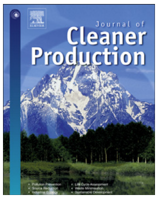
Journal of Cleaner Production 234 (2019) 1153–1164

Contents lists available at ScienceDirect

Journal of Cleaner Production

journal homepage: www.elsevier.com/locate/jclepro



State-of-charge estimation of lithium-ion battery using an improved neural network model and extended Kalman filter

Cheng Chen [a], Rui Xiong [a, *], Ruixin Yang [a, b], Weixiang Shen [b], Fengchun Sun [a]

[a] *Department of Vehicle Engineering, School of Mechanical Engineering, Beijing Institute of Technology, Beijing, 100081, China*
[b] *Faculty of Science, Engineering and Technology, Swinburne University of Technology, Hawthorn, Victoria, 3122, Australia*

ARTICLE INFO

Article history:
Received 31 March 2019
Received in revised form
20 June 2019
Accepted 23 June 2019
Available online 24 June 2019

Handling Editor: Prof. Jiri Jaromir Klemeš

Keywords:
Electric vehicles
Lithium-ion battery
State-of-charge
Neural network
Extended Kalman filter
Low temperature

ABSTRACT

Accurate state-of-charge (SoC) estimation is remarkably difficult due to nonlinear characteristics of batteries and complex application environment in electric vehicles (EVs), particularly low temperature and low SoC. In this paper, an improved battery model is first built using a feedforward neural network (FFNN) by introducing newly defined inputs. Based on the FFNN model and the extended Kalman filter algorithm, a FFNN-based SoC estimation method is designed, and its robustness is verified and discussed using the experimental data obtained at different temperatures. Finally, a hardware-in-loop test bench is built to further evaluate the real-time and generalization of the designed FFNN model. The results show that the SoC estimation can converge to the reference value at erroneous settings of an initial SoC error and an initial capacity error, and the SoC estimation errors can be stabilized within 2% after convergence, which applies to all the cases discussed in this paper, including low temperature and low SoC. This indicates that the FFNN-based method is an effective method to estimate SoC accurately in complex EV application environment.

1. Introduction

With the improvement of performance requirements for electric vehicles (EVs), such as longer driving range and faster speed, more powerful energy sources are needed. At present, lithium ion batteries are the most commonly used energy sources in EVs due to their advantage of higher energy density and longer lifetime than other batteries with different chemistries (Subburaj et al., 2015), but they still cannot provide sufficient energy to drive EVs as far as fossil fuels in traditional vehicles. Under these circumstances, it is crucial to fully utilize the energy stored in batteries for EVs through battery management systems (BMSs).

Battery state-of-charge (SoC) estimation is one of the main tasks of BMSs and its accuracy influences performances of other functions in BMSs, including charging control (Di Yin et al., 2016), balancing control (Ma et al., 2018), thermal management (Zhu et al., 2015), and safety management (Xiong et al., 2019a). Due to nonlinear characteristics and complex operation environments of batteries in EVs, it is very difficult to obtain accurate SoC, so a well-designed SoC estimation method is necessary for any BMSs.

1.1. Literature review

In the past, the most common approach for SoC estimation in BMSs was the ampere-hour method combined with initial value correction using looking-up table because of its low computation and stable output (Meng et al., 2018), (Zhang et al., 2014). However, this approach relies on accurate battery capacity, which is difficult to been obtained directly in practical applications. Although some capacity estimation methods are proposed, they usually need a specific and relatively stable environment which is different from that in EVs (Richardson et al., 2018), (Xiong et al., 2019b).

With the improvement of the processing power of on-board embedded system, model-based SoC estimation methods have become the focus in both academia and industry because of their sustained error correction mechanism through the closed-loop feedback of battery terminal voltage (Nejad et al., 2016), (Ramadan et al., 2017). This correction mechanism requires a more accurate battery model while allowing model-based SoC estimation methods to tolerate the error of battery capacity to a certain

* Corresponding author. Department of Vehicle Engineering, School of Mechanical Engineering, Beijing Institute of Technology, No. 5 South Zhongguancun Street, Haidian District, Beijing 100081, China.
E-mail address: rxiong@bit.edu.cn (R. Xiong).

https://doi.org/10.1016/j.jclepro.2019.06.273
0959-6526/© 2019 Elsevier Ltd. All rights reserved.

extent.

Common battery models can be divided into four categories: equivalent circuit models (ECMs), fractional order models (FOMs), electrochemical models (EMs) and black-box models. ECMs are the most commonly used models in EVs because of their low computation and relatively satisfactory accuracy (Xiong et al., 2019c), (Peng et al., 2017), but perform poorly at low temperature and low SoC (Huang et al., 2018), (Xia et al., 2018). FOMs can be viewed as the improved models of ECMs by giving model parameters a clearer physical significance based on electrochemical impedance spectroscopy (Xiong et al., 2019d); however, the improvement of model accuracy at low temperature and low SoC is insignificant. EMs are proposed based on battery internal reaction which usually have a higher accuracy than ECMs; however, a lot of model parameters are difficult to be obtained accurately (Gu et al., 2016), and inaccurate parameters may make such a complex model lose its advantages in describing electrochemical processes. Different from ECMs, FOMs and EMs, black-box models can describe battery external characteristic accurately through a generic network structure and a learning algorithm without knowing complex electrochemical reactions inside batteries (Dang et al., 2016), (Hussein, 2018).

Neural networks (NNs) are the most classic black-box models, and have been used in many fields, such as image recognition (Kim et al., 2018) and time-series prediction (Chandra, 2015). For battery modeling and SoC estimation, the most commonly used NNs are feedforward NNs (FFNNs) and feedback NNs (FBNNs). FFNNs can establish a direct mapping relationship between current, SoC and terminal voltage at a certain sampling point (Liu et al., 2015), (Hannan et al., 2018), and the accuracy of this mapping relationship is usually low because of battery polarization. One way to take the polarization into account is to increase the NN inputs from the data at a single sampling point to the data at multiple sampling points (Wang et al., 2016), (He et al., 2014). Although this approach can provide more battery information to train the NN model, it increases the complexity of the NN model. Another way is to add battery terminal voltage at the last sampling point as a NN input, which can improve model accuracy, reduce model complexity and also perform well in SoC estimation (Charkhgard and Farrokhi, 2010). However, this NN model can only be used for the data at fixed sampling interval. Furthermore, this NN model takes battery voltage as a NN input, which affects the stability of the NN model and the SoC estimation algorithm when the battery voltage is inaccurate or abnormal.

FBNNs include a feedback module that can save historical information. They are also able to describe battery polarization directly (Chaoui and Ibe-Ekeocha, 2017), (Chemali et al., 2018). However, FBNNs are not easy to be trained well, in particular when training data are discontinuous. Moreover, the initial state of a trained FBNN is usually set as constant which is difficult to adapt to different initial states of batteries in complex EV operation environments. Due to the fact that training data obtained in different situations, such as different temperatures, are usually fragmented, FFNNs are easier to be trained well than FBNNs. Thus, FFNNs are selected to model batteries in this paper.

1.2. Motivation and innovation

To ensure the accuracy of the FFNN while reducing its complexity, this paper improves the FFNN by adding the new NN inputs that can describe battery polarization characteristic. These new inputs can be calculated directly based on the original sampling data by the preprocessing method, which significantly reduces the inputs of the NN model. Then, a FFNN-based SoC estimation method is designed and verified based on the experimental data in different operating conditions. Finally, a BMS test bench is setup to verify the real-time and robustness of the proposed model and SoC estimation method.

1.3. Organization of this paper

The rest of this paper is organized as follows. The description of FFNN structure for battery modeling is presented in Section 2. Section 3 introduces the KF algorithm and its implementation flowchart based on the FFNN model. In Section 4, the experimental data are obtained and preprocessed to train and evaluate the model and the algorithm. The performances of the model and the algorithm are shown in Section 5 and Section 6, respectively, and the conclusions are drawn in Section 7.

2. Neural network for battery modeling

In this section, the structure and the training algorithms of a NN are first introduced as a preparation and then its application to battery modeling will be described in detail.

2.1. Feed-forward neural network

A FFNN used in this paper is shown in Fig. 1(a) and its neuron which is a basic computational unit is shown in Fig. 1(b).

According to Fig. 1(b), the output of the neuron is determined by

$$y = f\left(\sum_{i}^{n} w_i x_i - \theta\right) = f\left(\mathbf{w}^{\mathrm{T}}\mathbf{x} - \theta\right) \tag{1}$$

where n is the number of the input of the neuron, w_i is the weight between the input i and the neuron, θ is the threshold of the neuron, $f(\cdot)$ is the activation function, y is the output of the neuron, $\mathbf{w}$ is represented by a column vector $[w_1, w_2, \ldots, w_n]^{\mathrm{T}}$ and $\mathbf{x}$ is represented by a column vector $[x_1, x_2, \ldots, x_n]^{\mathrm{T}}$.

Substituting Eq. (1) into the structure of the FFNN shown in Fig. 1(a) gives

$$\begin{cases} \mathbf{y}_1 = f_1\left(\begin{bmatrix} \mathbf{w}_{1,m} & \mathbf{w}_{1,m} & \cdots & \mathbf{w}_{1,m} \end{bmatrix}^{\mathrm{T}} \mathbf{x} - \boldsymbol{\theta}_1\right) = f_1(\mathbf{W}_1\mathbf{x} - \boldsymbol{\theta}_1) = f_1(\boldsymbol{\varphi}_1) \\ \mathbf{y} = f_2\left(\begin{bmatrix} \mathbf{w}_{2,k} & \mathbf{w}_{2,k} & \cdots & \mathbf{w}_{2,k} \end{bmatrix}^{\mathrm{T}} \mathbf{y}_1 - \boldsymbol{\theta}_2\right) = f_2(\mathbf{W}_2\mathbf{y}_1 - \boldsymbol{\theta}_2) = f_2(\boldsymbol{\varphi}_2) \end{cases} \tag{2}$$

where $\mathbf{W}_1$ is the weight matrix between the input layer and the hidden layer, while $\mathbf{W}_2$ is the weight matrix between the hidden layer and the output layer; $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ are the threshold vectors of the hidden layer and the output layer; $f_1(\cdot)$ and $f_2(\cdot)$ are the

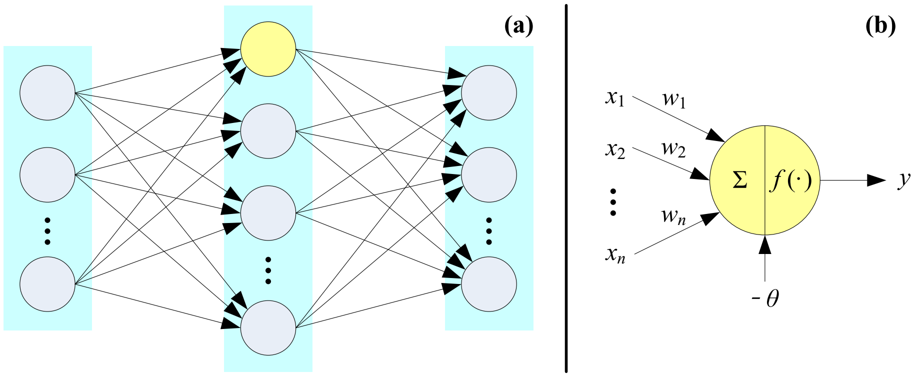


Fig. 1. A three-layer FFNN (a) and its neuron (b).

activation functions; $\mathbf{y}_1$ and $\mathbf{y}$ are the output vectors of the hidden layer and the output layer, respectively.

After the structure of the FFNN is determined, its parameters, including all of the weights and the thresholds, will be identified by learning algorithms. In this paper, back propagation algorithm (Hannan et al., 2018) is utilized to train the designed FFNN model.

2.2. Battery modeling and its improvement

The NN model of a battery should be designed carefully according to battery charge and discharge characteristics. Normally, battery terminal voltage is taken as the output and the current as the input in a NN model. Battery temperature has strong influence on the voltage response to current (Westerhoff et al., 2016), it will be taken as another input of the NN model. Furthermore, battery characteristics changes greatly in different states, including SoC, SoH (Westerhoff et al., 2016) and polarization state that significantly influences battery terminal voltage (Yang et al., 2018). Thus, these states will also need to be considered in the NN model. To ensure accurate and stable output of the NN model under different operating conditions, the inputs of the NN model should provide as much information as possible. However, too many inputs will cause a rapid increase in the number of model parameters, which easily lead to excessive calculation and even over-fitting of the NN model.

Based on the above analysis, battery terminal voltage is chosen as an only output of the NN model while battery current, surface temperature, SoC, and the newly-defined polarization state of a battery are selected as the inputs. Battery SoH is not considered in this paper, but it will be taken into consideration in the future research work. The polarization state of a battery can be defined in Eq. (3).

$$s_k = \exp(-\Delta t/\tau)s_{k-1} + (1-\exp(-\Delta t/\tau))i_{\mathrm{L},k} \tag{3}$$

where s_k is the polarization state at the sampling point k, which is different from the polarization voltage and can be calculated using only battery current; $i_{\mathrm{L},k}$ is the battery current; Δt is the sampling interval; τ is the time constant of the polarization state, which is a user-defined constant greater than 0.

This polarization state is suitable for data with variable sampling intervals, which means that the value of this polarization state is determined regardless of the size of sampling intervals when the current remains unchanged. The proof procedure is as following:

According to Eq. (3), the initial polarization state of a battery can be set as s_0, and the polarization state becomes s_1 after the battery is discharged with a constant current $i_{\mathrm{L}1}$ for a period of time Δt. The time Δt can be divided into N parts, named Δt_1 to Δt_N, and the polarization state s_1 also can be calculated recursively by Eq. (4).

$$\begin{cases} s_{0,\Delta t_1} = \exp(-\Delta t_1/\tau)s_0 + (1-\exp(-\Delta t_1/\tau))i_{\mathrm{L},1} \\ s_{0,\Delta t_1+\Delta t_2} = \exp(-\Delta t_2/\tau)s_{0,\Delta t_1} + (1-\exp(-\Delta t_2/\tau))i_{\mathrm{L},1} \\ \vdots \\ s_1 = \exp(-\Delta t_N/\tau)s_{0,\Delta t-\Delta t_N} + (1-\exp(-\Delta t_N/\tau))i_{\mathrm{L},1} \end{cases} \tag{4}$$

where $s_{0,\Delta t_1}$, $s_{0,\Delta t_1+\Delta t_2}$ and $s_{0,\Delta t-\Delta t_N}$ are the polarization state after the battery is discharged for the time Δt_1, Δt_1+Δt_2, and Δt-Δt_N, respectively.

Simplify the equations in Eq. (4), and substitute the polarization state calculated by the previous equation into the next equation:

$$s_1 = \exp\left(-\frac{1}{\tau}\sum_{i=1}^{N}\Delta t_i\right)s_0 + \left(1-\exp\left(-\frac{1}{\tau}\sum_{i=1}^{N}\Delta t_i\right)\right)i_{\mathrm{L},1} \tag{5}$$

Obviously, s_1 calculated by Eq. (3) is equal to s_1 calculated by Eq. (5), which proves the above conclusion.

The value of the time constant significantly affects the training process and the accuracy of the NN model. Small time constant makes polarization state change dramatically after sudden change in current, which is beneficial for a NN model to describe the polarization characteristic of a battery in a short time. However, when battery current becomes stable or zero, polarization state calculated with small time constant rapidly changes to a certain value and then remains unchanged. In this case, polarization state cannot describe the change of polarization voltage. On the contrary, large time constant is more suitable for describing the polarization characteristic of a battery after being left for a long time than a short time after sudden change in current. Since different time constants have their own advantages and disadvantages in different battery charging and discharging stages, more than one polarization states calculated with different time constants can be used as the inputs of the NN model to provide more information on battery polarization. But overmuch polarization states will make the model too complicated. Thus, the optimal combination of time constants should be determined.

The theoretical value range of time constant is (0, +∞). According to battery characteristics, the actual possible value range is usually (0, 1000]. For any combination of time constants, the training process is uncertain and time-consuming, which makes it almost impossible to find the optimal combination in the range of (0, 1000]. In the following, a selection method is proposed to reduce the value range of time constant.

Due to the fact that the polarization states calculated with the adjacent time constants are similar, the value of the time constant can be taken as the integer in the range (0, 1000]. To further reduce the number of polarization states, the correlation coefficient shown in Eq. (6) is used to evaluate the degree of similarity between different polarization states, and the threshold of the correlation coefficient is set in advance. If two polarization states whose correlation coefficients are greater than the set value, these two states are considered to be similar. In this way, the number of time constants will be greatly reduced. The specific selection process is introduced based on the experimental data in Section 4.2.

$$r(x,y)=\frac{Cov(x,y)}{\sqrt{Var[x]Var[y]}} \tag{6}$$

where x and y are the time series of different polarization states; $r(x,y)$ is the correlation coefficient of x and y; $Cov(x,y)$ is the covariance of x and y; $Var[x]$ and $Var[y]$ are the variances of x and y, respectively.

After the above process, several time constants are selected, and then any combination of these time constants will be used for modeling and model training based on test data in Section 5.1. Based on the training results, the final combination of time constants can be determined.

3. Extended Kalman filtering algorithm for SoC estimation

This section will introduce the basic procedure of the extended Kalman filter (EKF) and its application based on the NN model of a battery.

3.1. Extended Kalman filter

The classical EKF usually is divided into two steps, namely time update and measurement update, and its calculation procedure is shown in Table 1.

Table 1 Basic procedure of EKF

$$\text{Nonlinear system: } \begin{cases} \mathbf{x}_k = F(\mathbf{x}_{k-1},\mathbf{u}_{k-1})+\mathbf{w}_{k-1} \\ \mathbf{y}_k = G(\mathbf{x}_k,\mathbf{u}_k)+\mathbf{v}_k \end{cases} \tag{7}$$

$$\text{Definition: } \mathbf{A}_{k-1}=\left.\frac{\mathrm{d}F(\mathbf{x}_{k-1},\mathbf{u}_{k-1})}{\mathrm{d}\mathbf{x}_{k-1}}\right|_{\mathbf{x}_k=\widehat{\mathbf{x}}_{k-1}^{+}},\ \mathbf{C}_k=\left.\frac{\mathrm{d}G(\mathbf{x}_k,\mathbf{u}_k)}{\mathrm{d}\mathbf{x}_k}\right|_{\mathbf{x}_k=\widehat{\mathbf{x}}_k^{-}} \tag{8}$$

$$\text{Initialization: } \widehat{\mathbf{x}}_0=\mathrm{E}(\mathbf{x}_0),\ \mathbf{P}_0=\mathrm{E}\left[(\mathbf{x}_0-\widehat{\mathbf{x}}_0)(\mathbf{x}_0-\widehat{\mathbf{x}}_0)^{\mathrm{T}}\right] \tag{9}$$

$$\text{For } k\in\{1,2,\dots,\infty\},\ \text{calculation:} \tag{10}$$

Step 1: Time-update equations

$$\widehat{\mathbf{x}}_k^{-}=F\left(\widehat{\mathbf{x}}_{k-1}^{+},\mathbf{u}_{k-1}\right),\ \mathbf{P}_k^{-}=\mathbf{A}_{k-1}\mathbf{P}_{k-1}^{+}\mathbf{A}_{k-1}^{\mathrm{T}}+\mathbf{Q}_{k-1} \tag{11}$$

Step 2: Measurement-update equations

$$\begin{cases} \mathbf{K}_k=\mathbf{P}_k^{-}\left(\mathbf{I}+\mathbf{C}_k^{\mathrm{T}}\mathbf{R}_k^{-1}\mathbf{C}_k\mathbf{P}_k^{-}\right)^{-1}\mathbf{C}_k^{\mathrm{T}}\mathbf{R}_k^{-1} \\ \widehat{\mathbf{x}}_k^{+}=\widehat{\mathbf{x}}_{k-1}^{-}+\mathbf{K}_k\left(\mathbf{y}_k-G\left(\widehat{\mathbf{x}}_k^{-},\mathbf{u}_k\right)\right) \\ \mathbf{P}_k^{+}=\mathbf{P}_k^{-}\left(\mathbf{I}+\mathbf{C}_k^{\mathrm{T}}\mathbf{R}_k^{-1}\mathbf{C}_k\mathbf{P}_k^{-}\right)^{-1} \end{cases} \tag{12}$$

where $\mathbf{x}_k$ is the system state at the sampling point k and $\widehat{\mathbf{x}}_k$ is its guess value; $\mathbf{P}_k$ is the covariance matrix of state error; $\mathbf{x}_0$ is the initial system state and $\widehat{\mathbf{x}}_0$ is its guess value; $\mathbf{w}_k$ is the process noise vector and $\mathbf{Q}_{k\text{-}1}$ is its covariance matrix; $\mathbf{v}_k$ is the measurement noise vector and $\mathbf{R}_{k\text{-}1}$ is its covariance matrix.

3.2. State estimation algorithm

The state equation and the observation equation of a battery need to be determined. Normally, the recursive equation based on the ampere-hour method is considered as a state equation as shown below

$$z_k=z_{k-1}-\frac{\eta_k i_{\mathrm{L},k}\Delta t}{C_{\mathrm{a}}} \tag{13}$$

where z_k is the SoC, η_k is the coulomb efficiency, which is set to 1 in this paper and C_{a} is the battery capacity.

As for the observation equation, it can be described by the model built in Section 2 as follows

$$u_{t,k}=f(z_k,i_{L,k},s_k,T_k)\ =\ f_2\left(\mathbf{W}_2 f_1\left(\mathbf{W}_1[z_k,i_{L,k},s_k,T_k]^{\mathrm{T}}-\boldsymbol{\theta}_1\right)-\boldsymbol{\theta}_2\right) \tag{14}$$

where $u_{\mathrm{t},k}$ is the battery voltage, $f(\bullet)$ is the NN model and T_k is the battery temperature.

The matrix $\mathbf{A}_{k\text{-}1}$ and $\mathbf{C}_k$ in Eq. (8) can be deduced as

$$\begin{cases} \mathbf{A}_{k-1}=1 \\ \mathbf{C}_k=\dfrac{\mathrm{d}f(z_k,i_{L,k},s_k,T_k)}{\mathrm{d}z}=\dfrac{\partial\mathbf{y}}{\partial\boldsymbol{\varphi}_2}\mathbf{W}_2\dfrac{\partial\mathbf{y}_1}{\partial\boldsymbol{\varphi}_1}\mathbf{W}_1[1,0,0,\dots,0]^{\mathrm{T}} \end{cases} \tag{15}$$

According to the values of matrix $\mathbf{A}_{k\text{-}1}$ and $\mathbf{C}_k$, it is easy to prove that this system is observable. Fig. 2 shows the implementation flowchart of the EKF in Table 1 based on the NN model.

As long as the battery model is accurate enough, the algorithm in Fig. 2 can accurately estimate SoC even if the battery capacity has a certain error. Then the accuracy SoC obtained can be used to estimate the capacity accurately based on the inverse process of the ampere-hour method (Xiong et al., 2018) as

$$C_{\mathrm{a}}=\frac{\Delta Q}{\Delta z}=\frac{\sum_{i=0}^{k}(\eta_i i_{\mathrm{L},i}\Delta t)}{z_k-z_0} \tag{16}$$

where ΔQ is the accumulation of current and Δz is the variation of SoC. To ensure the stability of the algorithm, C_{a} should not be updated when Δz is too small.

4. Experiments and data preprocessing

4.1. Experiments

As shown in Fig. 3, the test bench, which is composed of a battery testing system, a thermal chamber and a computer, is built to obtain experimental data. The battery tested in this paper is the $\mathrm{Li(NiCoAl)O_2}$ battery, whose cut-off voltage is 2.5V/4.2V and rated capacity is 2.7Ah.

To obtain enough data to train the NN model, we design and perform a series of tests under different operating conditions and temperatures. Unlike the traditional testing procedure (Wang et al., 2016), (Zhu et al., 2017), the testing procedure used in this paper, as shown in Fig. 4, does not require capacity test, hybrid pulse test and OCV test, which simplifies the test procedure and shorten test

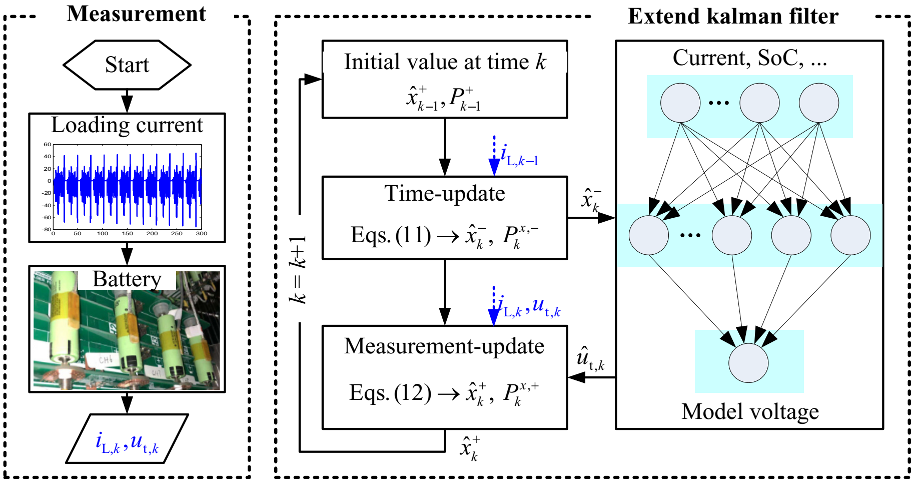


Fig. 2. Structure of state estimation algorithm.

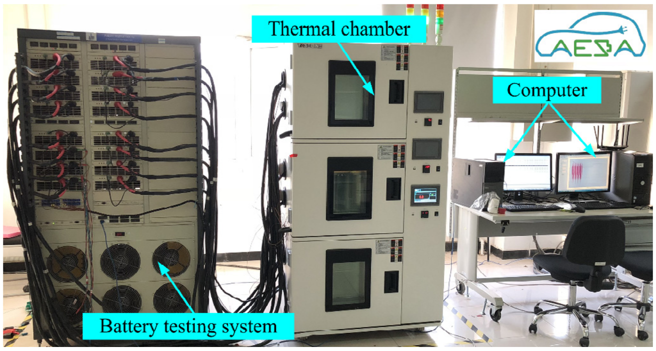


Fig. 3. Battery test bench.

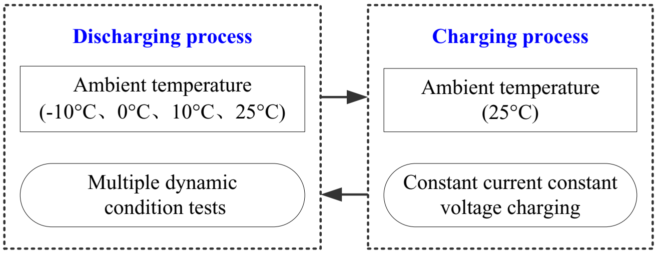


Fig. 4. Procedure of battery test.

period.

The loading current in multiple dynamic condition tests are determined by standard vehicle driving cycles (Hu et al., 2019), including EUDC, HL07, HWFET, LA92, NEDC, MANHATTAN, NYCC, REP05, SC03, and UNIF01. To ensure battery safety, battery will not be charged when battery temperature is not higher than 0 °C.

4.2. Data preprocessing

After the experimental data are obtained, the reference SoC and the polarization states can be calculated. The former can be calculated by Eq. (13) while the latter can be calculated with different time constants when the current and the corresponding sampling interval are determined. The partial experimental data and its preprocessing results are shown in Fig. 5. The polarization state is similar to battery current when time constant is equal to 1 while the polarization state changes slowly with battery current when time constant is equal to 1000.

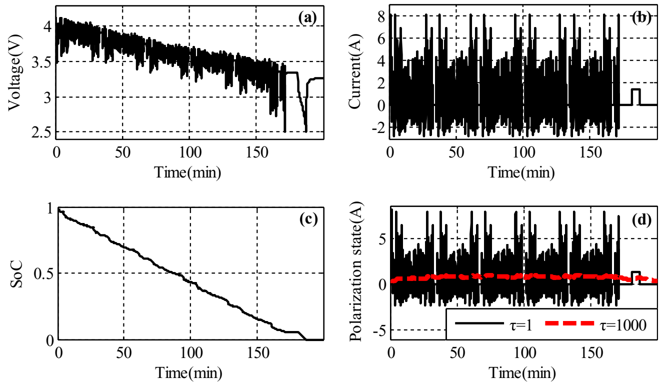


Fig. 5. Partial experimental data and its preprocessing results: (a) voltage; (b) current; (c) SoC; (d) polarization state.

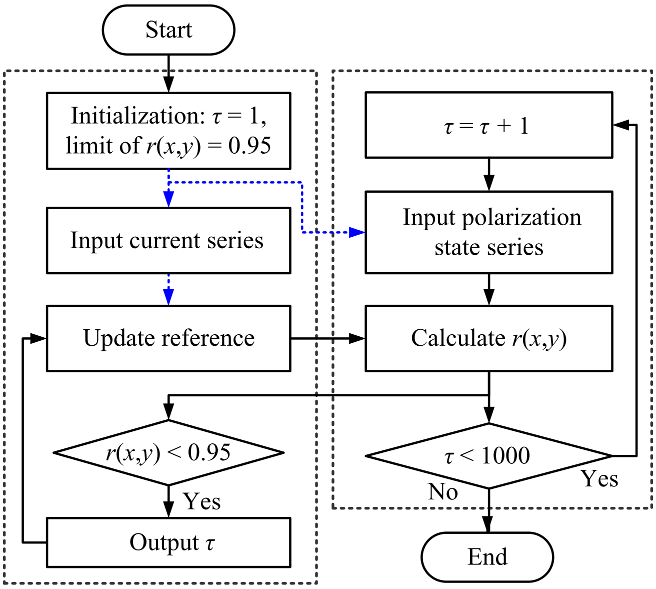


Fig. 6. Selection process of time constants.

As mentioned in Section 2, the value range of time constant should be further reduced based on the correlation coefficient shown in Eq. (6). As shown in Fig. 6, a selection method of time constant is designed.

The selection process is implemented in four steps. **Step 1**: set the threshold of the correlation coefficient and the initial time constant to 0.95 and 1, respectively, and due to the fact that polarization state will tend to be battery current when time constant tends to zero, set the initial reference series to battery current series. **Step 2**: input the polarization state series calculated by current time constant, and calculate the correlation coefficient between this polarization state series and the reference series. **Step 3**: if this correlation coefficient is lower than the limit of 0.95, consider the current time constant as the new reference to find the next time constant and the output of this selection process; otherwise, keep the previous reference series unchanged. **Step 4**: if the current time constant is equal to 1000, end the selection process; otherwise, increase the value of the time constant by 1 and return to Step 2.

Finally, eight time constants shown in Eq. (17) are selected to calculate the corresponding eight polarization state series that will be used for the following discussion. Note that these time constants are obtained in the above process based on only loading current, which means they can be used to other batteries with different chemistries.

$$\tau \in \{2, 6, 14, 32, 71, 155, 336, 688\} \tag{17}$$

5. Verification and discussion

Based on the test data, the performances of the NN models using different combinations of polarization states as the inputs and their applications to state estimation under different conditions are discussed in this section.

5.1. Evaluation of the NN models

The available inputs of the NN model include current, temperature, SoC, and the polarization states with different time constants. The first three inputs will remain the same and the number of polarization states as the inputs is varied.

5.1.1. Evaluation of the NN model based on the sectional experimental data

The roles of the polarization states calculated with different time constants in describing the polarization characteristic of a battery are analyzed qualitatively in Section 2.2. In the following, they will be discussed based on the sectional experimental data after the battery is discharged to the cutoff voltage, and three different stages of the data are carefully selected for an intuitive display, as shown in Fig. 7. The sampling interval of the data in the first 10 s is about 100 ms, and the sampling interval of the data after 10 s is 1s.

Three stages of the data are used for model training step by step, and the training results are shown in Fig. 8, where the $\tau = [2,688]$ represents the training results of the model using two polarization states as inputs.

Fig. 8(a) and (b) show the training results using the stage 1 data, which reflects the ability to describe the polarization characteristic of a battery in a short time after sudden change in current. The training error increases with the rise of time constant and reaches maximal when the time constant is equal to 688. Fig. 8(c) and (d) show the training results using the stage 2 data, which reflects the ability to describe the polarization characteristic after a battery is left for a long time. The training effect deteriorates as time constant decreases, and the effect is worst when the time constant is equal to 2. Obviously, these results are consistent with the qualitative analysis in Section 2.2, namely, small time constant is suitable for describing the polarization characteristic in a short time after sudden change in current while large time constant is suitable for describing the polarization characteristic after being left for a long time.

The stage 3 data, including the stage 1 and the stage 2, are used to evaluate the ability of the models using more than one time constants to describe the polarization characteristic during the entire rest stage. As shown in Fig. 8(e) and (f), the training result when two polarization states are added to the inputs of the NN model is more accurate than that when only one polarization state is added.

5.1.2. Evaluation of the NN model based on the complete experimental data

After selecting the time constants in Section 4.2, we can relatively easily determine the final combination of time constants by model training results based on the whole experimental data. The experimental data used here are obtained at −10 °C, 0 °C, 10 °C and 25 °C, and the data at any temperature are divided into two parts roughly in a ratio of four to one, about 80% of them used for model training and about 20% of them used for model validation.

The NN models whose inputs include only one polarization state are discussed first, and the statistical validation results of the fully trained models are shown in Fig. 9(a). The root mean square error (RMSE) of the models decreases first and then increases with the increase of time constant. The RMSE reaches minimum value when τ is equal to 14, which is viewed as the final RMSE of the NN models with one polarization state. The number of NN model inputs is

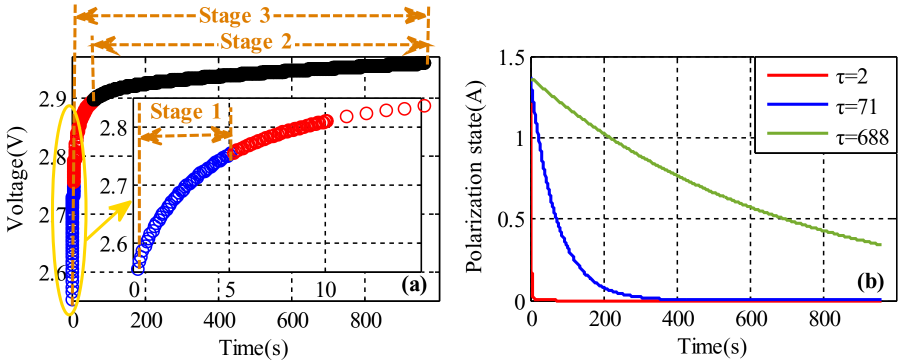


Fig. 7. Section experimental data at low SoC: (a) voltage; (b) polarization state.

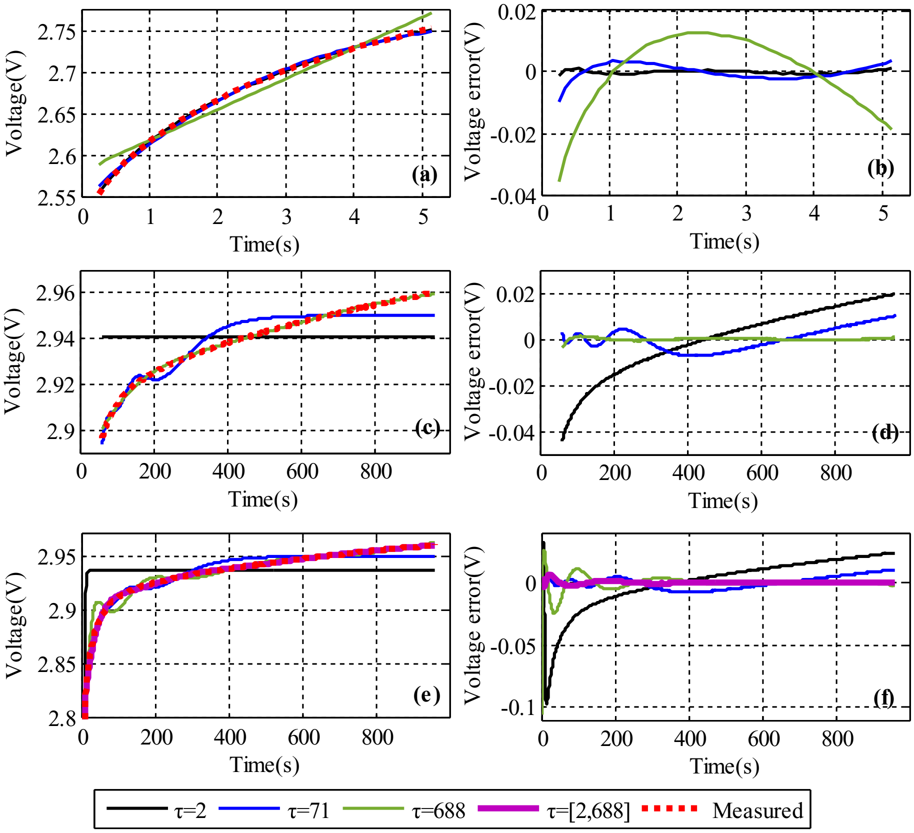


Fig. 8. Training results using different stages of data: (a) estimated and measured voltage of stage 1 and (b) corresponding error; (c) estimated and measured voltage of stage 2 and (d) corresponding error; (e) estimated and measured voltage of stage 3 and (f) corresponding error.

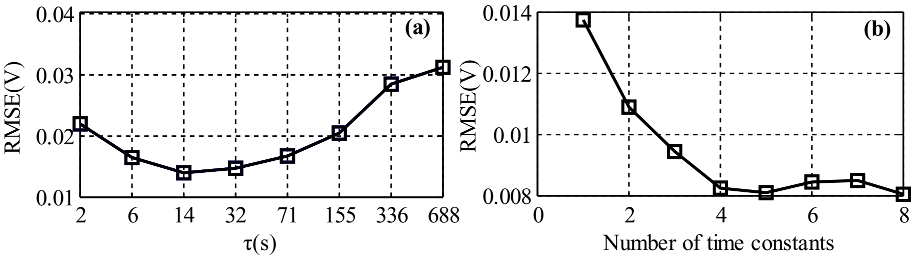


Fig. 9. Statistical validation results: (a) NN models with single polarization state; (b) NN models using different number of polarization states.

gradually increased, and the statistical results of the NN models using different number of inputs are shown in Fig. 9(b). The RMSE curve first drops rapidly and then remains essentially constant with increasing number of time constants. Considering the balance between accuracy and complexity of model, this paper recommends using four time constants, namely 2, 6, 32 and 155, as the final combination.

Actually, it is still not easy to obtain the optimal combination of eight time constants in Eq. (17), and after several attempts, we get the above results. Note that the four time constants are selected based on the experimental data including battery voltage, which means that the time constants for other batteries with different chemistries need to be reelected, but the above results also can give a reference for other batteries.

5.2. Evaluation of applications of NN models for state estimation

Based on the trained NN model, the EKF shown in Section 3 is used to estimate SoC. Considering that the stability of the algorithm is important for practical applications, we discuss the performances of the designed SoC estimation algorithm in different operating conditions. The experimental data used in this Section, namely Section 5.2, are not used for the model training in Section 5.1, but only used for the model validation.

To show the advantages of the proposed method, two common estimation methods are taken as a reference. The first one is the equivalent circuit model (ECM)-based method, and the second is the data-driven method based on FFNN named FFNN method. Similar to the FFNN model in Section 2, the FFNN method also establishes the relationship between battery voltage, current, temperature, SoC, and polarization state; however, this method views SoC as output.

5.2.1. Evaluation results at an inaccurate initial SoC value

Based on the experimental data of SC06 at 10 °C, the FFNN-based method, the ECM-based method and the FFNN method are

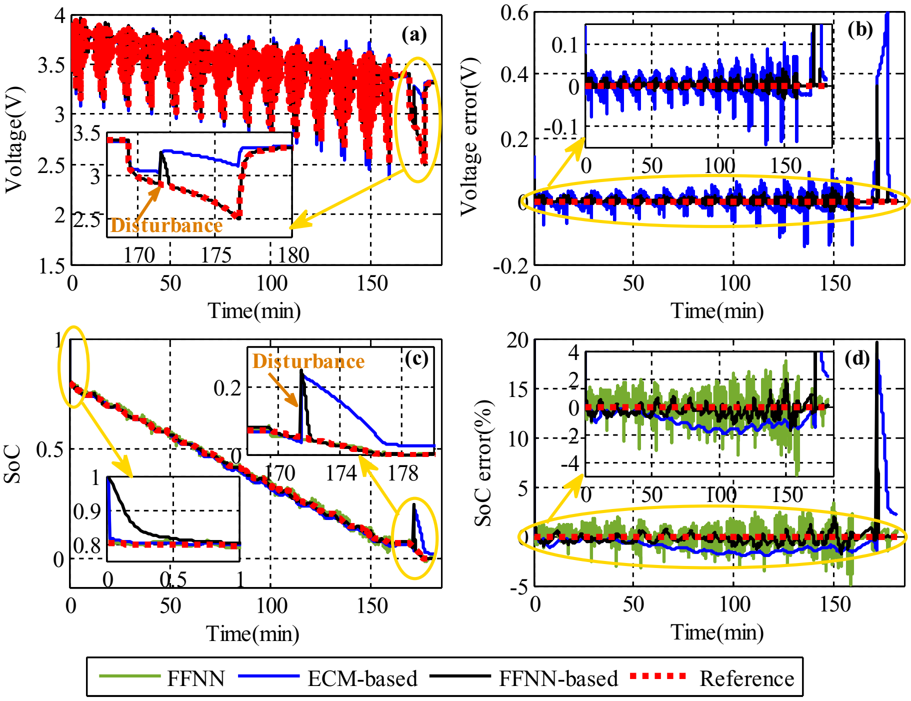


Fig. 10. Results of SoC estimation in cases of an inaccurate initial SoC value: (a) estimated and measured voltage and (b) voltage error; (c) estimated and reference SoC and (d) SoC error.

used to estimate SoC when the initial SoC value is erroneous. The initial SoC value of test data used here is about 80%, while the initial SoC value of SoC estimation method is set to 100%. The capacity of SoC estimation method is set accurately. To test the robustness of SoC estimation method at low SoC, a disturbance of 20% SoC is added when the reference SoC reaches about 5%. The terminal voltage and SoC estimation results are plotted in Fig. 10, where the Reference represents the measured voltage or reference SoC. Note that the only output of the FFNN method is the estimated SoC, and so Fig. 10(a) and (b) only show the voltage results of the ECM-based method and the FFNN-based method.

Fig. 10(a) and (b) show that the voltage estimation errors of the FFNN-based method are within 0.05V except the initial period and the disturbance period. Fig. 10(c) and (d) show that the SoC estimation results of the FFNN-based method can converges to reference value quickly, and its errors are maintained within 2% after convergence. Compared with the ECM-based method, the FFNN-based method has the more accurate estimation results of terminal voltage and a better robustness for the disturbance at low SoC because of the high accuracy of the FFNN model. For the FFNN method, although it have a faster convergence speed for the disturbance of SoC than the FFNN-based method, its estimation results fluctuate dramatically, while the results of the FFNN-based method are more stable and accurate owing to the EKF.

The run times of the three SoC estimation methods in MATLAB are recorded to evaluate their computation preliminarily. Note that these methods run continuously under the same conditions to ensure the validity of their comparison; the CPU used is Intel Core i5-4200H (2.80 GHz), the RAM used is 8.00 GB, and the version of MATLAB is 2014a. After repeated tests, the mean computational times of the FFNN-based method, the ECM-based method and the FFNN method are 1.2228s, 1.3443s, and 0.2265, respectively. The comparison results show that the run times of the FFNN-based method and the ECM-based method is similar although the FFNN model is much more complicated than the ECM. This is because that the FFNN-based method has only one state, meanwhile, the parallel operation of MATLAB is also beneficial for the FFNN-based method to save the run time. Considering that the existing universal chip for BMS does not have the function of parallel operation, the real-time of the FFNN-based method in BMS should be further verified.

5.2.2. Evaluation results at an inaccurate initial capacity value

Based on the experimental data of UNIF01 at 25 °C, the above three methods are used to estimate SoC when the initial value of SoC and capacity are erroneous. The initial SoC value of test data is about 90%, while the initial SoC value of SoC estimation method is set to 70%. The discharge capacity of test data is about 2.565Ah, while the initial capacity value of SoC estimation method is set to 2.309Ah. Similarly, a disturbance of −10% SoC is added when the reference SoC reaches about 10%. The terminal voltage, SoC and capacity estimation results are plotted in Fig. 11.

Fig. 11 shows that the FFNN-based method can estimate battery voltage, SoC and capacity accurately even when its initial capacity errors are set as 10%, and its estimation errors of voltage, SoC and capacity remain within 0.05V, 2% and 3%, respectively, after convergence. For the case that the battery capacity is erroneous, the model-based methods have to be setting to rely more on battery models to correct SoC; this approach is feasible for an accurate model, such as the above FFNN model, while it may lead to abnormal results of SoC estimation when the model error is large, such as the SoC estimation results of the ECM-based method after the disturbance of SoC is added at low SoC, as shown in Fig. 11(c) and (d).

5.2.3. Evaluation results at low temperature

The performances of SoC estimation methods at low temperature are discussed based on the test data of MANHATTAN at −10 °C. This discussion also considers effects of inaccurate initial SoC value, inaccurate initial capacity value and disturbance at low SoC.

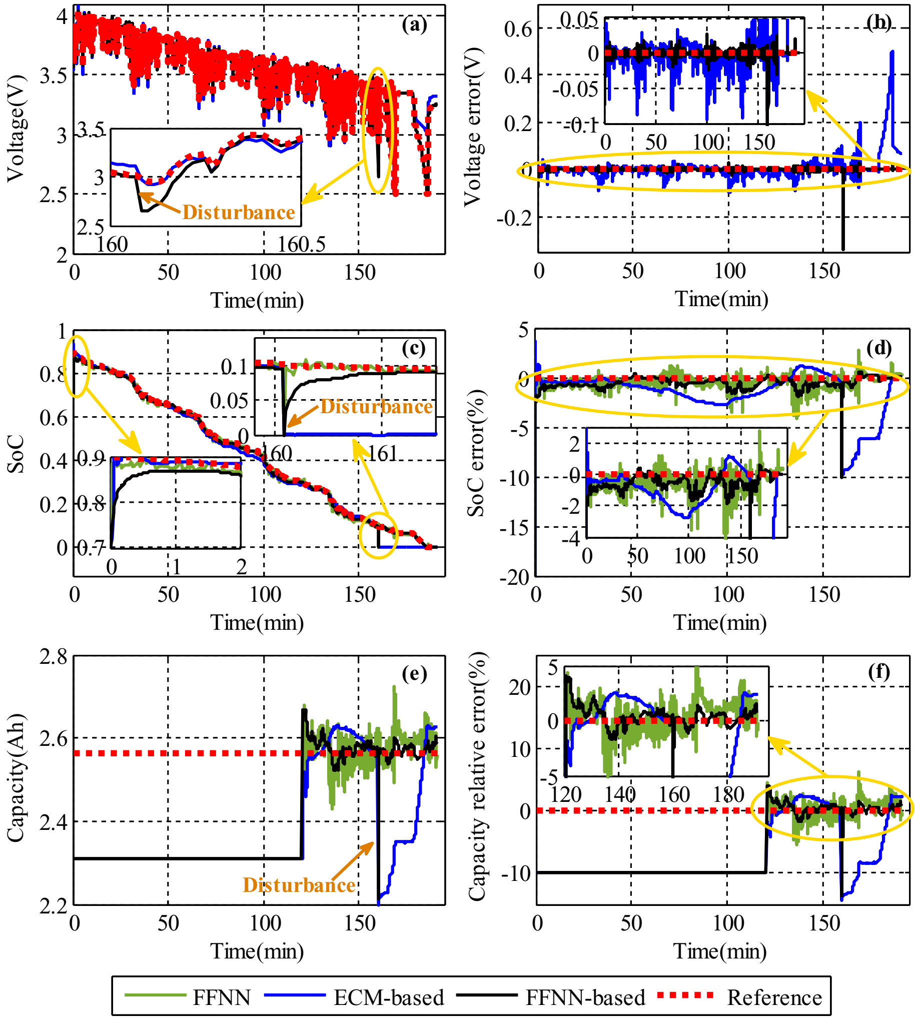


Fig. 11. Results of SoC estimation in cases of an inaccurate initial capacity value: (a) estimated and measured voltage and (b) voltage error; (c) estimated and reference SoC and (d) SoC error; (e) estimated and reference capacity and (f) capacity error.

Considering that the ECM-based method is difficult to accurately estimate SoC at low temperature (−10 °C) and large current (3C) because the accuracy of the ECM deteriorates sharply in this case, we only discuss the performances of the FFNN-based method and the FFNN method here. The terminal voltage, SoC and capacity estimation results are plotted in Fig. 12.

It shows that the FFNN-based method also has good estimation performances at −10 °C, and after convergence, its estimation errors of voltage, SoC and capacity remain within 0.06V, 2% and 3%, respectively. For the FFNN method, the fluctuation of SoC estimation results at −10 °C is more significant than 10 °C and 25 °C because of the dramatically fluctuating voltage in Fig. 12(a); although its SoC errors can be kept within 5%, the fluctuation of SoC greatly increases the capacity estimation error. As for the FFNN-based method, the errors of the FFNN model at −10 °C also increase slightly; however, after filtering, the influence of the voltage fluctuation can be weakened, and then the relatively stable and accurate estimation results of SoC and capacity can be obtained.

6. Hardware-in-Loop verification

In order to further verify the proposed FFNN-based method in real-time, a hardware-in-loop (HiL) test bench, including a real battery, is designed to closely simulate EV operation conditions.

6.1. Hardware-in-loop test bench

The hardware-in-loop test bench is shown in Fig. 13. It is setup on the basis of the test bench in Fig. 3 by adding some BMS modules, including a battery monitoring unit (BMU), a battery control unit (BCU), a current senor, a DC source, a download device and a CAN analyzer.

The BCU is used to calculate the battery voltage and battery SoC from the NN model. The BMU and the current sensor collect measured battery voltages and currents, respectively, which are sent to the BCU through the CAN bus. The computer is used to design the NN models and generate the executable codes, which are downloaded to the BCU for real-time operations through the download device. The DC source is used to power the current sensor, the BCU and the BMU.

6.2. Results of the FFNN-based method

A mixed operating condition consisting of the above conditions are used, and the ambient temperature during battery testing is set as about −5 °C, which are not used in the previous training and verification process. The battery capacity is not updated in the HiL test, namely it is inaccurate in the whole test process if its initial setting value has an error. The capacity setting is 2.5 Ah.

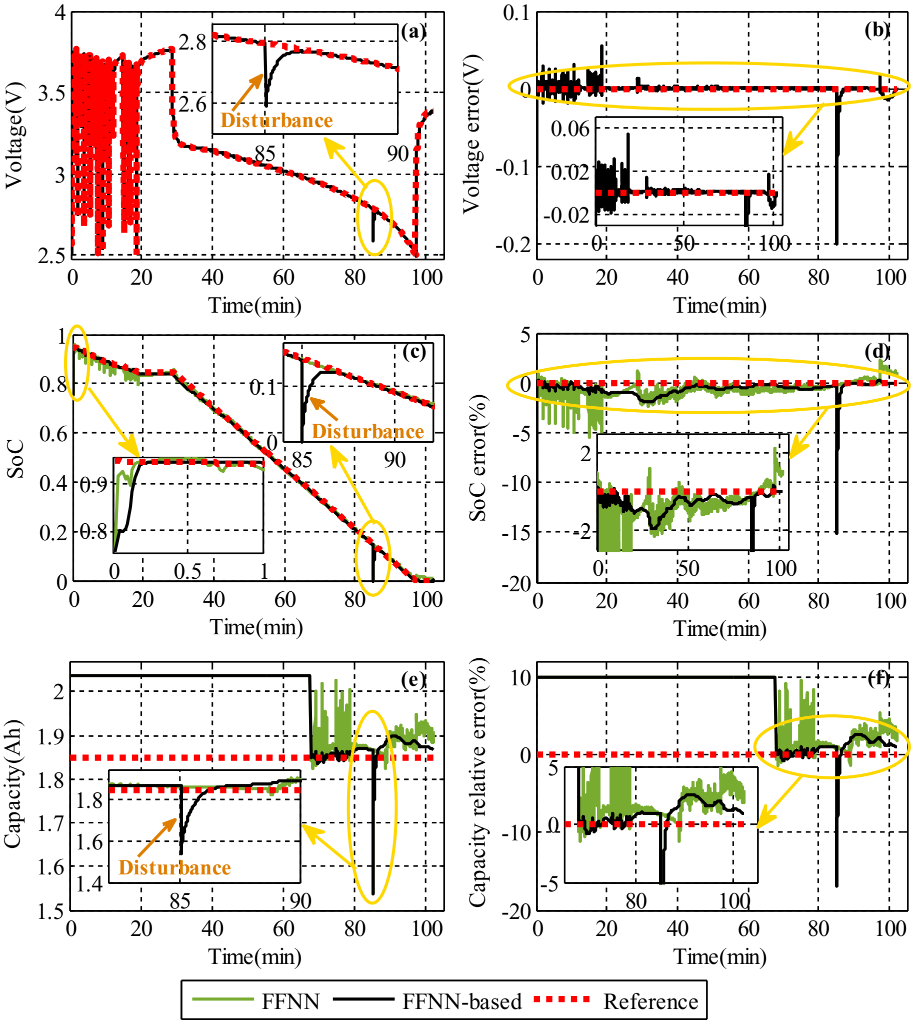


Fig. 12. Results of SoC estimation at −10 °C: (a) estimated and measured voltage and (b) voltage error; (c) estimated and reference SoC and (d) SoC error; (e) estimated and reference capacity and (f) capacity error.

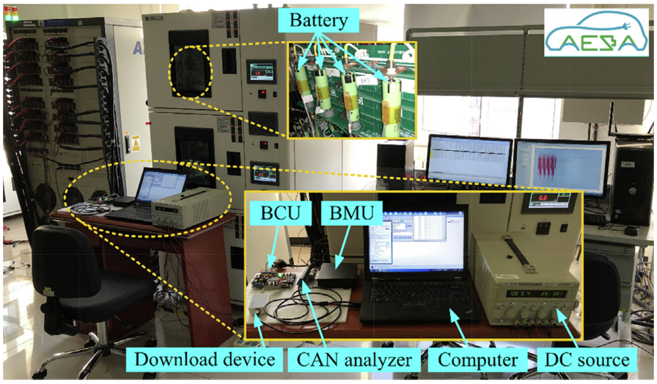


Fig. 13. Physical map of hardware-in-loop test bench.

In the preparation stage of the HiL test, a fully-charged battery is discharged by a constant current until its SoC is equal to approximately 90%. Then, the HiL test starts with the setting of initial SoC value of 100%, which is 10% different from the true value and loading the mixing operating conditions. The test results are shown in Fig. 14.

It shows that the SoC estimation can converge to the reference value gradually and the absolute value of the SoC estimation error is less than 2%. The real capacity value of battery at −5 °C can be calculated by the accumulation of current and is 2.087Ah, which means the setting capacity error of the algorithm in this HiL test is about 19.7% away from the true capacity of the battery. The above results demonstrate that the designed FFNN-based SoC estimation method meets the basic needs of EV applications.

7. Conclusions

This paper proposes an improved battery model based on the FFNN to accurately describe battery characteristics, in particular at low temperature and low SoC, and the model are further determined and improved based on the experimental data at −10 °C, 0 °C, 10 °C and 25 °C. Then, a FFNN-based SoC estimation method is designed, which is fully verified in the cases of the inaccurate initial SoC value, the inaccurate initial capacity value and low temperature. The results show that the SoC estimation can always converge to the true value quickly while keeping the low error within 2%, even under an inaccurate setting of capacity, which indicates that this FFNN based SoC estimation algorithm is more dependent on the battery model. For the capacity estimation, it can also be updated accurately if the capacity update requirement is met.

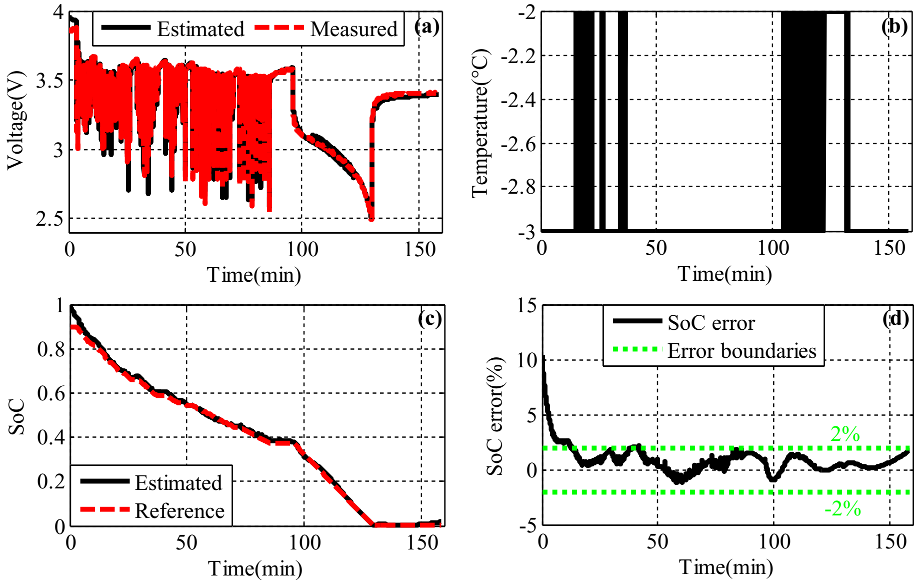


Fig. 14. Results of designed NN model in HiL test at −5 °C: (a) estimated and measured voltage; (b) battery temperature; (c) estimated and reference SoC and (d) SoC error.

Finally, the real-time test of the designed NN model for battery SoC estimation are further verified by the hardware-in-the-loop approach using a new mixing operating conditions and a new ambient temperature.

Acknowledgments

This work was supported by the National Key Research and Development Program of China (Grant 2018YFB0104100) and Graduate Technological Innovation Project of Beijing Institute of Technology (Grant No. 2018CX10003). The systemic experiments of the lithium-ion batteries were performed at the Advanced Energy Storage and Application (AESA) Group, Beijing Institute of Technology.

References

Chandra, R., 2015. Competition and collaboration in cooperative coevolution of elman recurrent neural networks for time-series prediction. IEEE Trans. Neural Networks Learn. Syst. 26 (12), 3123–3136.

Chaoui, H., Ibe-Ekeocha, C.C., 2017. State of charge and state of health estimation for lithium batteries using recurrent neural networks. IEEE Trans. Veh. Technol. 66 (10), 8773–8783.

Charkhgard, M., Farrokhi, M., 2010. State-of-Charge estimation for lithium-ion batteries using neural networks and EKF. IEEE Trans. Ind. Electron. 57 (12), 4178–4187.

Chemali, E., Kollmeyer, P.J., Preindl, M., Ahmed, R., Emadi, A., 2018. Long short-term memory networks for accurate state-of-charge estimation of Li-ion batteries. IEEE Trans. Ind. Electron. 65 (8), 6730–6739.

Dang, X., Yan, L., Xu, K., Wu, X., Jiang, H., Sun, H., 2016. Open-circuit voltage-based state of charge estimation of lithium-ion battery using dual neural network fusion battery model. Electrochim. Acta 188, 356–366.

Di Yin, M., Cho, J., Park, D., 2016. Pulse-based fast battery IoT charger using dynamic frequency and duty control techniques based on multi-sensing of polarization curve. Energies 9 (209), 1–20.

Gu, R., Malysz, P., Yang, H., Emadi, A., 2016. On the suitability of electrochemical-based modeling for lithium-ion batteries. IEEE Trans. Transp. Electrif. 2 (4), 417–431.

Hannan, M.A., Lipu, M.S.H., Hussain, A., Saad, M.H., Ayob, A., 2018. Neural network approach for estimating state of charge of lithium-ion battery using backtracking search algorithm. In: IEEE Access, vol. 6, pp. 10069–10079.

He, W., Williard, N., Chen, C., Pecht, M., 2014. State of charge estimation for Li-ion batteries using neural network modeling and unscented Kalman filter-based error cancellation. Int. J. Electr. Power Energy Syst. 62, 783–791.

Hu, J., Niu, X., Jiang, X., Zu, G., 2019. Energy management strategy based on driving pattern recognition for a dual-motor battery electric vehicle. Int. J. Energy Res. https://doi.org/10.1002/er.4474 to be published.

Huang, C., Wang, Z., Zhao, Z., Wang, L., Lai, C.S., Wang, D., 2018. Robustness evaluation of extended and unscented kalman filter for battery state of charge estimation. IEEE Access 6, 27617–27628.

Hussein, A.A., 2018. “Adaptive artificial neural network based models for instantaneous power estimation enhancement in electric vehicles' Li -ion batteries. IEEE Trans. Ind. Appl. 55 (1), 840–849.

Kim, J., Nguyen, A.D., Lee, S., 2018. Deep CNN-based blind image quality predictor. IEEE Trans. Neural Networks Learn. Syst. 30 (1), 11–24.

Liu, F., Liu, T., Fu, Y., 2015. An improved SoC estimation algorithm based on artificial neural network. In: 2015 8th Int. Symp. Comput. Intell. Des, vol. 2, pp. 152–155.

Ma, Y., Duan, P., Sun, Y., Chen, H., 2018. Equalization of lithium-ion battery pack based on fuzzy logic control in electric vehicle. IEEE Trans. Ind. Electron. 65 (8), 6762–6771.

Meng, J., Ricco, M., Luo, G., Swierczynski, M., Stroe, D.I., Stroe, A.I., Teodorescu, R., 2018. An overview and comparison of online implementable SOC estimation methods for lithium-ion battery. IEEE Trans. Ind. Appl. 54 (2), 1583–1591.

Nejad, S., Gladwin, D.T., Stone, D.A., 2016. A systematic review of lumped-parameter equivalent circuit models for real-time estimation of lithium-ion battery states. J. Power Sources 316, 183–196.

Peng, S., Chen, C., Shi, H., Yao, Z., 2017. State of charge estimation of battery energy storage systems based on adaptive unscented kalman filter with a noise statistics estimator. IEEE Access 5, 13202–13212.

Ramadan, H.S., Becherif, M., Claude, F., 2017. Extended kalman filter for accurate state of charge estimation of lithium-based batteries: a comparative analysis. Int. J. Hydrogen Energy 42 (48), 29033–29046.

Richardson, R.R., Birkl, C.R., Osborne, M.A., Howey, D., 2018. Gaussian process regression for in-situ capacity estimation of lithium-ion batteries. IEEE Trans. Ind. Inf. 15 (1), 127–138.

Subburaj, A.S., Pushpakaran, B.N., Bayne, S.B., 2015. Overview of grid connected renewable energy based battery projects in USA. Renew. Sustain. Energy Rev. 45, 219–234.

Wang, Y., Yang, D., Zhang, X., Chen, Z., 2016. Probability based remaining capacity estimation using data-driven and neural network model. J. Power Sources 315, 199–208.

Westerhoff, U., Kurbach, K., Lienesch, F., Kurrat, M., 2016. Analysis of lithium-ion battery models based on electrochemical impedance spectroscopy. Energy Technol. 4 (12), 1620–1630.

Xia, B., Zhang, Z., Lao, Z., Wang, W., Sun, W., Lai, Y., Wang, M., 2018. Strong tracking of a h-infinity filter in lithium-ion battery state of charge estimation. Energies 11 (6), 1481–1500.

Xiong, R., Yang, R.X., Chen, Z.Y., Shen, W.X., Sun, F.C., 2019a. Online fault diagnosis of external short circuit for lithium-ion battery pack. IEEE Tran. Ind. Electron. https://doi.org/10.1109/TIE.2019.2899565 to be published.

Xiong, R., Zhang, Y., Wang, J., He, H., 2019b. Lithium-ion battery health prognosis based on a real battery management system used in electric vehicles. IEEE Trans. Veh. Technol. 68 (5), 4110–4121.

Xiong, R., Yu, Q., Shen, W.X., Lin, C., Sun, F., 2019c. A sensor fault diagnosis method for a lithium-ion battery pack in electric vehicles. IEEE Trans. Power Electro. 34 (10) https://doi.org/10.1109/TPEL.2019.2893622 to be published.

Xiong, R., Tian, J., Shen, W.X., Sun, F., 2019d. A novel fractional order model for state of charge estimation in lithium ion batteries. IEEE Trans. Veh. Technol. 68 (5), 4130–4139.

Xiong, R., Li, L., Tian, J., 2018. Towards a smarter battery management system : a critical review on battery state of health monitoring methods. J. Power Sources 405 (5), 18–29.

Yang, J., Du, C., Wang, T., Gao, Y., Cheng, X., Zuo, P., Ma, Y., Wang, J., Yin, G., Xie, J., Lei, B., 2018. Rapid prediction of the open-circuit-voltage of lithium ion batteries based on an effective voltage relaxation model. Energies 11 (12), 3444, 1658.

Zhang, Y., Song, W., Lin, S., Feng, Z., 2014. A novel model of the initial state of charge estimation for $LiFePO_4$ batteries. J. Power Sources 248, 1028–1033.

Zhu, J.G., Sun, Z.C., Wei, X.Z., Dai, H.F., 2015. A new lithium-ion battery internal temperature on-line estimate method based on electrochemical impedance spectroscopy measurement. J. Power Sources 274, 990–1004.

Zhu, Q., Xiong, N., Yang, M.L., Sen Huang, R., Di Hu, G., 2017. "State of charge estimation for lithium-ion battery based on nonlinear observer: an $H\infty$ method. Energies 10 (5), 678–696.