A novel genetic weight-directed feed forward backpropagation neural network for state of charge estimation of lithium-ion batteries.

HAI, N., WANG, S., LIU, D., FERNANDEZ, C. and GUERRERO, J.M.

2024

A Novel Genetic Weight-directed Feed Forward Backpropagation Neural Network for State of Charge Estimation of Lithium-ion Batteries

Nan Hai[a], Shunli Wang[a*],Donglei Liu[a], Carlos Fernandez[b] , Josep M. Guerrero[c]

[a]*School of Information Engineering, Engineering and Technology Center, Southwest University of Science and Technology, Mianyang 621010, China.*

[b]*School of Pharmacy and Life Sciences, Robert Gordon University, Aberdeen AB10-7GJ, UK.*

[c]*Department of Energy Technology, Aalborg University, Pontoppidanstraede 111 9220 Aalborg East, Denmark*

Abstract: Precious estimation of state-of-charge has become a more important status to the lithium-ion batteries of electronic vehicles. Basically, a three-layer genetic algorithm based on feed forward backpropagation neural network model is established. Specifically, an adaptive genetic method that makes the p_c and p_m change and self-correct with the degree of adaptation F is proposed to improve the stability and accuracy. Then, the momentum volume $\Delta w_{ji}^{1}(\delta)$ and the inertial volume $\Delta w_{kj}^{2}(\mu)$ are introduced to the first and the second weight of the topology in weighting correction process of backpropagation to help reduce the convergence time and improve the matching of the system with the increase in data volume. Finally, a further performance comparison of variable algorithms based on the backpropagation neural network is made under different working conditions at variable temperatures with large data volumes to prove the effectiveness of the proposed methods. The experimental results showed that the maximum error reached 0.9%, 1.2% and 0.3% under BBDST at 35°C, 25°C and 0°C over 500000 data, similarly, it reached 0.18%, 0.1% and 0.69% under DST at 15°C, 25°C and 35°C over 200000 data.

Keywords: lithium-ion batteries; state of charge; weight-directed; feed-forward backpropagation; adaptive genetic algorithm

Corresponding author: Shunli Wang. Tel/fax: +86-15884655563 E-mail address: wangshunli@swust.edu.cn

1. Introduction

With the development of renewable energy utilization and energy storage system technology, the new energy vehicle has become an efficient way to alleviate the environmental pollution caused by gasoline[1, 2]. The lithium-ion batteries have become the best choice for vehicle power for their advantages such as large storage capacity, no memory effect, lightweight, low self-discharge rate, stable discharge voltage, environmental friendliness, etc[3, 4]. To avoid the potential safety hazards of lithium-ion vehicles, the battery management system (BMS) can monitor and control the state of the batteries during driving[5, 6]. The state of charge (SOC) plays a significant role in BMS, which can directly reflect the remaining power of the battery[7, 8].

For SOC estimation, it is defined as the ratio of the remaining capacity to the maximum available capacity of the battery. The accurate estimation of SOC is the basis for the safe and stable operation of lithium batteries. The electrochemical reaction process of power batteries is complex and is affected by temperature, humidity, aging, and many other factors[9, 10]. Based on the internal structure of batteries, traditional methods for SOC estimation include the Open-circuit voltage (OCV) method and the Ampere-hour (Ah) integral method[11, 12]. These methods need to clarify the electrochemical reaction inside the batteries, the characteristics of electrochemical material, and the failure mechanism in depth, to establish the physical failure model of the batteries[13, 14]. A scheme based on OCV-Ah optimized with Extended Kalman filter (EKF) is proposed to achieve the process of parameter identification[15], which is

particularly important for neural network models based on voltage and current as inputs. In the modeling process, many assumptions and empirical parameters are applied. The model accuracy is limited, the expression of the model is a multi-parameter partial differential equation system, and the solution process is very cumbersome. For the batteries in the actual working state, it is obvious that this method is difficult to meet the actual need[16, 17].

With the development of deep learning, using the neural network to estimate the SOC estimation of lithium-ion batteries is an increasing trend with its simple structure[18-20]. Correspondingly, an adaptive Kalman estimator based on GA-optimized extreme learning machine (ELM) has been constructed, showing both the root mean square error and the mean absolute error to be less than 1.2%[21], and combining filters and machine learning in this way can overcome the insensitivity to nonlinear non-Gaussian systems in conventional filtering in a relatively short time. Different from the GA, the differential evolution (DE) algorithm implements population perturbations by selecting particle differential information randomly[22, 23]. Then, a new modified DE algorithm enhances population diversity through a muli-angle searching strategy, which is presently one of the most powerful swarm intelligence optimization algorithms[24]. Then, a combined PSO and Least Squares Support Vector Machine (LSSVM) model (PSO-LSSVM) is constructed to estimate the SOC during non-constant current (CC) discharging with the maximum relative error standing at only 2.1%[25, 26], and with the help of particle optimization feature can effectively help the network to jump out of the local optimum, while LSSVM can achieve higher accuracy prediction, such a combination is to merge the advantages of the two, to improve the network adaptive ability. The Beetle Antennae search algorithm (BAS) is an intelligent single-unit search algorithm

proposed, and it optimized recurrent ELM to solve the modeling accuracy problem, which simulates the search of an aspen when it seeks food with high precision[27, 28].

Traditional algorithms for experiments and applications of the external conditions, such as the ampere-time integration method and the accuracy of the measurement equipment related to the measurement of the existence of errors, are more demanding than the current method for SOC estimation. Second, in the case of non-linear, non-Gaussian systems, the Kalman filter approach is computationally demanding and unstable. To increase prediction performance, select an appropriate optimization strategy from the variety of neural network algorithms available. Each neural network algorithm has unique properties, prediction content, while the second basic BP network is unable to satisfy the demands of high-precision prediction because of its poor accuracy. The selection, crossover, and mutation operations in traditional GA also have the problem of empirically taking values in the interval, which has a greater impact on the stability and matching of the network.

To solve the problem and improve the performance mentioned before, a novel genetic weight-directed algorithm feed forward backpropagation neural network (GWD-FFBPNN) is established in this paper. Particularly, an adaptive genetic algorithm that makes the p_c and p_m change and self-correct with the degree of adaptation F in the optimization process is proposed to improve the stability and accuracy of the three-layer GA-FFBPNN. Then, the inertial volume $\Delta w_{kj}^{2}(\mu)$ from the output layer to the hidden layer and the momentum volume $\Delta w_{ji}^{1}(\delta)$ from the hidden layer to the input layer are introduced in the weighting correction process of backpropagation, which can help to reduce the convergence time and improve the matching of the system. What's more, the performance of modified system GWD-FFBPNN

system is verified under DST, and BBDST working conditions, using mean absolute error (*MAE*), and the goodness of fit (R^2), root mean square error of prediction (*RMSEP*), and a further comparison with other algorithms is completed.

2. A novel GWD-FFBP modeling and mathematical analysis

2.1 Genetic algorithm based on backpropagation neural network

The typical multi-layer perception network is a three-layer hierarchical neural network, including the input layer, the hidden layer, and the output layer[29, 30]. According to the prediction content of lithium-ion batteries, the current and voltage are set as input layers, and the SOC is set as the output layer, in this way, a three-layer FFBPNN can be confirmed according to the Equation (1).

$$H = \sqrt{m+n} + \alpha \tag{1}$$

H, *n*, and *m* represent the number of the hidden layer, the input layer, and the output layer, respectively. Parameter α is set as 3 to build the FFBPNN with a framework of 2-5-1 pattern. The topology and its neuron can be observed in **Figure 1**.

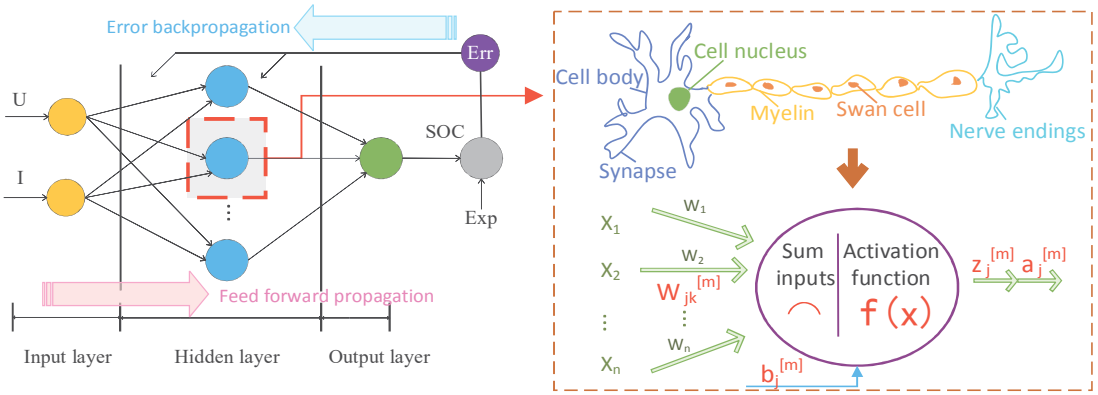


Figure 1. The topology and neurons of the FFBPNN

(1) Feed forward propagation

In **Figure 1**, a random neuron $N_j^{[m]}$ is the *j*th neuron from *m*th layer with *n* inputs is

selected to analyze. The input is $(x_1, x_2, x_3, x_4, \ldots x_n)^T$, its corresponding variable weight matrix is $(w_1, w_2, w_3, w_4, \ldots w_n)^T$ and the $b_j^{[m]}$ represents the deviation, then the linear input in summation is $z_j^{[m]} = w_1x_1 + w_2x_2 + w_3x_3 + \ldots + w_nx_n + b_j^{[m]}$, and $y_j^{[m]} = f(*)$, and the *f(*)* is an activation function[31]. The input $z_j^{[m]}$ of the network can be analyzed as shown in Equation (2).

$$Z_j^{[m]} = \sum_{i=1}^{n} w_{ji}^{[m]} x_{ji}^{[m]} + b_j^{[m]} = W_j X + b_j \tag{2}$$

Then the output of the system can be observed in Equation (3).

$$y_j = f(Z_j) = f(\sum_{i=i}^{n} w_{ji}^{[m]} * x_{ji}^{[m]}) = F(W_j X) \tag{3}$$

The sigmoid function is smooth and easy to derive, and its derivative function with concerning *x* can be expressed in terms of itself, which can be observed in Equation (4).

$$f(x) = \frac{1}{1 + e^{-x}} \tag{4}$$

(2) Error backpropagation

Gradient descent is an implementation of backward propagation of error that causes the weight of each training sample to vary along a negative gradient until E is minimized[32, 33]. According to the network structure in **Figure 1**, the nodes of the input layer, the implicit layer and the output layer are set as $X_i(n), H_j(n), Y_k(n)$, the connection weight of the *i*th neuron in the input layer to the *j*th neuron in the hidden layer is W_{ji}^1, similarly, the connection weight of the *j*th neuron in the hidden layer to the *k*th neuron in the output layer is W_{kj}^2. The specific steps of error backpropagation are in **Table 1**.

Table 1. Specific steps of error backpropagation

Step 1.	Confirming the output of the hidden layer and the output layer nodes, as shown in Equation (5). The θ_j and θ_k are the deviation of the hidden layer and the output layer, respectively, which is similar with the b_j with opposite direction.

$$\begin{cases} H_j(n) = f(\sum_{i=0}^{n} W_{ji}^{1} x_i - \theta_j) \\ Y_k(n) = f(\sum_{i=0}^{n} W_{kj}^{2} x_i - \theta_k) \end{cases} \tag{5}$$

Step 2. Calculating the error by the Equation (6). According to the neural network model established in **Figure 1**, the values of *m* and *n* can be determined as 1 and 2, and the $\hat{y}_{jk}$ means the expected output, and y_{jk} is the actual output of the system.

$$\begin{cases} E_j^{[m]} = \frac{1}{2}\sum_{k=1}^{n} e_j^2 \\ E = \sum_{j=1}^{N} E_j = \frac{1}{2}\sum_{i=1}^{N}\sum_{k=1}^{n} e_j^2 \\ e_j = \hat{y}_{jk} - y_{jk} \end{cases} \tag{6}$$

Step 3. Calculation of the error function and the partial derivatives of the output layer and the hidden layer nodes are shown in Equation (7), where the Y_k' represents the result of the output layer's derivation of the connection weights with the help of intermediate variables.

$$\begin{cases} \frac{\partial E}{\partial W_{kj}^{2}} = \sum_{k=1}^{l} \frac{\partial E}{\partial Y_k} * \frac{\partial Y_k}{\partial W_{kj}^{2}} = \frac{\partial E}{\partial Y_k} * \frac{\partial Y_k}{\partial W_{kj}^{2}} = -e_k * Y_k{}' * Y_j \\ \frac{\partial E}{\partial W_{ij}^{1}} = \sum_i \sum_j \frac{\partial E}{\partial Y_k} * \frac{\partial Y_k}{\partial H_j} * \frac{\partial H_j}{\partial W_{ij}^{1}} = -Y_j{}' * \sum_k e_k * Y_k{}' W_{jk}^{2} \end{cases} \tag{7}$$

Step 4. Using the gradient descent principle, let the connection weights be corrected according to the learning rate σ[28], and σ' is the learning rate for the next weight update. The H_j' is also the output of the implicit layer during the next weight update. The connection weights of the nodes in the output and hidden layers are updated into Equation (8).

$$\begin{cases} W_{kj}^{2}(m+1) = W_{kj}^{2}(m) + \Delta W_{kj}^{2} = W_{kj}^{2}(m) + \sigma e_k * Y_k{}' H_j \\ W_{ji}^{1}(m+1) = W_{ji}^{1}(m) + \Delta W_{ji}^{1} = W_{ji}^{2}(m) + \sigma' H_j{}' * e_k * Y_k{}' W_{kj}^{2}(m) * x_i \end{cases} \tag{8}$$

Step 5. Let m=m+1, until the error is less than the expected value, the neural network learning is finished.

According to the principle of survival of the fittest, the approximate solutions can be produced generation by generation after the emergence of the primary population with genetic selection and mechanisms in Darwin's theory of biological evolution[34, 35]. Based on the fitness size of the individual to confirm the selection, and with the help of genetic operators to finish the crossover and mutation to produce the new decompositions[36, 37].

In this article, the adaptation function and the objective function need to be converted, and the reciprocal of the sum of errors is designed as the fitness function, which is show as below:

$$F = \frac{1}{k\,|\sum_{i=1}^{n} e_j\,|} \tag{9}$$

The genetic algorithm specifies that the fitness function is non-negative and always evolves in the direction of the maximum value of the fitness function[38, 39], Moreover, according to the

regulation of the roulette wheel selection, the higher the chance of individuals with high adaptation to enter the next generation. The selection probability of the *i*th individual $\tilde{p}_i$ is

$$\begin{cases} p_i = kf_i \\ S_i = \dfrac{p_i}{\sum_{j=1}^{n} P_j}, i = 1,2,3...n, j = 1,2,3...n \end{cases} \tag{10}$$

where f_i means the fitness of a certain individual, k is the coefficient, n means the number of individuals, and S_i means the probability of selection. In this way, the diversity of individuals in the population is guaranteed.

In this paper, a new calculation of mutation is proposed, which will be finished in the last step and then the new population will be generated to start the next iteration, and it is shown in Equation (11). Mutation process of the *j*th gene of individual a_i is as below:

$$\begin{cases} a_{ij} = \begin{cases} a_{ij} + (a_{ij} - a_{\max}) * f(g), r > 0.5 \\ a_{ij} + (a_{\min} - a_{ij}) * f(g), r \leqslant 0.5 \end{cases} \\ f(g) = r(1 - g / G_{\max}) \end{cases} \tag{11}$$

where a_{max} and a_{min} are the upper and lower bound of gene a_{ij}, using a random number r in the interval [0,1] to determine the final model, g means the number of current iterations, and the G_{max} is the maximum of iterations. Similar to the p_c, the crossover probability p_m takes values in the range of (0.0001,0.1).

Problems that have arisen in the use of the traditional GA-FFBPNN to predict the SOC of lithium-ion batteries include the following three points:

(1) The computation of the traditional GA-FFBPNN for estimating the SOC of lithium-ion batteries becomes larger as the data set and the size of the neural network increase, and the training time increases.

(2) Empirical taking in the corresponding interval leads to increased uncertainty, and the

traditional GA-FFBPNN have a strong dependency on the initial weight threshold setting, which can influence the accuracy and the stability of the system.

(3) The number of the hidden layer in the topology is determined empirically for a given interval, which is not based on evidence and is subject to uncertainty.

2.2 A novel GWD-FFBP strategy

Analysis findings showed that the classic GWD-FFBPNN has a problem with low matching, which causes slow convergence, and a loss of stability and accuracy due to the sheer number of data. Therefore, an adaptive genetic method that makes the p_c and p_m change and self-correct with the degree of adaptation F is proposed to improve the stability and accuracy of the three-layer GWD-FFBPNN. Then, the inertial volume $\Delta w_{kj}^2(\mu)$ and the momentum volume $\Delta w_{ji}^1(\delta)$ are introduced to the first and the second weight of the topology in weighting correction process of backpropagation to help reduce the convergence time and improve the matching of the system with complex working conditions at variable temperatures. The flowchart of the novel GWD-FFBPNN is shown in **Figure 2**.

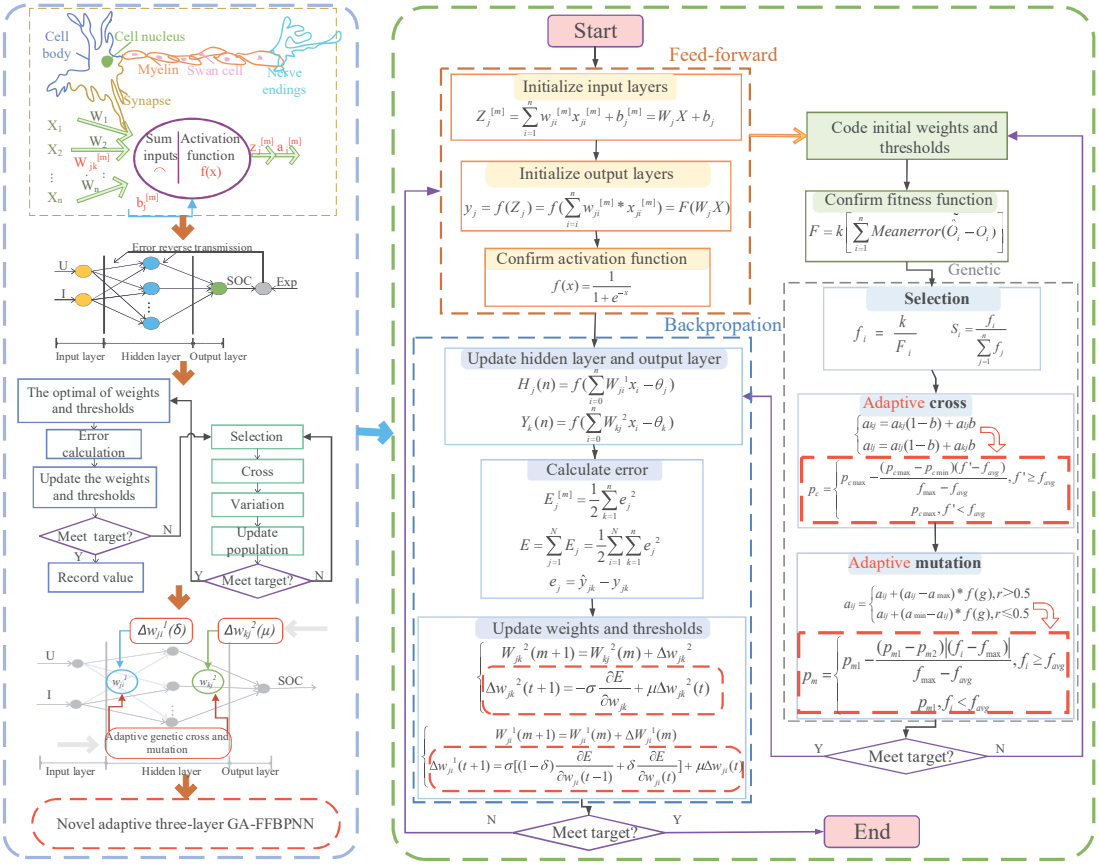


Figure 2. The flow chart of the novel GWD-FFBPNN

2.2.1 A novel adaptive genetic cross and mutation operation.

The network based on threshold training characteristics and initial weights indicates that the genetic approach can assist traditional FFBPNN in avoiding local minima. Among the parameters of the genetic algorithm, the choice of p_c and p_m is crucial to the behavior and performance of the genetic algorithm, which can influence the behavior and performance if the value is fixed. To address this problem, this paper proposes an adaptive genetic algorithm that makes the p_c and p_m change and self-correct with the degree of adaptation F in the optimization process. Adaptive adjustment is based on

$$p_c = \begin{cases} p_{c\max} - \dfrac{(p_{c\max} - p_{c\min})(f' - f_{avg})}{f_{\max} - f_{avg}}, f' \geq f_{avg} \\ p_{c\max}, f' < f_{avg} \end{cases} \tag{12}$$

where the p_{cmax} and p_{cmin} take the value of the upper bound and lower bound of the interval [0.4,0.99], respectively. f' is the larger value of fitness among the two individuals in the crossover, $f_{max}\ and\ f_{avg}$ represent the maximum and mean value of fitness of all individuals in the population, respectively. In this way, when the F of most individuals in the population tends to be locally optimal or concentrated, the value of the p_c and p_m automatically increase, and relatively, when the F of individuals in the population is distributed, the p_c self-corrects to a smaller value.

Similarly, the self-correction is achieved by combining the p_m and the fitness of individuals in the population, which is divided into two intervals by comparing the magnitude of the variant individual fitness with the f_{avg},

$$p_m = \begin{cases} p_{m\max} - \dfrac{(p_{m\max} - p_{m\min})(f - f_{avg})}{f_{\max} - f_{avg}}, f \geq f_{avg} \\ p_{m\max}, f < f_{avg} \end{cases} \tag{13}$$

where the f means the fitness of a certain individual, and the p_{mmax} and p_{mmin} take the value of the upper bound and lower bound of the interval [0.0001,0.1]. In this way, the probability of variation is adaptively adjusted by linking the maximum and minimum values of the interval to the fitness value of the population individuals.

2.2.2 Modified weighting correction strategy

In genetic algorithms, the crossover function and variogram are taken within a given interval. The empirical values are unclear when input data volume and topology vary, and a mismatch will cause the network to mature too quickly or become unstable. Simultaneously, matching is required to rectify the weight that the adaptive genetic algorithm optimized. Therefore, the paper proposed a kind of adaptive method which can be observed in **Figure 3**,

and a more detailed illustration follows.

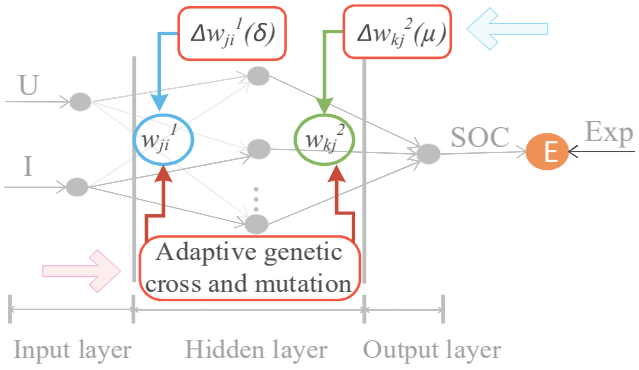


Figure 3. Adaptive implementation process map

(1) Introduce the inertial volume $\Delta w_{kj}^{2}(\mu)$

As analyzed by the backpropagation process of the FFBPNN, the error values are fed back to the output layer, the hidden layer, and the input layer in turn to help each layer continuously update its own weights and thresholds. However, it would take a lot of time and would not work with the adaptive adjustment genetic algorithm to alter each layer successively. To improve the adaptive matching, this paper introduced inertial volume in weighting correction process from the output layer to the hidden layer, which are described as:

$$\begin{cases} W_{kj}^{\ 2}(m+1) = W_{kj}^{\ 2}(m) + \Delta w_{kj}^{\ 2} \\ \Delta w_{kj}^{\ 2}(t+1) = -\sigma \dfrac{\partial E}{\partial w_{kj}} + \mu \Delta w_{kj}^{\ 2}(t) \end{cases} \tag{14}$$

where $\Delta w_{kj}^{2}(t+1)$ denotes the increment in connection weights that should be corrected at moment $t+1$ from the output layer to the hidden layer. The μ is inertia coefficient, which is set as a random variable within [0.1,0.99], and the upper bounds of its interval are taken from the maximum of the p_{cmax} and the p_{mmax}, respectively, in this way, the excellent dynamic characteristics at time $t+1$ will be achieved by adding the inertial characteristics at time t. From the expression, it is clear that the correction increments at moment $t+1$ is further increased when the connection weight increment is larger at moment t.

(2) Introduce the momentum volume $\Delta w_{ji}^{1}(\delta)$

To address the fact that operations limited to the negative gradient direction during error feedback prevent the network from stabilizing quickly when the amount of data becomes large, momentum volume are introduced in the weight corrections of the implicit and input layers and are related to the previous time.

$$\begin{cases} W_{ji}^{1}(m+1) = W_{ji}^{1}(m) + \Delta W_{ji}^{1} \\ \Delta w_{ji}^{1}(t+1) = \sigma[(1-\delta)\dfrac{\partial E}{\partial w_{ji}(t-1)} + \delta\dfrac{\partial E}{\partial w_{ji}(t)}] + \mu\Delta w_{ji}(t) \end{cases} \tag{15}$$

In Equation (15), $\Delta w_{ji}^{1}(t+1)$ denotes the increment in connection weights that should be corrected at moment *t+1* from the hidden layer to the input layer. The δ is momentum coefficient, which is set as a random variable within [0.1,0.99], and the upper bounds of its interval are taken from the maximum of the p_{cmax} and the p_{mmax}, respectively. It can be observed that the weight is adjusted according to the gradient values at both *t-1* and *t* moment. Hence, the introduction of the momentum volume δ correlates the preceding *t-1* and following moments *t+1* of the weight increments to achieve dynamic self-correction.

3. Experimental testing and analysis

3.1 Test platform establishment

In the experiment, the 3.7V/100 Ah ternary battery is set as the test object, and the Neware battery test equipment is the CT-4016-5V100A-NTFA, and the constant temperature box is the DGBELL BTT-331C. The experimental platform of the target lithium-ion battery test equipment is observed in **Figure 4**. The parameters of the tested battery are shown in **Table 1**. The parameter setting of the novel GWD-FFBPNN proposed during the experiment is displayed in **Table 2**.

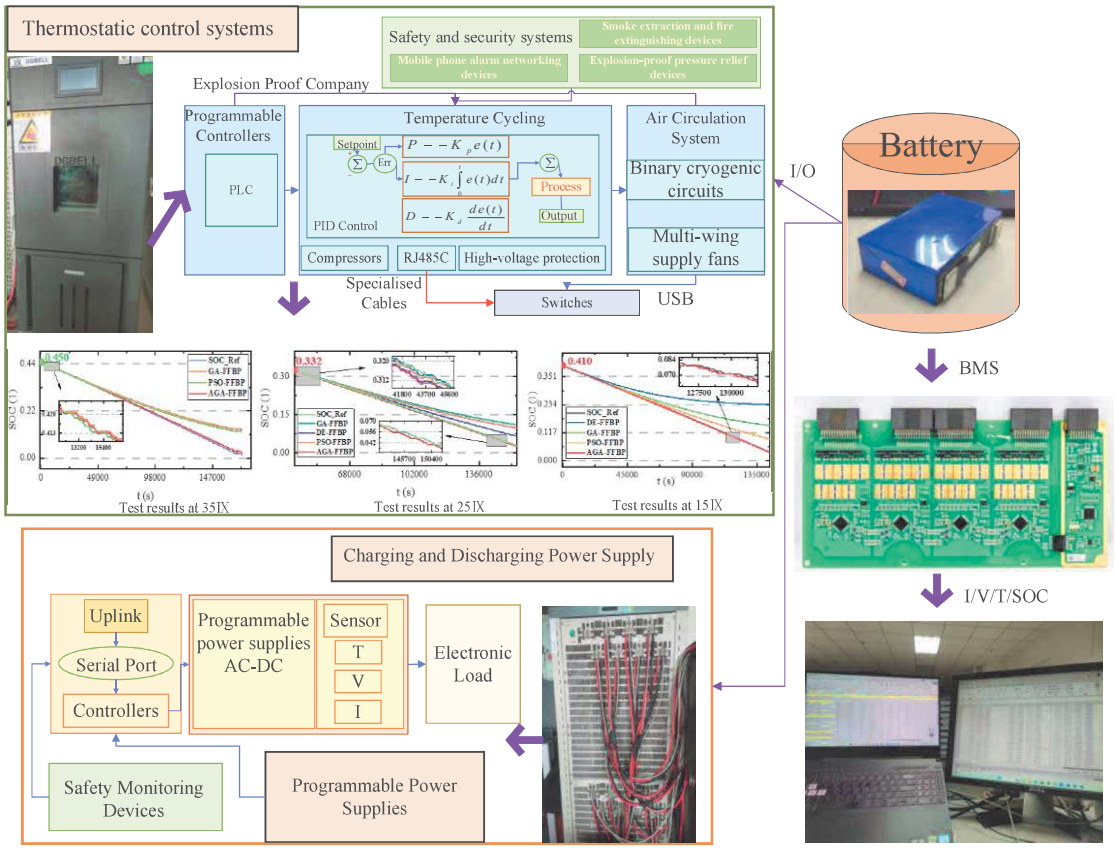


Figure 4. The experimental platform establishment

Table 1. The specification of 3.7V/100Ah lithium-ion battery

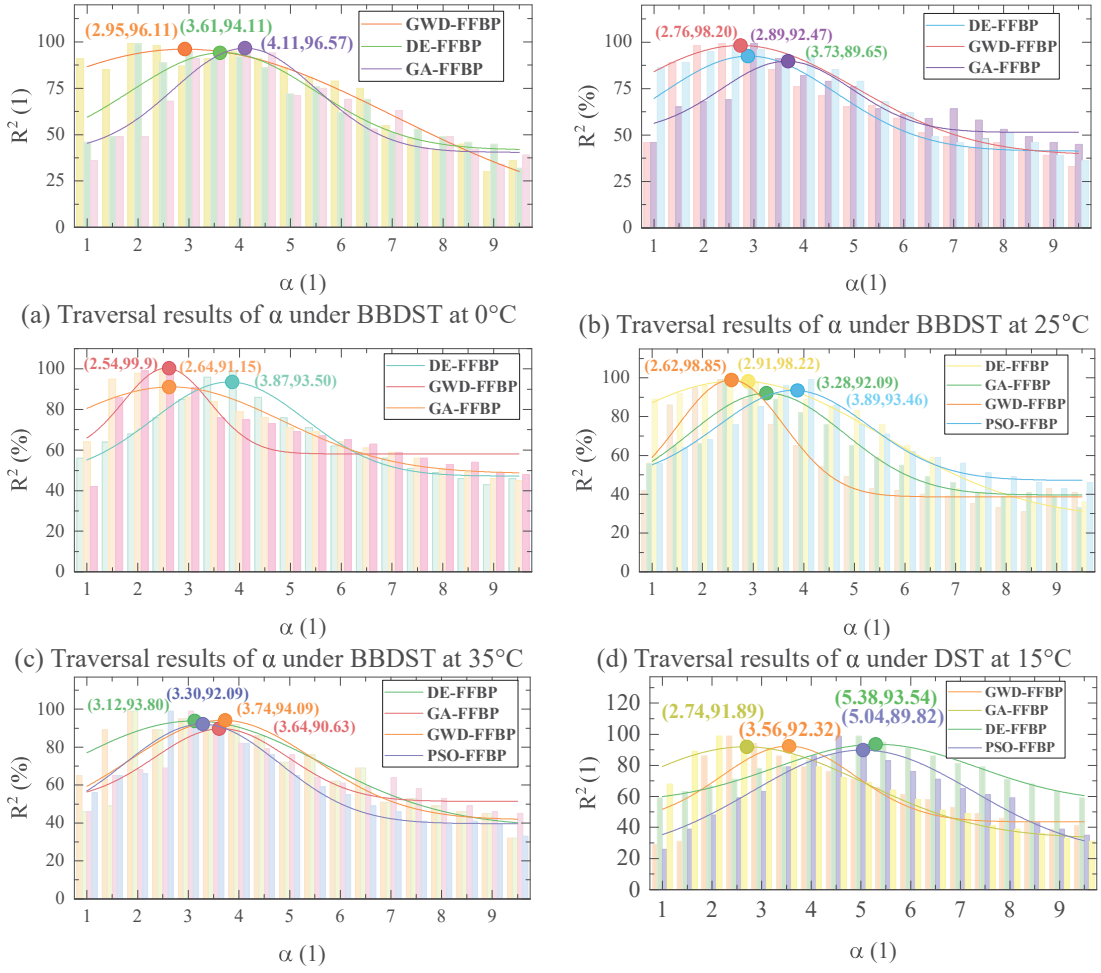
Cell nominal capacity (Ah)	100	Peak discharge current	3C
Cell nominal capacity (V)	3.7	Maximum load current	2C
Charge cut-off voltage (V)	4.5±0.05	Internal resistance (mΩ)	0.5-1
Discharge cut-off voltage (V)	2.75±0.05	Working temperature (℃)	-20-60
Standard charge current	1C	Dimension: 1*w*h (mm)	148*27*93

Table 2. The novel GWD-FFBP parameter settings

The maximum number of iterations	50	Number of training samples	>1000
Number of variables	31	Number of testing samples	>1000
Generation gap	0.92	Original learning rate	0.1
Cross-probability	0.43	Training method	Gradient descent
Mutation probability, migration probability	0.02,0.4	Population size	40

3.2 Determination of the hidden layers number by traversal method

Generally, the parameter setting is critical for the number of hidden layers in the topology can be determined according to Equation (1), and the range of parameters will affect the training effect of the network directly. The iterative approach is used to determine the optimal values for different temperatures and different working conditions, and the traversal results are shown in **Figure 4**.



(a) Traversal results of α under BBDST at 0°C

(b) Traversal results of α under BBDST at 25°C

(c) Traversal results of α under BBDST at 35°C

(d) Traversal results of α under DST at 15°C

(e) Traversal results of α under DST at 25°C

(f) Traversal results of α under DST at 35°C

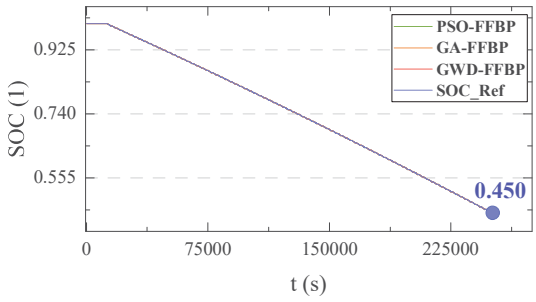
Figure 4. Traverse results of α at different operating conditions and different temperatures

In **Figure 4**, the value of the parameter α under different working conditions at variable temperatures are confirmed through traversal in corresponding internal [1,9]. For BBDST working condition, the traversal value is determined as 2.95,2.76,2.54 at 0°C, 25°C, and 35°C,

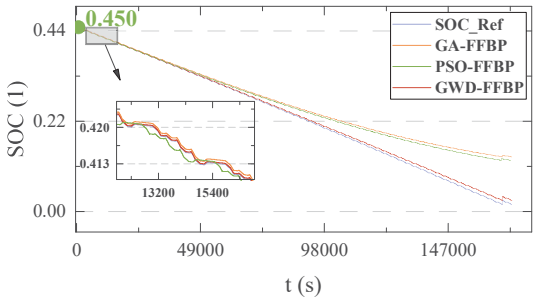
respectively from Figure **4** (a)(b)(c), so it is taken the corresponding integer to be 3. Similarly, for DST working condition, the traversal value of α is 2.6,3.74,3.56 at 15°C, 25°C, and 35°C in Figure **4** (d)(e)(f), then the value is taken the corresponding integer to be 3 and 4. The values for the other neural networks can be determined in the same way.

3.3 Analysis of BBDST working condition

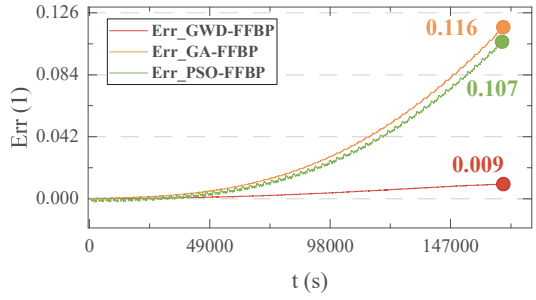
The performance verification of the novel GWD-FFBPNN is completed at various temperatures includes 35°C, 25°C and 0°C under the BBDST working conditions, comparing with the neural networks of the traditional GA-FFBP, DE-FFBP and PSO-FFBP. The details are shown in **Figure 5**.



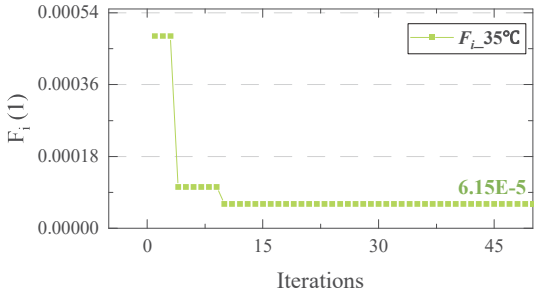
(a-1) Training results



(a-2) Test results



(a-3) Error curve of test

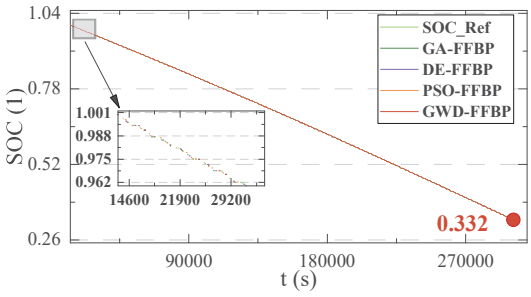


(a-4) Fitness curve

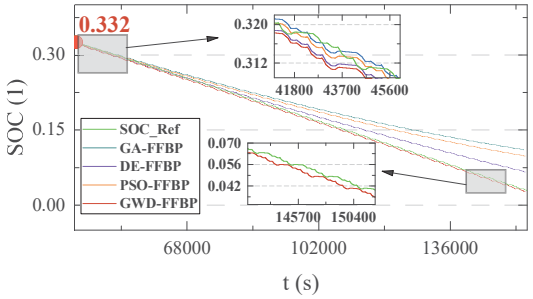
Figure 5 (a). Results curves under BBDST at 35°C

In Figure 5 (a-1), the training data set is close to 300,000, which is 60% of total data, and the training result curves show that the three different optimization algorithms acting on BP neural networks with high accuracy and stability for large training sets compared to the

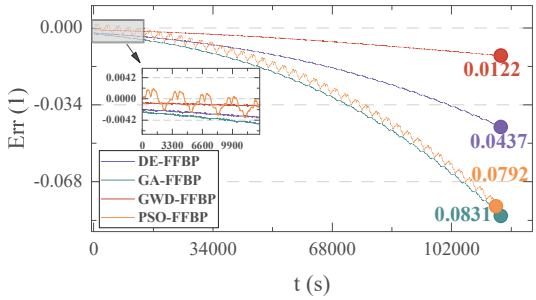
SOC_Ref. However, for predictions with data volumes close to 200,000, which is 40% of total data, the three algorithms show large differences in Figure 5 (a-2) and (a-3). It can be concluded from Figure 5 (a-2) that the four algorithms basically overlap in the resultant curves in the pre-test period, but as the amount of data increases, the errors of GA-FFBP and PSO-FFBP are getting larger, which is reflected in the fact that they deviate further from the SOC_Ref, whereas the GWD-FFBP always stays close to the SOC_Ref, which reflects its good stability and accuracy. The maximum error of the novel GWD-FFBP reached to 0.009, which achieved a high precision compared to 0.107 and 0.116 of the PSO-FFBP and the traditional GA-FFBP, respectively. In the error curves, GA-FFBP has the largest error maximum, but the fluctuations and peaks of its error curves are smaller, and the overall curves are smoother, while the jaggedness of the error curves of PSO-FFBP is more obvious. In comparison, GWD-FFBP has the smallest error and the smoothest curve without obvious peaks and jaggedness. The fitness of a neural network is presented in its inverse form F_i, the greater the fitness, the smaller the value in the graph Figure 5 (a-4), which reached to 6.15E-5. In addition to this, the verification at 25°C is presented in Figure 5 (b).



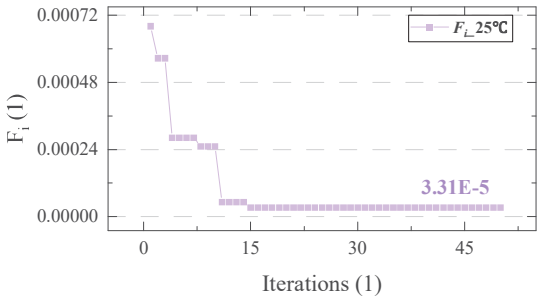
(b-1) Training results



(b-2) Test results



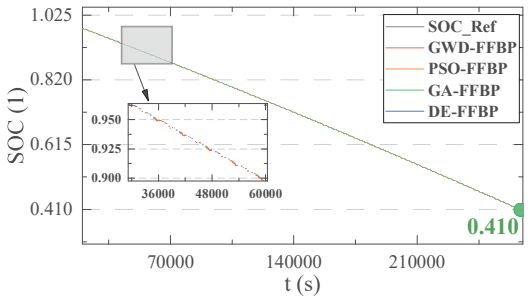
(b-3) Error curve of test



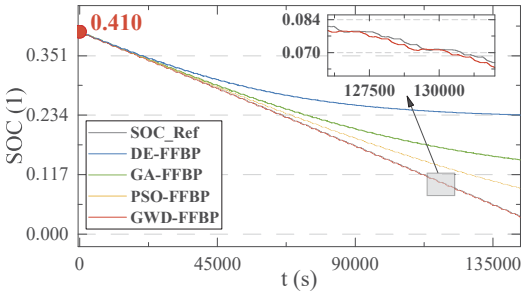
(b-4) Fitness curve

Figure 5 (b). Results curves under BBDST at 25°C

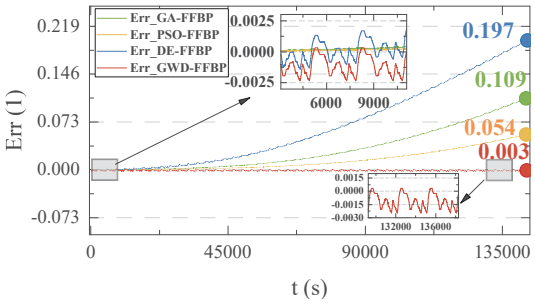
Compared to the test data at 35°C under BBDST, the 25°C test training data is significantly higher at over 350,000, and the ratio of training data and test data is 5:2. The training results of each network are stable and highly accurate in the training of huge amount of data, which is reflected in the fact that they all coincide with the SOC_Ref curve in Figure 5 (b-1), and their result curves largely overlap with the reference SOC_Ref. Test volumes up to 120,000 show that the DE, the traditional GA and the PSO algorithm has increasing errors with increasing data volumes of 0.0437, 0.0831 and 0.0792 respectively compared to the novel GWD-FFBP error value of 0.0122 from Figure 5 (b-2) and (b-3). The characteristics of the test result curves of each algorithm are different, in which GA-FFBP deviates the furthest from the SOC_Ref, but the error curve is smoother without obvious gears and peaks. PSO-FFBP has a larger error and the error curve has obvious regular fluctuations, with poor stability and matching. The GWD-FFBP not only has the smallest error, but also has a smoother error curve, which is a smooth arc with a downward trend near the 0 straight line. The F_i is shown in Figure 5 (b-4), which reached at 3.31E-5. Other than that, the verification of the novel GWD-FFBP at 0°C is presented in Figure 5 (c).



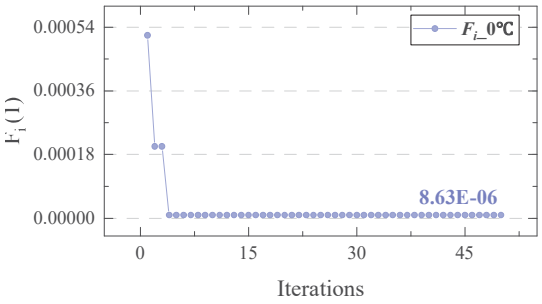
(c-1) Training results



(c-2) Test results



(c-3) Error curve of test



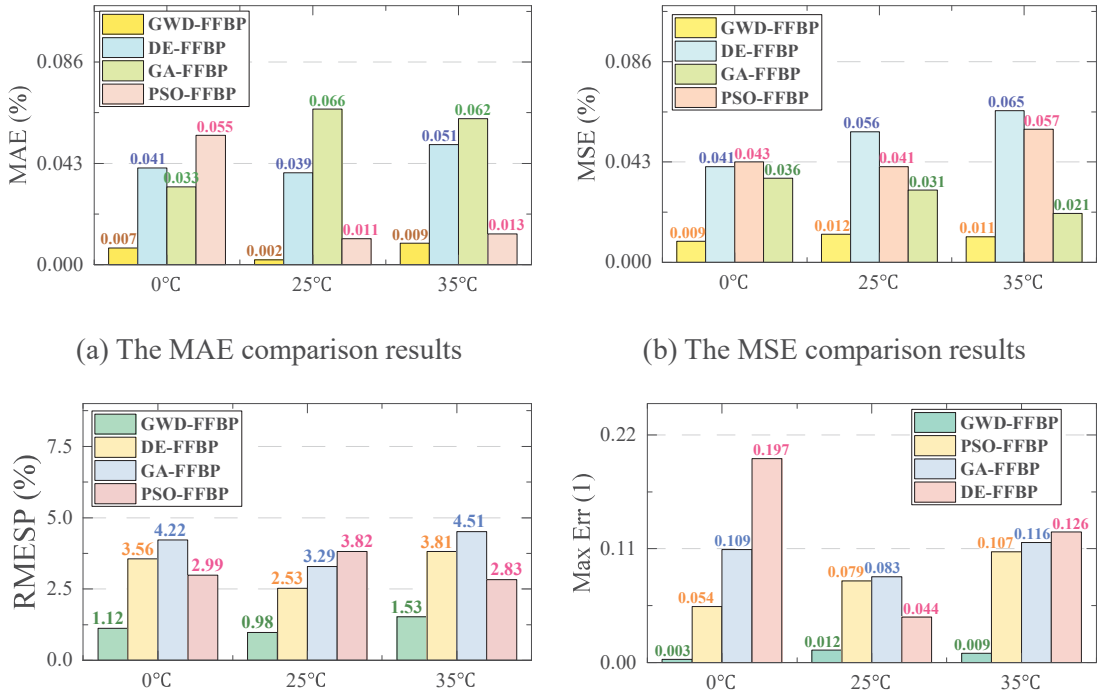
(c-4) Fitness curve

Figure 5 (c). Results curves under BBDST at 0°C

Figure 5. Verification results at variable temperature under BBDST

When the temperature dropped, the dataset—which included over 280,000 training data—was modified for use in experiments at 0°C until the SOC value reached 0.410. As can be observed from Figure 5(c-1) the training results roughly match the SOC_Ref curve and do not differ considerably between networks. Nonetheless, it is evident from the result plot Figure 5(c-2) for test data above 140,000 that the result curves differ significantly, and the ratio of training data and test data is 2:1. While GWD-FFBP is still close to the SOC_Ref curve and corresponds with it, indicating a high accuracy, DE-FFBP has the highest inaccuracy among them, which is shown in the furthest distance from the SOC_Ref. The erroneous results in Figure 5(c-3) also show this. As can be seen from the figure, the maximum error of DE-FFBP reaches 0.197, which is nearly 66 times higher than the 0.003 of GWD-FFBP. The second PSO-FFBP has a smaller error of 0.054, which is 18 times that of the GWD-FFBP. In this

experimental setting, although the error curve of GWD-FFBP is a curve that coincides with a straight line of 0, it has obvious regular fluctuations. And from Figure 5(c-3), the novel GWD-FFBP has a maximum error of 0.003, which is more accurate than the errors of 0.197, 0.109 and 0.054 for the DE, GA and PSO algorithms. The F_i is shown in Figure 5(c-4), which reached at 8.63E-6. Further error comparison of different algorithms based on FFBPNN are recorded in the form of the chart column in the Figure 6.



(a) The MAE comparison results

(b) The MSE comparison results

(c) The RMSEP comparison results

(d) The Max Err comparison results

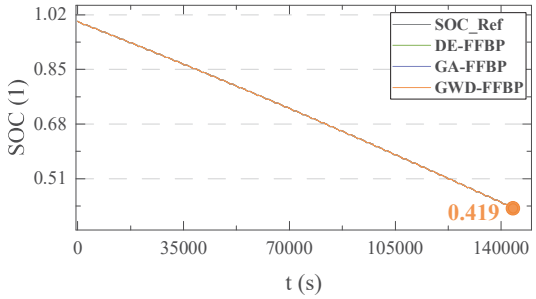
Figure 6. Error comparison results of MAE, MSE, RMSEP and Max Err under BBDST

In Figure 6, the MAE, MSE, RMESP, and the maximum error of the novel GWD-FFBP are all minimal compared with the GA, DE and the PSO algorithms, which indicates that the novel GWD-FFBP overcomes the shortcomings of the traditional algorithms and provides high precision and stability over large data volumes and at different temperatures.

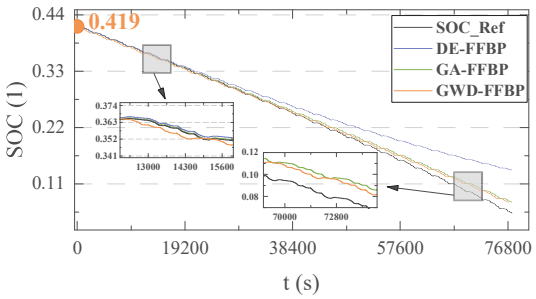
3.4 Analysis of DST working conditions

The performance verification of the novel GWD-FFBP is completed at various

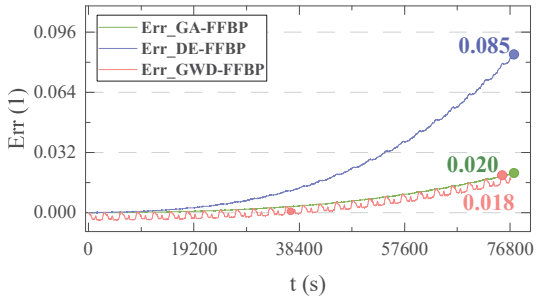
temperatures includes 15°C, 25°C and 35°C under the DST working conditions, comparing with the neural networks of the traditional GA-FFBP and DE-FFBP. The details are shown in Figure 7.



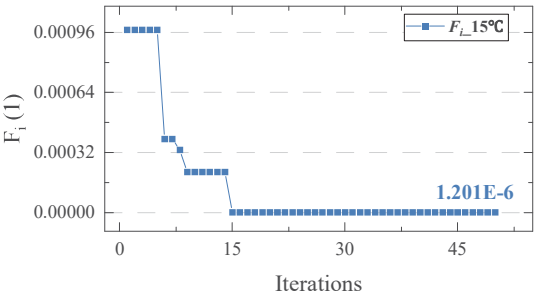
(a-1) Training results



(a-2) Test results



(a-3) Error curve of test

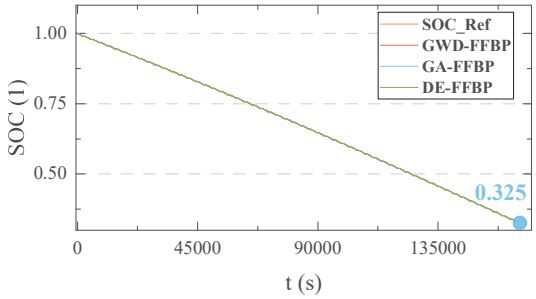


(a-4) Fitness curve

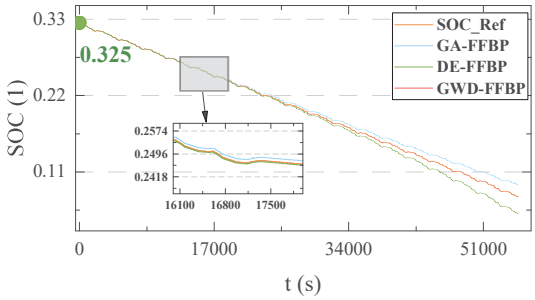
Figure 7 (a). Results curves under DST at 15°C

Overall data volume exceeds 210,000, with training and testing data ratio closed to 2:1. In Figure 7(a-1), the training for the novel GWD-FFBP, DE-FFBP, and the traditional GA-FFBP, and the results all showed high accuracy and stability. In a test environment close to room temperature 15°C, the training results of the DE-FFBP are not significantly different from other algorithms under DST, but in a test of 1/3, the performance is poor, showing a greater distance from the SOC_Ref curve. From the error curve in Figure 7(a-3) it can be further analyzed that the three BP-based optimization algorithms have predictions larger than SOC_Ref, with the curve above SOC_Ref, where the DE algorithm has the greatest error, reaching 0.085, which is 4.7 times of that of the GWD. The GA and GWD have similar errors,

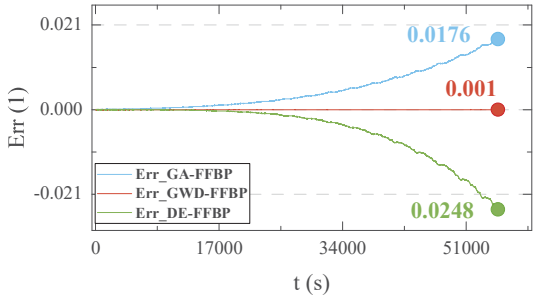
divided into 0.020 and 0.018. The error curve of the GWD-FFBP is also accompanied by a clearly regular gear, while the other two algorithms are smoother. The F_i is shown in Figure 7 (a-4), which reached at 1.201E-6. Apart from this, the verification at 25°C under DST is completed in Figure 7 (b).



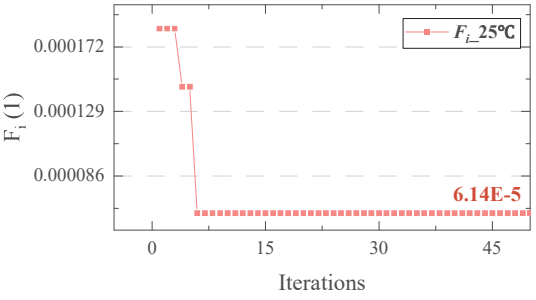
(b-1) Training results



(b-2) Test results



(b-3) Error curve of test

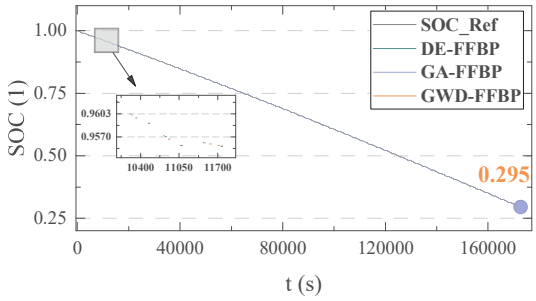


(b-4) Fitness curve

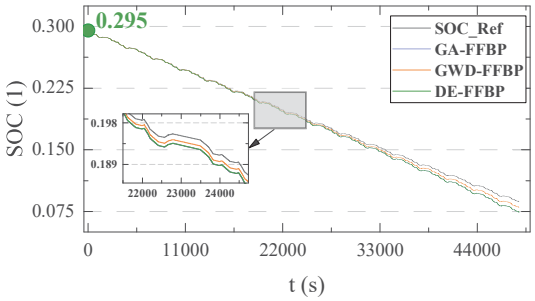
Figure 7 (b). Results curves under DST at 25°C

Over 160,000 data were used as training terms for the networks, which accounts for the total 72%, and the training results for the novel GWD-FFBP, GA, and the DE largely overlapped with the reference SOC_Ref in the Figure 7(b-1). The training results of the three optimization algorithms were smoother in the DST test environment at room temperature of 25°C, but the gap was more noticeable in 28% test of all data. From the test curve of Figure 7 (b-2), it can be further analyzed that the test result curves of GA and DE are located on the upper and lower sides of the SOC_Ref, respectively, whereas the GWD-FFBP basically overlaps with it. In Figure 7(b-3), the maximum error of GA at 0.0176, which is 17times than

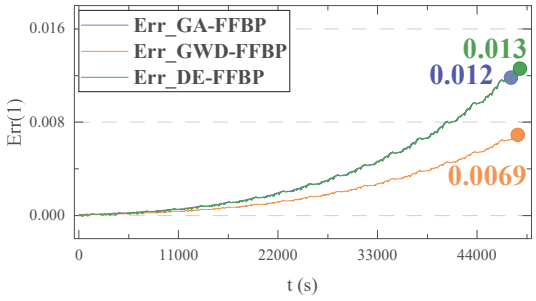
that of the GWD, showing an upward trend, while the DE algorithm's maximum error reaches 0.0248, which is 24 times than that of the improved method with a marked downward trend and more noticeable fluctuations. GWD-FFBP's error curve is basically a straight line that overlaps with a 0 line, with a maximum error of 0.001. The F_i is shown in Figure 7(b-4), which reached at 6.14E-6. Apart from this, the verification at 35°C under DST is completed in Figure 7 (c).



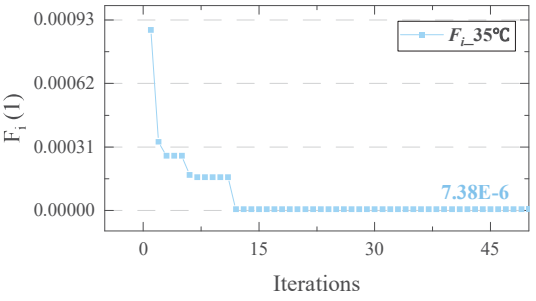
(c-1) Training results



(c-2) Test results



(c-3) Error curve of test



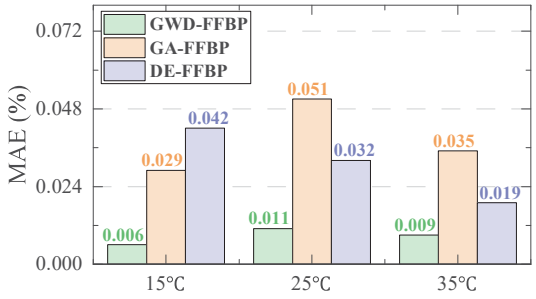
(c-4) Fitness curve

Figure 7 (c). Results curves under DST at 35°C

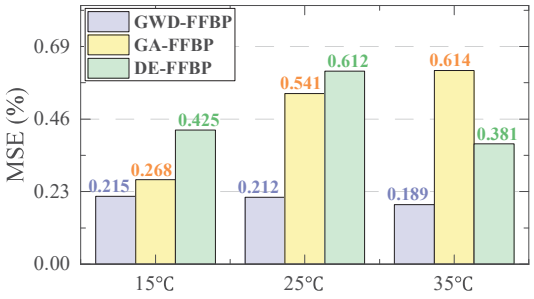
Figure 7. Verification results at variable temperature under DST

During the DST operation, the data volume at 35°C exceeds 23,000, of which the training and test data ratio is 18:5. From Figure 7(c-1), you can see a small difference between the algorithms. In Figure 7(c-2), the DE algorithm has the greatest deviation. In Figure 7(c-3) the maximum error of DE reached 0.013 and GA's maximum error was 0.012, respectively 1.88 and 1.73 times GWD-FFBP max error of 0.0069, and the error curve of GWD had no

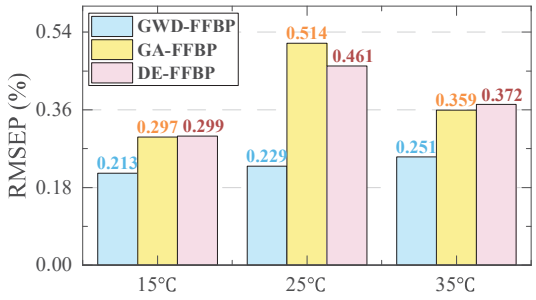
noticeable fluctuations and peak values, and was a flat ascending curve. The stability and accuracy of the GWD-FFBP remain at high temperatures. The F_i reached at 7.38E-6 in Figure 7(c-4). Further error comparison of different algorithms based on FFBPNN are recorded in the form of the chart column in the Figure 8.



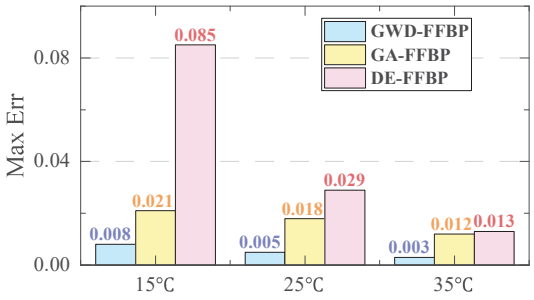
(a) The MAE comparison results



(b) The MSE comparison results



(c) The RMSEP comparison results



(d) The Max Err comparison results

Figure 8. Error comparison results of MAE, MSE, RMSEP and Max Err under DST

From Figure 8 (a), the MAE of the novel GWD-FFBP is 0.029%, 0.011% and 0.009% at 15°C, 25°C, and 35°C, respectively, which performances a better accuracy compared with GA-FFBP and DE-FFBP. Meanwhile, the value of the MSE and the RMSEP of the novel GWD-FFBP are also lower than the GA-FFBP and the DE-FFBP at variable temperatures in Figure 8 (b) and (c). Furthermore, the maximum error of the novel GWD-FFBP is recorded as 0.008,0.005, and 0.003 at 15°C, 25°C and 35°C in Figure 8 (d), which are all lower than other neural networks include the GA-FFBP and the DE-FFBP. The test times for different temperatures at different operating conditions are collected in Table 3.

Table 3. The time consumption summary

T(s)	BBDST			DST		
	35℃	25℃	0℃	15℃	25℃	35℃
GWD-FFBP	1017	986	1212	596	432	284
GA-FFBP	1023	946	1243	526	482	293
DE-FFBP	1215	983	1198	684	491	299
PSO-FFBP	1301	991	1610	549	512	303

It is evident from Table 3 that the test temperature, test circumstances, and data volume all affect test time. At 35°C, the performance of various algorithms changes under the two circumstances. While the DST condition has more than 180,000 and 50,000 respectively in volume of training and testing data, the test times are all within 6 minutes, with PSO-FFBP having the longest test time of 303s. The BBDST condition has more than 300,000 training data, while the training data is more than 200,000, and the time is more than 16 minutes for all of them. Among them, the test time is the shortest for GWD-FFBP and GA-FFBP, which are 1017s and 1023s, respectively. Similarly, under BBDST conditions, the shortest test time for GA-FFBP at 25°C is 946s. GWD-FFBP and PAO-FFBP have test times that are closer together, at 986s and 991s. The GWD-FFBP and PSO-FFBP test timings under DST conditions are 432s and 512s, respectively, for the shortest and longest test times. Even though GWD-FFBP's test time is not the quickest when compared to the other algorithms, it performs better in the shorter amount of time at 0°C and 15°C.

4. Conclusion

In this research, to achieve high-precision SOC estimation of lithium-ion batteries at variable temperatures under complex working conditions, a novel genetic weight-directed feed forward backpropagation neural network is established. To reduce uncertainty in

experience values of the parameter, the hidden layer is determined through traversal method. Specially, an adaptive genetic method that makes the p_c and p_m change and self-correct with the degree of adaptation *F* is proposed to improve the stability and accuracy of the three-layer GWD-FFBPNN at variable temperatures. Then, the inertial volume $\Delta w_{kj}^{2}(\mu)$ and the momentum volume $\Delta w_{ji}^{1}(\delta)$ are introduced to help reduce the convergence time and improve the matching of the system with the increase in data volume. Finally, a further performance comparison of the novel GWD-FFBP, DE-FFBP, GA-FFBP and the PSO-FFBP is completed and the maximum error of the novel GWD-FFBP reached 0.9%, 1.2% and 0.3% under BBDST at 35°C, 25°C and 0°C over 500000 data, similarly, it reached 0.18%, 0.1% and 0.69% under DST at 15°C, 25°C and 35°C over 200000 data. The inability to prevent the impact of the battery's temperature on the forecast findings and the neglect to take into account additional real-world energy storage scenarios and battery types are two of the work's drawbacks. As the data increased, no consideration was given to whether different test instruments affected on the predicted results, while the difference in temperature between on-board batteries and those in energy storage power stations requires further comparison and study. Future work can focus on the following parts:

(1) Research on SOC estimation of series-parallel battery systems.

(2) Research on SOC estimation to meet higher real-time requirements.

(3) Research on SOC estimation under more types of batteries.

Acknowledgment

The work is supported by the National Natural Science Foundation of China (No. 62173281, 52377217, U23A20651), Sichuan Science and Technology Program (No.

23ZDYF0734, 23NSFSC1436), Dazhou City School Cooperation Project (No. DZXQHZ006), Technopole Talent Summit Project (No. KJCRCFH08). Thanks to the sponsors. Carlos Fernandez also expresses his profound gratitude to Robert Gordon University for its support.

References

[1] Wang, Y., et al., *A comprehensive review of battery modeling and state estimation approaches for advanced battery management systems.* Renewable & Sustainable Energy Reviews, 2020. **131**: p. 101-114.

[2] Yang, X., et al., *A novel fuzzy adaptive cubature Kalman filtering method for the state of charge and state of energy co-estimation of lithium-ion batteries.* Electrochimica Acta, 2022. **415**: p. 181-194.

[3] Xin, Z., et al., *A method for compensating random errors in MEMS gyroscopes based on interval empirical mode decomposition and ARMA.* Measurement Science and Technology, 2024. **35**(1) : p. 2221-2229.

[4] Lee, S.T. and J.H. Lee, *Neuromorphic Computing Using Random Synaptic Feedback Weights for Error Backpropagation in NAND Flash Memory-Based Synaptic Devices.* Ieee Transactions on Electron Devices, 2023. **70**(3): p. 1019-1024.

[5] Yang, Z.-F., et al., *GABP transcription factor is required for myeloid differentiation, in part, through its control of Gfi-1 expression.* Blood, 2011. **118**(8): p. 2243-2253.

[6] Huichao, Z., et al., *A Fast Signal-Processing Method for Electromagnetic Ultrasonic Thickness Measurement of Pipelines Based on UKF and SMO.* Energies, 2022. **15**(18) : p. 881-894.

[7] Jin, S., et al., *A novel robust back propagation neural networkdual extended Kalman filter model for state-ofcharge and state-of-health co-estimation of lithiumion batteries.* 2023: p. 1-5.

[8] Hai, N., et al., *An improved random drift particle swarm optimization-feed forward backpropagation neural network for high-precision state-of-charge estimation of lithium-ion batteries.* Journal of Energy Storage, 2023: p. 1101-1114.

[9] Yangshuo, L., et al., *Diesel engine small-sample transfer learning fault diagnosis algorithm based on STFT time–frequency image and hyperparameter autonomous optimization deep convolutional network improved by PSO–GWO–BPNN surrogate model.* Open Physics, 2022. **20**(1) : p. 661-674.

[10] Yuqin, S., et al., *Bayesian analysis of multiple break-points threshold ARMA model with exogenous inputs.* Communications in Statistics - Theory and Methods, 2023. **52**(24) : p. 291-314.

[11] Sangeetha, E.P., et al., *Validation of EKF based SoC estimation using vehicle dynamic modelling for range prediction.* Electric Power Systems Research, 2024. **226**: p. 221-234.

[12] Shen, J.Q., et al., *Pseudo single lithium-ion conductors enabled by a metal-organic framework with biomimetic lithium-ion chains for lithium metal batteries.* Materials Chemistry Frontiers, 2023. **7**(12): p. 2436-2442.

[13] Wang, S., et al., *A novel collaborative multiscale weighting factor-adaptive Kalman filtering*

method for the time-varying whole-life-cycle state of charge estimation of lithium-ion batteries. International Journal of Energy Research, 2022. **46**(6): p. 7704-7721.

[14] Han, Y.J., et al., *Capacity Consistency Prediction and Process Parameter Optimization of Lithium-Ion Battery based on Neural Network and Particle Swarm Optimization Algorithm.* Advanced Theory and Simulations, 2023: p. 14-24.

[15] Fariña, B., J. Toledo, and L. Acosta, *Sensor Fusion Algorithm Selection for an Autonomous Wheelchair Based on EKF/UKF Comparison.* International Journal of Mechanical Engineering and Robotics Research, 2023. **12**(1) : p. 884-894.

[16] Yu, W., et al., *Novel Composite Separators Based on Heterometallic Metal-Organic Frameworks Improve the Performance of Lithium-Ion Batteries.* Advanced Energy Materials, 2023. **13**(22): p. 8-16.

[17] Bingjie, J., et al., *Hybrid LSTM–BPNN-to-BPNN Model Considering Multi-Source Information for Forecasting Medium- and Long-Term Electricity Peak Load.* Energies, 2022. **15**(20) : p. 665-669.

[18] Zhang, H. *BP Neural Network and Its Improved Algorithm In the Power System Transformer Fault Diagnosis*. in *International Conference on Applied Mechatronics and Android Robotics (ICAMAR 2013)*. 2013. Taipei, TAIWAN.

[19] Chen, X., et al., *A deep learning framework for wind pressure super-resolution reconstruction.* Wind and Structures, 2023. **36**(6): p. 405-421.

[20] Yu, L., et al., *Prediction of Tunnelling Parameters for Underwater Shield Tunnels, Based on the GA-BPNN Method.* Sustainability, 2022. **14**(20) : p. 999-1014.

[21] Huang, W.C., et al., *Railway dangerous goods transportation system risk identification: Comparisons among SVM, PSO-SVM, GA-SVM and GS-SVM.* Applied Soft Computing, 2021. **109**: p. 16-27.

[22] Pan, W., et al., *UKF Estimation Method of Centroid Slip Angle for Vehicle Stability Control.* International Journal of Control, Automation and Systems, 2023. **21**(7) : p. 889-900.

[23] Vera, J.F. and J.M. Angulo, *An MDS-based unifying approach to time series K-means clustering: application in the dynamic time warping framework.* Stochastic Environmental Research and Risk Assessment, 2023: p. 12-21.

[24] Kakani, V., et al., *Implementation of Field-Programmable Gate Array Platform for Object Classification Tasks Using Spike-Based Backpropagated Deep Convolutional Spiking Neural Networks.* Micromachines, 2023. **14**(7): p. 24-36.

[25] Almomani, A., et al., *Age and Gender Classification Using Backpropagation and Bagging Algorithms.* Cmc-Computers Materials & Continua, 2023. **74**(2): p. 3045-3062.

[26] D, S., A. G, and S. T, *Data-driven modeling of hydrodynamic loading using NARX and harmonic probing.* Journal of Physics: Conference Series, 2023. **2626**(1) : p. 442-454.

[27] Harifi, S. and R. Mohamaddoust, *Zigzag mutation: a new mutation operator to improve the genetic algorithm.* Multimedia Tools and Applications, 2023: p. 22-34.

[28] Lai, Y., *Project engineering management evaluation based on GABP neural network and artificial intelligence %J Soft Computing.* 2023. **27**(10) : p. 2006-2016.

[29] Ding, C.Y., et al., *Performance prediction for a fuel cell air compressor based on the combination*

of backpropagation neural network optimized by genetic algorithm (GA-BP) and support vector machine (SVM) algorithms. Thermal Science and Engineering Progress, 2023. **44**: p. 10-20.

[30] Cui, L.G., et al., *BBO-BPNN and AMPSO-BPNN for multiple-criteria inventory classification.* Expert Systems with Applications, 2021. **175**: p. 16-31.

[31] Soudan, B., M. Saad, and Ieee. *An Evolutionary Dynamic Population Size PSO Implementation*. in *3rd International Conference on Information and Communication Technologies*. 2008. Damascus, SYRIA.

[32] Shao, J. and S. Kumar, *Optical backpropagation for fiber-optic communications using optical phase conjugation at the receiver.* Optics Letters, 2012. **37**(15): p. 3012-3014.

[33] Gaoqi, L., et al., *A BP neural network-Ant Lion Optimizer and UKF method for SOC estimation of lithium-ion batteries.* Journal of Physics: Conference Series, 2022. **2369**(1) : p. 1012-1030.

[34] Yang, Y.L., et al., *State of Health Assessment of Lithium-ion Batteries Based on Deep Gaussian Process Regression Considering Heterogeneous Features.* Journal of Energy Storage, 2023. **61**: p. 16-31.

[35] Murugesan, V.P., *Impact of new seed and performance criteria in proposed rough k-means clustering.* Multimedia Tools and Applications, 2023: p. 30-39.

[36] Matsubara, T., Y. Miyatake, and T. Yaguchi, *The Symplectic Adjoint Method: Memory-Efficient Backpropagation of Neural-Network-Based Differential Equations.* Ieee Transactions on Neural Networks and Learning Systems, 2023: p. 13-32.

[37] Jingyi, Z., et al., *Improved particle filter remaining life prediction method for switch mode power supply using variance of case temperature.* Microelectronics Reliability, 2023. **142**: p. 193-212.

[38] Mahdi, A.M., et al., *Augmenting inertial motion capture with SLAM using EKF and SRUKF data fusion algorithms.* Measurement, 2023. **222**: p. 613-626.

[39] Ali, D.S., et al., *Defining the optimal conditions using FFNNs and NARX neural networks for modelling the extraction of Sc from aqueous solution by Cryptand-2.2.1 and Cryptand-2.1.1.* Heliyon, 2023. **9**(11) : p. 83-98.