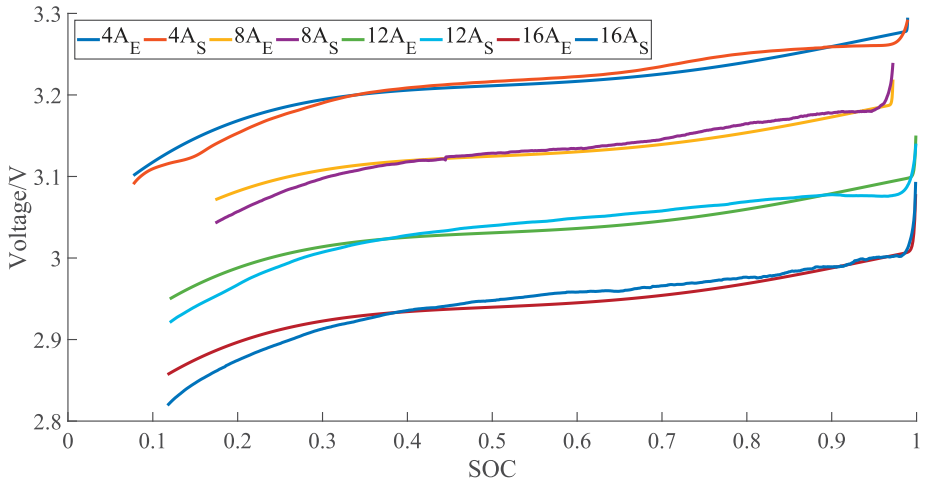

FIGURE 7 | *OCV–SOC* curve diagram.

In the range of *SOC* from 0 to 1, the computed value always maintains a reasonable range with the actual value and the changes of the two are consistent. The output voltage calculated using the identified parameters is the same as the actual voltage, which can correspond to the battery's *SOC*. In other words, the voltage value obtained from the parameters identified by DOLADE could be used to calculate the value of *SOC*.

5.2 Statistical and Convergence Results

In the **section 5.1**, the parameters were extracted and RMSE values by the DOLADE algorithm are demonstrated, and these results are compared with the results of other algorithms. Then, the estimated *OCV* and the *OCV–SOC* relationship curves are described. This subsection clearly compares the statistical values of the DOLADE algorithm operation results with the other algorithms. Comparisons in many aspects verify the effectiveness

TABLE 5 | Statistical results of RMSE for six algorithms under different currents.

Current (A)	Algorithm	RMSE			
		Min	Max	Mean	SD
4	DOLADE	**9.002779604474E-03**	**9.003036924617E-03**	**9.002900438728E-03**	**4.0085440807E-07**
	JAYA	9.381464717889E-03	1.248375845678E-02	1.0061516166523E-02	2.7360440588E-03
	GWO	9.404958797718E-03	1.272236168406E-02	1.0496140634186E-02	3.7135546654E-03
	MFO	9.696240840670E-03	1.239799339603E-02	1.0907584146048E-02	2.7914947868E-03
	ETLBO	9.360809868522E-03	1.253619568585E-02	1.0716641473875E-02	3.3960578471E-03
	cfPSO	9.369193236454E-03	1.293643767615E-02	1.0340315076050E-02	3.3818424672E-03
8	DOLADE	**1.001275689544E-02**	**1.001472477164E-02**	**1.001310251552E-02**	**1.9240576126E-06**
	JAYA	1.089265275704E-02	1.398963354066E-02	1.217063058029E-02	3.3163154120E-03
	GWO	1.049558421637E-02	1.416023243213E-02	1.230147096075E-02	4.0859148179E-03
	MFO	1.095252511418E-02	1.39398712214E-02	1.248013430019E-02	3.0134578055E-03
	ETLBO	1.030286343850E-02	1.298322545988E-02	1.140655701609E-02	2.8231964951E-03
	cfPSO	1.042041381206E-02	1.348205049525E-02	1.171551763142E-02	3.1958857126E-03
12	DOLADE	**1.230992627917E-02**	**1.230992670458E-02**	**1.230992940061E-02**	**1.2057888928E-08**
	JAYA	1.488883631287E-02	2.047012014682E-02	1.674145194711E-02	5.3988957502E-03
	GWO	1.273347388381E-02	2.003350064990E-02	1.446686085325E-02	7.5483356836E-03
	MFO	1.305045245166E-02	2.255298141265E-02	1.595985213139E-02	1.0870695028E-02
	ETLBO	1.335345837345E-02	2.260783718486E-02	1.647466474546E-02	8.0596580374E-03
	cfPSO	1.534569169212E-02	1.941982082972E-02	1.715229998765E-02	4.7247962405E-03
16	DOLADE	**1.271079787252E-02**	**1.271300579296E-02**	**1.271147766604E-02**	**2.1680060963E-06**
	JAYA	1.460045750031E-02	1.940613349739E-02	1.695675224236E-02	5.2194996489E-03
	GWO	1.358133156823E-02	1.918546631544E-02	1.583551650674E-02	6.2422604240E-03
	MFO	1.321713331073E-02	1.838186001229E-02	1.429187641740E-02	4.6580291525E-03
	ETLBO	1.429006976219E-02	2.044814378224E-02	1.643003079588E-02	6.3255093371E-03
	cfPSO	1.364959094527E-02	1.952808863702E-02	1.531984831305E-02	5.3307481357E-03

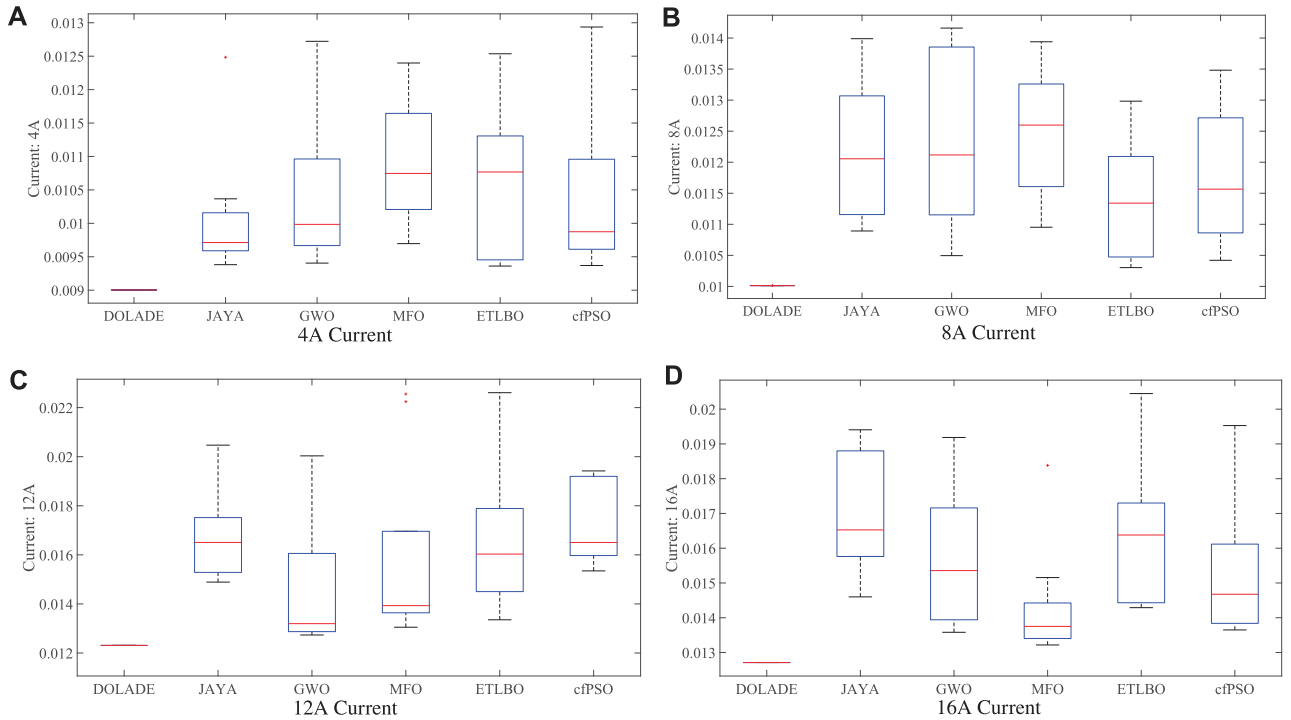

FIGURE 8 | Best RMSE boxplot in 30 runs of six algorithms under different currents.

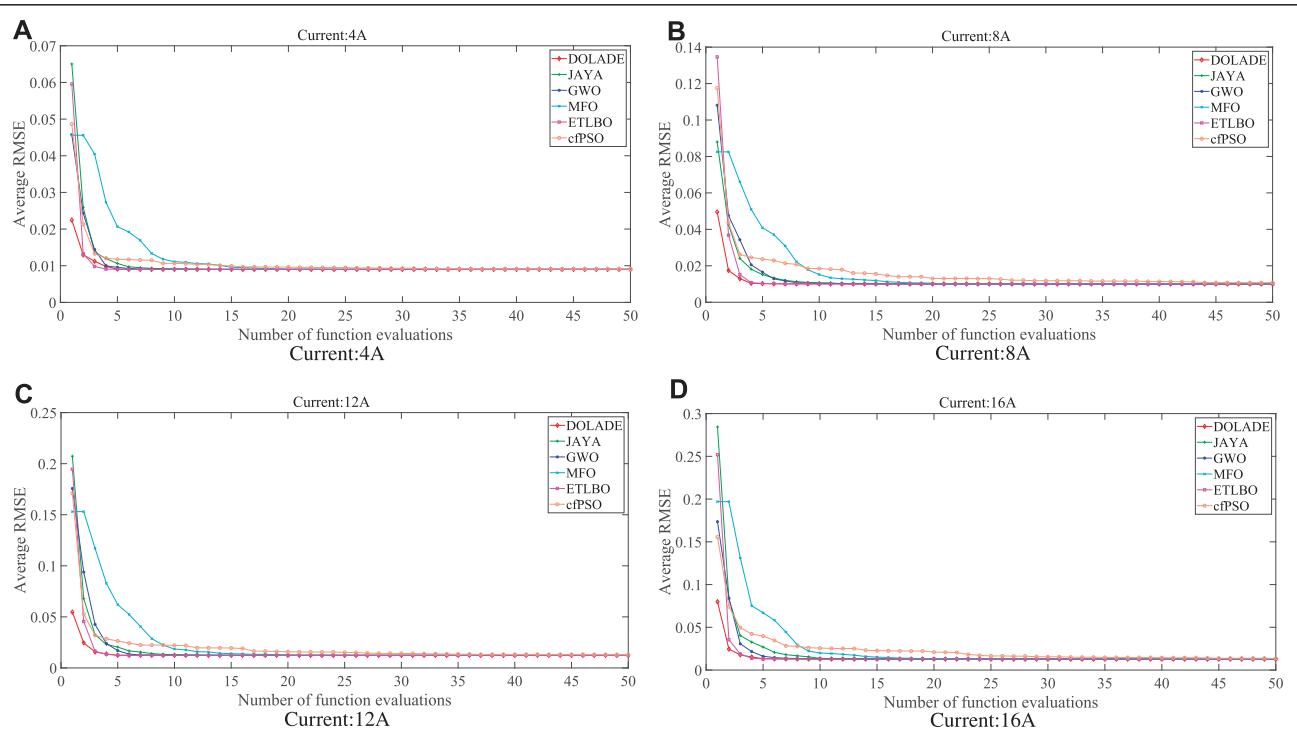

FIGURE 9 | Convergence curves of six algorithms under different currents.

TABLE 6 | Statistical results of RMSE for different sets under different currents.

Current (A)	Algorithm	RMSE			
		Min	Max	Mean	SD
4	DOLADE	**9.002779604474E-03**	**9.003036924617E-03**	**9.002900438728E-03**	**4.0085440807E-07**
	DOLADE-P	9.006873642750E-03	9.208613137926E-03	9.059873373755E-03	2.0301615695E-04
	DOLADE-A	9.003909498853E-03	9.340338840274E-03	9.080590003377E-03	3.6390649543E-04
8	DOLADE	**1.001275689544E-02**	**1.001472477164E-02**	**1.001310251552E-02**	**1.9240576126E-06**
	DOLADE-P	1.004285214575E-02	1.079385060983E-02	1.019450328621E-02	6.4687995847E-04
	DOLADE-A	1.005437508360E-02	1.229197189748E-02	1.030825889359E-02	2.0923814950E-03
12	DOLADE	**1.230992627917E-02**	**1.230992670458E-02**	**1.230992940061E-02**	**1.2057888928E-08**
	DOLADE-P	1.233039003441E-02	1.292689237338E-02	1.247640096522E-02	5.4748932388E-04
	DOLADE-A	1.232212259000E-02	1.241616262287E-02	1.235961177111E-02	9.6878474133E-05
16	DOLADE	**1.27(4)1079787252E-02**	**1.271300579296E-02**	**1.271147766604E-02**	**2.1680060963E-06**
	DOLADE-P	1.272895421090E-02	1.315425056924E-02	1.292100419026E-02	3.8106791039E-04
	DOLADE-A	1.271100984529E-02	1.294206960349E-02	1.275907011799E-02	2.1255567559E-04

of the DOL strategy, and it reflects the convergence speed and stability of the DOLADE algorithm. The comprehensive results of all algorithms after 30 independent runs are listed in **Table 5**. In addition to the minimum error, the table also lists the maximum error, the average value of the error, and the standard deviation reflecting the error distribution. For each current, the overall best results among the six algorithms are highlighted in bold.

Among the statistical results listed in **Table 5**, the six algorithms under different conditions and the performance indicators of the DOLADE algorithm all show strong competitiveness. In all the statistical results, the difference between the maximum and minimum values of the DOLADE algorithm is also the smallest, which indicates that the DOLADE algorithm perfectly performs the particles' exploration and exploration capabilities through the DOL strategy when solving problems. On the average of all results, the average value of the DOLADE algorithm is close to the global optimal value, which shows the reliability and stability of the DOLADE algorithm in solving problems without excessive waste of resources. Finally, the standard deviation further verifies the aforementioned findings from this side, providing strong evidence for the DOLADE algorithm to be more convincing.

In **Figure 8**, the boxplot shows the distribution of the results of the aforementioned six different algorithms under 30 independent executions, demonstrating the dispersion of the statistical results of all algorithms under different currents. As can be seen from the boxplot, the DOLADE algorithm results are stable and have a superior performance by comparing with the other five algorithms, and the advantages are very obvious. It is verified once again that the DOLADE algorithm is superior to other algorithms in terms of accuracy and reliability, has an outstanding performance in enhancing particle capabilities, and can be trusted as an optimizer.

Figure 9 demonstrates the convergence curves of the six algorithms under different discharging currents in the SECM. The figures depict the convergence graph when different algorithms run independently 30 times, where the number of runs of each algorithm is 50. It is not difficult to obtain from the figure that the speed of converging to the optimal value using the DOLADE algorithm is very considerable. Compared with the other five algorithms, DOLADE has a wonderful optimization process, fewer times of iteration when the algorithm converges to the optimal value, and the initial value of the optimization is closest to the global optimum, which is highly competitive. Therefore, it could be concluded that DOLADE has an excellent performance in solving the parameter identification of the battery model in BESS, which provides a new choice for solving this kind of problem.

5.3 Effectiveness Evaluation in Different Sets

The DOL strategy is applied to the population P (elite individuals) that the individuals' performance well and the external archive A (general individuals) that conserves the eliminated individuals, respectively. The excellent performance of DOL strategy in DE algorithm is verified by experimental results. The elite dynamic opposite learning for the elite particles in the population and the general dynamic opposite learning for the ordinary particles are used, and then the results are compared with the existing results of the combination of the two, that is, elite dynamic opposite learning and general dynamic opposite learning. The comparison results are shown in **Table 6**, and the best results are highlighted in bold. As can be seen from the data in the table, when the elite or general individuals in the population perform the dynamic opposite learning, the effect is not as good as when both the elite and general individuals are allowed to conduct dynamic opposite learning at the same time. These can be reflected from the data in the table, including minimum, maximum, mean, and standard deviation. Therefore, removing the DOL strategy from any set will not yield the best results. Only when the strategy is applied to both elite and general individuals, the following optimization

process can satisfy the need of generating global optimal solutions.

6 CONCLUSION

In this study, a new method to solve the problem of identifying battery model parameters in BESS is proposed. This method can accurately obtain the internal parameters of the battery model, which is of great significance for the coordination work of PV-BESS. As a variant of the DE algorithm, the DOLADE algorithm introduces the DOL strategy to expand the number of particles in the population and improve the exploration and exploitation capability, achieving a faster convergence rate, accurately obtaining the global optimum and avoiding falling into the local optimum. First, based on the battery equivalent circuit model in BESS, the relationship between *OCV* and *SOC* is studied, and the parameters that need to be identified, the objective function, and the evaluation function are determined. Then, the parameters under various discharging currents are extracted, and the battery's *OCV* is estimated according to the identified parameters. Finally, the estimated *OCV* is compared with the experimental *OCV*, and the experimental voltage-estimated voltage curve and the *OCV–SOC* curve are plotted. It can be known from the aforementioned figures that all the parameters and indicators are in line with the expected values and within a reasonable and acceptable range, which further verifies the effectiveness and accuracy of the DOLADE algorithm. Furthermore, compared with the well-established algorithm, the boxplot and the convergence curves of the algorithms are plotted, which shows that the advantages of the DOLADE algorithm are available. In general, the DOL strategy can significantly improve the diversity of the population and enhance the search efficiency of particles by endowing new searching abilities to individuals in the population. Therefore, DOLADE is one of the ideal methods to identify unknown parameters of the battery model. In future plans, research to address this type of problem will continue, and improvements will be made to the DOLADE algorithm to explore its potential for solving more complex problems.

DATA AVAILABILITY STATEMENT

The raw data supporting the conclusion of this article will be made available by the authors, without undue reservation.

AUTHOR CONTRIBUTIONS

JZ: conceptualization, methodology, software, formal analysis, data curation, validation, and writing—original draft. YuZ: investigation and formal analysis. YG: formal analysis and validation. MM: resources and writing—review and editing. WF: funding acquisition. YaZ: writing—review and editing, supervision, project administration, and funding acquisition.

FUNDING

The authors are grateful for the financial support from the Shenzhen science and technology project (No. JCYJ 20210324115606017), the Science and Technology Innovation Commission of Shenzhen (No. ZDSYS20190902093209795), the National Natural Science Foundation of China (No. U1813222), Guangdong Frontier and Key Technological Innovation (No. 2017B090910013), and the Science and Technology Innovation Commission of Shenzhen (Nos. JCYJ20180507182239617 and JCYJ2018050718 223961).

REFERENCES

Alam, D. F., Yousri, D. A., and Eteiba, M. B. (2015). Flower Pollination Algorithm Based Solar Pv Parameter Estimation. *Energ. Convers. Manage.* 101, 410–422. doi:10.1016/j.enconman.2015.05.074

Brest, J., Maučec, M. S., and Bošković, B. (2020). "Differential Evolution Algorithm for Single Objective Bound-Constrained Optimization: Algorithm J2020," in 2020 IEEE Congress on Evolutionary Computation (CEC) (Glasgow, UK: IEEE), 1–8. doi:10.1109/cec48606.2020.9185551

Bruch, M., Millet, L., Kowal, J., and Vetter, M. (2021). Novel Method for the Parameterization of a Reliable Equivalent Circuit Model for the Precise Simulation of a Battery Cell's Electric Behavior. *J. Power Sourc.* 490, 229513. doi:10.1016/j.jpowsour.2021.229513

Bruen, T., and Marco, J. (2016). Modelling and Experimental Evaluation of Parallel Connected Lithium Ion Cells for an Electric Vehicle Battery System. *J. Power Sourc.* 310, 91–101. doi:10.1016/j.jpowsour.2016.01.001

Bullich-Massagué, E., Cifuentes-García, F.-J., Glenny-Crende, I., Cheah-Mañé, M., Aragüés-Peñalba, M., Díaz-González, F., et al. (2020). A Review of Energy Storage Technologies for Large Scale Photovoltaic Power Plants. *Appl. Energ.* 274, 115213. doi:10.1016/j.apenergy.2020.115213

Chen, X., Xu, B., Mei, C., Ding, Y., and Li, K. (2018). Teaching-learning-based Artificial Bee colony for Solar Photovoltaic Parameter Estimation. *Appl. Energ.* 212, 1578–1588. doi:10.1016/j.apenergy.2017.12.115

Cheng, S., Zhao, G., Gao, M., Shi, Y., Huang, M., and Marefati, M. (2021). A New Hybrid Solar Photovoltaic/phosphoric Acid Fuel Cell and Energy Storage System; Energy and Exergy Performance. *Int. J. Hydrogen Energ.* 46, 8048. doi:10.1016/j.ijhydene.2020.11.282

Choi, Y. Y., Kim, S., Kim, S., and Choi, J.-I. (2020). Multiple Parameter Identification Using Genetic Algorithm in Vanadium Redox Flow Batteries. *J. Power Sourc.* 450, 227684. doi:10.1016/j.jpowsour.2019.227684

El-Sehiemy, R. A., Hamida, M. A., and Mesbahi, T. (2020). Parameter Identification and State-Of-Charge Estimation for Lithium-Polymer Battery Cells Using Enhanced sunflower Optimization Algorithm. *Int. J. Hydrogen Energ.* 45 (15), 8833–8842. doi:10.1016/j.ijhydene.2020.01.067

Farhadi, B., Marriam, I., Yang, S., Zhang, H., Tebyetekerwa, M., Zhu, M., et al. (2019). Highly Efficient Photovoltaic Energy Storage Hybrid System Based on Ultrathin Carbon Electrodes Designed for a Portable and Flexible Power Source. *J. Power Sourc.* 422, 196–207. doi:10.1016/j.jpowsour.2019.02.091

Gao, S., Wang, K., Tao, S., Jin, T., Dai, H., and Cheng, J. (2021). A State-Of-The-Art Differential Evolution Algorithm for Parameter Estimation of Solar Photovoltaic Models. *Energ. Convers. Manage.* 230, 113784. doi:10.1016/j.enconman.2020.113784

Heidari, A. A., Mirjalili, S., Faris, H., Aljarah, I., Mafarja, M., and Chen, H. (2019). Harris Hawks Optimization: Algorithm and Applications. *Future generation Comput. Syst.* 97, 849–872. doi:10.1016/j.future.2019.02.028

Ikeda, S., and Ooka, R. (2019). Application of Differential Evolution-Based Constrained Optimization Methods to District Energy Optimization and Comparison with Dynamic Programming. *Appl. Energ.* 254, 113670. doi:10.1016/j.apenergy.2019.113670

Jiang, L. L., MaskellMaskell, D. L., and Patra, J. C. (2013). Parameter Estimation of Solar Cells and Modules Using an Improved Adaptive Differential Evolution Algorithm. *Appl. Energ.* 112, 185–193. doi:10.1016/j.apenergy.2013.06.004

Kang, J., Yan, F., Zhang, P., and Du, C. (2012). A Novel Way to Calculate Energy Efficiency for Rechargeable Batteries. *J. Power Sourc.* 206, 310–314. doi:10.1016/j.jpowsour.2012.01.105

Laquai, F., Chowanetz, F., and Rigoll, G. (2011). "A Large-Scale Led Array to Support Anticipatory Driving," in 2011 IEEE International Conference on Systems, Man, and Cybernetics (Anchorage, AK, USA: IEEE), 2087–2092. doi:10.1109/icsmc.2011.6083980

Li, R., Li, W., Zhang, H., Zhou, Y., and Tian, W. (2021). On-line Estimation Method of Lithium-Ion Battery Health Status Based on Pso-Svm. *Front. Energ. Res.* 401, 693249. doi:10.3389/fenrg.2021.693249

Liang, J., Qiao, K., Yuan, M., Yu, K., Qu, B., Ge, S., et al. (2020). Evolutionary Multi-Task Optimization for Parameters Extraction of Photovoltaic Models. *Energ. Convers. Manage.* 207, 112509. doi:10.1016/j.enconman.2020.112509

Lin, X., and Wu, Y. (2020). Parameters Identification of Photovoltaic Models Using Niche-Based Particle Swarm Optimization in Parallel Computing Architecture. *Energy* 196, 117054. doi:10.1016/j.energy.2020.117054

Liu, R., and Zhang, C. (2021). An Active Balancing Method Based on Soc and Capacitance for Lithium-Ion Batteries in Electric Vehicles. *Front. Energ. Res.* 9, 773838. doi:10.3389/fenrg.2021.773838

Maghyereh, A., and Abdoh, H. (2021). The Impact of Extreme Structural Oil-price Shocks on Clean Energy and Oil Stocks. *Energy* 225, 120209. doi:10.1016/j.energy.2021.120209

Mirjalili, S., Mirjalili, S. M., and Lewis, A. (2014). Grey Wolf Optimizer. *Adv. Eng. Softw.* 69, 46–61. doi:10.1016/j.advengsoft.2013.12.007

Mirjalili, S. (2015). Moth-flame Optimization Algorithm: A Novel Nature-Inspired Heuristic Paradigm. *Knowledge-based Syst.* 89, 228–249. doi:10.1016/j.knosys.2015.07.006

Pathak, V. K., and Singh, A. K. (2017). Form Error Evaluation of Noncontact Scan Data Using Constriction Factor Particle Swarm Optimization. *J. Adv. Manuf. Syst.* 16 (03), 205–226. doi:10.1142/s0219686717500135

Peng, F., Tang, K., Chen, G., and Yao, X. (2009). "Multi-start Jade with Knowledge Transfer for Numerical Optimization," in 2009 IEEE Congress on Evolutionary Computation (Trondheim: IEEE), 1889–1895. doi:10.1109/cec.2009.4983171

Qais, M. H., Hasanien, H. M., and Alghuwainem, S. (2019). Identification of Electrical Parameters for Three-Diode Photovoltaic Model Using Analytical and sunflower Optimization Algorithm. *Appl. Energ.* 250, 109–117. doi:10.1016/j.apenergy.2019.05.013

Rahnamayan, S., Tizhoosh, H. R., and Salama, M. M. A. (2007). "Quasi-oppositional Differential Evolution," in 2007 IEEE congress on evolutionary computation (Singapore: IEEE), 2229–2236. doi:10.1109/cec.2007.4424748

Rao, R. V., and Patel, V. (2012). An Elitist Teaching-Learning-Based Optimization Algorithm for Solving Complex Constrained Optimization Problems. *Int. J. Ind. Eng. Computations* 3 (4), 535–560. doi:10.5267/j.ijiec.2012.03.007

Siecker, J., Kusakana, K., and Numbi, B. P. (2017). A Review of Solar Photovoltaic Systems Cooling Technologies. *Renew. Sustain. Energ. Rev.* 79, 192–203. doi:10.1016/j.rser.2017.05.053

Storn, R., and Price, K. (1997). Differential Evolution–A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. *J. Glob. optimization* 11 (4), 341–359. doi:10.1023/a:1008202821328

Tan, X., Zhan, D., Lyu, P., Rao, J., and Fan, Y. (2021). Online State-Of-Health Estimation of Lithium-Ion Battery Based on Dynamic Parameter Identification at Multi Timescale and Support Vector Regression. *J. Power Sourc.* 484, 229233. doi:10.1016/j.jpowsour.2020.229233

Tizhoosh, H. R. (2005). "Opposition-based Learning: a New Scheme for Machine Intelligence," in International conference on computational intelligence for modelling, control and automation and international conference on intelligent agents, web technologies and internet commerce (CIMCA-IAWTIC'06), volume 1 (Vienna, Austria: IEEE), 695–701.

Tong, K., Yao, J., Jin, M., Yang, S., and Duong, T. (2018). A Novel Improved Cuckoo Search Algorithm for Parameter Estimation of Photovoltaic (Pv) Models. *Energies* 11 (5), 1060.

Venkata Rao, R. (2016). Jaya: A Simple and New Optimization Algorithm for Solving Constrained and Unconstrained Optimization Problems. *Int. J. Ind. Eng. Computations* 7 (1), 19–34. doi:10.5267/j.ijiec.2015.8.004

Wang, H., Wu, Z., Rahnamayan, S., Liu, Y., and Ventresca, M. (2011). Enhancing Particle Swarm Optimization Using Generalized Opposition-Based Learning. *Inf. Sci.* 181 (20), 4699–4714. doi:10.1016/j.ins.2011.03.016

Wang, Q., Kang, J., Tan, Z., and Luo, M. (2018). An Online Method to Simultaneously Identify the Parameters and Estimate States for Lithium Ion Batteries. *Electrochimica Acta* 289, 376–388. doi:10.1016/j.electacta.2018.08.076

Wang, Q., Wang, J., Zhao, P., Kang, J., Yan, F., and Du, C. (2017). Correlation between the Model Accuracy and Model-Based Soc Estimation. *Electrochimica Acta* 228, 146–159. doi:10.1016/j.electacta.2017.01.057

Wang, S., Liu, S., Yang, F., Bai, X., Wang, S., and Yue, C. (2020). Novel Power Allocation Approach in a Battery Storage Power Station for Aging Minimization. *Front. Energ. Res.* 7, 166. doi:10.3389/fenrg.2019.00166

Xia, Z., and Xiang, Z. (2020). Catalysts for Clean Energy Conversion and Storage. *Front. Mater.* 7, 43. doi:10.3389/fmats.2020.00043

Xiong, R., He, H., Sun, F., and Zhao, K. (2012). Evaluation on State of Charge Estimation of Batteries with Adaptive Extended Kalman Filter by experiment Approach. *IEEE Trans. Vehicular Technol.* 62 (1), 108–117.

Yang, Z., Liu, Q., Zhang, L., Dai, J., and Razmjooy, N. (2020). Model Parameter Estimation of the Pemfcs Using Improved Barnacles Mating Optimization Algorithm. *Energy* 212, 118738. doi:10.1016/j.energy.2020.118738

Zhang, J., and Sanderson, A. C. (2009). Jade: Adaptive Differential Evolution with Optional External Archive. *IEEE Trans. Evol. Computat.* 13 (5), 945–958. doi:10.1109/tevc.2009.2014613

Zhang, Y., Huang, C., and Jin, Z. (2020). Backtracking Search Algorithm with Reusing Differential Vectors for Parameter Identification of Photovoltaic Models. *Energ. Convers. Manage.* 223, 113266. doi:10.1016/j.enconman.2020.113266

Zhou, J., Zhang, Y., Zhang, S., Guo, Y., Yang, Z., Feng, W., et al. (2021). A Novel Maximum Power point Tracking Strategy Based on Enhanced Real-Time Adaptive Step-Size Modified Control for Photovoltaic Systems. *Front. Energ. Res.* 9, 685415. doi:10.3389/fenrg.2021.685415

Zhou, S., Liu, X., Hua, Y., Zhou, X., and Yang, S. (2021). Adaptive Model Parameter Identification for Lithium-Ion Batteries Based on Improved Coupling Hybrid Adaptive Particle Swarm Optimization- Simulated Annealing Method. *J. Power Sourc.* 482, 228951. doi:10.1016/j.jpowsour.2020.228951

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

GLOSSARY

A_E experiment current
A_S calculated current
BESS battery energy storage system
C_d concentration polarization capacitance (C)
C_p electrochemical polarization capacitance (C)
cfPSO constriction factor particle swarm optimization
D dimension of problem
DE differential evolution
DOL dynamic opposite learning
DOLADE adaptive differential evolution algorithm with a dynamic opposite learning strategy
ETLBO elitist teaching–learning-based optimization
G_{max} maximal number of generation
GWO grey wolf optimizer
I_L load current (A)
JADE adaptive differential evolution algorithm with an optional external archive
Max maximum value
Mean average value
MFO moth-flame optimization
Min minimum value
N number of experimental data
Np population size
OCV open circuit voltage
P population
PV photovoltaic
R_0 internal resistance (Ω)
R_d concentration polarization resistance (Ω)
R_p electrochemical polarization resistance (Ω)
RMSE root mean square error
$rand$ random numbers in (0, 1)
SD standard deviation
SECM second-order RC equivalent circuit model
SOC state of charge
T sampling time (s)
U_0 resistance voltage (V)
U_d second RC network's voltage (V)
U_L terminal voltage (V)
U_{OCV} open circuit voltage (V)
U_p first RC network's voltage (V)
μCR crossover factor
μF mutation factor