Received August 1, 2019, accepted August 22, 2019, date of publication August 26, 2019, date of current version September 12, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2937550

A Virtual Sample Generation Method Based on Differential Evolution Algorithm for Overall Trend of Small Sample Data: Used for Lithium-ion Battery Capacity Degradation Data

GUOQING KANG[1,2], LIFENG WU[1,2], (Member, IEEE), YONG GUAN[1,2], (Member, IEEE), AND ZHEN PENG[3]

[1]College of Information Engineering, Capital Normal University, Beijing 100048, China
[2]Beijing Key Laboratory of Electronic System Reliability Technology, Capital Normal University, Beijing 100048, China
[3]Beijing Institute of Petrochemical Technology, Beijing 102617, China

Corresponding author: Lifeng Wu (wulifeng@ cnu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61873175 and Grant 71601022, in part by the Key Project B Class of Beijing Natural Science Fund under Grant KZ201710028028, in part by the Capacity Building for Sci-Tech Innovation—Fundamental Scientific Research Funds under Grant 025185305000-187, in part by the Youth Innovative Research Team of Capital Normal University, and in part by the Beijing Youth Talent Support Program under Grant CIT&TCD201804036.

ABSTRACT Considering the wide application of lithium-ion battery in life, the prediction of the remaining life of lithium-ion battery has become a research hotspot. Studies show, due to the improvement of the technology level of lithium-ion battery, its life is getting longer and longer. Even under the condition of accelerated life test, it is difficult to obtain enough available data for research in a short term. In order to solve the problem of how to accurately predict the residual life with the data-driven method under the condition of small sample size, an overall trend virtual sample generation method based on differential evolution (OT-DEVSG) is proposed. This method uses a differential evolution algorithm with better optimization performance, and improves the original mega-trend-diffusion (MTD) method, the range of virtual samples is effectively constrained and the trend of samples can be estimated more accurately. The method can effectively generate a virtual sample data sequence with time parameters, and adapt the virtual sample to the real-life sample trend, which solves the problem of insufficient degradation data of the lithium-ion batteries. Finally, we validate the effectiveness of the OT-DEVSG method with three existing data sets. The experimental results show that the proposed OT-DEVSG method is effective for solving the problem of long-term life prediction of lithium-ion batteries.

INDEX TERMS Lithium-ion battery, overall trend virtual sample generation method based on differential evolution (OT-DEVSG), small data.

I. INTRODUCTION

Lithium-ion battery is an environmentally-friendly high-energy rechargeable battery. Because of its various advantages, such as high capacity, low self-discharge rate, high safety and long cycle life, it is widely used in areas like electronic communication engineering, transportation and aerospace [1], [2]. However, after a number of charge and discharge cycles, the capacity of the lithium-ion battery will gradually decrease. This degradation of performance will affect the normal use of the equipment, and even cause serious accidents [3]. Therefore, from the perspective of safety, reliability and economy, it is especially important to identify the long-term safe and effective way of operating of lithium-ion batteries for avoiding potential accidents [4]. In recent years, the prediction of the remaining service life of lithium-ion batteries has become a research hotspot [5]–[7].

There are two methods for predicting the remaining life of lithium-ion batteries, model-driven methods and data-driven methods [8]. However, due to the difficulty in detecting

The associate editor coordinating the review of this article and approving it for publication was Jason Gu.

and controlling the physical working mechanism and various chemical reactions inside the battery, the model-based prediction method is relatively complicated and difficult to implement [9]. Data-driven method can analyze the current health status and residual life of lithium-ion batteries based on existing data, avoiding the shortcomings of model-driven method. At present, there are many methods for predicting the remaining life of lithium-ion battery based on the data-driven method. For example, Li *et al.* [18] proposed a method for predicting the remaining life of indirect lithium-ion batteries based on Elman neural network. Cadini *et al.* [11] proposed a particle filter based residual life prediction diagnosis method for lithium-ion batteries. Li *et al.* [8] proposed a hybrid remaining useful life prediction method for lithium-ion batteries based on a mixture of long-short-time memory and Herman neural network. Zhang *et al.* [20] proposed a method for predicting the remaining useful life of lithium-ion batteries using an improved UPF method based on MCMC. Zhao *et al.* [14] proposed a life prediction method for lithium-ion battery based on support vector machine. Wang *et al.* [15] proposed a battery life prediction method based on relevant vector machine under uncertain conditions. However, the results from previous studies are seriously undermined by the lack of large sample of data, which is necessary for the data-driven methods to yield accurate predictions [16]. Fu-Kwun Wang et al. proposed a method for predicting the remaining life of lithium-ion batteries based on SVR and differential evolution algorithms [19]. Moreover, with the improvement of design and product technology level, the life of lithium-ion battery is getting longer and longer. According to the test methods and requirements for the cycle life of lithium-ion batteries and battery packs, it is difficult to obtain sufficient degradation data in a short period of time even under accelerated life test conditions. In this regard, it takes a half year or more to complete a set of cycle life tests, which makes the existing battery data becomes not available. Therefore, it is so difficult to accurately predict battery life with a small amount of battery data, which solving the problem of small samples becomes the key to solving the problem of accurately predicting battery life [16].

Small sample refers to the situation in which the number of samples is small, which is the standard for judging the quality of results in existing studies. In practical applications, the threshold value of small sample problem is usually defined as 30 [11], [17], [18]. Specifically, in fields of medical diagnosis or industrial manufacturing, the problem of small data volume exists due to the lack of prior experience data or the difficulty in obtaining available data [8]. On the other hand, in the fields of machine learning and pattern recognition, expanding sample sizes has also become a research hotspot. In order to solve the small sample problem, some methods for expanding sample size have been proposed, Romero F. A. B. de Morais et al. proposed an undersampling method to solve small sample problems [30]. Der-Chiang Li et al. proposed a method of estimating the sample first and then reconstructing the sample [31]. Soman, Sumit et al. proposed a non-iterative technique to add small samples and samples [32]. Yan-Lin He et al. proposed a nonlinear interpolation virtual sample generation method to enhance sample information [13]. and the virtual sample generation technology is the most advanced and most popular method [20]. According to Niyogi *et al.* [21], who proposed the virtual sample generation method, the attempt for using prior knowledge of a given small training set to create virtual samples to improve recognition performance succeeded. They generated a fresh view of a given 3D object from applying other directions through mathematical transformation, and call the newly generated sample a virtual sample [21]. Inspired by this innovation, many VSG-based methods were then proposed. Yang *et al.* [23] proposed a virtual sample generation method based on Gaussian distribution. Li *et al.* [24] proposed a small sample generation method based on genetic algorithm. Chen *et al.* [16] proposed a virtual sample generation method based on PSO. However, when it comes to analyzing the degradation data of lithium-ion battery with time attribute, these methods are not applicable. They cannot describe the overall trend of data well or solve the problem of inaccurate prediction of the remaining life of lithium-ion battery in the long term.

All in all, in this paper, a method of generating overall trend virtual samples based on differential evolution algorithm is proposed. The main contribution of this paper is proposing a new virtual sample generation method and considering the overall trend of data with time series, improving the existing virtual sample generation method with the DE algorithm that has better performance than previous attempts of other methodologies. The OT-DEVSG method given in this paper can generate a virtual sample sequence effectively, It solves the existing small sample problem very well and improve the predicting accuracy of Back Propagation Neural Networks (BPNN) which achieves the accurate long-term prediction of the remaining life of the lithium-ion battery at any time point. Moreover, the validity of the proposed method is verified by a comparison between it and the PSOVSG method as well as experiments under three different data sets.

II. PROPOSED METHOD

In this paper, a method of generating overall trend virtual samples based on differential evolution algorithm is proposed for small samples, and BPNN is established to test the reliability of virtual samples. This section will introduce the basic principle of differential evolution algorithm, summarize the BP neural network, and explain the proposed method in detail.

A. DIFFERENTIAL EVOLUTION

The Differential Evolution (DE) was proposed by Rainer Storn and Kenneth Price on the basis of evolutionary ideas such as genetic algorithms in 1997, the essence is a multi-objective (continuous variable) evolutionary algorithm (MOEAs) for solving the global optimal solution in multidimensional space. Like other evolutionary algorithms, DE is a stochastic model that simulates the evolution of organisms, and through repeated iterations, individuals who adapt to the environment are preserved [25]. Compared with the genetic algorithm, the differential evolution algorithm generates the initial population and defines the fitness value of each individual in the population randomly. The main process also includes three steps: mutation, intersection and selection. In the differential evolution algorithm, a vector that is generated from two parent vectors that is different from their offspring vector, then the offspring vector and a vector from the older generation that is not one of its' parent vectors generate a new vector, who later compared with one of its' parent vectors for preserving the one with better fitness. The differential evolution algorithm preserves the population-based global search strategy, using real-coded, differential-based simple mutation operations and one-to-one competitive survival strategies, for reducing the complexity of genetic operations. The unique memory of the differential evolution algorithm makes it possible to dynamically track the current search situation to adjust its search strategy. In addition, it has strong global convergence ability and robustness, and do not need to use the characteristic information of the problem that is suitable for solving some use of conventional mathematical programming methods which cannot solve the optimization problems in complex environment. Therefore, it is widely used in data mining, pattern recognition, digital filter designing, artificial neural networks, electromagnetics and other fields. In the first International Competition on Evolutionary Optimization, (ICEO) held in Nagoya, Japan in 1996, the differential evolution algorithm proved to be the fastest evolutionary algorithm. Obviously, the approximation effect of the differential evolution algorithm relative to the genetic algorithm is more significant [26], [27].

The differential evolution algorithm is mainly used to solve the global optimization problem of continuous variables. Its main working steps are basically the same as those of other evolutionary algorithms, including Mutation, Crossover and Selection. The basic idea of the algorithm is to start from a randomly generated initial group, and the offspring vector generated from two individuals randomly selected from the population as the random variation source of the third individual, the offspring vector is weighted and summed with a third individual according to a certain rule to generate a new individual, this operation is called mutation. Then, the mutated individual is mixed with a predetermined target individual to generate a test individual, and this process is called crossover. If the fitness value of the test individual is better than the fitness value of the target individual, the test individual replaces the target individual in the next generation, otherwise the target individual is still preserved, and the operation is called selection. In each evolutionary process, each individual vector is used as the target individual. The algorithm continuously calculates it, retains the individuals with higher fitness level, eliminates the individuals with lower level of fitness, and guides the search process to the global optimal solution.

The differential evolution algorithm mainly includes the following four steps:

Step 1: Initialization

Initialization includes determining the boundaries of the population range, the number of individuals, and the dimensions of the individual population, which are generally initialized by the formula (1).

$$X_i(0)=X_{i,1}(0)+X_{i,2}(0)+\ldots+X_{i,n}(0), \quad i=1,2,3,\ldots M \tag{1}$$

$$X_{i,j}(0) = L_j^{\min} + rand(0,1) * (L_j^{\max} - L_j^{\min}) \tag{2}$$

where, $X_i(0)$ represents the i-th individual of the initial population, M represents the number of individuals produced, n represents the dimension of each individual, and $X_{i,j}(0)$ represents the j-th gene of the i-th individual of the 0th generation, $L_j^{\min}$,$L_j^{\max}$ is the initial range boundary of the population.

Step 2: Mutation

In the g-th iteration, three individuals of $X_{p1}(g)$, $X_{p2}(g)$, $X_{p3}(g)$ are randomly selected from the population as parents. Two of the individuals perform vector difference generation to generate a new vector, and then the new vector sums with the third individual to generate an experimental individual that is called the first intermediate vector, and the expression is as shown in (3)

$$H_i(g) = X_{p1}(g) + F * (X_{p2}(g) - X_{p3}(g)) \tag{3}$$

where $p1 \neq p2 \neq p3 \neq i$, F is the scaling factor, generally between [0, 2], F mainly affects the global locating ability of the algorithm. The smaller the F is, the better the local search ability is. The larger the F is, the more the algorithm can jump out of the local minimum, but the convergence speed will be slower.

Step 3: Crossover

Crossing can increase the diversity of the population. In the g-th iteration, each individual and the intermediate vector generated by the mutation are crossed. Specifically, each allele of each individual is selected according to a certain probability. Select the allele of the first intermediate vector to cross and generate the second intermediate vector, the expression is as shown in (4)

$$v_{i,j} = \begin{cases} h_{i,j}(g), & rand(0,1) \leq P_{cr} \\ x_{i,j}(g), & else \end{cases} \tag{4}$$

where P_{cr} is the rate of crossover.

Step 4: Selection

The selection process is based on the greedy choice strategy, according to the fitness function value, from the second intermediate vector $V_i(g)$ and the original vector $X_i(g)$ of each individual in the g-th iteration, the higher the fitness is selected in the next generation. The selection process will make each individual's $X_i(g+1)$ better in fitness than $X_i(g)$, and finally converge to some best or local best. The mutation and crossover operation will help to jump out of the local optimum to achieve global optimality. The expression of the selection process is as shown in (5)

$$v_{i,j} = \begin{cases} V_i(g), & if\ f(V_i(g)) > f(X_i(g)) \\ X_i(g), & else \end{cases} \quad (5)$$

where $f()$ represents the fitness function.

The maximum number of iterations set in this paper is 200. Before reaching the maximum number of iterations, formula (5) will be used for selection and judgment. When the maximum number of iterations is reached, the optimal value for the 200th iteration is output. The algorithm flow chart shown in Fig. (1).

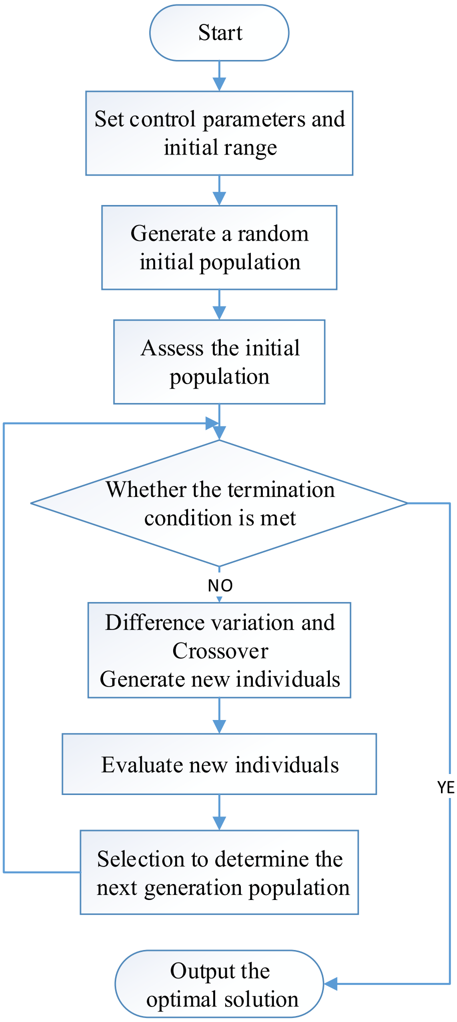

Start
Set control parameters and initial range
Generate a random initial population
Assess the initial population
Whether the termination condition is met
NO
Difference variation and Crossover Generate new individuals
Evaluate new individuals
YES
Selection to determine the next generation population
Output the optimal solution

FIGURE 1. DE algorithm flow chart.

B. BPNN

BPNN is a multi-layer feedforward neural network. The main characteristics of this neural network are forward propagation of signals and back propagation of errors. In forward transfer, the input signal is processed layer by layer from the input layer through the hidden layer until the output layer, and the neuron state value of each layer affects the state of the next layer of neurons. If the output layer does not get the expected output, the error between the prediction and the expectation is backpropagated, and the network weight and threshold are adjusted according to the prediction error, so that the BPNN prediction output is continuously approaching the desired output. The network topology of BPNN is shown in Fig. (2). In the figure, I1, I2, and I3 are input values of BPNN, θ is the offset, H1, H2, H3, H4, and H5 are hidden layer nodes, and Out is the output.

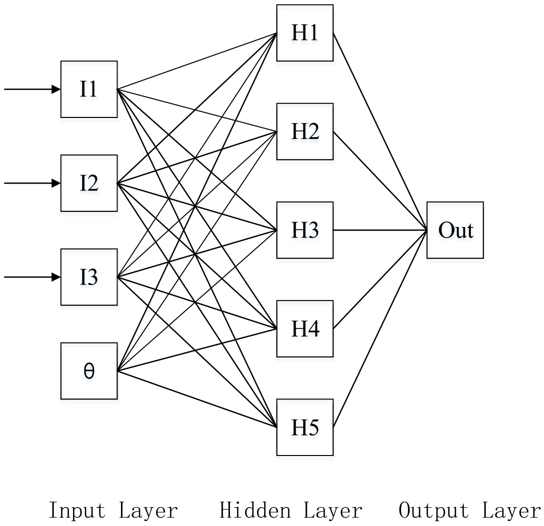


FIGURE 2. BPNN topology diagram.

BPNN must first be trained by large amount of data, so that the network has the ability to associate memory and prediction. BPNN training first carries out the feedforward process, determines the input layer node, the output layer node, the hidden layer node of the network according to the input (X, Y) of the system, and initializes the connection weights of the input layer, the hidden layer and the output layer, and Parameters such as threshold and learning rate are outputs. Then, X is sequentially sent to the input neuron, and the corresponding value obtained by the hidden layer neuron is determined according to the input data and the connection weight. According to the hidden layer neurons and the connection weights, the regression value of the final output is obtained after the activation function. The process of BP is to compare the predicted values obtained from feedforward with the reference value and adjust the connection weight according to the error. Then we use a trained network parameter to predict the test machine data to obtain the target prediction value.

C. PROPOSED METHOD

This paper is to achieve a long-term accurate prediction of the remaining life of lithium-ion batteries by increasing the sample size. Here we propose a method based on differential evolution algorithm for overall trend virtual sample generation (OT-DEVSG). This method can generate a new sample with the same overall trend as the original small sample data, and with the same time attribute within an acceptable range. Next, the proposed OT-DEVSG will be described in detail.

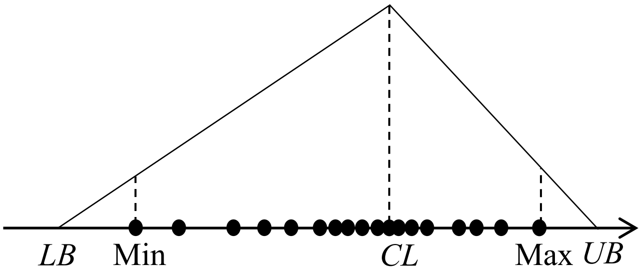


FIGURE 3. **Schematic diagram of trend diffusion.**

For OT-DEVSG proposed method, first the original sample have been divided into training and test sets, then the overall trend of the original training set with small sample data were acquired, and finally calculate the estimated distribution of data points and data sets attribute acceptable range. Li *et al.* [28] proposed the mega-trend-diffusion (MTD) to estimate the trend of the data and roughly determine the acceptable range of the data. However, the MTD technology uses a triangular distribution to describe the overall distribution. It is difficult to describe the overall detailed characteristics of the data with complex distribution, and the data from the center point in the overall trend will have a greater impact on the estimation of the acceptable range. On the basis of MTD, Zhu *et al.* [29] proposed a multi-distribution overall trend diffusion technology (MD-MTD), which solved the some problems mentioned previously to some extent. In this paper, based on the MD-MTD method, the size of the adaptive data set can be determined, for calculating the value of the correction factor, which enable us to have acceptable range of datasets. The schematic diagram of the trend diffusion is shown in Figuer(3), where Min is the minimum value of the small sample data, Max is the maximum value of the small sample data, *LB* is the lower bound of the extended acceptable range, and *UB* is the expanded Accept the upper bound of the range.

During extended range, the original training set to a small sample of $X = \{x_1, x_2, \ldots, x_m\}$ by substantially the MD-MTD method, according to the formula for calculating the acceptable range boundaries *UB* and *LB*.

$$LB = \begin{cases} CL - LS \times \sqrt{-2\frac{\hat{S}_x}{N_L} \times \ln(10^{-20})} & L \leq \min \\ \min & L > \min \end{cases} \tag{6}$$

$$UB = \begin{cases} CL + RS \times \sqrt{-2\frac{\hat{S}_x}{N_U} \times \ln(10^{-20})} & L \geq \max \\ \min & L < \max \end{cases} \tag{7}$$

$$CL = \begin{cases} x_{[(n+1)/2]} & \textit{if } n \textit{ is odd} \\ \frac{1}{2}(x_{[n/2]} + x_{[n/2+1]}) & \textit{if } n \textit{ is odd} \end{cases} \tag{8}$$

$$LS = \frac{N_L}{N_L + N_U + m} \tag{9}$$

$$RS = \frac{NU}{N_L + N_U + m} \tag{10}$$

$$\hat{S}_x = \frac{\sum_{i=1}^{n} (X_i - \bar{X})^2}{n-1} \tag{11}$$

where m is the number of small sample data, CL is the data center, N_L is the number of sample data that with values less than CL, N_U is the number of sample data values that with values greater than CL, LS is the left skewness that describes the asymmetric diffusion characteristics of the data, and RS is the right skewness that describes the asymmetric diffusion characteristics of the data, and $\hat{S_x}$ is the variance of the original sample data. Here, the CL is the calculation method of the MD-MTD method which is modified to overcome the influence of outliers in the center of datasets and to optimally estimate the trend of the data. Due to the existence of outliers, the values of NL and NU are too large, which result in an overestimation of the left and right skewness LS and RS that makes the acceptable range of sample estimation excessively increase. Here, the distribution in the left and right skewness calculation formula increases the correction factor m, which is m=n/10 in this paper.

Fig. (4) is a schematic diagram of a collection of small samples and virtual samples of the training set. The blue square points shown in the figure are a set of original small sample data points, and the area between the two brown curves is a collection of all virtual samples, the area between two black lines is the range of acceptable virtual samples. Obviously, adding dummy samples to the gaps of the original samples reduces the gap between the original small sample data and increases the amount of information in the sample set. But not all generated virtual sample data help to enrich the information obtained from a small sample set. The generated virtual samples can be divided into two types: a suitable virtual sample (green triangle) and an inappropriate virtual sample (brown star), as shown. Appropriate virtual samples will improve the prediction accuracy of the prediction model, while inappropriate virtual samples will have a negative impact on the prediction accuracy of the prediction model.

The OT-DEVSG proposed in this paper can perform virtual sample generation on sample data sequences with time attributes. At each time point, the original small sample is expanded to generate n new virtual sample data points, and the set of the i-th sample data with all the time points is regarded as a new virtual sample data sequence $X_i = \{x_{i1}, x_{i2}, \ldots, x_{im}\}$, m is the number of sets of virtual sample sequence data. The method can constrain the generated virtual samples, constrain the virtual samples to the ideal confidence interval, and adapt the trend of the virtual sample data sequence closer to the trend of the original sample data. This method can be regarded as a multi-dimensional nonlinear overall trend constraint optimization under certain circumstances. The constraint conditions can be described by the following expressions.

$$LB \leq x_{ij} \leq UB, \quad j = 1, 2, \ldots, m \tag{12}$$

$$G(X_i) < 5\%, \quad i = 1, 2, \ldots, n \tag{13}$$

$$G(X_i) = \frac{1}{n} \sum_{j=1}^{n} \left| x_j - x_{ij} \right| \times 100\% \tag{14}$$

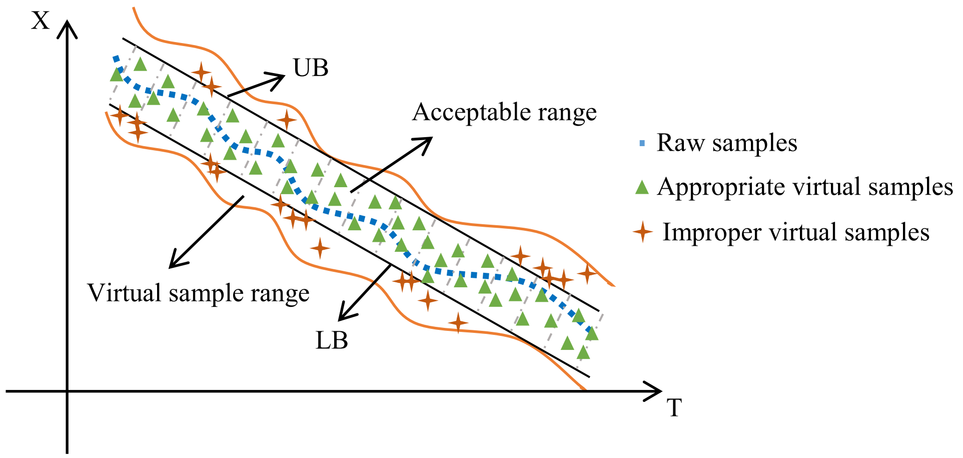


FIGURE 4. Diagram of the relationship between small sample data and virtual sample data.

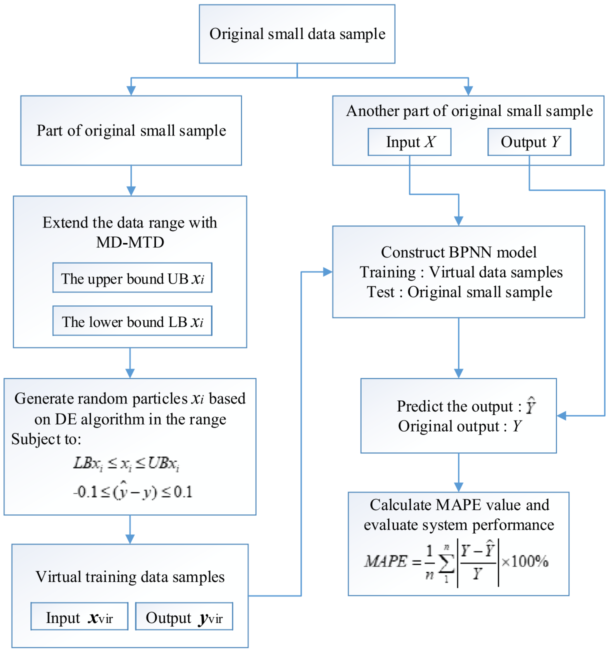

Original small data sample
Part of original small sample
Another part of original small sample
Input X
Output Y
Extend the data range with MD-MTD
The upper bound UB x_i
The lower bound LB x_i
Construct BPNN model
Training : Virtual data samples
Test : Original small sample
Generate random particles x_i based on DE algorithm in the range
Subject to:
$LBx_i \le x_i \le UBx_i$
$-0.1 \le (\hat{y} - y) \le 0.1$
Predict the output : $\hat{Y}$
Original output : Y
Calculate MAPE value and evaluate system performance
$MAPE = \frac{1}{n}\sum_1^n \left|\frac{Y-\hat{Y}}{Y}\right| \times 100\%$
Virtual training data samples
Input x_{vir}
Output y_{vir}

FIGURE 5. OT-DEVSG method flow chart.

where LB, UB represent the upper and lower boundaries of the generated virtual sample data point x_{ij}, and $G(X_i)$ represents the fitness value of the overall trend of the i-th virtual sample data sequence and the trend of the original small sample data.

After obtaining the appropriate virtual samples, combine with the original small samples of the training set for generating a new training set, which is used for training the BPNN with the new training set and the network with the training set with a large number of virtual samples, in order to optimize the prediction performance of the network.

Finally, the original small sample test set data is tested with the trained network, and the predicted results are used to evaluate the performance of the network, thereby verifying the validity of the virtual sample. The flow chart of the method proposed in this paper is shown in Fig. (5).

TABLE 1. The parameter settings for DE.

Notation	Value	Parameter
NP	100	Number of particles
CR	0.7	Crossover probability
G	200	Number of genetic iterations
F	0.6	Differential scaling factor
Fitness	10^{-3}	Expected population fitness value

III. EXPERIMENT AND DISCUSSION

In order to verify the rationality and effectiveness of the OT-DEVSG method, three different common data sets with only a small amount of sample data were selected for experimental verification, including battery data sets held by NASA, CALCE, and Oxford University. The NASA data set includes 4 complete battery degradation data, B5, B6, B7, B18; CALCE battery data has A3, A5, A8, A12, which are 4 complete battery degradation data; The Oxford University battery degradation dataset contains complete battery degradation data for eight SLPB533459H4 lithium-ion battery cells c1-c8 from Kokam COLTD. In order to measure the effectiveness of the method, the root mean square error (RMSE) and the average percentage (MAPE) of the full cycle battery capacity prediction and the absolute error (AE) of the predicted remaining cycle number are used as evaluation criteria. The relevant parameter settings of DE are shown in Table (1).

$$RMSE = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(y_{test}^{(i)} - \hat{y}_{test}^{(i)})^2} \tag{15}$$

$$MAPE = \frac{1}{n_o}\sum_{n=1}^{n_o}\left|\frac{y-\hat{y}}{y}\right| \times 100\% \tag{16}$$

$$AE = \left|RUL_{true} - RUL_{predicted}\right| \tag{17}$$

A. CALCE

This section uses battery test data from the University of Maryland Center for Advanced Life Cycle

Engineering (CALCE), which is the battery capacity degradation data of A3, A5, A8, and A12 lithium-ion batteries tested at room temperature [14]. The initial capacity of this group of lithium-ion batteries is 0.9Ah, and a constant current discharge of 0.45A is used. After a plurality of charge and discharge cycles, when the rated capacity of the battery is reduced by 30% (from 0.9Ah to 0.63Ah), the battery is considered to have reached the end of life standard. Fig. (6) shows the capacity degradation data curve of four batteries.

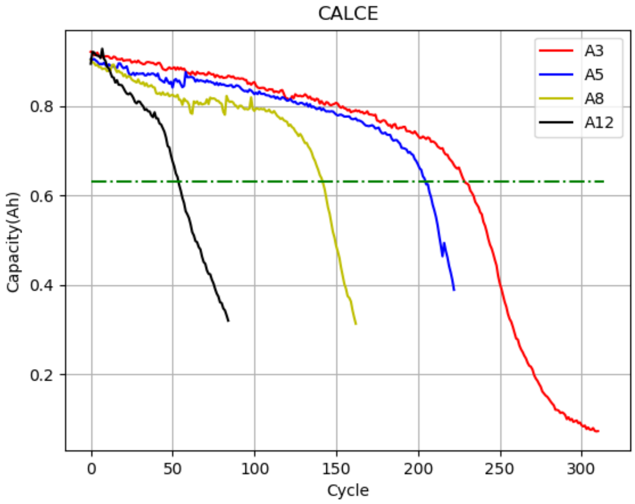


FIGURE 6. **CALCE raw data.**

(1) Only the A3 raw sample data is used as the training set for BPNN loop training, and the remaining A5, A8, and A12 groups are used as the test. The predicted result is shown in Fig. (7).

(2) The A3 original sample data was used as the training set to expand and generate virtual samples, and the remaining A5, A8, and A12 groups were tested. Firstly, the training set is spread according to the formula, and the upper and lower bounds of the extended data are obtained. The range is taken as the range of the random initial population of the differential evolution algorithm, and the random population is initialized, and then the initialized particles are processed according to the formula. Perform mutation, crossover, and selection processes to obtain an optimal new individual value, and perform each of the above data points of the A3 original sample to obtain a new complete sample. The above experiment was performed 100 times to form 100 sets of complete virtual sample data sequences. The 100 sets of virtual samples are used as the training sets of the BPNN network, and the remaining 3 sets of original sample data are used as the test sets for predicting purpose. In this experiment, a sliding window with a capacity of 10 data points and a step size of 10 data points was used for continuous test and prediction. The 10 data in the window were test data input into the trained BPNN, and the output would obtain the predicted battery capacity for the next 10 times, testing processes are repeated continuously until the end of battery life. The training set data, the generated virtual sample data shown in Fig. (8), the test set data and the corresponding predicted values are shown in Fig.(9).

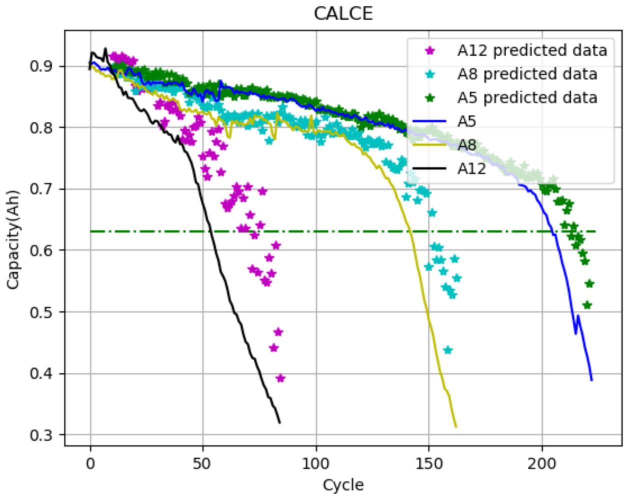


FIGURE 7. **Prediction results before CALCE data is added to a small sample.**

B. NASA

This section uses NASA lithium-ion battery data for experimental verification. NASA battery data is held by the NASA PCoE research center test data of lithium-ion battery [33], the battery has the rated capacity of 2 Ah commercial 18650 lithium-ion batteries, this group of data with four groups of lithium-ion battery (5, 6, 7 and 18) at room temperature in 24 degrees Celsius 3 different job characteristics (charging, discharge, and impedance). Charge with 1.5A current in constant current mode until the battery voltage reaches 4.2V, then continue to charge in the constant voltage mode, and the charging ends when the charging current drops to 20 mA. The discharge is a constant current discharge at a current of 2 A. When the voltages of the No. 5, No. 6, No. 7, and No. 18 batteries are reduced to 2.7 V, 2.5 V, 2.2 V, and 2.5 V, respectively, the discharge is stopped. The above-mentioned charge and discharge cycle of the battery is performed several times to deteriorate the battery. When the rated capacity of the battery is reduced by 30% (from 2Ah to 1.4Ah), the battery is considered to have reached the end of life standard, and the experimental data collection is stopped. In this experiment, the experimental data was verified by three sets of data of B5, B6 and B7. Fig. (10) shows the capacity degradation data curve of three batteries.

(1) Only B5, B6 raw sample data is used as the training set for BPNN training, and B7 is used as the test. The prediction result is shown in Fig. (11).

(2) In this experiment, we will extract the original data of B5 and B6 batteries in the NASA data set for training purpose, and adopt the B7 battery data as the test set. The B5 and B6 sample data are generated in accordance with the steps in Section 3.1, respectively. The B5 and B6 data sets

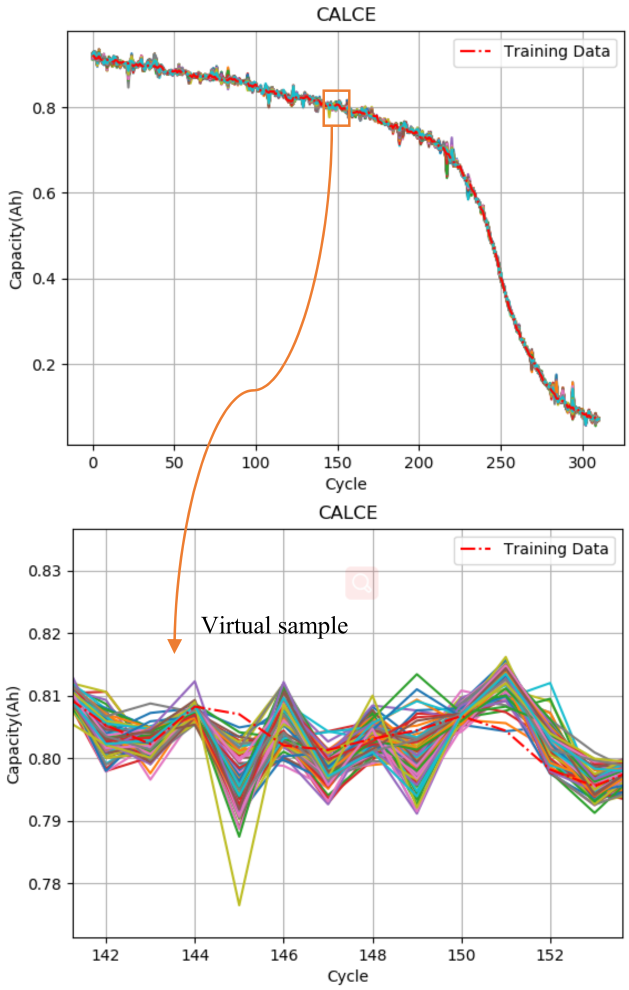


FIGURE 8. A3 raw sample data and virtual sample data sequence in CALCE.

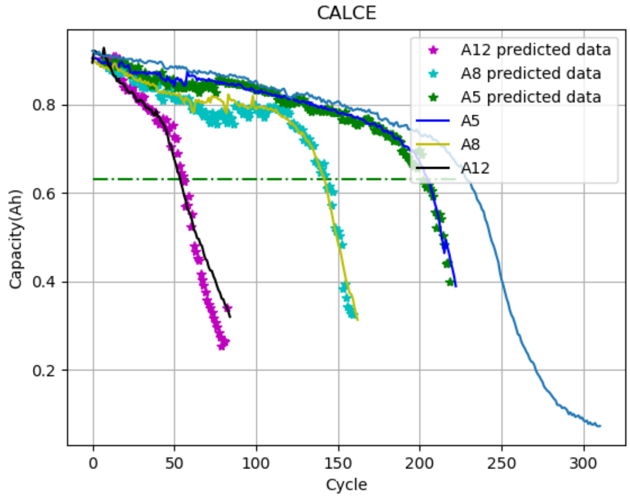


FIGURE 9. Prediction result graph after adding virtual samples to the CALCE data set.

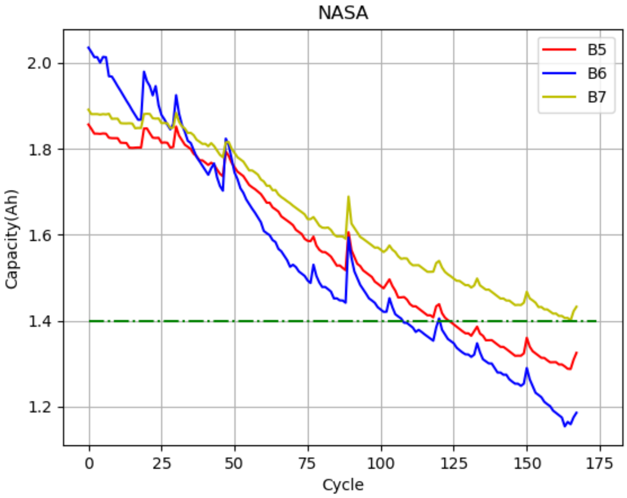


FIGURE 10. NASA raw data.

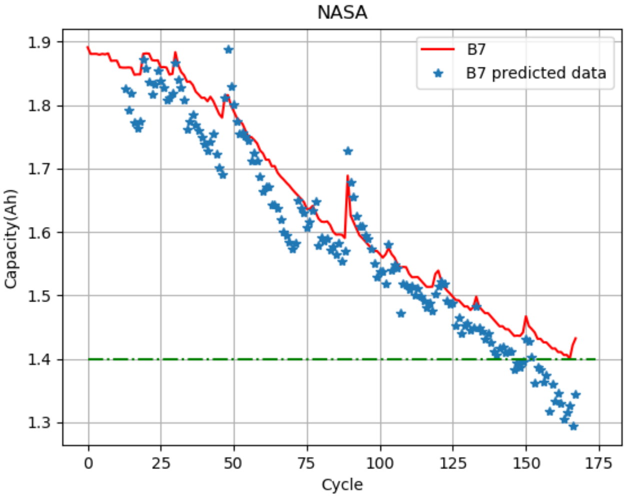


FIGURE 11. Prediction results before NASA data is added to the virtual sample.

respectively form corresponding 100 sets of complete virtual sample data sequences, and the 200 sets of virtual samples are used as training sets of the BPNN network. In this experiment, a sliding window with a capacity of 10 data points and a step size of 10 data points were used for continuous testing and predicting. The 10 data in the window were test data input into the trained BPNN, and the output would obtain the predicted battery capacity for the next 10 times, testing is continued until the end of battery life. The training set data, the generated virtual sample data shown in Fig. (12), the test set data and the corresponding predicted values are shown in Fig. (13).

C. OXFORD

This section uses the Oxford Battery Degradation Dataset, the battery life cycle data from the Oxford Battery Degradation Data Set for verifying experimental results. This data set contains 8 small lithium-ion batteries C1-C8, at 40° temperature test charge 1 - C, 1 C discharge, pseudo OCV charge (OCVch), pseudo OCV discharge (OCVdc) time, voltage,

TABLE 2. Prediction error comparison table before and after adding virtual samples.

Data set		MSE			MAPE			RUL AE(times)	
		Before	After	Improvement %	Before	After	Improvement %	Before	After
CALCE	A5	0.0504	0.0154	69.4	2.0569	1.2598	38.8	11	1
	A8	0.3101	0.0499	85.5	4.2904	1.9820	53.8	17	2
	A12	1.1334	0.2171	80.8	15.4117	6.5056	57.8	20	1
NASA	B7	0.2794	0.0449	83.9	3.5181	1.1005	68.7	31	2
Oxford	C4	0.01951	0.00195	90.0	2.0907	0.5841	72.1	15	2

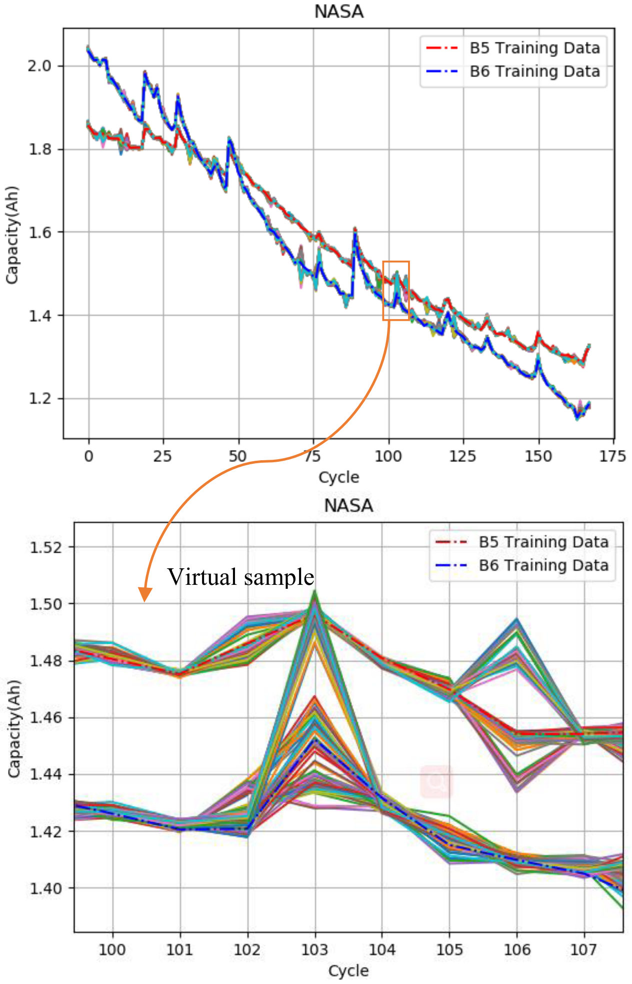


FIGURE 12. B5, B6 raw sample data and virtual sample data sequence in NASA.

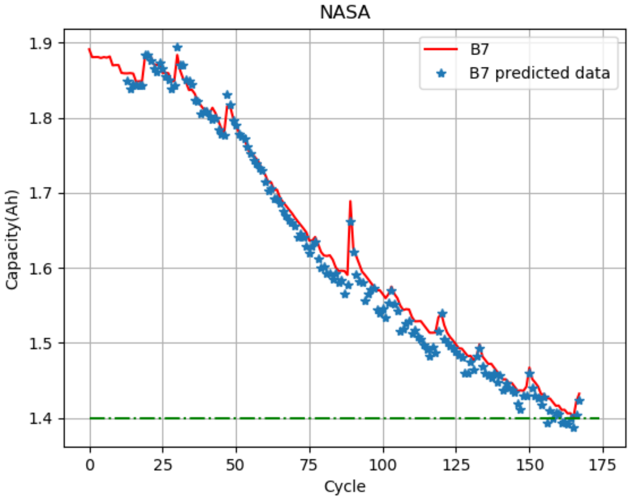


FIGURE 13. Prediction results after NASA data is added to the virtual sample.

charge and temperature degradation data of the measured values. This section uses battery charge data as a battery degradation reference to make battery life predictions. The rated capacity of the battery is 0.74Ah, and a constant current discharge of 0.74A is used. After multiple charge and discharge cycles, the battery is considered to be ineffective when the battery capacity is reduced by 30% (from 0.74 Ah to 0.52Ah). This section uses C1, C2, C3, and C4 for experimental verifications. Fig. (14) shows the capacity degradation data curve of four batteries.

(1) The C1, C2, C3 raw sample data is used as the training set for BPNN training, and C4 is used as the test. The predicted result is shown in Fig. (15).

(2) In order to verify the impact of the number of training set samples on the prediction accuracy, we will extract the original data of C1, C2, C3 batteries as training sets, and the C4 battery data as the test set. The training sets C1, C2, and C3 are respectively subjected to virtual sample generation according to the steps shown in Section 3.1. The C1, C2, and C3 data sets respectively form corresponding 100 sets of complete virtual sample data sequences, and the 300 sets of virtual samples are used as training sets of the BPNN network. In this experiment, a sliding window with a capacity of 15 data points and a step size of 15 data points was used for continuous testing and predicting. The 15 data in the window are test data that are inputted into the trained BPNN, and in the nest 15 times, the output of the predicted battery capacity would be obtained, continue to test until the end of battery life. The training set data, the generated virtual sample data

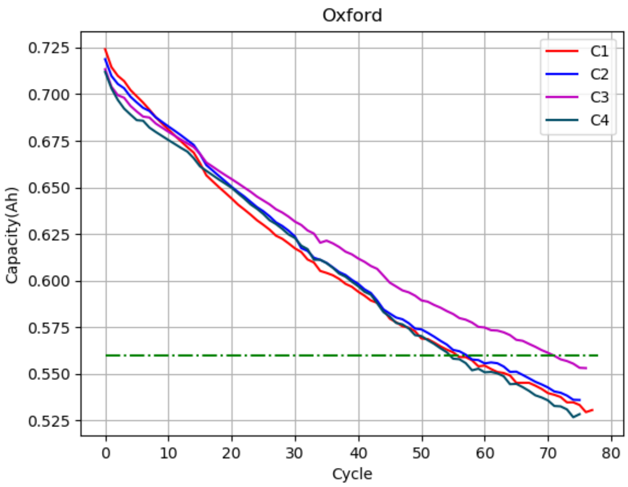


FIGURE 14. Oxford raw data.

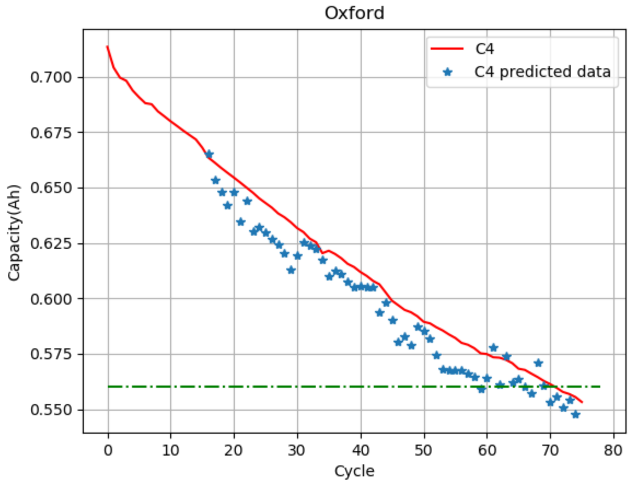


FIGURE 15. Prediction results before Oxford C4 data is added to the virtual sample.

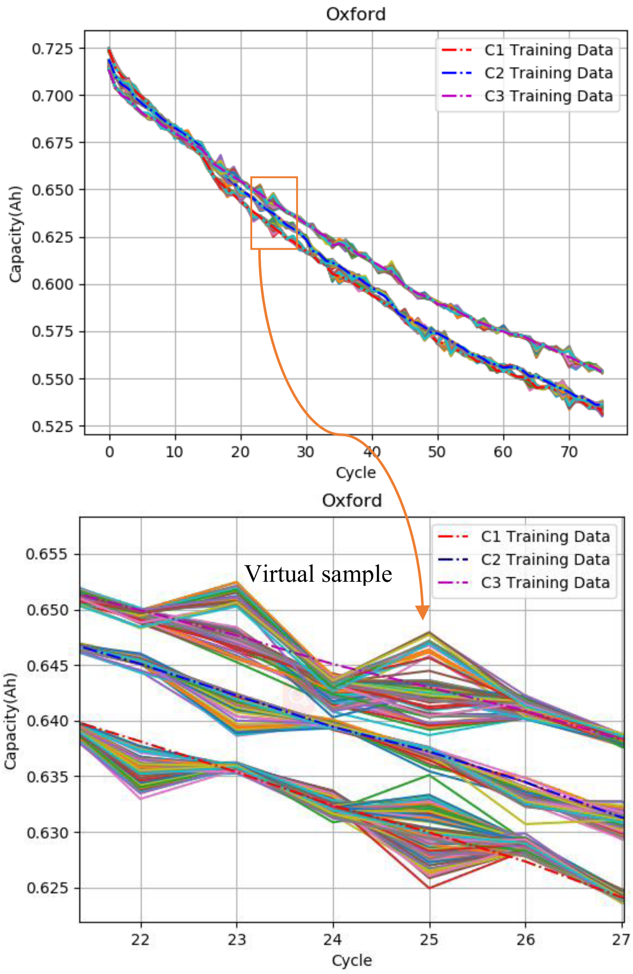


FIGURE 16. C1, C2, C3 raw sample data and virtual sample data sequence in Oxford.

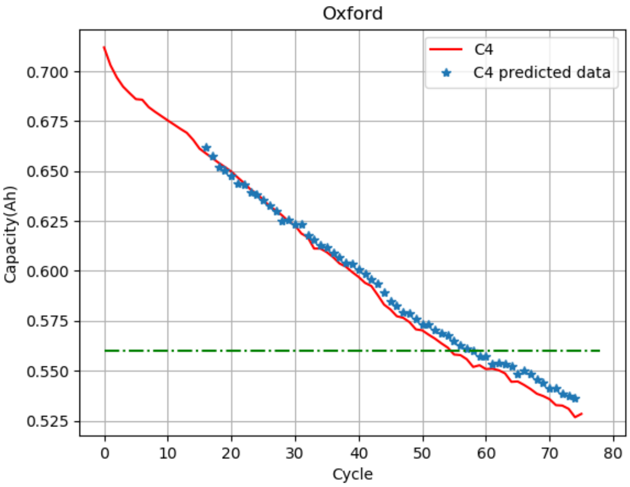


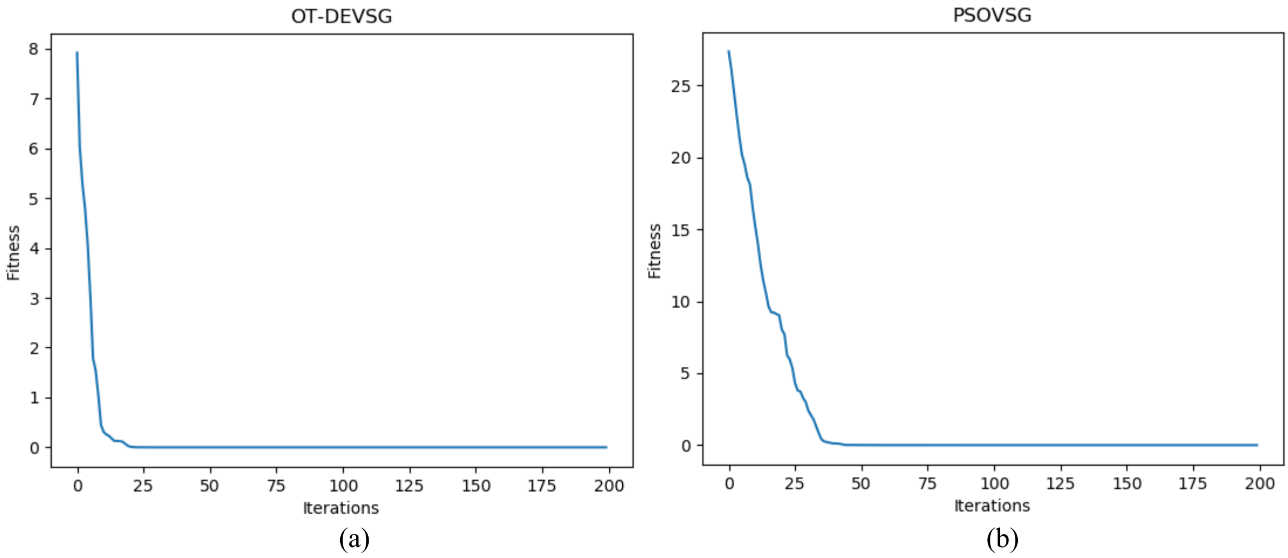
FIGURE 17. Prediction results after Oxford C4 data is added to the virtual sample.

shown in Fig. (16), the test set data and the corresponding predicted values are shown in Fig.(17).

D. DISCUSSION

In order to reflect the faster convergence speed and higher learning efficiency of the OT-DEVSG method proposed in this paper, here we will compare the proposed method with the PSOVSG method on the NASA dataset. Convergence curve Fitness value are shown in Fig. 18. It can be seen from the figure that the fitness value of the OT-DEVSG method proposed in this paper is reduced from the initial training of 7.92 after 23 iterations to about 0.001. However, the initial value of the fitness of the PSOVSG method is 27.34, which is reduced to about 0.001 after 50 iterations. This shows that OT-DEVSG has higher convergence speed and learning efficiency.

In order to prove that BPNN has better learning efficiency with sufficient training data, the convergence performance of BPNN is verified under NASA data set, and the loss function is shown in Fig. 19. The loss function value is reduced from the initial 7.513 through 15 iterations to 0.0001. It can be



(a) (b)

FIGURE 18. **Fitness value decline curve. (a)OT-DEVSG (iterations=200). (b)PSOVSG (iterations=200).**

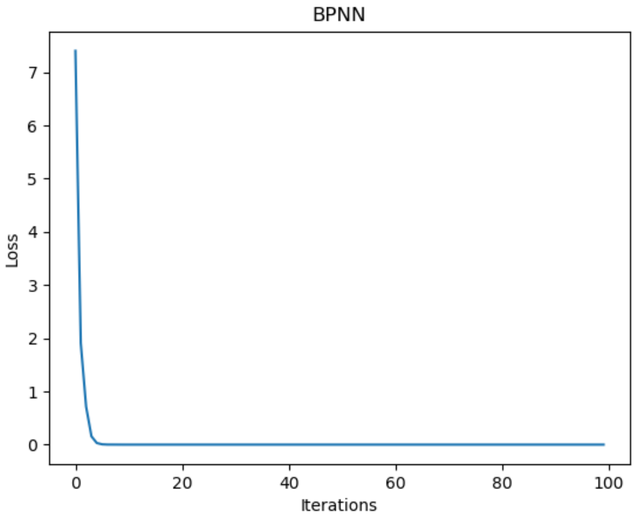


FIGURE 19. **Loss function decline curve.**

seen from the male figure that the training of the BPNN model is effective when there is a sufficiently large training set.

Table (2) gives the comparison of the prediction error data after adding the virtual sample and the prediction error data with the original small sample data. It can be seen that, compared with the prediction error value without virtual samples, after adding a large number of virtual samples, the value of MSE, MAPE and RUL AE have been significantly improved, which enables accurate prediction of the future long-term residual life. The more virtual samples are added, the smaller the MAPE value of the prediction result will be. It can be seen from the diagrams (20), (21) and (22) that the prediction accuracy has been greatly improved after adding a large number of virtual samples, and the OT-DEVSG effectively improves the prediction performance of the small sample set.

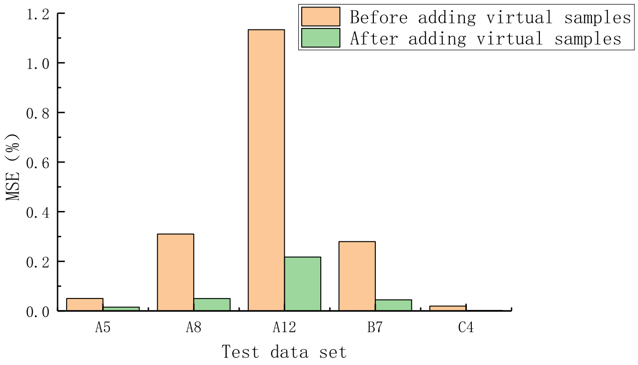


FIGURE 20. **Comparison of predicted MSE before and after adding virtual samples.**

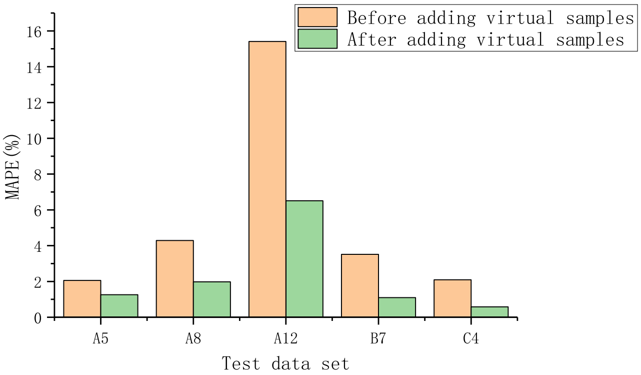


FIGURE 21. **Comparison of predicted MAPE before and after adding virtual samples.**

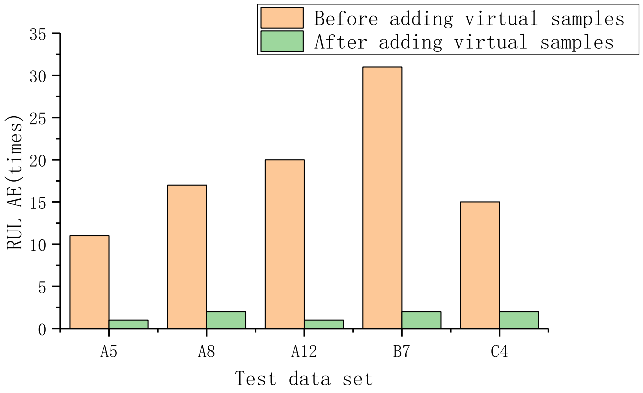


FIGURE 22. Comparison of predicted RUL AE before and after adding virtual samples.

IV. CONCLUSION

Due to the insufficient data of existing training samples, the performance of machine learning algorithm model is poor, and it is difficult to obtain robust prediction results and enhance prediction accuracy. A new OT-DEVSG method is proposed to solve the problem of poor robustness of prediction model caused by insufficient training samples of data-driven method. This method generates a large number of virtual samples within an acceptable range, fills the information gap between the original small sample data by adding virtual samples, increases the amount of training sample data, and improves the accuracy of the prediction model. In addition, the method adaptively expands the acceptable range of the overall trend, makes the deviation between the virtual sample and the actual sample insignificant, which improves the applicability of the virtual sample. In this paper, three different small sample datasets are tested. By adding virtual samples generated by this method, the prediction accuracy is improved, and the effectiveness of this method is supported. In conclusion, the OT-DEVSG method is reasonable and effective for solving small sample prediction problems.

REFERENCES

[1] W. Chen, J. Liang, Z. Yang, and G. Li, "A review of lithium-ion battery for electric vehicle applications and beyond," *Energy Procedia*, vol. 158, pp. 4363–4368, Feb. 2019.

[2] W. He, N. Williard, M. Osterman, and M. Pecht, "Prognostics of lithium-ion batteries based on Dempster–Shafer theory and the Bayesian Monte Carlo method," *J. Power Sources*, vol. 196, no. 23, pp. 10314–10321, 2011.

[3] F. Li and J. Xu, "A new prognostics method for state of health estimation of lithium-ion batteries based on a mixture of Gaussian process models and particle filter," *Microelectron. Rel.*, vol. 55, no. 7, pp. 1035–1045, Jun. 2015.

[4] A. Nuhic, T. Terzimehic, T. Soczka-Guth, M. Buchholz, and K. Dietmayer, "Health diagnosis and remaining useful life prognostics of lithium-ion batteries using data-driven methods," *J. Power Sources*, vol. 239, pp. 680–688, Oct. 2013.

[5] G. Ning, R. E. White, and B. N. Popov, "A generalized cycle life model of rechargeable Li-ion batteries," *Electrchimica Acta*, vol. 51, pp. 2012–2022, Feb. 2006.

[6] Y. Xing, N. Williard, K.-L. Tsui, and M. Pecht, "A comparative review of prognostics-based reliability methods for Lithium batteries," in *Proc. Prognostics Syst. Health Manage. Conf.*, May 2011, pp. 1–6.

[7] H. J. Ploehn, P. Ramadass, and R. E. White, "Solvent diffusion model for aging of lithium-ion battery cells," *J. Electrochem. Soc.*, vol. 151, no. 3, pp. A456–A462, 2004.

[8] X. Li, L. Zhang, Z. Wang, and P. Dong, "Remaining useful life prediction for lithium-ion batteries based on a hybrid model combining the long short-term memory and Elman neural networks," *J. Energy Storage*, vol. 21, pp. 510–518, Feb. 2019.

[9] Y. Song, D. Liu, C. Yang, and Y. Peng, "Data-driven hybrid remaining useful life estimation approach for spacecraft lithium-ion battery," *Microelectron. Rel.*, vol. 75, pp. 142–153, Aug. 2017.

[10] Z.-H. Zhou, "Learning with unlabeled data and its application to image retrieval," in *Proc. PRICAI*. Berlin, Germany: Springer, 2006, pp. 5–10.

[11] F. Cadini, C. Sbarufatti, F. Cancelliere, and M. Giglio, "State-of-life prognosis and diagnosis of lithium-ion batteries by data-driven particle filters," *Appl. Energy*, vol. 235, pp. 661–672, Feb. 2019.

[12] C.-J. Chang, D.-C. Li, Y.-H. Huang, and C.-C. Chen, "A novel gray forecasting model based on the box plot for small manufacturing data sets," *Appl. Math. Comput.*, vol. 265, pp. 400–408, Aug. 2015.

[13] Y.-L. He, P. J. Wang, M. Q. Zhang, Q.-X. Zhu, and Y. Xu, "A novel and effective nonlinear interpolation virtual sample generation method for enhancing energy prediction and analysis on small data problem: A case study of Ethylene industry," *Energy*, vol. 147, pp. 418–427, Mar. 2018.

[14] Q. Zhao, X. Qin, H. Zhao, and W. Feng, "A novel prediction method based on the support vector regression for the remaining useful life of lithium-ion batteries," *Microelectron. Rel.*, vol. 85, pp. 99–108, Jun. 2018.

[15] X. Wang, B. Jiang, and N. Lu, "Adaptive relevant vector machine based RUL prediction under uncertain conditions," *ISA Trans.*, vol. 87, pp. 217–224, Apr. 2019.

[16] Z.-S. Chen, B. Zhu, Y.-L. He, and L.-A. Yu, "A PSO based virtual sample generation method for small sample sets: Applications to regression datasets," *Eng. Appl. Artif. Intell.*, vol. 59, pp. 236–243, Mar. 2017.

[17] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, 2nd ed. New York, NY, USA: Academic, 1990, pp. 24–37.

[18] W. Li, Z. Jiao, L. Du, W. Fan, and Y. Zhu, "An indirect RUL prognosis for lithium-ion battery under vibration stress using Elman neural network," *Int. J. Hydrogen Energy*, vol. 44, no. 23, pp. 12270–12276, Apr. 2019.

[19] F.-K. Wang and T. Mamo, "A hybrid model based on support vector regression and differential evolution for remaining useful lifetime prediction of lithium-ion batteries," *J. Power Sources*, vol. 401, pp. 49–54, Oct. 2018.

[20] X. Zhang, Q. Miao, and Z. Liu, "Remaining useful life prediction of lithium-ion battery using an improved UPF method based on MCMC," *Microelectron. Rel.*, vol. 75, pp. 288–295, Aug. 2017.

[21] P. Niyogi, F. Girosi, and T. Poggio, "Incorporating prior information in machine learning by creating virtual examples," *Proc. IEEE*, vol. 86, no. 11, pp. 2196–2209, Nov. 1998.

[22] D.-C. Li, L.-S. Chen, and Y.-S. Lin, "Using functional virtual population as assistance to learn scheduling knowledge in dynamic manufacturing environments," *Int. J. Prod. Res.*, vol. 41, no. 17, pp. 4011–4024, 2003.

[23] J. Yang, X. Yu, Z.-Q. Xie, and J.-P. Zhang, "A novel virtual sample generation method based on Gaussian distribution," *Knowl.-Based Syst.*, vol. 24, no. 6, pp. 740–748, 2011.

[24] D.-C. Li and I.-H. Wen, "A genetic algorithm-based virtual sample generation technique to improve small data set learning," *Neurocomputing*, vol. 143, pp. 222–230, Nov. 2014.

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

[26] S. Das, S. S. Mullick, and P. N. Suganthan, "Recent advances in differential evolution—An updated survey," *Swarm Evol. Comput.*, vol. 27, pp. 1–30, Apr. 2016.

[27] F. Neri and V. Tirronen, "Recent advances in differential evolution: A survey and experimental analysis," *Artif. Intell. Rev.*, vol. 33, nos. 1–2, pp. 61–106, 2010.

[28] D.-C. Li, C.-S. Wu, T.-I. Tsai, and Y.-S. Lina, "Using mega-trend-diffusion and artificial samples in small data set learning for early flexible manufacturing system scheduling knowledge," *Comput. Oper. Res.*, vol. 34, no. 4, pp. 966–982, 2007.

[29] B. Zhu, Z. Chen, and L. Yu, "A novel mega-trend-diffusion for small sample," *CIESC J.*, vol. 67, no. 3, pp. 820–826, 2016.

[30] R. F. A. B. de Morais and G. C. Vasconcelos, "Boosting the performance of over-sampling algorithms through under-sampling the minority class," *Neurocomputing*, vol. 343, pp. 3–18, May 2019.

[31] D.-L. Li, W.-K. Lin, C.-C. Chen, H.-Y. Chen, and L.-S. Lin, "Rebuilding sample distributions for small dataset learning," *Decis. Support Syst.*, vol. 105, pp. 66–76, Jan. 2018.

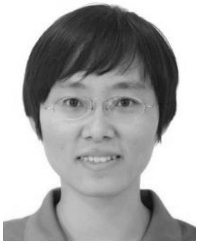
[32] Jayadeva, S. Soman, and S. Saxena, "EigenSample: A non-iterative technique for adding samples to small datasets," *Appl. Soft. Comput.*, vol. 70, pp. 1064–1077, Sep. 2018.

[33] B. Saha and K. Goebel. (2007). Battery data set. NASA Ames Prognostics Data Repository, Nat. Aeronaut. Space Admin., Ames Res. Center, Mountain View, CA, USA. [Online]. Available: http://ti.arc.nasa.gov/tech/dash/pcoe/prognostic-data-repository/

GUOQING KANG was born in Inner Mongolia, China, in 1995. He is currently pursuing the master's degree with the School of Information Engineering, Capital Normal University of China. His research interests include power electronics and electrical vehicles, data-driven modeling, and deep learning.

LIFENG WU (M'12) received the B.S. degree in applied physics from the China University of Mining and Technology, in 2002, the M.S. degree in detection technology and automation device from Northeast Electric Power University, in 2005, and the Ph.D. degree in physical electronics from the Beijing University of Posts and Telecommunications, in 2010. From 2012 to 2013, he was a Visiting Scholar with Tsinghua University. From 2014 to 2015, he was a Postdoctoral with University of Marylan, College Park, USA. From 2017 to 2018, he was a Visiting Scholar with Peking University. He is currently a Professor with Capital Normal University. His research interests include data-driven modeling, estimation and filtering, fault diagnosis, power electronics, and electrical vehicles.

YONG GUAN (M'12) received the Ph.D. degree in information and telecommunication engineering from the China University of Mining and Technology, in 2003. He is currently a Professor with Capital Normal University. His main research interests include formal verification techniques, fault diagnosis, power electronics, and electrical vehicles.



ZHEN PENG received the B.S. and M.S. degrees in computer applications technology from Shan Dong University, Jinan, China, in 2002 and 2005, respectively, and the Ph.D. degree in computer applications technology from the University of Science and Technology Beijing, Beijing, China, in 2011. She is currently a Professor with the Beijing Institute of Petrochemical Technology. Her research interests include data mining, fault diagnosis, power electronics, and electrical vehicles.