Optimized multi-hidden layer long short-term memory modeling and suboptimal fading extended Kalman filtering strategies for the synthetic state of charge estimation of lithium-ion batteries.

XIE, Y., WANG, S., ZHANG, G., FAN, Y., FERNANDEZ, C. and BLAABJERG, F.

2023

Optimized multi-hidden layer long short-term memory modeling and suboptimal fading extended Kalman filtering strategies for the synthetic state of charge estimation of lithium-ion batteries

Yanxin Xie[a], Shunli Wang[a, b*], Gexiang Zhang[a], Yongcun Fan[a], Carlos Fernandez[c], Frede Blaabjerg[d]

[a]School of Information Engineering, Southwest University of Science and Technology, Mianyang 621010, China; [b]School of Electrical Engineering, Sichuan University, Chengdu 610065; [c]School of Pharmacy and Life Sciences, Robert Gordon University, Aberdeen AB10-7GJ, UK; [d]Department of Energy Technology, Aalborg University, Pontoppidanstraede 111 9220 Aalborg East, Denmark.

Abstract: With the demand for high-endurance lithium-ion batteries in new energy vehicles, communication and portable devices, high energy density lithium-ion batteries have become the main research direction of the battery industry. State of Charge (SoC), as a state parameter that must be accurately evaluated by the battery management system, enables online safety monitoring of the battery operation, and prolongs its service life. In this paper, an improved algorithm based on multi-hidden layer long short-term memory (MHLSTM) neural network and suboptimal fading extended Kalman filtering (SFEKF) is proposed for synthetic SoC estimation. First, the battery external measurable information is captured. The battery real data properties are matched with the network topology without additional battery model construction, and the battery SoC is roughly evaluated using an MHLSTM network. Then, a suboptimal fading factor is inserted into the extended Kalman filter (EKF) algorithm for iterative recursion and adaptive handling to smooth the prediction results of the MHLSTM network and enhance the accuracy of state estimation, system stability, and generality. Three customized electric vehicle (EV) driving conditions datasets are categorized into training and testing sets to fulfill the efficient estimation of synthetic SoC by the fusion algorithm and solve the time series problem. Using the maximum error (ME), mean absolute error (MAE), root mean squared error (RMSE), and mean absolute percentage error (MAPE), the results show that the maximum bias of the fusion algorithm to estimate the synthetic SoC is limited to within 1.2%, even under the abrupt change of the system. It can converge to the real value quickly and maintains an excellent tracking capability for data changes, reflecting the high accuracy estimation capability and the robustness possessed by the system.

Key words: Ternary lithium-ion battery; long time series; long short-term memory network; hyper-parameter selection; suboptimal fading factor extended Kalman filtering algorithm; custom driving conditions

***Corresponding author**: Shun-li Wang. Tel: +86-15884655563. E-mail address: 497420789@qq.com.

1. Introduction

With the speedy expansion of high-end emerging industries such as new energy vehicles, aerospace and rail transportation, lithium-ion batteries management and real-time safety monitoring have become the core technologies of energy systems to reliably alleviate the energy crisis [1, 2]. Lithium-ion batteries as the material of choice for green batteries. Due to its advantages of high energy density and vibrancy density, long cycle life and low cost, the end-use market is rapidly diversifying outward [3-6]. Simultaneously, production applications benefit from market demand, high growth dividends, technology costs, and other factors driving control. Energy storage technology gradually advances to the core of the battery's multi-dimensional key technology at high speed, with a very broad development space [1]. The State of Charge (SoC) refers to the ratio of the remaining capacity in the battery and its fully charged state after using it for a period or leaving it unused for a long time, which can directly impact the service life of the battery and the safe operation and overall performance of the electric vehicle (EV) [7, 8]. The SoC not only provides more intuitive information to the user, but is also very important and challenging for the safe management of batteries, improving range, and reducing costs [9-11].

The SoC estimation methods are classified into four main categories: direct measurement methods, ampere-hour (Ah) integration methods, model-based methods, and artificial intelligence (AI) methods. The direct measurement methods are based on parameter characterization, which includes methods such as the open circuit voltage (OCV) method, electrochemical impedance spectroscopy (EIS), etc. [12-15]. The chief merit of this method is its simplicity and ease of execution. Nevertheless, the OCV method is time-consuming, more significantly affected by changes in environmental factors such as temperature and noise, and cannot be used for batteries in operating conditions [16, 17]. The Ah integration method relies excessively on the initial given SoC

value of the system, which always fails to eliminate the influence caused by the accumulated errors and deviates seriously from the true value [18, 19]. Thus, if this method is applied to the actual battery operating condition, a correction factor needs to be introduced to resolve the cumulative deviation.

The model-based method refers to the process of using a mathematical model to characterize the dynamic system for battery charging and discharging from the battery operating mechanism, and then calculating the battery SoC based on this model. Although the model parameters can be processed with fixed values, the model parameters are adaptively and dynamically adjusted in actual on-board conditions, which are influenced by environmental noise, complex working conditions, and the degree of battery aging [20, 21]. The Kalman filtering (KF) algorithm is the most frequently used model-based prognostic method, which is an optimal autoregressive data filtering method that not only corrects the initial error but also suppresses system noise. Since the battery is a strongly nonlinear system, the basic KF algorithm can only be applied to the online system, so a series of variants are operated on it, such as the extended Kalman filtering (EKF) algorithm, unscented Kalman filtering algorithm (UKF), etc. [22-25]. In this regard, Zhang et al. [26] proposed an improved particle swarm optimization and EKF method to weaken the strong cross-talk between model parameters and SoC. This method can update the model parameters periodically, but it cannot get rid of the errors arising from the identification of the model parameters, which makes the accuracy of the SoC estimation constrained. Chen et al. [27] modeled a fractional-order second-order RC circuit and developed an adaptive fractional-order square-root unscented Kalman filter (AFSR-UKF) for high-precision SoC estimation using a particle swarm optimization algorithm parameter identification with a linear differential descent strategy. But this approach does not take into account factors affecting battery parameters such as hysteresis, aging, and self-discharge characteristics in battery modeling, making it impossible to pair the model with consistent internal battery characteristics. Zheng et al. [28] construct a universal cell model based on differential voltage and particle filtering methods to estimate SoC, which updates the particle weights with high accuracy and a wide range of applications. However,

this requires a large number of samples for iterative update processing. When the number of samples increases, accordingly, the complexity of the algorithm grows, and the resampling process is prone to impoverishment. Besides, among the model-based methods, Chen et al. [29] proposed the H∞ filtering method and other adaptive methods [15, 30-32]. It is notable that these algorithms often construct a battery model. Although this enables higher accuracy state estimation, we know that the model construction does not characterize the internal microcosm of the battery well, resulting in less reliable estimation. Moreover, the algorithms involved are complex and computationally intensive, requiring strong arithmetic power, which requires more consideration in the process of use.

In contrast, AI methods do not require in-depth knowledge of the cell structure and simply obtain a large number of input and output samples with relevant characteristics from the target battery. This type of approach has attracted widespread attention from researchers. The main methods in this category include neural network (NN), fuzzy logic control, support vector machine (SVM), machine learning, etc. [33-37]. It is a class of AI approaches that leans on a large amount of data for learning and training and have a high level of self-learning capability. With the jump in the development of NNs, recurrent neural network (RNN) introduced the concept of sequences, such as time series and spatial sequences, which became one of the valuable methods to estimate the battery SoC [38, 39]. Feng et al. [40] proposed a new RNN structure, the clock recurrent neural network (CWRNN), for estimating the battery SoC. It specifies different pointer speeds for each independent hidden layer module, reducing training and computational costs and obtaining satisfactory estimation results. Yet, the method needs to obtain a large amount of upfront data related to the battery characteristics upfront, which is highly reliable data. Meanwhile, it cannot directly solve the problem of long-term series loss. To tackle this problem, Guo et al. [33] compared in detail the advantages and disadvantages of fully connected neural networks (FCNN), long short-term memory (LSTM), gate recurrent units (GRU), and transient convolutional networks (TCN) in computing SoC. The results show that LSTM, GRU, and TCN outperform FCNN in

precision. The LSTM network, evolved from RNN, compensates for the lack of memory for earlier sequences and directly solves the long-term dependence problem through the gate control structure. Although a large amount of data must be trained in the early stages, post-processing becomes simple, which effectively avoids the errors introduced by the second type of method in the linear processing problem and allows for high-accuracy online battery state estimation [41].

To cope with the above problems, this paper combines an AI method with an improved filtering algorithm in the form of fusion and proposes a synthetic SoC estimation method using a multi-hidden layer long short-term memory (MHLSTM) network with a suboptimal fading extended Kalman filtering (SFEKF). The main contributions of this paper can be summarized in the following three aspects that set it apart from published studies.

(1) A network structure with multiple hidden layers is devised for the LSTM network. The advantage of this structure is the multi-level abstraction over the input features and obtaining additional transformations of the input values. The MHLSTM network topology is matched with the actual battery measurement data. We reasonably initialize the weights to effectively avoid the zones where the gradients vanish.

(2) The SFEKF algorithm is introduced to enhance the resilience to noise and to vigorously track the filtering results. Using this methodology, we do not restrict the noise from forming unduly and can estimate and filter the target state and smooth the a priori results. Thus, it bridges the gap of EKF in the target tracking process, broadens its scope of application, and greatly boosts the accuracy of estimation and the robustness of the system.

(3) In the proposed synthetic SoC estimation method, we neither require an accurate equivalent circuit model for the battery nor overly model sophisticated parameters for the MHLSTM network. This greatly reduces the tedious problem of choosing hyper-parameters. The results show that the proposed algorithm exhibits good adaptability to different driving cycles. Thereby, the feasibility and effectiveness of the algorithm are verified.

The paper is organized as follows: Section 2 presents the basic concepts, the hyper-parameters selection of the

MHLSTM network, and the computational steps of the related algorithms in detail. Section 3 constructs the testbed required for the experiments and analyzes the experimental results in multiple dimensions. Section 4 summarizes the paper.

2. Methodology

2.1. Structure of the proposed MHLSTM network

To solve a series of serialized data modeling problems for efficient learning of nonlinear features on a temporal scale, RNN have been widely applied. However, their units suffer from gradient disappearance and explosion when dealing with long sequential data and only possess short-term memory [42, 43]. To tackle and forecast broader sequential information, a particular RNN structural unit, i.e., MHLSTM, is introduced. It constructs a temporal feedback network by adding memory neurons and refining the structure of hidden layer neurons to generate gradient long-term continuous flow paths, which effectively solves the long-term dependence problem and remedies the fact that RNNs do not have a memory function for earlier sequences to avoid severe memory data leakage. The principle structure of the MHLSTM network and the forward propagation process of the input data *Xt* in the structural cell are shown in Fig. 1.

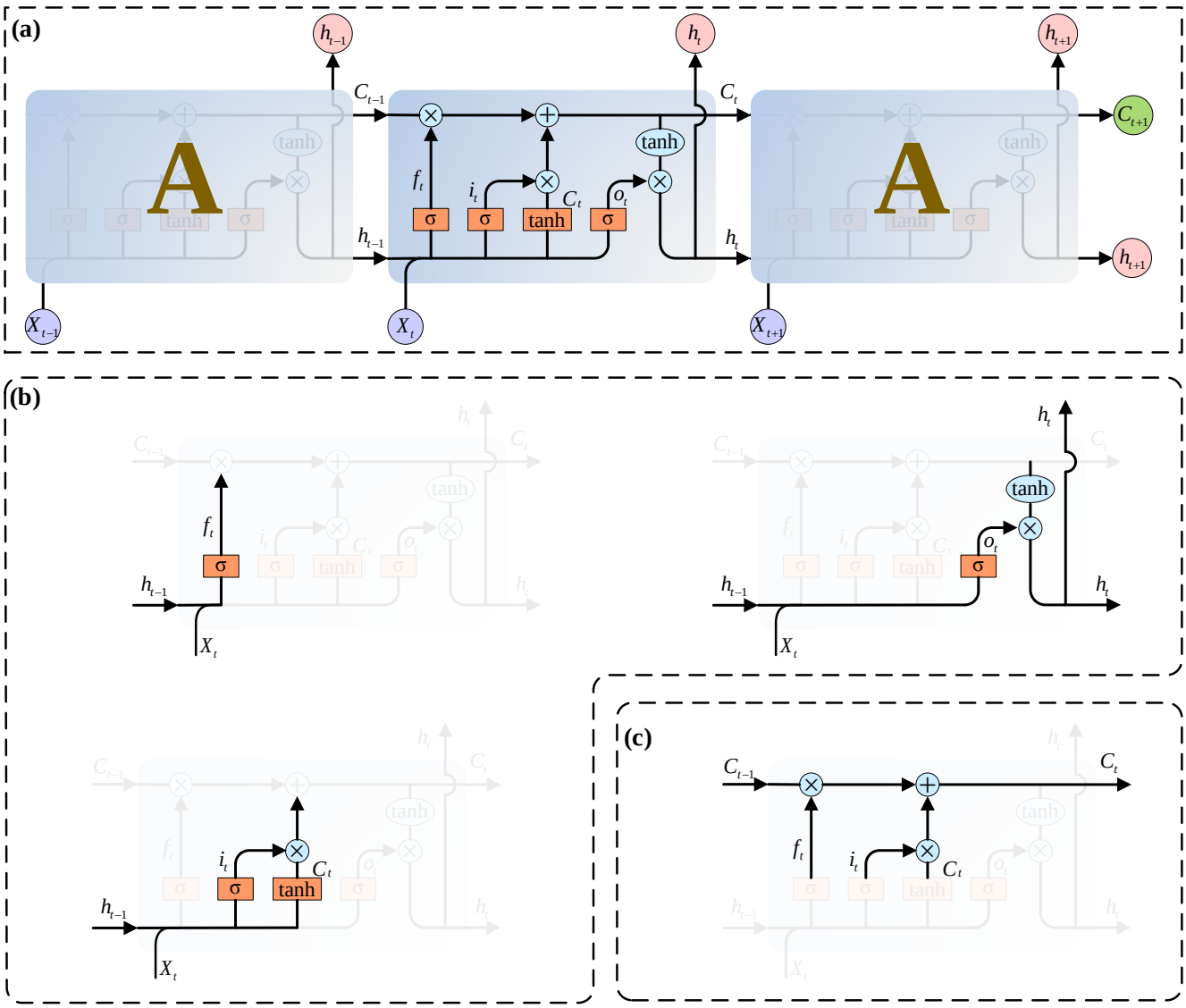


Fig. 1. MHLSTM network topology. (a) The principle structure of the MHLSTM network. (b) The data forward transmission process of the hidden layer. (c) The data forward transmission process of the memory cell.

In Fig. 1(a), the MHLSTM unit contains the memory cell C_t, the forgetting gate f_t, the input gate i_t and the output gate o_t, which are used to selectively remember the modified parameters of the error function with gradient descent for the feedback, thereby achieving control of the information flowing into the neuron. σ and *tanh* denote the Sigmoid function and the hyperbolic tangent function, respectively, and are calculated as shown in Eq. (1).

$$\begin{cases} \sigma(x) = \dfrac{1}{1+e^{-x}} \\ tanh(x) = \dfrac{1-e^{-2x}}{1+e^{-2x}} \end{cases} \tag{1}$$

The MHLSTM network acts as a decision maker for the increase or decrease of the cell state information whenever the history information moves to the next repetition of structure A. The structure of this interaction mode control information, which is the gate, is adopted by the Sigmoid activation function so that the information takes on a value mapping to the [0, 1] interval. 1 means that the information is retained intact, and 0 means that the information is forgotten. Where the forgetting gate can alleviate the conflicting information update problem, the last moment hidden layer state h_{t-1} and the current data Xt are taken as input to obtain the equation of forgetting gate f_t.

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \tag{2}$$

The input gate i_t is transmitted to the memory cell by controlling the effective amount of input information at the current moment, calculated as shown in Eq. (3).

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{3}$$

The output gate o_t exports the final message by controlling the valid information that the memory cell needs to transmit to the exterior at the current moment, which is calculated, as shown in Eq. (4).

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{4}$$

In the three gates controlling the information flow, W_f, W_i, and W_o indicate the weights of each gate layer state, and b_f, b_i, and b_o are the bias matrices in the generation procedure of each gate layer, respectively. The candidate memory cells of the MHLSTM cell utilize *tanh* as the activation function, as shown in Eq. (5).

$$\tilde{C}_t = tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{5}$$

In Eq. 5, W_C is the weight of the candidate memory cells, and b_C is the bias matrix in calculating the candidate memory cells. Thus, according to the data forward transmission process of the memory cell shown in Fig. 1(c), the cell state update at moment t is obtained, as in Eq. (6).

$$C_t = f_t C_{t-1} + i_t \tilde{C}_t \tag{6}$$

The ultimate state of the memory cell C_t at the current moment incorporates both the state C_{t-1} from the

previous moment and, at the same time, injects the newest information of the candidate memory cell. The flow of messages forward is controlled by this step through forgetting and input gates. Then, based on Eqs. (4) and (6), keep updating the information in the hidden state h_t to obtain the external output information.

$$h_t = o_t \cdot tanh(C_t) \tag{7}$$

In Eq. (7), the *tanh* layer updates the alternate information and determines the output value of the final hidden layer state as the input for the next moment to achieve the data generation and iterative update.

2.2. Suboptimal fading extended Kalman filtering algorithm

The derivation procedure of the EKF algorithm reinforces the role of the first-order term in the Taylor expansion, ignoring the effect of higher-order terms, which is less tolerant and may make the filtering scattered and biased [44]. Also, the EKF algorithm incorporates a Jacobian matrix, which makes the error in linearization processing large and the covariance prone to morbid. Hence, this paper introduces a strong tracking filtering (STF) algorithm to optimize the traditional EKF, so as to achieve strong tracking of system mutation.

The STF algorithm dynamically and adaptively adjusts the gain matrix by introducing a suboptimal fading factor in the a priori estimation of the error covariance calculation, resulting in excellent robustness and anti-interference capability, also known as an SFEKF. The derivation formula of the algorithm is as follows.

Step 1. Let k = 0, choose the initial value of state estimation $\hat{x}_{0|0}$ and the initial value of estimation variance $P_{0|0}$. Select the appropriate weakening factor β to make the state estimation smoother by reflecting the cumulative error of the filter.

$$\beta: min\left(\sum_{k=0}^{L}\sum_{i=1}^{n}\left|x_{i,k} - \hat{x}_{i,k|k}\right|\right) \tag{8}$$

In Eq. (8), L is the simulation step number.

Step 2. The state prediction values are obtained.

$$\hat{x}_{k+1|k+1} = \hat{x}_{k+1|k} + K_{k+1}\gamma_{k+1} \tag{9}$$

Step 3. Residual covariance array.

$$V_{k+1}=\begin{cases}\gamma_1{\gamma_1}^T, k=0\\ \dfrac{\rho V_k+\gamma_{k+1}{\gamma_{k+1}}^T}{1+\rho}, k\geq 1\end{cases} \tag{10}$$

In Eq. (10), ρ denotes the forgetting factor, which is a number in the interval of [0, 1], and is taken as ρ = 0.95.

Step 4. Calculate the time-varying suboptimal asymptotic elimination factor.

$$\begin{cases}\lambda_{k+1}=\begin{cases}\lambda_0, \lambda_0\geq 1\\ 1, \lambda_0<1\end{cases}\\ N_{k+1}=V_{k+1}-G_{k+1}\Gamma_k Q_k{\Gamma_k}^T{G_{k+1}}^T-\beta\cdot R_{k+1}\\ M_{k+1}=G_{k+1}F_k{F_k}^T{G_{k+1}}^T\\ \lambda_0=tr(N_{k+1})/tr(M_{k+1})\end{cases} \tag{11}$$

In Eq. (11), *tr* is the trace operation on the matrix, which is the sum of all elements on the main diagonal of the matrix.

Step 5. Obtain a priori estimates of the error covariance.

$$P_{k+1|k}=\lambda_{k+1}F_k P_{k|k}{F_k}^T+\Gamma_k Q_k{\Gamma_k}^T \tag{12}$$

Step 6. Calculate the gain matrix.

$$K_{k+1}=P_{k+1|k}{G_{k+1}}^T\left(G_{k+1}P_{k+1|k}{G_{k+1}}^T+R_{k+1}\right)^{-1} \tag{13}$$

Step 7. Obtain the final state estimate.

$$\hat{x}_{k+1|k+1}=\hat{x}_{k+1|k}+K_{k+1}\gamma_{k+1} \tag{14}$$

Step 8. Error covariance posterior estimates.

$$P_{k+1|k+1}=(I-K_{k+1}G_{k+1})P_{k+1|k} \tag{15}$$

Step 9. Repeat *steps* 2 to 8 until the state estimation is completed for all moments and the loop is jumped out.

The SFEKF algorithm compensates for the lack of adaptive tracking for the sudden changes of the system state in the traditional KF algorithm and employs the "predict-correct-update" principle to enhance the robustness of the system, thus improving the state tracking capability and the filtering effect.

2.3. Converged architecture for the synthetic SoC estimation

The SoC is a physical quantity that reflects how much energy remains in the battery. The current widely used definition of SoC is defined from a power perspective by the United States Advanced Battery Consortium manual for EVs. It consists of the ratio of the discharge loss rate multiplied by the remaining charge and its rated

capacity at a certain discharge multiplier, as shown in Eq. (16).

$$SoC(t) = SoC(t_0) \pm \int_{t_0}^{t} \frac{\eta_c \cdot i(\tau) d\tau}{C_n(T)} \tag{16}$$

In Eq. (16), the SoC value at moment t is defined as $SoC(t)$. $SoC(t_0)$ is the SoC value at the initial moment. $i(\tau)$ represents the current magnitude at the moment τ. η_c is the Coulomb efficiency parameter. $C_n(T)$ shows the maximum available capacity at the different temperatures. It is worth noting that the open-loop estimation method can only guarantee the estimation accuracy of the remaining battery power for a short time. Once the measured current or the initial value of SoC is inaccurate, the estimation error of SoC will gradually accumulate and become larger and larger in the long-term use of the battery.

LSTM, which takes memory neuron as its basic construction unit, is a typical time-series feedback network sharing the same weight, and has the advantages of learning long-term dependence and processing time series. Therefore, the introduction of the MHLSTM-SFEKF algorithm into the SoC estimation model can not only recall the memory of network states at different times, but also enhance the robustness of the system while avoiding the divergence caused by estimation bias. Therefore, the whole procedure is loosely divided into four components: data collection and partitioning of the collected datasets into training and testing datasets; iterative updating and inverse normalization; NN training on the data to achieve SoC estimation; and adding filtering applications to robustly estimate the SoC of the battery. The multi-algorithm synthetic SoC estimation framework is shown in Fig. 2.

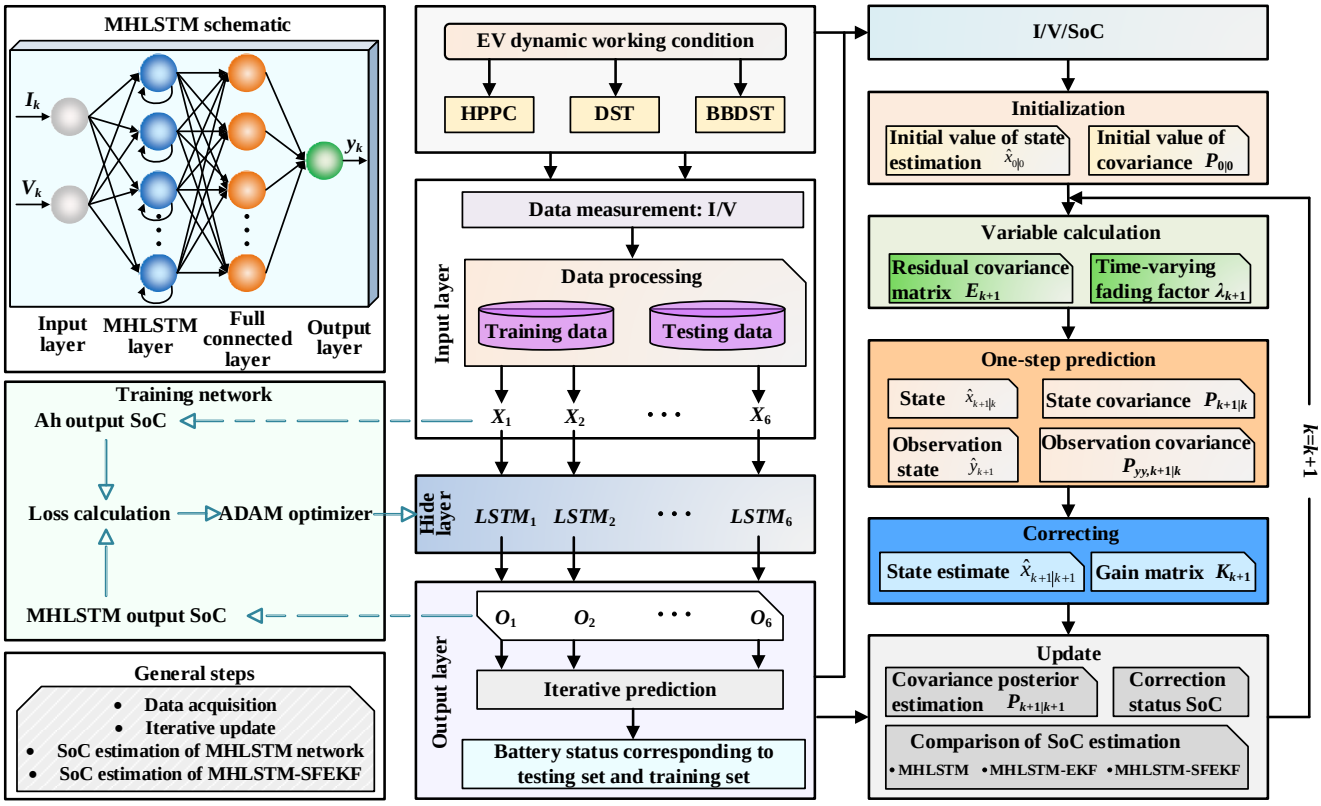


Fig. 2. Synthetic SoC estimation framework

As shown in Fig. 2, different charge and discharge tests are performed on the lithium-ion battery to obtain the real current and voltage data under different operating conditions, and the measured data are used as the global input for the synthesis of SoC estimation by the fusion algorithm. The input data are manipulated and partitioned into two data sets, which are the training and testing datasets. On this basis, a certain working condition data is fixed as a test set to train the working condition data under different situations, to verify the effectiveness of the MHLSTM network on data training, and to obtain the real-time SoC about the battery. Using deep learning, the data are trained to substitute the battery equivalent model and output the target state to determine the correlation between the measured values and SoC. The whole learning cycle is deemed a black-box model, thus reducing the bias caused by model parameter identification and improving the accuracy of battery state prediction. Besides, to achieve smoother and more stable SoC estimation, the STF method

application is integrated to further filter and optimize the output of the MHLSTM algorithm.

2.4. Performance evaluation indexes

To measure the excellent performance of the model in estimating the SoC from the battery, a quantitative description is needed to obtain evaluation metrics for the prediction class of models, except for the explicit trend of the data. In this paper, four evaluation criteria, maximum error (ME), mean absolute error (MAE), root mean squared error (RMSE), and mean absolute percentage error (MAPE), are used to better characterize the degree of model strengths and weaknesses, and the calculation formula is shown in Eq. (17).

$$\begin{cases} ME = max(|SoC'(t) - SoC(t)|) \\ MAE = \frac{1}{N}\sum_{i=1}^{N}|SoC'(t) - SoC(t)| \\ RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(SoC'^{(t)}(t) - SoC(t)\right)^2} \\ MAPE = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{SoC'(t) - SoC(t)}{SoC(t)}\right| \end{cases} \tag{17}$$

In Eq. (17), *SoC*(*t*) is the actual state value and *SoC*′(*t*) is the estimated value using the algorithm. In the analysis section of the paper, we will use basic and validation experiments to test the feasibility of the proposed algorithm and analyze the comprehensive performance of the SoC estimation.

3. Experimental test bench

This paper opts for a ternary lithium-ion battery, which combines the advantages of lithium cobaltate, lithium nickelate, and lithium manganate with high capacity, low cost, and good safety performance. The battery has the following properties: a rated capacity of 72 Ah, dimensions of 149*40*103 mm, an internal impedance of 0.6~0.7 mΩ, a lower cut-off voltage of 2.75 V, an upper cut-off voltage of 4.2 V, a constant current and voltage for charge and discharge testing. The battery was aged for 800 cycles.

To acquire the working environment data under different driving conditions for batteries, such as current, voltage, temperature, and other real data, it is essential to build an experimental test platform to collect the time-

varying data of each parameter, as shown in Fig. 3.

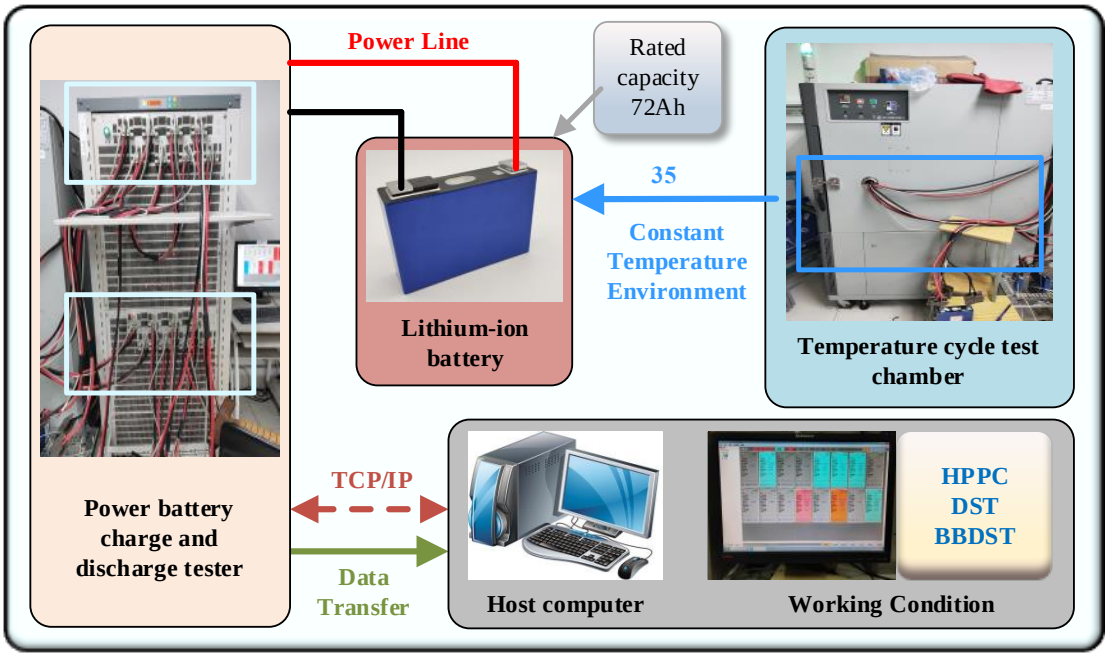


Fig. 3. Experimental test platform

The experimental equipment includes a power cell charge/discharge tester (CT-4016-5V100A-NTFA), temperature cycle test chamber (BTKS5-150C), monitoring upper computer, etc. Set the working condition steps under each driving condition in the host computer, drive the power cell charge/discharge tester to charge/discharge test the battery, collect it with a 0.1 s sampling frequency, obtain various data required for the study and transmit it back to the host computer. The temperature cycling test chamber provides a constant temperature environment of 35 ℃ for this experiment.

4. Results and discussion

4.1. Hyper-parameter design of the MHLSTM network

Before loading the training data, define the MHLSTM network structure parameters, normalize the pre-processing, and set the hyper-parameter values. MHLSTM network tuning parameters often affect the rate of deep learning in the model training course and the outcome achieved, including self-learning and adaptive ability to fit arbitrarily limited input-output mapping problems. The elements of the input data are 3, including I, V, and SoC, and the output data are set to be the SoC values. It is a multiple-input single-output network

structure.

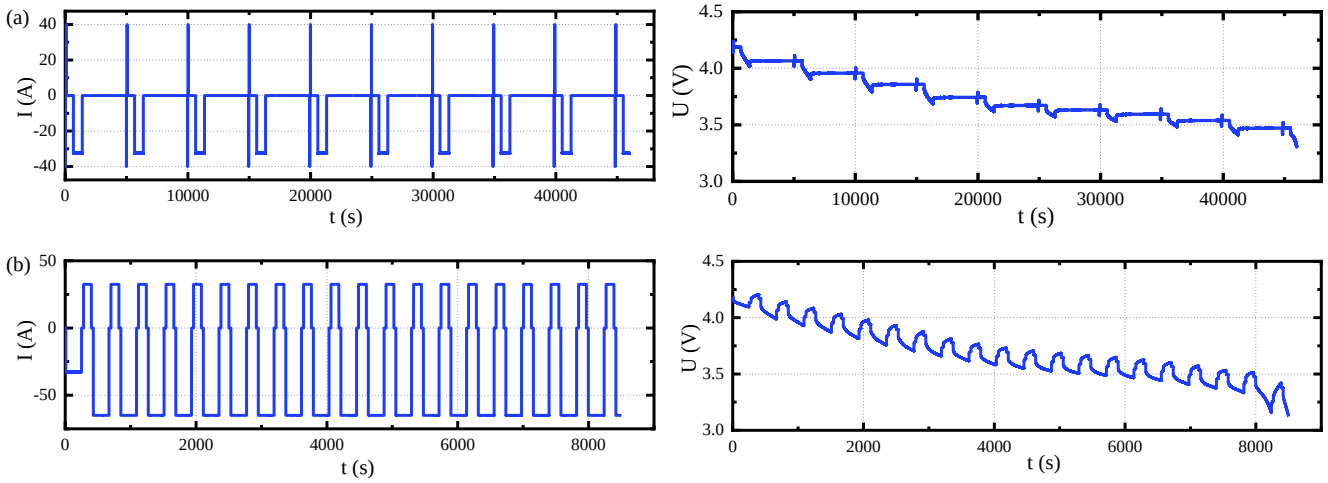
The architecture of the MHLSTM network requires the specification of several hyper-parameter values. The first one is the number of hidden layers. To perform better linear partitioning for different types of data, a multi-hidden layer MHLSTM is used, and the stacking of individual MHLSTM structures is set to 6 layers. A deep NN set to 6 layers outperforms a shallow NN, better approximates an arbitrary nonlinear mapping space, and can predict the SoC of the battery. Further, while increasing the number of hidden layers can solve complex tasks more efficiently, the large amount of data and arithmetic power make the computation too large and costly. Therefore, 6 layers are chosen based on the trade-off between computational accuracy and efficiency. Each layer is specified as an MHLSTM layer with 30 hidden units, the number of neurons in the hidden layer is 30, and the last MHLSTM layer joins the fully connected layer with a node count of 1. The second one is the gradient threshold. In MHLSTM networks with long input time series, gradient explosion may still occur. To guard against gradient explosion, gradient truncation is used, and the size of the gradient is limited to 1 during the learning procedure. The third one is the learning rate. Any NN model, which needs to avoid the occurrence of overfitting and underfitting, can be tweaked by the learning rate. A global learning rate of 0.01 is assigned to the parameter, and the weights are updated in accordance with the gradient at the end of each batch under control. The fourth hyper-parameter is to set the rotation of the training samples to 150 times. Also, we tried to use other values, but no dramatic change in network performance occurred. Therefore, the number of rounds is reduced as much as possible to reduce the training burden of the MHLSTM network, and 150 epochs are used. The number of samples between model weight updates is 32, i.e., the minimum step size is 32. Thus, we obtain the entire MHLSTM network structure.

The entire MHLSTM network hyper-parameters are set without traversing all the possible values but only roughly setting the various parameter values so that the training results are within an acceptable accuracy range. Then, we transfer the SoC estimates trained by the MHLSTM network to the SFEKF algorithm. Training tests

are executed under different driving conditions to further improve the SoC estimation accuracy and to enhance the robustness and generalization of the battery system. Then, we transfer the SoC estimates trained by the MHLSTM network to the SFEKF algorithm. Training tests are executed under different driving conditions to further improve the SoC estimation accuracy and enhance the robustness and generalization of the system.

4.2. SoC estimation with the MHLSTM network

Test experiments are conducted to provide real data with different work steps, to verify the feasibility of the algorithm and the strong adaptability features of the model in a dynamic environment. The hybrid power pulse characterization (HPPC) test verifies to verify the pulse power performance of battery charging and discharging at varying depths of discharge [45]. We customize the battery to conduct charge/discharge pulse test experiments every time the battery is discharged to 10% of its capacity and finish switching between different SoC taking values of the battery, so ten pulse cycles are gained. Highly accurate prediction of the remaining mileage during EV driving with reference to the custom dynamic stress test (DST) and Beijing Bus Dynamic Stress Test (BBDST) working conditions [46-48]. The experimental data are available by processing the actual operation of the EV during starting, accelerating, coasting, and braking. The current and terminal voltage profiles are plotted for the experimental duration are shown in Fig. 4.



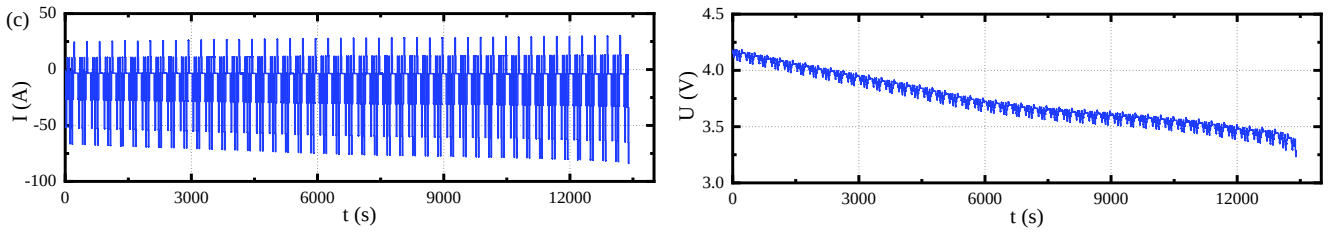
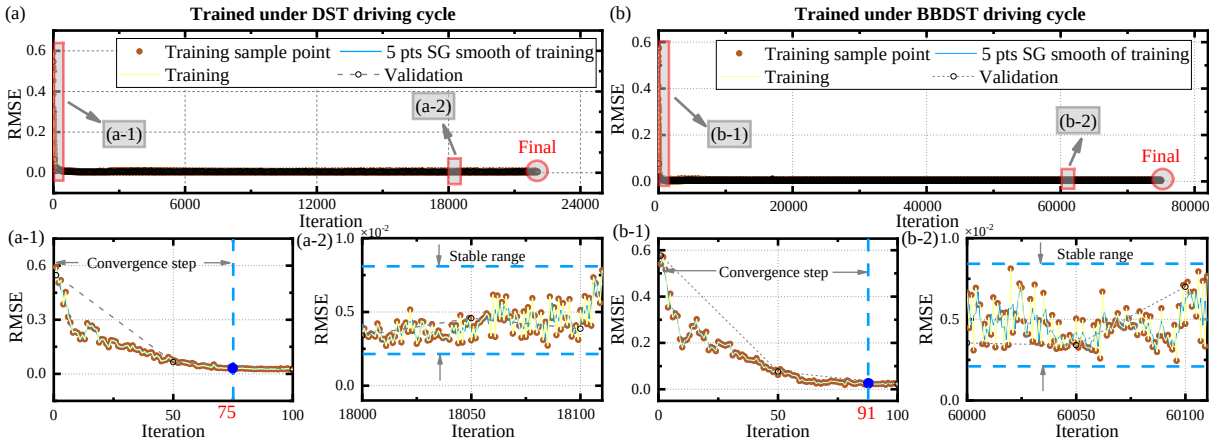
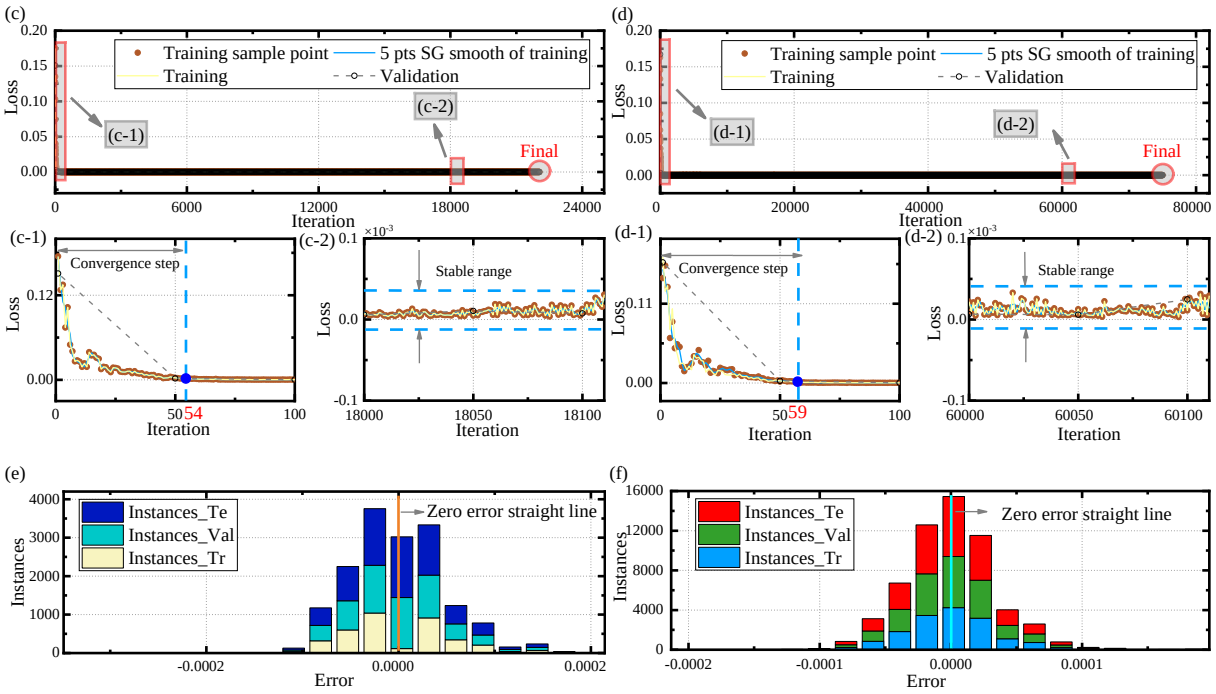


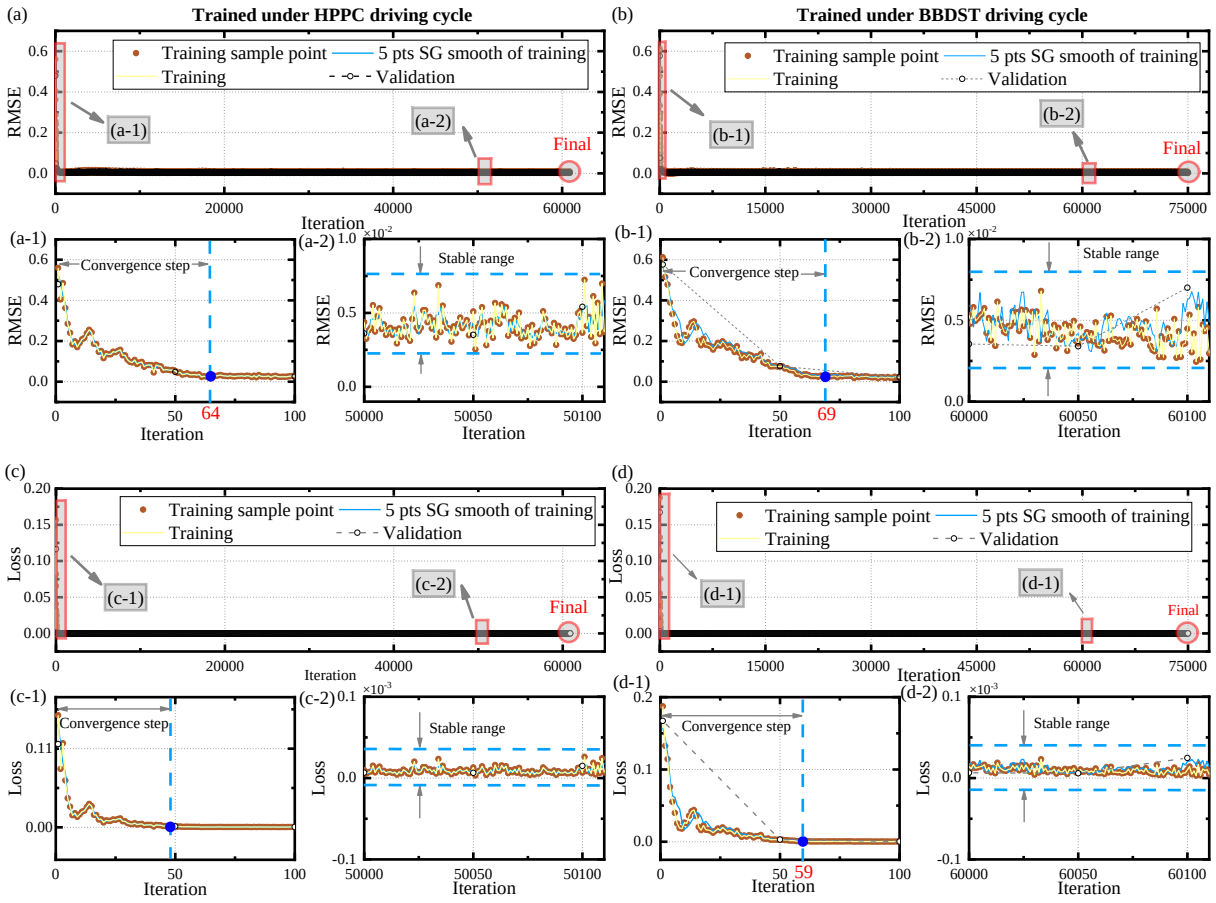
Fig. 4. Driving conditions simulating EV operation. (a) Current and terminal voltage curves under HPPC working condition. (b) Current and terminal voltage curves under DST working condition. (c) Current and terminal voltage curves under BBDST working condition.

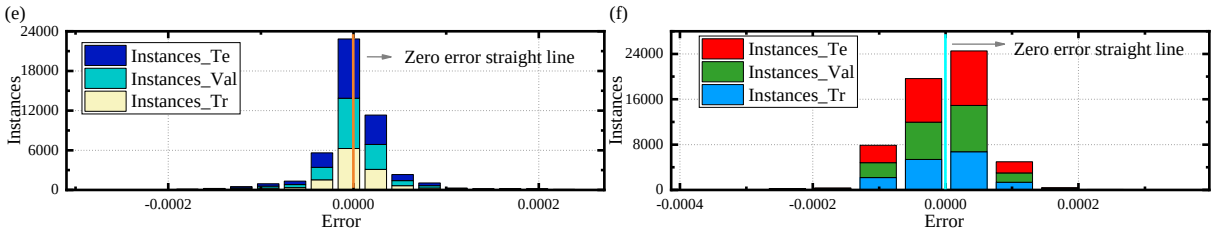
Aiming to verify the estimation effectiveness of the MHLSTM network for SoC, the analysis is performed by combining HPPC, DST, and BBDST working situations. The loss function can assess the inconsistency between the predicted and true values of the model to quantify how well the SoC prediction classifies the input data. During the training process, the loss is reduced as quickly as possible. While, on the one hand, it can reduce the iterative update of hyper-parameters, reduce the number of experiments, and shorten the time required for experiments, on the other hand, a suitable loss value can avoid over-fitting the network, thereby improving the accuracy and generality of the model. The loss error is inserted in the evaluation metrics to analyze the integrated performance of the network, as shown in Fig. 5.



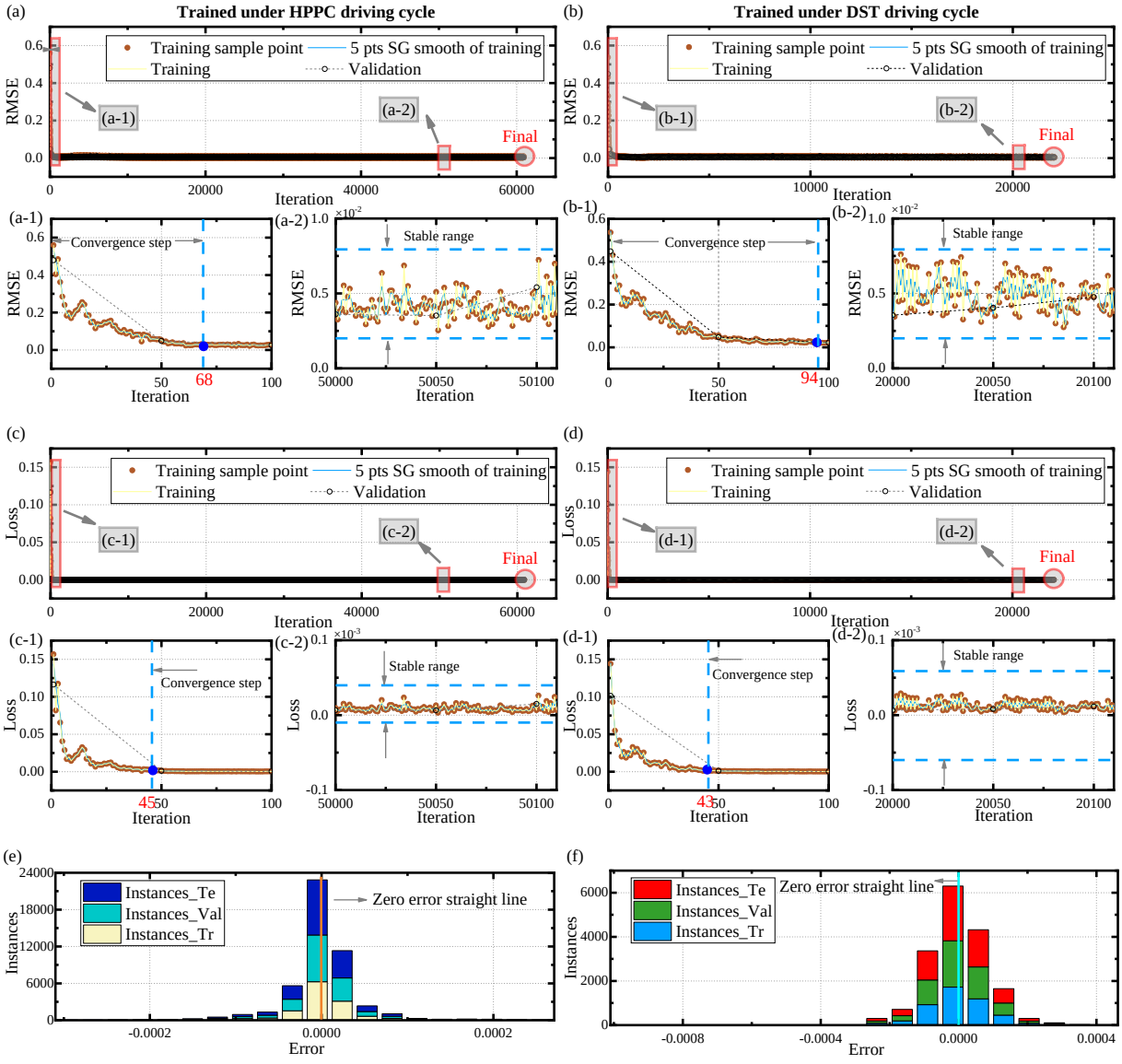


(A) Test set is HPPC cycle data. (a), (c), and (e) are the RMSE, Loss, and error histograms gained by training under the DST driving condition, respectively. (b), (d), and (f) are the RMSE, Loss, and error histograms gained by training under the BBDST driving condition, respectively.





(B) Test set is DST cycle data. (a), (c), and (e) are the RMSE, Loss, and error histograms gained by training under the HPPC driving condition, respectively. (b), (d), and (f) are the RMSE, Loss, and error histograms gained by training under the BBDST driving condition, respectively.



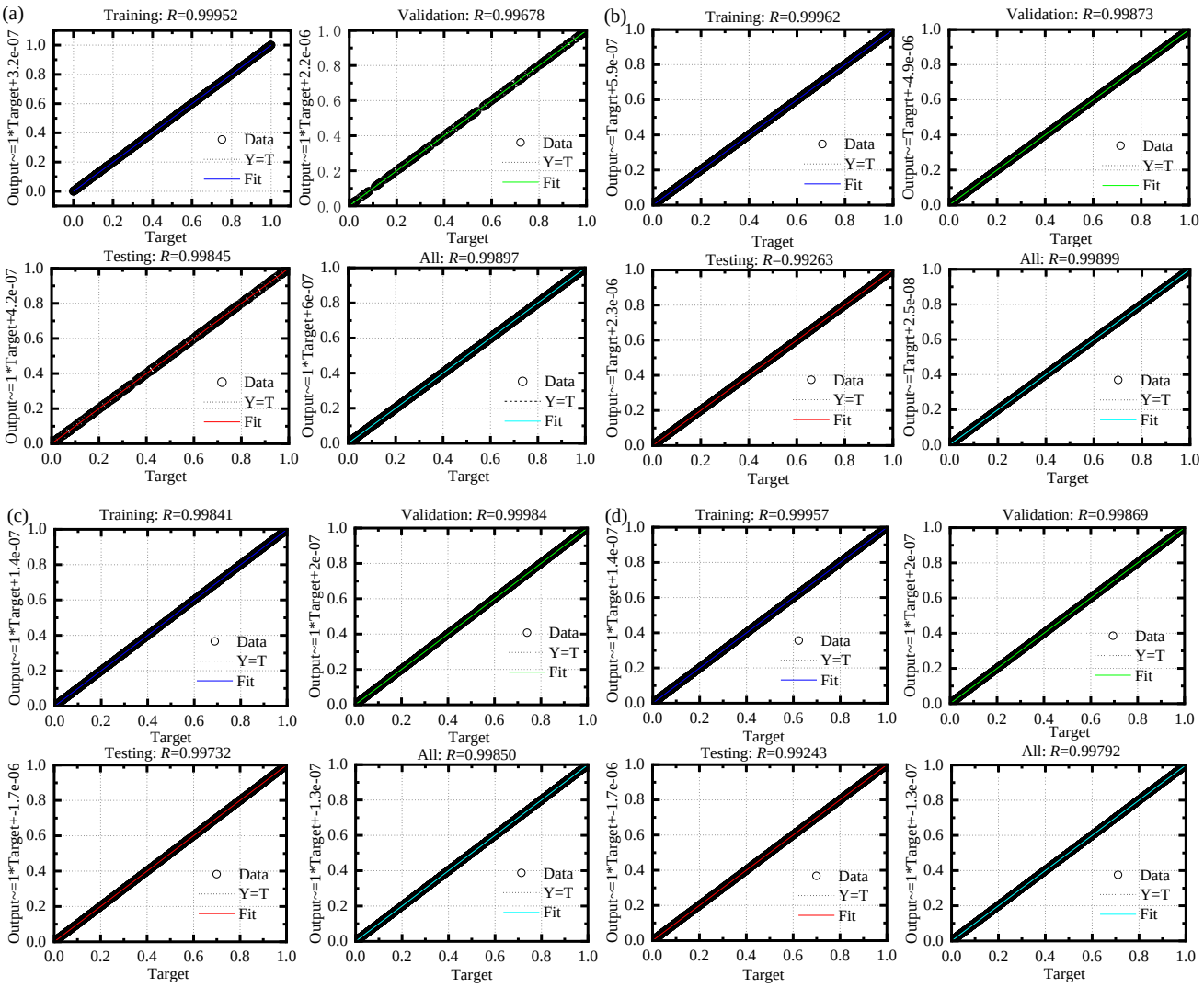
(C) Test set is BBDST cycle data. (a), (c), and (e) are the RMSE, Loss, and error histograms gained by training under the HPPC driving condition, respectively. (b), (d), and (f) are the RMSE, Loss, and error histograms gained by training under the DST driving condition, respectively.

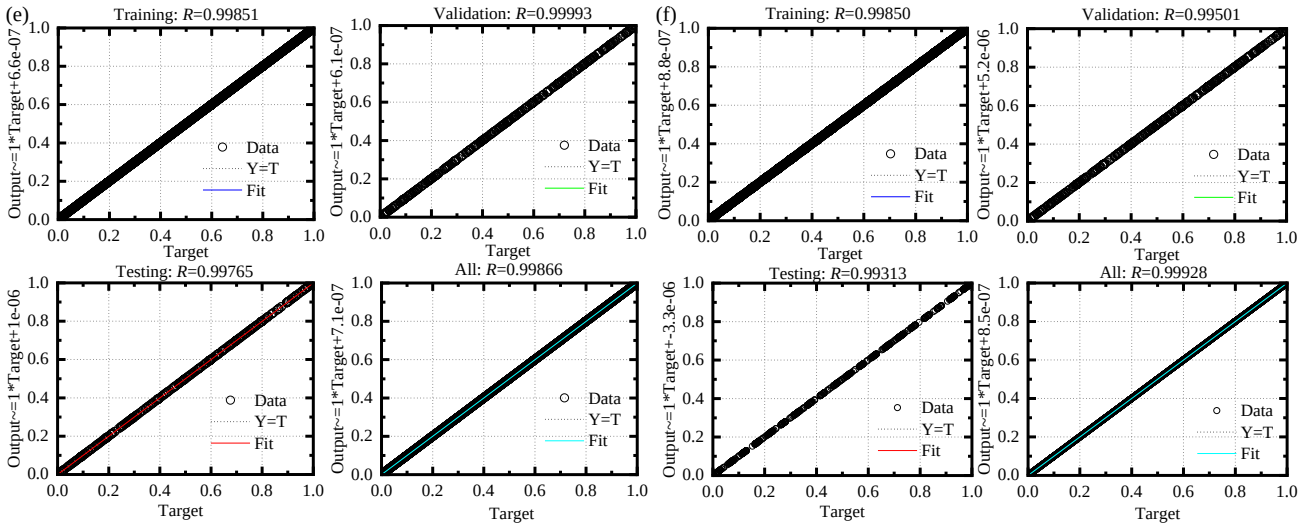
Fig. 5. Performance analysis of the MHLSTM network under different test conditions

First, HPPC driving cycles are used as the test set, while DST and BBDST driving cycles are taken as the training sets, respectively, to evaluate the training capability of the MHLSTM network. The training process is shown in Fig. 5(A). The diagram shows the RMSE and loss values for training and validation in a complex dynamic environment. In the DST cycle, the RMSE largely converges at the 75th iteration and the loss converges at the 54th iteration. In the BBDST cycle, the RMSE converges basically at the 91st iteration, and the loss converges at the 59th. The convergence is all done in the first 100 iterations of training, with training and validation changing for the same trend, achieving fast convergence of the model. Fig. 5(A-e/f) displays the error histogram, visualizing the error between the prediction and the target, which is basically concentrated in the [-0.0002, 0.0002] interval in the DST cycle. The accuracy improves in the BBDST cycle, mainly in the [-0.0001, 0.0001] interval. Similarly, we use the DST and BBDST cycles as test sets to obtain the training results illustrated in Figs. 5(B) and 5(C). From the graph, we can see that the training and validation have different distributions, but both are basically stable during the first 100 iterations. Surely, we are aware that the RMSE and loss values cannot be consistent under different dataset tests, but we can observe from the graph that they both maintain similar trends, demonstrating that the MHLSTM model is characterized by strong generalizability. Additionally, in the process of fulfilling the maximum number of rounds, all types of metrics achieve rapid convergence at an early stage. Therefore, balancing the computational complexity as well as the running time, the final iteration data may not be selected as the model parameters if the approximate SoC estimation accuracy requirements are met and the better-quality values are chosen by integrating the number of iterations and the loss values.

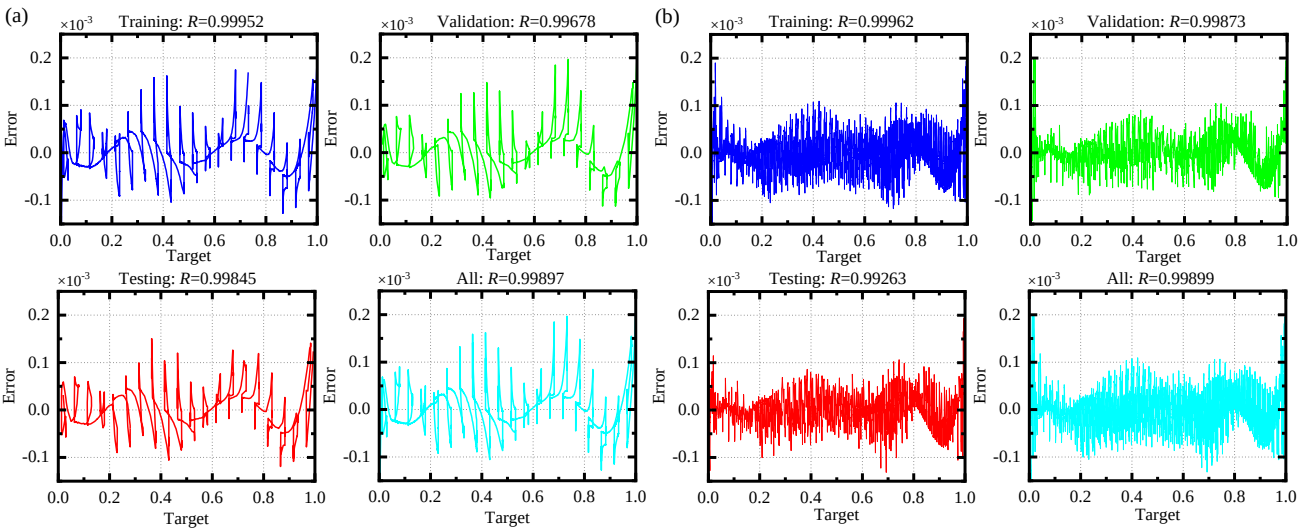
As shown in Fig. 5, which captures the trend of the data training process, it can be seen that the MHLSTM network shows good behavior under different training conditions. The losses obtained from training with different iteration steps under different test sets were not overfitted and underfitted. The loss values were able to converge to the minimum relatively quickly. When executing the script, the training loss is measured and

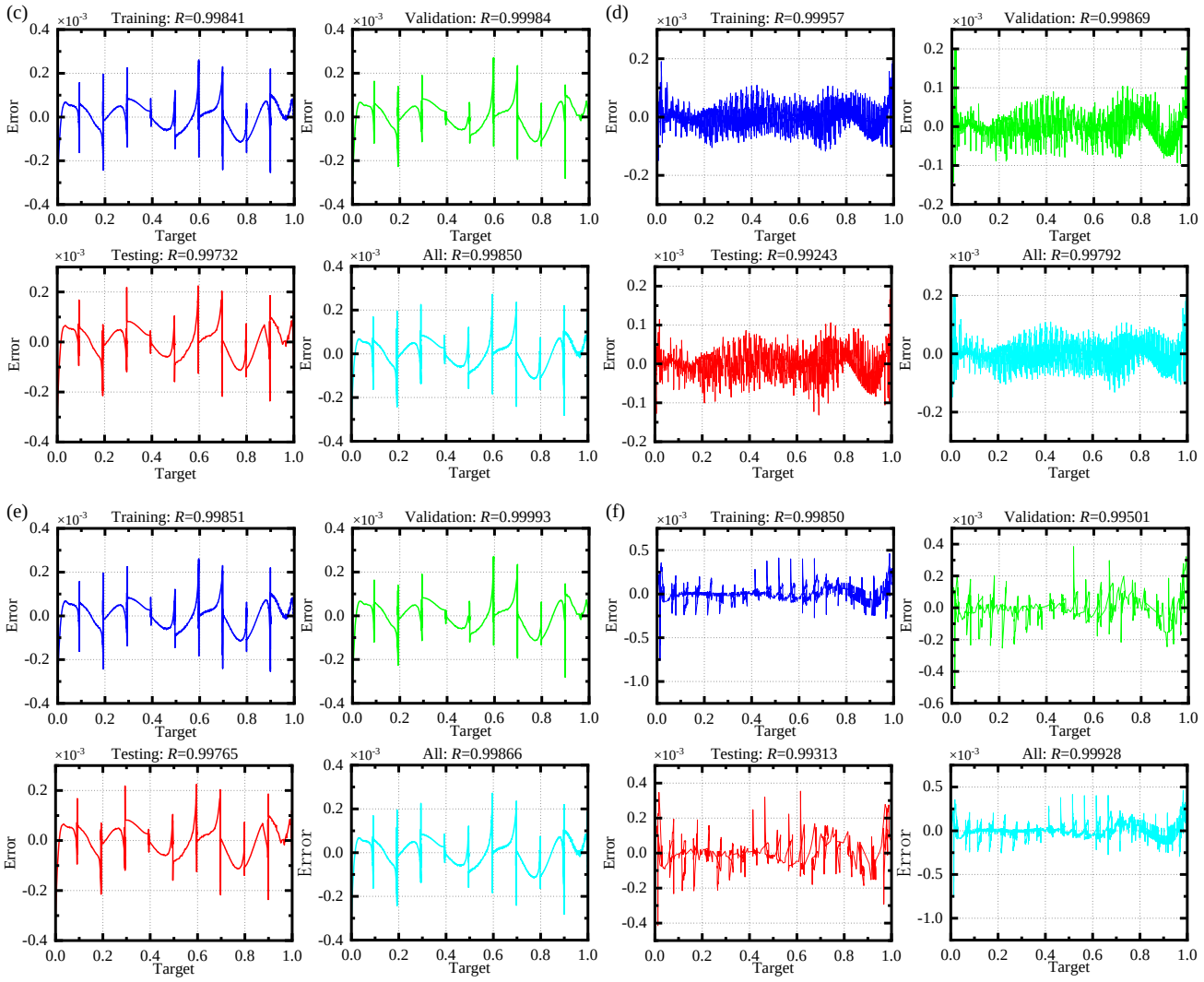
calculated within each epoch, while the validation loss is measured and calculated at the end of each epoch. This causes the loss of the validation set at the beginning of each iteration to be smaller than the loss of the training set. To further analyze the capability of the MHLSTM network in estimating the charge state. The *R* regression coefficient performance metrics are plotted, as shown in Fig. 6.





(A) Input output correlation curve. (a). The testing set is HPPC cycle data, and the training set is DST cycle data. (b). The testing set is HPPC cycle data, and the training set is BBDST cycle data. (c). The testing set is DST cycle data, and the training set is HPPC cycle data. (d). The testing set is DST cycle data, and the training set is BBDST cycle data. (e). The testing set is BBDST cycle data, and the training set is HPPC cycle data. (f). The testing set is BBDST cycle data, and the training set is DST cycle data.





(B) Linear regression error. (a). The testing set is HPPC cycle data, and the training set is DST cycle data. (b). The testing set is HPPC cycle data, and the training set is BBDST cycle data. (c). The testing set is DST cycle data, and the training set is HPPC cycle data. (d). The testing set is DST cycle data, and the training set is BBDST cycle data. (e). The testing set is BBDST cycle data, and the training set is HPPC cycle data. (f). The testing set is BBDST cycle data, and the training set is DST cycle data.

Fig. 6. MHLSTM network linear regression

The relevancy relationship of this method focuses on the training set, the test set, the validation set, and the overall learning results. Based on the linear regression curve shown in Fig. 6(A), the horizontal coordinate is the target output, and the vertical coordinate is the fit function between the prediction and the target output. The *R* coefficient is close to 1, which is a good indication of the strong correlation between the prediction and the target output. Depending on the training results of the MHLSTM network, each performance index of the

network can be derived, as shown in Table 1.

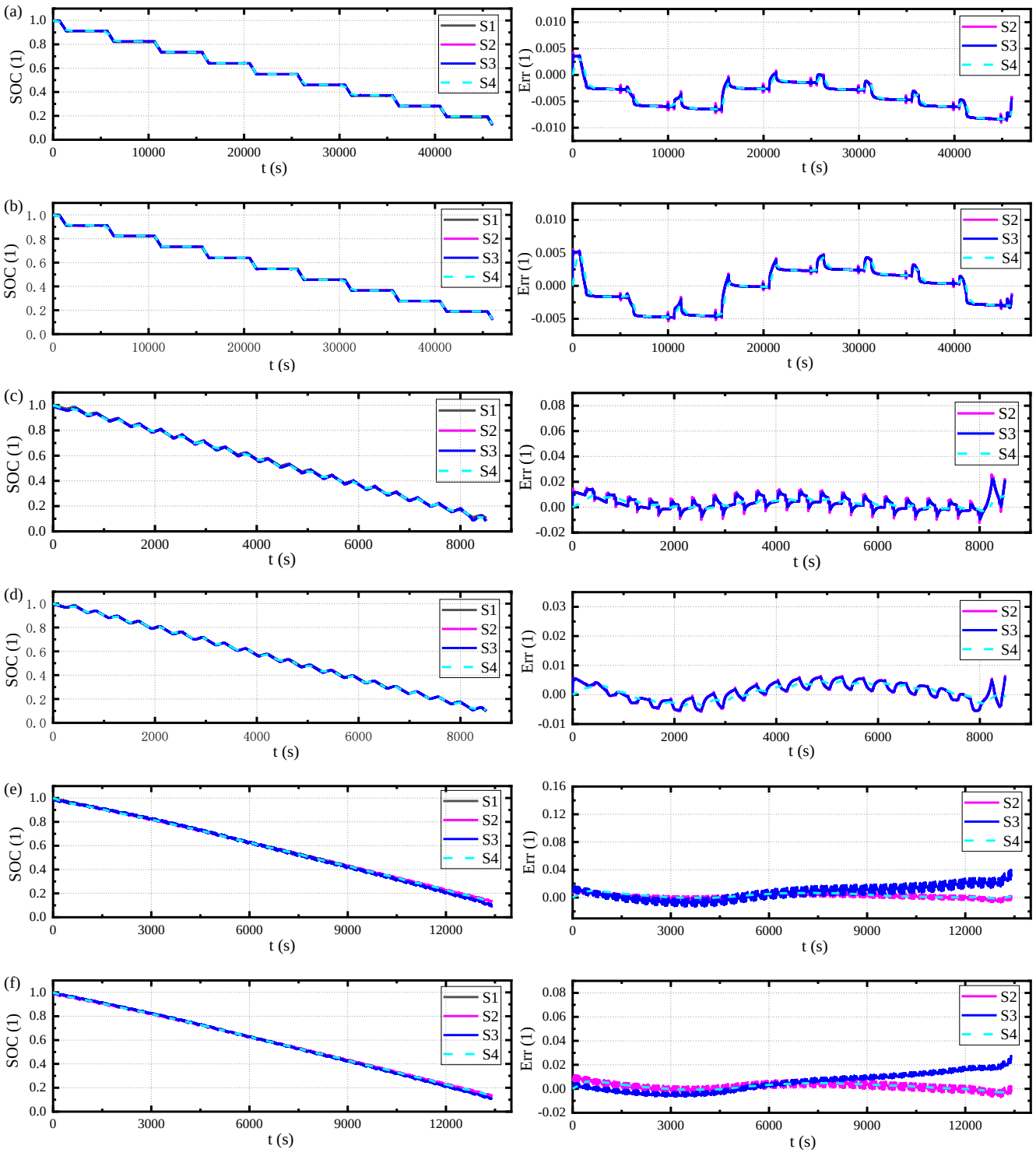
Table 1. Accuracy of SoC estimation after convergence

Driving Cycle Conditions	*RMSE*			*Loss*		
	HPPC	DST	BBDST	HPPC	DST	BBDST
HPPC	-	0.0045623	0.0035407	-	1.05E-05	6.52E-06
DST	0.0045679	-	0.0037296	1.07E-05	-	7.18E-06
BBDST	0.0040361	0.0042965	-	8.38e-06	9.40E-06	-
Driving Cycle Conditions	*Iteration*			*Learn Rate*		
	HPPC	DST	BBDST	HPPC	DST	BBDST
HPPC	-	22050	75000	0.01	0.01	0.01
DST	60900	-	75000	0.01	0.01	0.01
BBDST	60900	22050	-	0.01	0.01	0.01

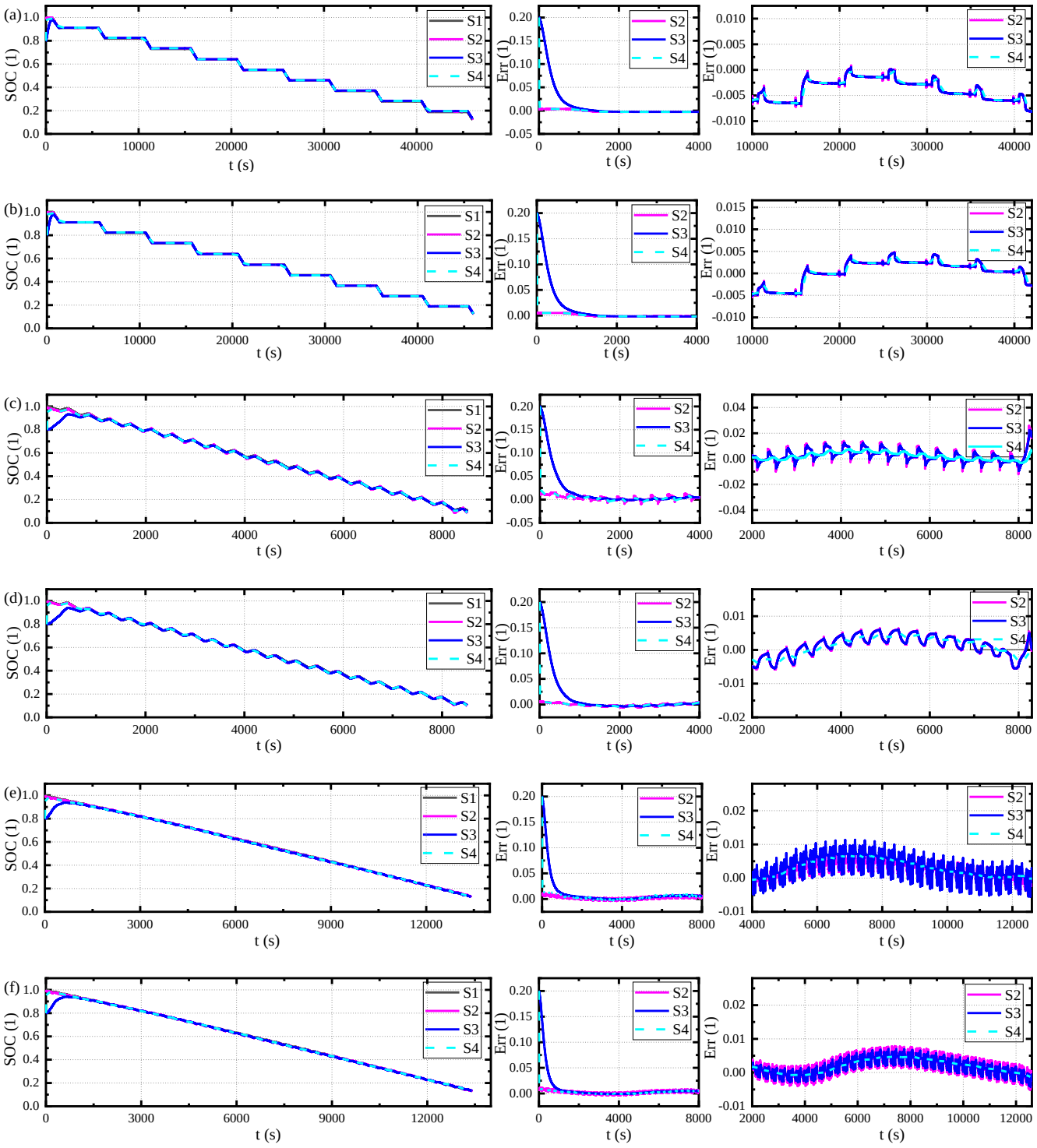
It is known from the data in Table 1 that the MHLSTM network can well emulate the experimental data under different test environments, suits the influence brought by different data, and anticipate the changing trend of SoC. The design of the MHLSTM network is often complicated and time-consuming in the choice of hyper-parameters. It takes continuous searching and training to settle on the best parameter values. Hence, we choose the MHLSTM parameters whose SoC estimation errors are within the tolerable scope to coarsely estimate the SoC values. The estimated SoC is utilized as the global input to the SFEKF algorithm for STF as a way to enhance the model estimation accuracy.

4.3. SoC estimation after SFEKF processing

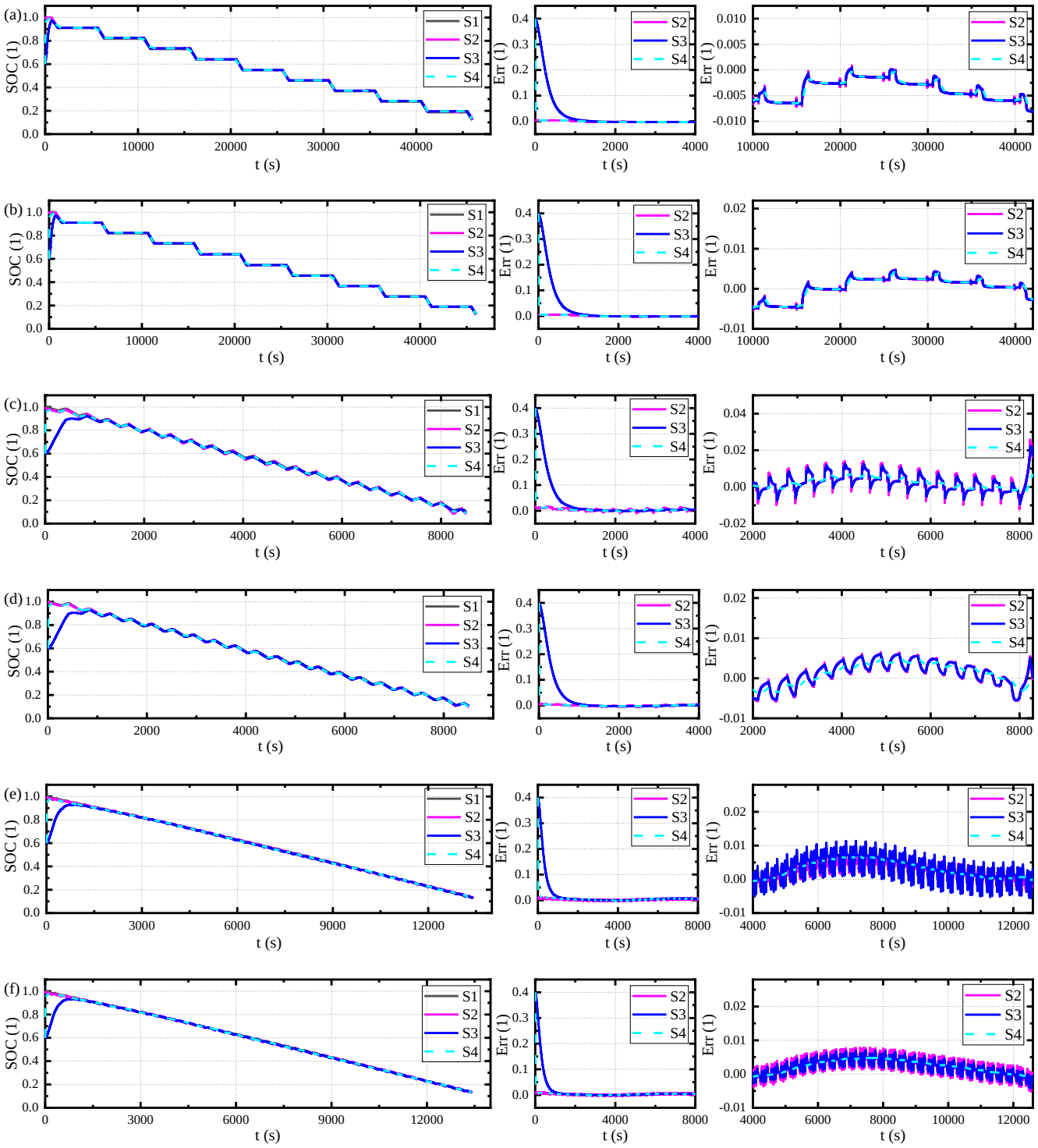
The outcome of the above MHLSTM network trained on the data is valid and can be used to obtain better SoC estimates. To further promote the accuracy and system stability of the model, the SFEKF algorithm is introduced to filter the MHLSTM model prediction results for a highly accurate expression of the remaining mileage. The custom three driving conditions cycle data along with the SoC values estimated by the MHLSTM network are input into the MHLSTM-SFEKF model. In this paper, the initial values of SoC are set as 1, 0.8, and 0.6, respectively, and the estimated curve and error tracking curve of synthetic SoC are obtained from the training data, as shown in Fig. 7.



(A) SoC₀=1. (a) The HPPC cycle data is used as the test set for the DST driving cycle data. (b) The HPPC cycle data is used as the test set for the BBDST driving cycle data. (c) The DST cycle data is used as the test set for the HPPC driving cycle data. (d) The DST cycle data is used as the test set for the BBDST driving cycle data. (e) The BBDST cycle data is used as the test set for the HPPC driving cycle data. (f) The BBDST cycle data is used as the test set for the DST driving cycle data.



(B) SoC$_0$=0.8. (a) The HPPC cycle data is used as the test set for the DST driving cycle data. (b) The HPPC cycle data is used as the test set for the BBDST driving cycle data. (c) The DST cycle data is used as the test set for the HPPC driving cycle data. (d) The DST cycle data is used as the test set for the BBDST driving cycle data. (e) The BBDST cycle data is used as the test set for the HPPC driving cycle data. (f) The BBDST cycle data is used as the test set for the DST driving cycle data.



(C) SoC$_0$=0.6. (a) The HPPC cycle data is used as the test set for the DST driving cycle data. (b) The HPPC cycle data is used as the test set for the BBDST driving cycle data. (c) The DST cycle data is used as the test set for the HPPC driving cycle data. (d) The DST cycle data is used as the test set for the BBDST driving cycle data. (e) The BBDST cycle data is used as the test set for the HPPC driving cycle data. (f) The BBDST cycle data is used as the test set for the DST driving cycle data.

Fig. 7 Synthetic SoC estimation curves and their error tracking curves under different test sets

As shown in Fig. 7, S1 denotes the real value, S2 denotes the MHLSTM algorithm, S3 denotes the MHLSTM-EKF algorithm, and S4 denotes the MHLSTM-SFEKF algorithm. It can be observed that the MHLSTM, MHLSTM-EKF and MHLSTM-SFEKF can all track the battery charge and discharge states well. All have strong generalization capabilities when estimating the battery SoC. In the visual analysis among the three algorithms, it is noticeably seen that the MHLSTM-SFEKF algorithm has higher reliability and a better ability to accommodate sudden changes in the system compared to the other two algorithms when dealing with long-term series problems.

4.4. Reliability assessment of fusion models

To analyze the battery SoC estimation results more intuitively, take SoC_0=1, and the outcomes of the three algorithms are compared. Four evaluation metrics, ME, MAE, RMSE, and MAPE, are employed to provide an in-depth analysis of the advantages offered by the fusion model. The comparison results of the evaluation metrics under different test sets are shown in Table 2.

Table 2. Reliability assessment of models

Driving Cycle Conditions		*ME*			*MAE*		
		MHLSTM	MHLSTM-EKF	MHLSTM-SFEKF	MHLSTM	MHLSTM-EKF	MHLSTM-SFEKF
HPPC	DST	0.00905657	0.008693766	0.008427263	0.004191676	0.004192406	0.00414684
	BBDST	0.005526002	0.005264787	0.004910026	0.002397089	0.002396182	0.002356223
DST	HPPC	0.025721108	0.022336386	0.009881843	0.004541591	0.004488086	0.003257627
	BBDST	0.006774908	0.006137341	0.004919091	0.002697396	0.002685914	0.002350432
BBDST	HPPC	0.011308024	0.039534331	0.00985654	0.002758152	0.009960662	0.003209137
	DST	0.011308024	0.027415913	0.006497557	0.002758152	0.007494755	0.002452616
Driving Cycle Conditions		*RMSE*			*MAPE*		
		MHLSTM	MHLSTM-EKF	MHLSTM-SFEKF	MHLSTM	MHLSTM-EKF	MHLSTM-SFEKF
HPPC	DST	2.24951E-05	2.25007E-05	2.20844E-05	0.011429225	0.011432607	0.011335299
	BBDST	7.95973E-06	7.95296E-06	7.64219E-06	0.00516154	0.005160729	0.005089728
DST	HPPC	3.56753E-05	3.4669E-05	1.71277E-05	0.012236141	0.012040334	0.00760301
	BBDST	1.00805E-05	9.9753E-06	7.17727E-06	0.006601494	0.006557442	0.005436701
BBDST	HPPC	1.17955E-05	0.000147395	1.75783E-05	0.00569677	0.031519024	0.005611608
	DST	1.17955E-05	8.73284E-05	9.06914E-06	0.00569677	0.024643695	0.005101305

Notes: *ME*, *MAE*, *RMSE* and *MAPE* are commonly used evaluation indicators in regression algorithms.

Table 2 summarizes the numerical analysis results of the MHLSTM method, EKF, and SFEKF integrated with the MHLSTM method under different working conditions, respectively. The findings show that the constructed MHLSTM-SFEKF model obtains a substantial improvement in its comprehensive performance indexes, good stability, high robustness, and fast convergence to the true value compared with both the independent MHLSTM model and the MHLSTM-EKF model. As can be seen from the data in the table, the maximum values of the four evaluation metrics are 2.572%, 0.0454%, 0.0035%, and 1.2236% when the battery SoC is estimated by the independent MHLSTM algorithm when tested under different operating conditions. The MHLSTM-EKF algorithm has a maximum of 3.953%, 0.996%, 0.0147%, and 3.152% for the four categories of metrics, respectively. The MHLSTM-SFEKF algorithm has a maximum of 0.998%, 0.415%, 0.0022%, and 1.134% for the four types of targets, respectively. Throughout the training, it appears that the MHLSTM-EKF algorithm error is larger than that of the independent MHLSTM algorithm. It is due to the scattered and biased estimation when the EKF algorithm is used for the filtering process under BBDST driving conditions. During the EKF linearization operation, the calculation of the covariance matrix appears pathological, leading to increased errors and system instability. A comprehensive study of three different algorithms for estimating the SoC reveals that the prediction results of the MHLSTM-SFEKF model have smaller errors and smoother and more stable curves.

For further analysis of Table 2, when using the DST driving cycle data as the test set, the MHLSTM-SFEKF obtained the greatest ME evaluation index with 0.988% of the synthetic SoC value under HPPC conditions. The maximum skew is reduced by 1.584% compared to the stand-alone MHLSTM method. When contrasted with the MHLSTM-EKF method, the maximum bias is lessened by 1.246%. Concerning the three evaluation criteria, MAE, RMSE, and MAPE, the maximum deviations estimated by the MHLSTM-SFEKF algorithm all appear in the test set with HPPC driving cycle data and in the training set with DST driving cycle data, with error values of 0.415%, 2.21*10-3%, and 1.134%, respectively. Because HPPC tests using the power switching cyclic pulse

test method abruptly give a pulsing shock, making sudden changes in voltage, the phenomenon of voltage plunges and rises, leading to an increase in error. In addition, because a 0.1 s interval is used for sampling, the data are numerous and the inconsistency is high, which indirectly increases the estimation bias. Also, as can be seen in both Fig. 7(A-a/b), the HPPC condition itself causes an increase in the estimated initial error, which increases the error magnitude to some extent.

Overall, from Fig. 7 and Table 2, the maximum deviation of the MHLSTM-SFEKF algorithm to estimate the synthetic SoC is limited to 1.2%, which means that the filtering of the MHLSTM network results is effective. The introduction of the SFEKF algorithm not only achieves high-precision synthetic SoC estimation in the case of abrupt system changes but also maintains a very strong tracking capability, which enhances the robustness and versatility of the system.

5. Conclusion

In this paper, a synthetic SoC estimation is proposed by combining an MHLSTM network with an SFEKF algorithm to solve the long-term sequence problem and prevent serious memory data leakage using traditional AI algorithms. To mitigate the gradient disappearance or gradient explosion phenomenon and capture the semantic association between long time sequences, the MHLSTM network is used to obtain approximate SoC estimation. Without a deep dissection of the internal electrochemical reactions of the battery, model errors caused by discrepancies and parameter errors caused by the identification of equivalent model parameters are avoided. The maximum ME, MAE, RMSE, and MAPE of the approximate SoC values obtained by the individual MHLSTM algorithm were 2.572%, 0.0454%, 0.0035%, and 1.2236%, respectively, which were able to stay within 3%. Aiming at raising the prediction accuracy of SoC and speeding up the global convergence, the training results of the MHLSTM network are filtered with unbiased parameters to smooth the algorithm estimation results and remove the noise effects. The suboptimal asymptotic elimination factor is added to the filtering algorithm to achieve stronger robustness about the system uncertainty and a better tracking ability of

the system's SoC. The results show that the MHLSTM-SFEKF algorithm is trained using data collected from HPPC, DST, and BBDST cycles, adaptively adjusted to obtain online estimates of the unknown covariates. The maximum deviations of the fusion algorithm used to estimate the synthetic SoC are all limited to within 1.2%. For the independent MHLSTM and MHLSTM-EKF algorithms, the error is reduced by more than half. As a result, this work confirms that the proposed algorithm has strong generalization and obtains satisfactory results that provide a fruitful exploration for the accurate SoC evaluation.

Acknowledgments

The work is supported by the National Natural Science Foundation of China (No. 62173281, 61801407), Sichuan science and technology program (No. 2019YFG0427), China Scholarship Council (No. 201908515099), and Fund of Robot Technology Used for Special Environment Key Laboratory of Sichuan Province (No. 18kftk03).

References

1. Zhang, L.P., et al., *High-safety separators for lithium-ion batteries and sodium-ion batteries: advances and perspective.* Energy Storage Materials, 2021. **41**: p. 522-545.
2. Zhang, X.H., et al., *A review on thermal management of lithium-ion batteries for electric vehicles.* Energy, 2022. **238**.
3. Huang, B., et al., *Recycling of lithium-ion batteries: Recent advances and perspectives.* Journal Of Power Sources, 2018. **399**: p. 274-286.
4. Ma, S., et al., *Temperature effect and thermal impact in lithium-ion batteries: A review.* Progress In Natural Science-Materials International, 2018. **28**(6): p. 653-666.
5. Zhang, C., et al., *Anion-Sorbent Composite Separators for High-Rate Lithium-Ion Batteries.* Advanced Materials, 2019. **31**(21).
6. Li, J.L., et al., *From Materials to Cell: State-of-the-Art and Prospective Technologies for Lithium-Ion Battery Electrode Processing.* Chemical Reviews, 2022. **122**(1): p. 903-956.
7. Ouyang, T.C., et al., *Coestimation of State-of-Charge and State-of-Health for Power Batteries Based on Multithread Dynamic Optimization Method.* Ieee Transactions on Industrial Electronics, 2022. **69**(2): p. 1157-1166.
8. Bian, X.L., et al., *State-of-Health Estimation of Lithium-Ion Batteries by Fusing an Open Circuit Voltage Model and Incremental Capacity Analysis.* Ieee Transactions on Power Electronics, 2022. **37**(2): p. 2226-2236.
9. Xu, P.H., et al., *State-of-Charge Estimation for Lithium-ion Batteries Based on Fuzzy Information Granulation and Asymmetric Gaussian Membership Function.* Ieee Transactions on Industrial Electronics, 2022. **69**(7): p. 6635-6644.
10. Qiao, D.D., et al., *Online quantitative diagnosis of internal short circuit for lithium-ion batteries using incremental capacity method.* Energy, 2022. **243**.
11. Liu, H.Q., et al., *Thermal issues about Li-ion batteries and recent progress in battery thermal management systems: A review.* Energy Conversion And Management, 2017. **150**: p. 304-330.
12. Xu, J.M., et al., *Studies of interfacial reaction characteristics for high power lithium-ion battery.* Electrochimica

Acta, 2022. **435**.

13. Chen, J.X., et al., *SOC estimation for lithium-ion battery using the LSTM-RNN with extended input and constrained output.* Energy, 2023. **262**.
14. Wang, S.L., et al., *A novel feedback correction-adaptive Kalman filtering method for the whole-life-cycle state of charge and closed-circuit voltage prediction of lithium-ion batteries based on the second-order electrical equivalent circuit model.* International Journal Of Electrical Power & Energy Systems, 2022. **139**.
15. Rodriguez-Iturriaga, P., et al., *A novel Dual Fractional-Order Extended Kalman Filter for the improved estimation of battery state of charge.* Journal Of Energy Storage, 2022. **56**.
16. Wang, S.-L., et al., *Open circuit voltage and state of charge relationship functional optimization for the working state monitoring of the aerial lithium-ion battery pack.* Journal Of Cleaner Production, 2018. **198**: p. 1090-1104.
17. Pan, H., et al., *State of charge estimation of lithium-ion batteries using a grey extended Kalman filter and a novel open-circuit voltage model.* Energy, 2017. **138**: p. 764-775.
18. Ren, Y.T., J. Ding, and T.Y. Hao, *GA-ELM-based adaptive Kalman estimator for SOC of lithium-ion batteries.* Ionics, 2022. **28**(12): p. 5377-5388.
19. Ren, L.C., et al., *Comparison of robustness of different state of charge estimation algorithms.* Journal Of Power Sources, 2020. **478**.
20. Sheng, C., et al., *Energy management strategy based on health state for a PEMFC/ Lithium-ion batteries hybrid power system.* Energy Conversion And Management, 2022. **271**.
21. Mao, S.Y., et al., *An Electrical-Thermal Coupling Model with Artificial Intelligence for State of Charge and Residual Available Energy Co-Estimation of LiFePO4 Battery System under Various Temperatures.* Batteries-Basel, 2022. **8**(10).
22. Yang, F., D.L. Shi, and K.H. Lam, *Modified extended Kalman filtering algorithm for precise voltage and state-of-charge estimations of rechargeable batteries.* Journal Of Energy Storage, 2022. **56**.
23. Zhao, X.B., et al., *State of charge estimation of lithium-ion battery based on multi-input extreme learning machine using online model parameter identification.* Journal Of Energy Storage, 2022. **56**.
24. Liu, E.H., et al., *Uncertainty Management in Lebesgue-Sampling-Based Li-Ion Battery SFP Model for SOC Estimation and RDT Prediction.* Ieee-Asme Transactions on Mechatronics, 2022.
25. Yang, X., et al., *A novel fuzzy adaptive cubature Kalman filtering method for the state of charge and state of energy co-estimation of lithium-ion batteries.* Electrochimica Acta, 2022. **415**.
26. Zhang, S.Z., et al., *State-of-charge estimation for lithium-ion battery during constant current charging process based on model parameters updated periodically.* Energy, 2022. **257**.
27. Chen, L.P., et al., *Adaptive state-of-charge estimation of lithium-ion batteries based on square-root unscented Kalman filter.* Energy, 2022. **252**.
28. Zheng, L.F., et al., *Differential voltage analysis based state of charge estimation methods for lithium-ion batteries using extended Kalman filter and particle filter.* Energy, 2018. **158**: p. 1028-1037.
29. Chen, Z.G., et al., *State-of-charge estimation of lithium-ion batteries based on improved H in fi nity fi lter algorithm and its novel equalization method.* Journal Of Cleaner Production, 2021. **290**.
30. Pang, H., et al., *A Composite State of Charge Estimation for Electric Vehicle Lithium-Ion Batteries Using Back-Propagation Neural Network and Extended Kalman Particle Filter.* Journal Of the Electrochemical Society, 2022. **169**(11).
31. Jiang, B., et al., *An adaptive capacity estimation approach for lithium-ion battery using 10-min relaxation voltage within high state of charge range.* Energy, 2023. **263**.
32. Guo, R.H. and W.X. Shen, *Online state of charge and state of power co-estimation of lithium-ion batteries based on fractional-order calculus and model predictive control theory.* Applied Energy, 2022. **327**.
33. Guo, S.S. and L. Ma, *A comparative study of different deep learning algorithms for lithium-ion batteries on state-of-charge estimation.* Energy, 2023. **263**.

34. Liu, X.T., et al., *Bayesian information criterion based data-driven state of charge estimation for lithium-ion battery.* Journal Of Energy Storage, 2022. **55**.
35. Lin, Q.B., et al., *Simultaneous and rapid estimation of state of health and state of charge for lithium-ion battery based on response characteristics of load surges.* Journal Of Energy Storage, 2022. **55**.
36. Hannan, M.A., et al., *Toward Enhanced State of Charge Estimation of Lithium-ion Batteries Using Optimized Machine Learning Techniques.* Scientific Reports, 2020. **10**(1).
37. Li, K.Q., Y.J. Wang, and Z.H. Chen, *A comparative study of battery state-of-health estimation based on empirical mode decomposition and neural network.* Journal Of Energy Storage, 2022. **54**.
38. Shu, X., et al., *Stage of Charge Estimation of Lithium-Ion Battery Packs Based on Improved Cubature Kalman Filter With Long Short-Term Memory Model.* Ieee Transactions on Transportation Electrification, 2021. **7**(3): p. 1271-1284.
39. Gong, Q.R., et al., *A Method for Estimating State of Charge of Lithium-Ion Batteries Based on Deep Learning.* Journal Of the Electrochemical Society, 2021. **168**(11).
40. Feng, X., et al., *State-of-charge estimation of lithium-ion battery based on clockwork recurrent neural network.* Energy, 2021. **236**.
41. Ren, X.Q., et al., *A method for state-of-charge estimation of lithium-ion batteries based on PSO-LSTM.* Energy, 2021. **234**.
42. Eddahech, A., et al., *Behavior and state-of-health monitoring of Li-ion batteries using impedence spectroscopy and recurrent neural networks.* International Journal Of Electrical Power & Energy Systems, 2012. **42**(1): p. 487-494.
43. Li, C.R., F. Xiao, and Y.X. Fan, *An Approach to State of Charge Estimation of Lithium-Ion Batteries Based on Recurrent Neural Networks with Gated Recurrent Unit.* Energies, 2019. **12**(9).
44. Xiong, B., et al., *Extended Kalman filter method for state of charge estimation of vanadium redox flow battery using thermal-dependent electrical model.* Journal Of Power Sources, 2014. **262**: p. 50-61.
45. Xiong, R., et al., *A Novel Joint Estimation Method of State of Charge and State of Health Based on the Strong Tracking-Dual Adaptive Extended Kalman Filter Algorithm for the Electric Vehicle Lithium-Ion Batteries.* International Journal Of Electrochemical Science, 2021. **16**(11).
46. Xu, H., et al., *A novel Drosophila-back propagation method for the lithium-ion battery state of charge estimation adaptive to complex working conditions.* International Journal Of Energy Research, 2022. **46**(11): p. 15864-15880.
47. Tang, H., et al., *Design of power lithium battery management system based on digital twin.* Journal Of Energy Storage, 2022. **47**.
48. Dang, X., et al., *Open-Circuit Voltage-Based State of Charge Estimation of Lithium-ion Battery Using Dual Neural Network Fusion Battery Model.* Electrochimica Acta, 2016. **188**: p. 356-366.

Declaration of interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

- A fusion model is presented for synthetic SoC estimation.
- The MHLSTM network topology is matched with the actual battery measurement data.
- The SFEKF algorithm is developed to adaptively smooth the MHLSTM model results.
- The accuracy of the model is verified by simulation tests on different data sets.
- The method features excellent generalization ability and can improve the estimation accuracy.

Credit Author Statement

Yanxin Xie: Conceptualization, Methodology, Writing - Original Draft. **Shunli Wang:** Data curation, Funding acquisition. **Gexiang Zhang**: Validation, Supervision. **Yongcun Fan:** Investigation, Methodology. **Carlos Fernandez**: Formal analysis, Writing - Review & Editing. **Frede Blaabjerg:** Software, Writing - Review & Editing.