



State-of-charge estimation of lithium-ion batteries based on gated recurrent neural network



Fangfang Yang ^{a, b}, Weihua Li ^b, Chuan Li ^a, Qiang Miao ^{a,*}

^a School of Mechanical Engineering, Dongguan University of Technology, Dongguan 523808, China

^b School of Mechanical and Automotive Engineering, South China University of Technology, Guangzhou, 510641, China

ARTICLE INFO

Article history:

Received 10 December 2018

Received in revised form

7 March 2019

Accepted 9 March 2019

Available online 11 March 2019

Keywords:

Lithium-ion batteries

State-of-charge estimation

Recurrent neural network

Gated recurrent unit

Ambient temperature

ABSTRACT

Accurate state-of-charge (SOC) estimation, which is critical to ensure the safe and reliable operation of battery management systems in electric vehicles, is still a challenging task due to sophisticated battery dynamics and ever-changing ambient conditions. In contrast to model-based SOC estimation methods, whose performance rely heavily on the quality of battery models, neural network-based methods are purely data-driven and model-free, and can be easily extended. Recently, with the ever-increasing computing power provided by graphic processing units, the neural network-based methods have gained more and more attentions. In this paper, a recurrent neural network with gated recurrent unit is proposed to estimate the battery SOC from measured current, voltage, and temperature signals. Compared with traditional feed-forward neural networks, the proposed method exploits information of the previous SOCs and measurements and yields better estimation accuracy. The proposed method presents satisfying estimation results on data collected from two mainstream lithium-ion batteries under dynamic loading profiles. Moreover, the proposed method is robust against unknown initial SOC values and can be trained to learn the influence of ambient temperatures. The proposed method can estimate the SOC at varying temperatures with root mean square errors within 3.5% and works under untrained temperatures.

© 2019 Elsevier Ltd. All rights reserved.

1. Introduction

Due to the energy crisis and the global commitment to reduce greenhouse gas emissions, the electric vehicle (EV) has become one of the most promising candidates to gradually replace traditional engine vehicles [1]. Lithium-ion batteries, which feature a high energy/power density and extended cycle life, have been recognized as the main energy storage device for EVs [2]. An advanced battery management system (BMS) that can effectively monitor the battery status is therefore of the utmost importance to ensure safe, reliable, and efficient operations of EVs. State-of-charge (SOC), as one of the key status in the BMS, quantifies remaining energy of a battery at the present cycle and indicates how long the battery can sustain before recharge is needed [3]. Accurate knowledge of the SOC is necessary for EV range determination and trip planning, and it is particularly helpful for the BMS to guarantee cells working

within safe operating windows and to extend cycle life. However, direct measurement of the SOC is not possible, and it can only be inferred from measurable cell parameters such as current and voltage. Accurate SOC estimation is still a challenging task due to complex battery dynamics as well as varying operation conditions, such as ambient temperature, self-discharge rate, hysteresis, regeneration, and cell aging [4].

The existing methods reported for SOC estimation mainly fall into the following categories. (i) The open-circuit voltage (OCV) method approximates the SOC via the monotonic relationship between the SOC and its OCV [5]. Measuring the OCV requires the battery to be disconnected from the loads for a period longer than 2 h to reach a steady state, thus the method is not suitable for on-board implementation. (ii) The Coulomb counting method estimates the battery SOC by integrating the discharge current over time [6]. It can be easily implemented, but this open-loop estimation may be biased due to inaccurate SOC initialization. Moreover, the estimation error accumulates as a result of low sensor precision. (iii) The model-based filtering method, such as widely used extended Kalman filter (EKF), unscented Kalman filter (UKF), and

* Corresponding author.

E-mail addresses: yangfang619@dgut.edu.cn (F. Yang), whlee@scut.edu.cn (W. Li), chuanli@scut.edu.cn (C. Li), 2017831@dgut.edu.cn (Q. Miao).

particle filter (PF) [7], combines the previous two methods with nonlinear filtering techniques. A battery state-space model is needed to update SOC estimation with new measurements. The model-based method achieves acceptable estimation performance, but its performance depends heavily on the quality of battery models. For this reason, many models have been investigated. A simple model, a combined model, a zero-state hysteresis model, a one-state hysteresis model, and an enhanced self-correcting model were reviewed for SOC estimation of lithium-ion batteries [8]. A resistance-capacitance network based equivalent circuit model was investigated for SOC estimation of two types of lithium-ion batteries [9]. A temperature-based model was developed to estimate the SOC of lithium-ion batteries under different temperatures [10]. A partial differential equation-based model was proposed to capture the battery dynamics [11]. These models perform well in strict conditions, including constant temperature, specified battery type, etc. Researchers are still in pursuit of a comprehensive model that can integrate the complicated dynamic behaviors of electrochemical, electro-thermal, and degradation inside the battery under actual working environment. (iv) The data-driven method treats the battery as a pure black-box and identifies the nonlinear relationship between battery SOC and measurable variables (eg. current, voltage, temperature) directly with large amounts of training data. Salkind et al. [12] applied a fuzzy logic to estimate the SOC and the state-of-health of lithium-ion batteries. Anton et al. [13] employed a support vector machine to estimate the battery SOC from cell current, voltage, and temperature measurements. Kang et al. [14] proposed an artificial neural network (NN) to estimation the SOC of lithium-ion batteries under varying aging levels. This kind of method does not require an explicit battery model. Hence, temperature and other factors may be considered in a much easier way. On the other side, the estimation accuracy turns out to depend greatly on the quality and quantity of training data and the training process may take a lot of time.

Recently, with the ever-increasing computing power provided by graphic processing units (GPUs), the artificial NN-based methods have gained more and more attentions from the research world. Huge amounts of training data may be stored and collected either by continuously transferring battery field data from online BMSs to offline data servers, or by battery simulation tests using dynamic driving profiles. As to the training process, under a GPU environment, the training process on large amounts of data is shortened to several quarters. Once the model is trained, the estimation can be computed in few milliseconds, meeting the needs of onboard EV applications. He et al. [15] proposed an artificial NN to estimate the battery SOC from measured current and voltage and added a UKF behind the network to improve the estimation accuracy. Sahinoglu et al. [16] developed a recurrent NN (RNN) to estimate the SOC of lithium-ion batteries. Chaoui et al. [17] applied the RNN to estimate the SOC and the state-of-health of lithium-ion batteries and evaluated its performance using two different kinds of batteries. The experimental results showed that the RNN is robust to battery's aging, hysteresis, dynamic current profile, nonlinear dynamic nature, and parametric uncertainties. Compared with traditional feed-forward NNs, the RNN with recurrent units makes use of historical information, therefore shows more power on battery SOC estimation, as the present battery SOC strongly relates to previous SOCs and measurements. Nonetheless, RNN itself has been revealed to be incapable of capturing long-term dependencies, as a result of the so-called gradient vanishing phenomenon, which emerges in the traditional back-propagation training process [18].

The gated recurrent unit (GRU) was proposed by Cho et al. to deal with the gradient vanishing problem in [19], where the GRU-RNN architecture was adopted to learn the mapping from one

sequence to another in the phrase-based English/French statistical machine translation system. The new architecture improved both the memory capacity and the ease of training, and finally improved the overall translation performance. In contrast to standard RNN forcing the recurrent unit to update in an exponential-moving-average way, the GRU employs a reset gate and an update gate to decide what to forget and what to remember. By using gating mechanism, the reset gate drops unwanted information and the update gate propagates useful information from previous hidden states. In other words, the GRU identifies information essential to estimate the current status and is therefore capable of addressing long-term dependencies. Besides machine translation, the GRU has been successfully applied to other time-series applications, such as speed recognition [20] and healthcare [21].

In this paper, we propose to apply the GRU-RNN for SOC estimation of lithium-ion batteries. Battery variables such as voltage, current, and temperature are used as the input to train the GRU-RNN network, while the SOC is used as the network output. A step-by-step searching strategy is provided to find the optimal GRU structure for SOC estimation. The impact of network parameters on SOC estimation is explored. The performance of the proposed estimation method is evaluated using data collected from dynamic testing profiles including dynamic stress test and federal urban driving schedule under different temperatures. The robustness of the network is investigated with respect to unknown initial states. To see the applicability of the proposed model, SOC estimation of different lithium-ion batteries are also considered. Experiments show that the proposed method yields satisfying SOC estimation accuracy under different ambient temperatures, even if no training data has been provided at that temperature.

The rest of this paper is outlined as follows. Section 2 presents the details of standard RNN and GRU and the implemented procedure. Section 3 introduces the experiment design and data collection. Results and discussions are drawn in Section 4. Section 5 finally gives a conclusion.

2. Gated recurrent neural network for online SOC estimation

This section first introduces the standard RNN and GRU in detail, and then presents the implementation procedure of SOC estimation using the GRU-RNN network.

2.1. Recurrent neural network

A traditional feed-forward neural network lacks the ability to use historical information, which appears critical in sequential information handling, such as phrase-to-phrase or sentence-to-sentence translation, generating titles for video input et al. These applications also feature variable input and output sizes, which cannot be handled by the conventional neural network as well.

Unlike feed-forward neural networks, RNN [22] uses recurrent units to store information of past information to handle sequential input. As shown in Fig. 1, the recurrent unit has a feedback loop, and if unfurled, the hidden state h_k is updated by

$$h_k = \begin{cases} 0, & k = 0 \\ \Phi(h_{k-1} | x_k), & \text{otherwise} \end{cases} \quad (1)$$

where x_k is a k -th input vector, and Φ is a nonlinear function such as composition of a logistic sigmoid with an affine transformation. Practically, the update of the hidden state in Equation (1) is implemented as

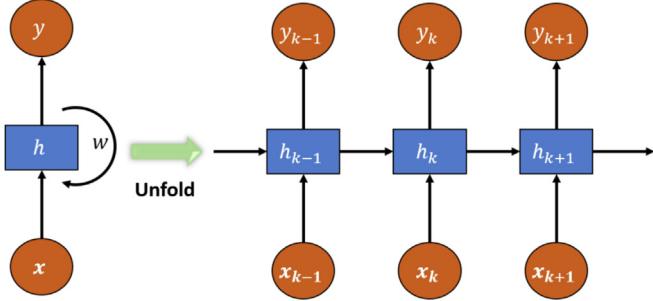


Fig. 1. Structure of the RNN.

$$h_k = g(W_h \cdot [h_{k-1} \ x_k] + b_h), \quad (2)$$

where g is a smooth, bounded activation function such as a logistic sigmoid function or a hyperbolic tangent function; W_h is a weight matrix between a hidden layer and an input layer; b_h is a bias parameter representing an offset of each node. Then the corresponding output y_k can be obtained by

$$y_k = g(W_y \cdot h_k + b_y). \quad (3)$$

The time-step recurrent edge makes RNN excels at handling tasks such as connected handwriting recognition or speech recognition. Nonetheless, as pointed out by Benigo et al. [18], when the sequence gets longer, it is harder to train an RNN to capture long-term dependency, as a result of gradient vanishing or rarely, gradient explosion. This phenomenon makes gradient-based training methods inapplicable. To settle this problem, there are two mainly explored directions. One direction focuses on improving traditional gradient-based learning methods, such as using simple gradient-clipping [23], identity initialization [24], second-order methods et al. The other direction aims at designing a new activation function that weakens gradient vanishing rate intrinsically. Long short-term memory (LSTM) unit [25] and gated recurrent unit (GRU) [19] are the main results in this direction. Compared to LSTM, GRU employs a simpler structure with less parameters and has been reported to outperform LSTM on smaller datasets [26], and is therefore investigated in this paper to implement SOC estimation.

2.2. Gated recurrent unit

The GRU [19] is proposed to settle the long-term dependencies problem. In the GRU, the previous hidden state h_{k-1} and the current input x_k doesn't directly affect the current hidden state h_k as in standard RNN. Instead, the GRU uses an update gate to determine how much information of the past should be let through, and a reset gate to decide how much information to be discarded.

Fig. 2 represents a general structure of the GRU, where the activation of the GRU is a linear combination of the previous state h_{k-1} and a candidate state \tilde{h}_k :

$$h_k = (1 - z_k) \cdot h_{k-1} + z_k \cdot \tilde{h}_k. \quad (4)$$

Here, z_k is called the update gate, which is activated by a sigmoid function:

$$z_k = \sigma(W_z \cdot [h_{k-1} \ x_k]). \quad (5)$$

If an entry of z_k is close to 1, the current state depends more on the candidate state, while if an entry is close to 0, then the current state relies more on the previous state. Intuitively, z_k determines

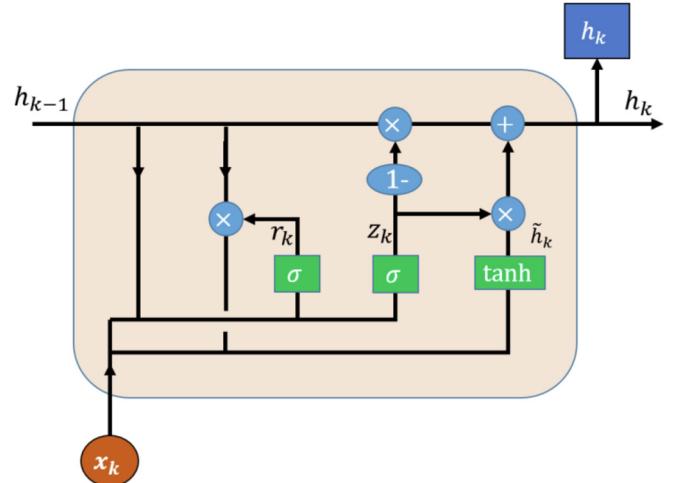


Fig. 2. Structure of the GRU.

how much proportion of the candidate state should be accepted.

The candidate state of the GRU is calculated as follows:

$$\tilde{h}_k = \tanh(W_h \cdot [r_k \cdot h_{k-1} \ x_k]), \quad (6)$$

where r_k is the reset gate activated by a sigmoid function:

$$r_k = \sigma(W_r \cdot [h_{k-1} \ x_k]). \quad (7)$$

Similar to the update gate, the reset gate restricts the impact of the previous hidden state on the candidate state.

While the standard RNN forces the recurrent unit to update in an exponential-moving-average way, the GRU gets to decide what's to update, what's to reset. Moreover, if the gating factor is close to 1, the hidden unit output flows along expanded chain for a very long distance. Meanwhile, gradient vanishing phenomenon can also be greatly avoided, since errors can also back propagate many temporal steps. Subsequently, capturing of long-term dependencies becomes possible.

2.3. SOC estimation based on the proposed network

The architecture of the proposed GRU-RNN network for SOC estimation is illustrated in Fig. 3. The network starts with a

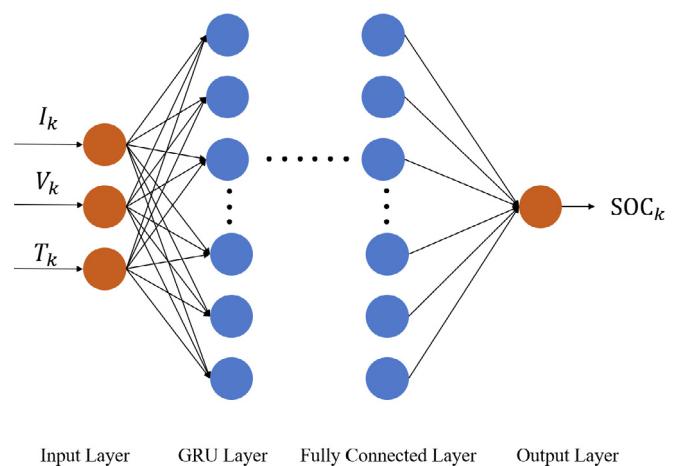


Fig. 3. Architecture of the proposed GRU-RNN network.

sequence input layer, where the network input vector is the measured signals including current, voltage, and temperature: $x_k = [I_k, V_k, T_k]$. A GRU layer is followed to learn the dependence on previous inputs. Multiple GRU layers can be stacked to deep the network. Finally, a fully connected layer and a regression output layer are used to estimate the battery SOC output: $y_k = [\text{SOC}_k]$. At each time step k , hidden state h_k in the GRU layer is updated by the input data x_k , the hidden state at previous time step h_{k-1} , the update gate z_k , and the reset gate r_k , according to Equations (4)–(7). The performance of the proposed network is evaluated under root mean square error (RMSE) and mean absolute error (MAE) criteria:

$$\text{RMSE} = \sqrt{\frac{1}{K} \sum_{k=1}^K (y_k - \hat{y}_k)^2}, \quad \text{MAE} = \frac{1}{K} \sum_{k=1}^K |y_k - \hat{y}_k|. \quad (8)$$

The RMSE indicates the robustness of the estimation while the MAE indicates the accuracy of the estimation.

The overall SOC estimation consists of an offline training process and an online testing process. In the offline training process, large amounts of historical data are fed into the network to learn the nonlinearity between the output and the input. The training database can be constructed by both field data and data collected from simulation tests. To prevent the network from overfitting, one dropout layer [27] with masking probability set to 0.5 is added between the GRU layer and the fully-connected layer. With a well-trained network, the SOC estimation can then be performed on the online testing process.

3. Battery experiments and data collection

In the experiments, three cylindrical lithium nickel manganese cobalt oxide (NMC) batteries were used, with brief specifications shown in Table 1. To mimic the complex EV battery loading behaviors, test subjects were exposed to two dynamic testing profiles, including dynamic stress test (DST) and federal urban driving schedule (FUDS). Both profiles are designed to simulate a discharge mode of an EV battery by the US Advanced Battery Consortium [28]. Fig. 4(a) plots the current profiles of the DST and FUDS profiles. Specifically, the DST takes regenerative charging into account and is composed of a variety of current steps with different lengths and amplitudes, while the FUDS focuses on simulating urban driving profile with rapid speed fluctuations.

In both DST and FUDS tests, the subject undergoes a complete charge-discharge cycle. In the charge process, constant-current and constant-voltage charging profile is applied to fully charge the battery. In the constant current stage, the battery is charged using a constant current of 1C. Once the battery charge voltage (4.2 V as in specification) is reached, the charge process is then switched to the constant voltage stage. This stage features gradually decreasing charging current, the battery is eventually considered fully charged when the current drops below 0.01C. The discharge process kicks off right upon completion of the charge process, in which the abovementioned testing profiles are applied to simulate a real-

world battery usage. The discharge process terminates when the battery is regarded as fully discharged, i.e. the battery's voltage drops below the discharge cut-off voltage 2.5 V.

For each battery sample, the procedure was repeated under 0°C, 10°C, 20°C, 30°C, 40°C, 50°C, and room temperature (RT, around 27°C) to construct the training and testing dataset under different temperatures. After setting the chamber temperature, there was a 3-h gap to reinstate the battery to a steady state before starting the tests. Sensor signals including current, voltage, temperature, etc., were recorded every 1 s during the whole tests. Fig. 4(b) plots the measured voltages at RT for the DST and FUDS tests, respectively. The discharge voltage versus the cumulative capacity at varying temperatures is presented in Fig. 5. Clearly, battery deliverable capacities are heavily constrained under low temperatures.

4. Estimation results and discussion

The input vector fed into the GRU-RNN network is given by $x_k = [V_k, I_k, T_k]$, where V_k , I_k , and T_k represent the measured voltage, current, and temperature of the battery at time step k , respectively. The output of the GRU-RNN network is the estimated SOC at the same time step, $y_k = [\text{SOC}_k]$. For method evaluation, true SOC is calculated by Coulomb counting method [6]. Some of the network parameters during the training process are initialized as follows:

- initial learning rate: 0.01;
- mini batch size: 60;
- gradient threshold: 1;
- number of neurons in fully connected layer: 80.

All the training processes are run on a computer with a single GeForce GTX 1080Ti GPU.

4.1. Estimation at room temperature

Fig. 6 shows the SOC estimation for the DST test under RT. A GRU-RNN network with a single GRU layer (150 hidden neurons) is trained over 2000 epochs using data from the FUDS test. Under a GPU environment, the training time is about 20 min. The tracking results are presented in Fig. 6(a) and the estimation errors, evaluated as true value minus estimated value, are plotted in more detail in Fig. 6(b). The overall performance of the GRU-RNN network is satisfactory. The estimation error starts from 4% and then quickly converges within 2%. The RMSE and MAE of the estimation are 1.05% and 0.77%, respectively. The average computation time at each timestamp is about 0.06 ms on our lab computer (OS: Win 10 64-bit, Memory: 8 GB, CPU: Intel i7-7500U @2.70 GHz). Therefore, the proposed method can be applied to onboard applications, despite poorer hardware settings.

Number of training epochs, number of hidden GRU layers, and number of hidden neurons in each GRU layer are three critical parameters that may influence the performance of the GRU-RNN network on SOC estimation. In the next, various experiments are conducted to explore the impact of these parameters on the GRU-RNN network, especially in the application of battery SOC estimation.

In the first experiment, the same GRU-RNN network as above is used to test the influence of the training epoch. The network is trained on the training dataset (FUDS data with 8300 data points) along with varying epochs and then its performance is tested on the validation dataset (DST data with 8500 data points). Table 2 summarizes the training time, RMSE, and MAE results for both training and validation sets. To gain visual intuition, the RMSEs are plotted in Fig. 7. The RMSEs of the training and validation processes decrease sharply in the first few hundreds of epochs, and keep

Table 1
Specifications of the battery cells.

Type	BAK B18650CD
Cathode	Li(NiMnCo) O ₂
Anode	Si/C
Nominal capacity	1.3Ah
Nominal voltage	3.6 V
Charge voltage	4.2 V
Discharge cut-off voltage	2.5 V
End-of-charge current	0.01C

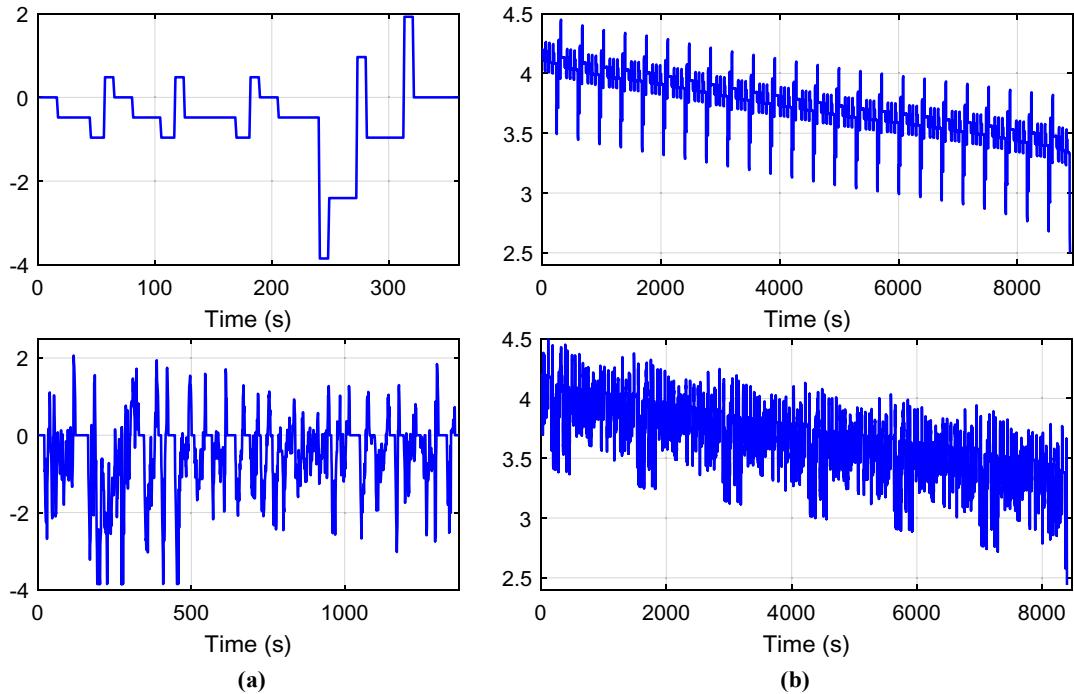


Fig. 4. Current profile and measured voltage in one discharge cycle: (a) current profiles of DST (top) and FUDS (bottom); (b) measured voltages during DST test (top) and FUDS test (bottom).

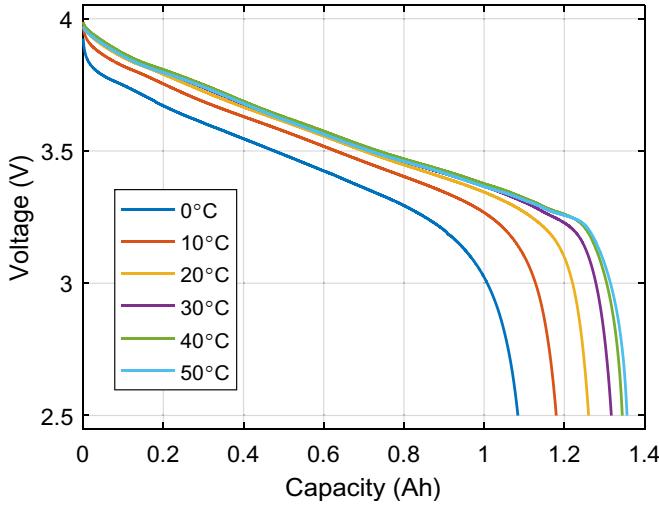


Fig. 5. Discharge curves of a battery under different temperatures.

relatively steady after 1500 epochs. At epoch 2000, the validation result reaches its local optimal, with RMSE of 1.05% and MAE of 0.77%. Performance after epoch 2000 is almost the same, but slightly increasing trends are observed for the RMSE and MAE of the validation dataset, which indicates that larger epoch number may lead to an over-fitting. Trading off between testing performance and training time, which extends linearly with training epochs, 2000 epochs presents an optimal choice, the training of which is about 20 min in our settings.

Next, the performance of the proposed network is tested with hidden GRU neurons varying among [25 50 100 150 200 300 400 600]. All the results are tabulated in Table 3, the RMSEs are also plotted in Fig. 8. While generally using more hidden neurons yields a smaller training error, too many of which also leads to an

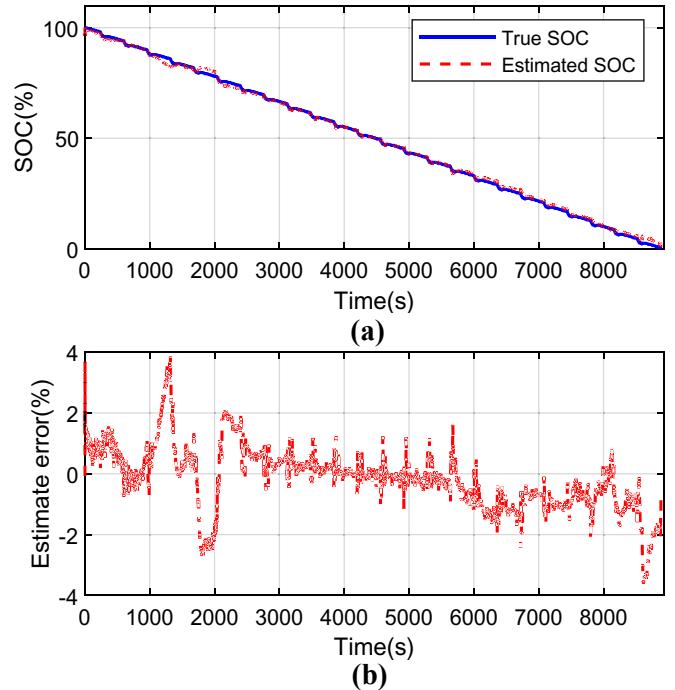


Fig. 6. Results of SOC estimation under RT: (a) SOC tracking; (b) estimation error.

overfitting, i.e. a large testing error. On the other hand, too few neurons would fail to capture the underlying dynamics. From this perspective, 100 to 150 hidden neurons are recommended for the network.

Finally, additional hidden GRU layers can be added to make a deeper network. In the third test, a list of stacked GRU-RNN networks are constructed and compared regarding the performance of

Table 2

Results of SOC estimation using different epoch numbers.

	Epoch	25	100	200	500	1000	1500	2000	2500	3000	4000	6000
Training	Time (min)	0.47	1.24	2.12	5.05	10.03	14.95	20.63	26.01	33.28	47.88	79.14
	RMSE (%)	16.21	7.65	3.10	0.78	0.46	1.59	0.86	0.69	0.70	0.76	0.84
Testing	MAE (%)	12.98	7.32	2.74	0.66	0.38	1.44	0.73	0.58	0.60	0.65	0.71
	RMSE (%)	18.30	8.54	3.87	1.53	1.25	2.07	1.05	1.28	1.29	1.42	1.45
	MAE (%)	14.17	8.33	3.53	1.34	1.10	1.94	0.77	1.19	0.98	1.31	1.01

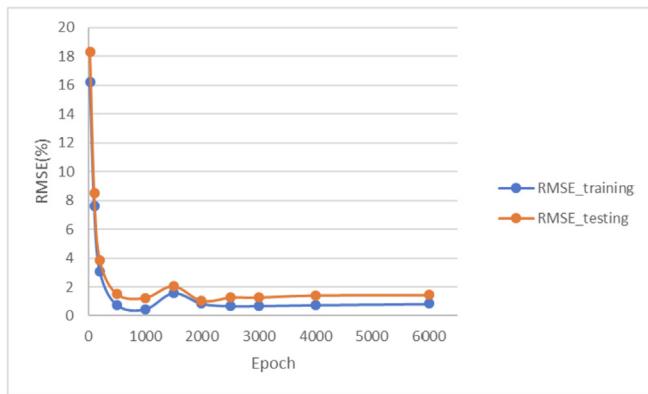


Fig. 7. RMSEs of training and validation sets with respect to the number of epochs.

Table 3

Results of SOC estimation using different numbers of hidden neurons.

	Hidden neuron	25	50	100	150	200	300	400	600
Training	RMSE (%)	1.06	0.64	0.75	0.86	1.20	0.75	1.30	1.11
	MAE (%)	0.90	0.55	0.65	0.73	1.01	0.68	1.12	1.02
Testing	RMSE (%)	1.25	1.39	1.04	1.05	1.74	1.66	1.28	2.24
	MAE (%)	0.92	1.28	0.75	0.77	1.27	1.48	0.99	2.09

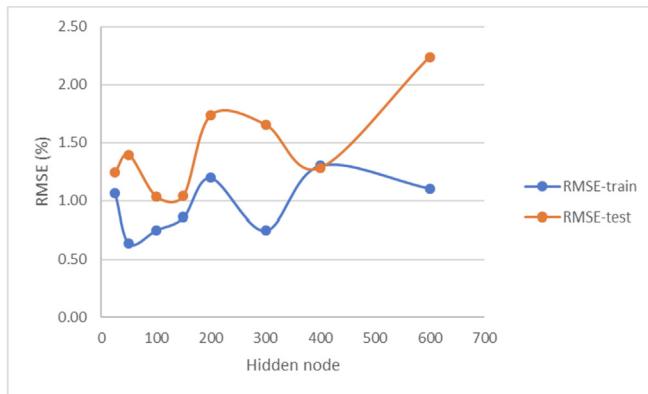


Fig. 8. RMSEs of training and validation sets with respect to the number of hidden neurons.

SOC estimation. Table 4 lists the details of these networks as well as their training and testing results. It seems that stacked networks with much fewer neurons, such as networks 2 and 3, show comparable results with the network of single GRU layer (network 1). Although they have fewer hidden neurons, they can recombine the learned representation from prior layers and create new representations at high levels of abstraction. For networks 4 and 5 with more layers, they provide better RMSE training results, but their testing error is much larger. The SOC tracking and estimation errors of the previous three networks are plotted in Fig. 9. Results of

Table 4

Comparison of stacked GRU-RNN networks. '150', '16', '32', '75', and '50' refer to the number of hidden neurons in the GRU layers.

Network	Model	Training		Testing	
		RMSE (%)	MAE (%)	RMSE (%)	MAE (%)
1	GRU 150	0.86	0.73	1.05	0.77
2	GRU 16-16	1.01	0.86	1.30	0.89
3	GRU 32-16	1.19	0.98	1.17	0.75
4	GRU 75-75	0.84	0.71	3.42	1.89
5	GRU 50-50-50	0.88	0.76	2.42	1.98

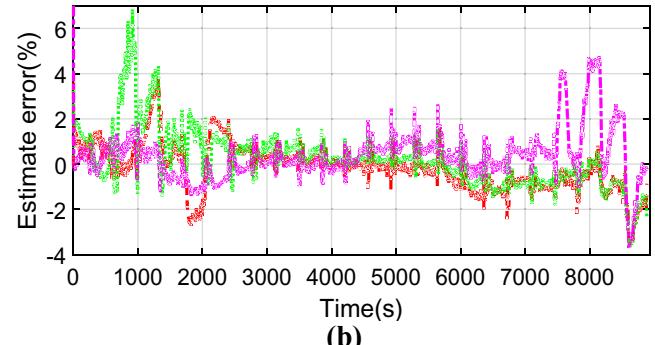
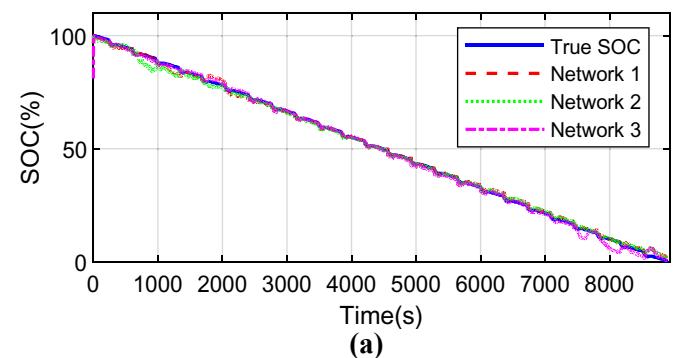


Fig. 9. Results of SOC estimation using different GRU-RNN networks: (a) SOC tracking; (b) estimation error.

deeper networks fluctuate more, especially at the beginning (network 2 in green dash line) and end (network 3 in purple dash line) of SOC estimation, implying the existence of over-fitting. Still, it can be concluded that for SOC estimation, a network with single GRU layer is sufficient.

The discussion on epoch numbers, hidden neurons, and hidden GRU layers indicates that the GRU-RNN network with one GRU layer and few hidden neurons performs well on SOC estimation. One reason is that dimension of the network input is small, so that few GRU neurons are enough to depict the nonlinearity inside the battery. Consequently, complex structures with more hidden GRU neurons or more hidden layers tend to over-fit. In the following experiment, one GRU-layer with 150 neurons is adopted unless

stated otherwise.

4.2. Estimation with unknown initial state

In real applications, the battery is not always fully charged initially. In other words, the initial SOC is not necessarily 100%. Table 5 summarizes the experimental results with different initial SOC values (100%, 80%, 60%, 40%, 20%), where the performance of the proposed method is compared with some state-of-the-art methods including EKF, UKF, and PF. For the three model-based filtering methods, a combined model [8] is employed to construct the state-space model, and the variances of state function and measurement function are set to 1e-5 and 1e-2, respectively.

When the initial SOC value is 100%, the performance of the proposed method is comparable with the EKF, the UKF, and the PF, with a slightly larger RMSE (1.05%) and a slightly smaller MAE (0.77%). The total computation time over 8500 timestamps are 0.6s (GRU-RNN), 0.14s (EKF), 0.31s (UKF), and 1.004s (PF), respectively.

When the initial SOC value varies from 100% to 20% (but keep the initial guess of model-based filtering methods at 100%), the RMSE and MAE values increase, especially for the EKF and the PF, as in Table 5. Fig. 10 plots the estimation results when the initial SOC is 60%. It takes some time for the model-based filtering methods to track the true SOC. While for the proposed method, it tracks the true SOC almost from the second step. For the proposed method, in all cases, the RMSE and MAE values vary around 1%, indicating that, compared with model-based filtering methods, the proposed method has stronger robustness against unknown initial SOC values.

4.3. Estimation with varying temperature

Temperature is a critical parameter in modelling battery dynamics and has been proven to have a significant influence on SOC estimation. The battery may operate under varying ambient temperatures, including seasonal change and night-day shift. The heat generation during charge-discharge processes will also influence the battery surface temperatures. In this section, the GRU-RNN network is trained using data collected under varying ambient temperatures.

The network is trained using FUDS datasets collected under 0°C, 10°C, 20°C, 30°C, 40°C, and 50°C. The training process reaches its optimal status at epoch 3000 and it takes around 34 min. The performance of the proposed network is then tested on the DST datasets under 0°C, 10°C, 20°C, 30°C, 40°C, 50°C, and RT, respectively. As it is impractical to train the network under all temperatures, data under RT are used here to test the robustness of the GRU-RNN network against untrained temperatures.

The RMSEs and MAEs of all cases are tabulated in Table 6. Overall, the proposed network provides an acceptable estimation result, with RMSEs within 2.5% and MAEs within 2%. Fig. 11 presents the SOC estimation under 0°C, RT, and 40°C, respectively. As in Fig. 11(c), under normal and high ambient temperatures, the

network gives quite a satisfying SOC estimation. The estimated SOCs are close to the true value, with almost all the estimation errors within 2%. The RMSE and MAE at 40°C are 1.07% and 0.9%, respectively. For the SOC estimation under low temperatures (0°C, 10°C, 20°C), RMSEs vary from 1.5% to 2.5%. Moreover, it is easily seen from Fig. 11(a) that the estimated SOCs slightly deviates from the true values at the end of the estimation. This is because battery dynamics at low temperature are more complicated, making SOC estimation much more difficult [29]. To improve the estimation performance at low temperatures, one practical solution is to train networks particularly for battery working at low temperatures. The SOC estimation under RT is shown in Fig. 11(b). During the discharge process, the temperature of battery surface varies between 26.5°C and 29.1°C. The estimated SOC approaches the true SOC, but fluctuates around the true value, with estimation errors sometimes as high as 5%. The corresponding RMSE and MAE are 2.15% and 1.65%, respectively. Overall, the trained network shows the ability to adapt untrained temperatures. For the fluctuation, a UKF [15] can be added right behind the network to eliminate the unwanted noises.

To conclude, the proposed GRU-RNN network can well learn the influence of temperature variations, which is usually hard to capture in the model-based filtering methods. Moreover, the proposed network can be applied to SOC estimation at different ambient temperatures, even if no data at that temperature has been used for training.

4.4. Estimation on battery of a different material

Batteries of different materials may present different charge-discharge behaviors. For instance, lithium iron phosphate (LFP) cell, as one of the mainstream lithium-ion batteries, has a flatter OCV-SOC curve compared with the NMC cell, especially for SOC between 40% and 70% [10]. In this section, to demonstrate the effectiveness of the proposed method, SOC estimation is conducted for the LFP cell.

For data collection, the same experiment procedure as in Section 3 was repeated on a set of cylindrical 18650 LFP batteries from A123 Systems, LLC, with nominal capacity 1.1Ah and cut-off voltage 2/3.6 V. The collected data under varying ambient temperatures, excluding RT, were then used to train the GRU-RNN network. The network parameters are determined by the strategy presented in Section 4.1. Finally, a GRU-RNN network with a single GRU layer (300 hidden neurons) is constructed, with over 7000 training epochs. The whole training time is 2.3 h.

Fig. 12 presents the SOC estimation under 0°C, RT, and 40°C, respectively. The top figures show the results when the initial SOC is 100% while the bottom figures show the results when the initial SOC is 60%. The proposed network presents satisfying estimation results under varying ambient temperatures, in terms of both convergence ability and tracking accuracy. When the SOC starts from 60%, it takes some time for the network to converge to the true SOCs, which roots in the flat region ranging from 70% SOC to 40% SOC in the OCV-SOC curve of the LFP battery. The RMSEs and MAEs of all cases are tabulated in Table 7. Overall, the proposed network provides an acceptable estimation result, with RMSEs within 3.5% and MAEs within 2.5%. The estimation results demonstrate the robustness of the proposed method against different battery materials.

5. Conclusions

In this paper, a GRU-based RNN network has been proposed to estimate the SOC of lithium-ion batteries. Data collected from dynamic loading profiles were used to train and test the proposed

Table 5
RMSEs and MAEs of SOC estimation when initial SOC starts from various values.

Initial SOC (%)	RMSE (%)				MAE (%)			
	GRU-RNN	EKF	UKF	PF	GRU-RNN	EKF	UKF	PF
100	1.05	1.03	1.02	1.00	0.77	0.79	0.79	0.80
80	1.20	1.85	1.09	2.06	0.81	0.86	0.86	1.13
60	0.98	3.63	1.16	5.41	0.70	0.92	0.92	2.00
40	1.18	6.80	1.35	11.59	0.96	1.14	1.14	5.00
20	1.34	13.67	1.56	25.54	1.10	1.98	1.98	14.65

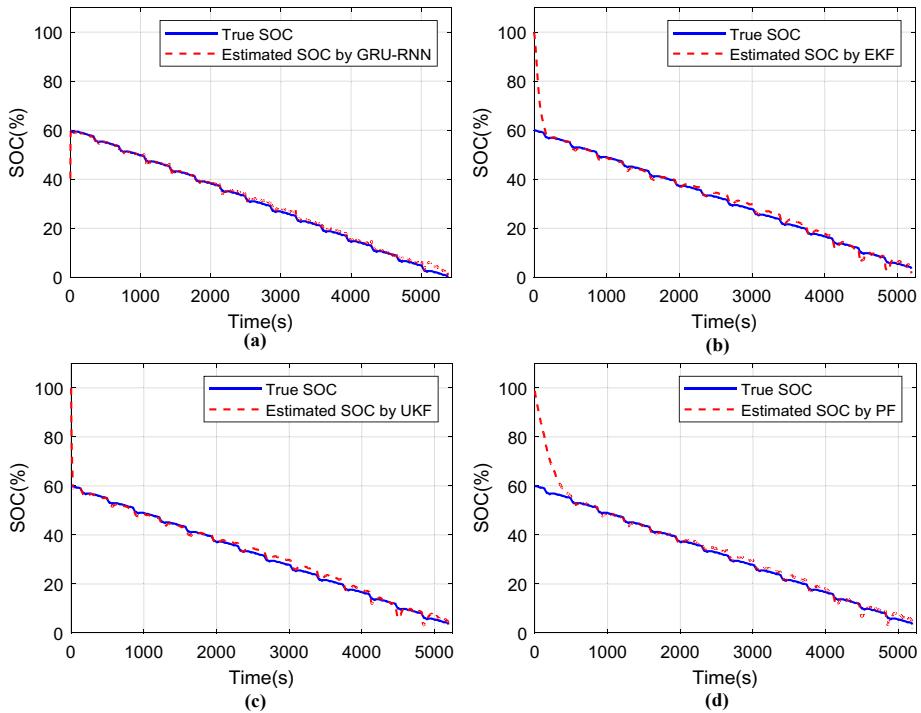


Fig. 10. SOC estimation when initial SOC = 60% and initial guess = 100%: (a) GRU-RNN; (b) EKF; (c) UKF; (d) PF.

Table 6
RMSEs and MAEs of SOC estimation under varying temperatures.

Temperature (°C)	Testing	
	RMSE (%)	MAE (%)
0	2.33	1.94
10	1.53	1.30
20	2.44	1.99
30	1.28	1.17
40	1.07	0.90
50	1.81	1.50
RT	2.15	1.65

method. The training process took several hours under a GPU environment, but the testing time was very short, even under a CPU environment. Therefore, it is suitable for real-time on-board applications. The effectiveness of the proposed method was verified on two mainstream lithium-ion batteries. Experimental results showed that simple GRU-RNN architectures with few layers and hidden neurons provided satisfying estimation results for both NMC and LFP batteries. Nevertheless, compared with the NMC battery, the LFP battery required more hidden neurons to capture the complex battery dynamics owing to the flat region existing in the OCV-SOC curve. It was also shown that the proposed method

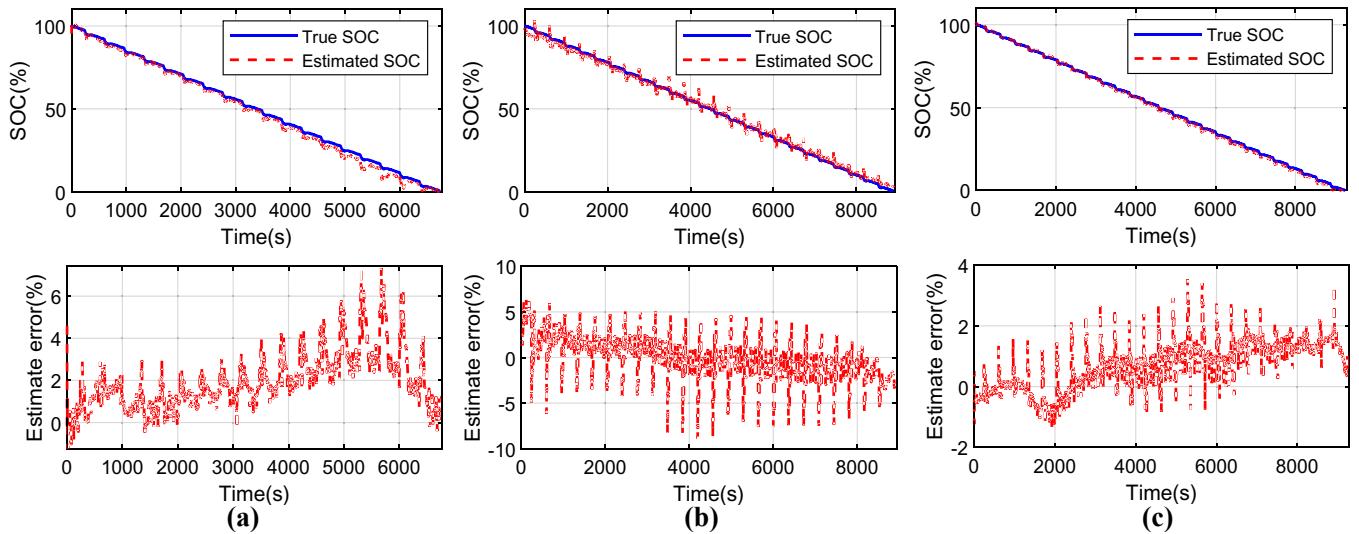


Fig. 11. Results of SOC estimation under different temperatures: (a) 0°C; (b) RT; (c) 40°C.

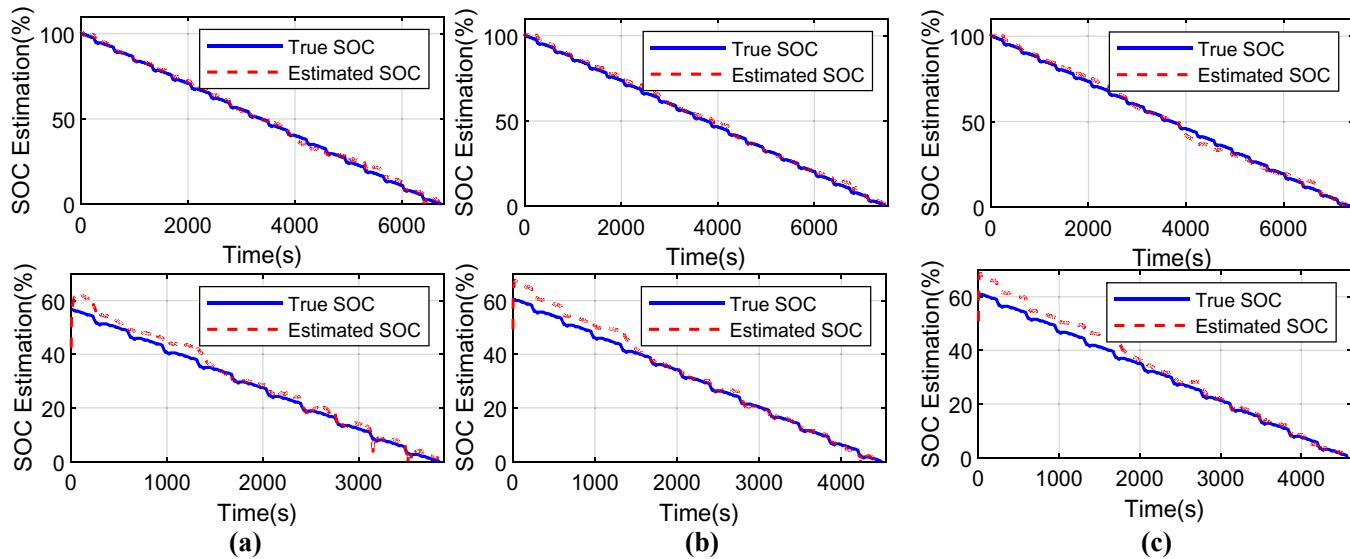


Fig. 12. Results of SOC estimation on the LFP battery under different temperatures (top row: initial SOC = 100%; bottom row: initial SOC = 60%): (a) 0°C; (b) RT; (c) 40°C.

Table 7
RMSEs and MAEs of SOC estimation on the LFP battery.

Temperature (°C)	Initial SOC = 100%		Initial SOC = 60%	
	RMSE (%)	MAE (%)	RMSE (%)	MAE (%)
0	1.68	1.36	2.60	2.06
10	1.87	1.32	2.35	1.78
20	1.92	1.68	2.40	2.04
30	1.97	1.58	2.68	1.79
40	1.55	1.26	3.34	2.50
50	1.55	1.26	3.22	2.36
RT	2.11	1.72	2.74	1.85

was robust against inaccurate initial SOC values and could well learn the influence of ambient temperatures. Under varying temperatures, the RMSEs of SOC estimation were less than 2.5% for the NMC battery and less than 3.5% for the LFP battery.

Compared with traditional model-based estimation methods, the proposed method is totally data-driven. There are no assumptions on battery materials or models. Thus, it can be readily applied to BMSs of different battery types for EVs. Besides ambient temperature, the proposed method can be easily extended to consider more factors, such as regeneration, hysteresis, and degradation.

Some suggestions are given as following to improve the performance of the proposed method on SOC estimation in the future.

1. To reduce the estimation fluctuation, especially when the network is used to estimate SOC under an untrained ambient temperature, a UKF can be added behind the network to filter out the fluctuations in the estimation;
2. Use more data for training. It is practicable because massive historical data can be obtained during EV battery running. And most of them are started from full SOC, which means true SOC can be calculated. The running data can be transferred to the data server for offline network training and validation;
3. Upon repeated usage, degradation may influence the battery dynamics. A practical operation is to update the network every 2 months. According to our observation, the influence of battery aging is not as important as that of ambient temperature, therefore, longer update gap is also acceptable.

Acknowledgment

This research is supported by the DGUT Grant (GC300501-27) and the National Natural Science Foundation of China (51875208, 51475170).

References

- [1] Zhang X, Wang Y, Yang D, Chen Z. An on-line estimation of battery pack parameters and state-of-charge using dual filters based on pack model. *Energy* 2016;115:219–29.
- [2] Manthiram A. An outlook on lithium ion battery technology. *ACS Cent Sci* 2017;3(10):1063–9.
- [3] Hannan MA, Lipu MH, Hussain A, Mohamed A. A review of lithium-ion battery state of charge estimation and management system in electric vehicle applications: Challenges and recommendations. *Renew Sustain Energy Rev* 2017;78:834–54.
- [4] Lu L, Han X, Li J, Hua J, Ouyang M. A review on the key issues for lithium-ion battery management in electric vehicles. *J Power Sources* 2013;226:272–88.
- [5] He H, Zhang X, Xiong R, Xu Y, Guo H. Online model-based estimation of state-of-charge and open-circuit voltage of lithium-ion batteries in electric vehicles. *Energy* 2012;39(1):310–8.
- [6] Ng KS, Moo C-S, Chen Y-P, Hsieh Y-C. Enhanced coulomb counting method for estimating state-of-charge and state-of-health of lithium-ion batteries. *Appl Energy* 2009;86(9):1506–11.
- [7] Yang F, Xing Y, Wang D, Tsui K-L. A comparative study of three model-based algorithms for estimating state-of-charge of lithium-ion batteries under a new combined dynamic loading profile. *Appl Energy* 2016;164:387–99.
- [8] Plett GL. Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs: Part 2. Modeling and identification. *J Power Sources* 2004;134(2):262–76.
- [9] Hu X, Li S, Peng H, Sun F. Robustness analysis of State-of-Charge estimation methods for two types of Li-ion batteries. *J Power Sources* 2012;217(0):209–19.
- [10] Xing Y, He W, Pecht M, Tsui KL. State of charge estimation of lithium-ion batteries using the open-circuit voltage at various ambient temperatures. *Appl Energy* 2014;113(0):106–15.
- [11] Zou C, Manzie C, Nesić D. A framework for simplification of PDE-based lithium-ion battery models. *IEEE Trans Control Syst Technol* 2016;24(5):1594–609.
- [12] Salkind AJ, Fennie C, Singh P, Atwater T, Reisner DE. Determination of state-of-charge and state-of-health of batteries by fuzzy logic methodology. *J Power Sources* 1999;80(1–2):293–300.
- [13] Anton JA, Nieto PG, Viejo CB, Vilan JV. Support vector machines used to estimate the battery state of charge. *IEEE Trans Power Electron* 2013;28(12):5919–26.
- [14] Kang L, Zhao X, Ma J. A new neural network model for the state-of-charge estimation in the battery degradation process. *Appl Energy* 2014;121(0):20–7.
- [15] He W, Williard N, Chen C, Pecht M. 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* 2014;62:783–91.

- [16] Sahinoglu GO, Pajovic M, Sahinoglu Z, Wang Y, Orlik PV, Wada T. Battery state-of-charge estimation based on regular/recurrent Gaussian process regression. *IEEE Trans Ind Electron* 2018;65(5):4311–21.
- [17] Chaoui H, Ibe-Ekeocha CC. State of charge and state of health estimation for lithium batteries using recurrent neural networks. *IEEE Trans Veh Technol* 2017;66(10):8773–83.
- [18] Bengio Y, Simard P, Frasconi P. Learning long-term dependencies with gradient descent is difficult. *IEEE Trans Neural Netw* 1994;5(2):157–66.
- [19] Cho K, Van Merriënboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation. 2014. arXiv preprint arXiv:1406.1078.
- [20] Ravanelli M, Brakel P, Omologo M, Bengio Y. Improving speech recognition by revising gated recurrent units. 2017. arXiv preprint arXiv:1710.00641.
- [21] Choi E, Bahadori MT, Schuetz A, Stewart WF, Sun J. Doctor ai: Predicting clinical events via recurrent neural networks. 2015. arXiv preprint arXiv: 1511.05942.
- [22] Liu P, Qiu X, Huang X. Recurrent neural network for text classification with multi-task learning. 2016. arXiv preprint arXiv:1605.05101.
- [23] Goodfellow I, Bengio Y, Courville A, Bengio Y. Deep learning. Cambridge: MIT press; 2016.
- [24] Le QV, Jaitly N, Hinton GE. A simple way to initialize recurrent networks of rectified linear units. 2015. arXiv preprint arXiv:1504.00941.
- [25] Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Comput* 1997;9(8):1735–80.
- [26] Chung J, Gulcehre C, Cho K, Bengio Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. 2014. p. 3555. arXiv preprint arXiv: 1412.
- [27] Hinton GE, Srivastava N, Krizhevsky A, Sutskever I, Salakhutdinov RR. Improving neural networks by preventing co-adaptation of feature detectors. 2012. arXiv preprint arXiv:1207.0580.
- [28] Hunt G. USABC electric vehicle battery test procedures manual. Washington, DC, USA: United States Department of Energy; 1996.
- [29] Waldmann T, Wilka M, Kasper M, Fleischhammer M, Wohlfahrt-Mehrens M. Temperature dependent ageing mechanisms in Lithium-ion batteries—A Post-Mortem study. *J Power Sources* 2014;262:129–35.