IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON ADVANCES IN PROGNOSTICS AND SYSTEM HEALTH MANAGEMENT

Received March 3, 2019, accepted April 18, 2019, date of publication April 23, 2019, date of current version May 3, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2912803

State-of-Charge Estimation of Lithium-Ion Batteries via Long Short-Term Memory Network

FANGFANG YANG[1], XIANGBAO SONG[2], FAN XU[1], AND KWOK-LEUNG TSUI[1]

[1]School of Data Science, City University of Hong Kong, Hong Kong
[2]Googol Technology (Shenzhen) Ltd., Shenzhen 518000, China

Corresponding author: Xiangbao Song (xbao.song@connect.ust.hk)

This work was supported in part by the Research Grants Council Theme-Based Research Scheme under Project T32-101/15-R and in part by the General Research Fund under Project CityU 11206417.

ABSTRACT Accurate state-of-charge (SOC) estimation is critical for driving range prediction of electric vehicles and optimal charge control of batteries. In this paper, a stacked long short-term memory network is proposed to model the complex dynamics of lithium iron phosphate batteries and infer battery SOC from current, voltage, and temperature measurements. The proposed network is trained and tested using data collected from the dynamic stress test, US06 test, and federal urban driving schedule. The performance on SOC estimation is evaluated regarding tracking accuracy, computation time, robustness against unknown initial states, and compared with results from the model-based filtering approach (unscented Kalman filter). Moreover, different training and testing data sets are constructed to test its robustness against varying loading profiles. The experimental results show that the proposed network well captures the nonlinear correlation between SOC and measurable signals and provides better tracking performance than the unscented Kalman filter. In case of inaccurate initial SOCs, the proposed network presents quick convergence to the true SOC, with root mean square errors within 2% and mean average errors within 1%. Moreover, the estimation time at each time step is sub-millisecond, making it appropriate for real-time applications.

INDEX TERMS State-of-charge estimation, lithium iron phosphate batteries, long short-term memory, recurrent neural network, unscented Kalman filter.

I. INTRODUCTION

Due to long cycle life and high energy/power density, lithium-ion batteries have become the main energy storage device for electric vehicles (EVs). To ensure the safe, reliable and efficient operation of EVs, it is critical to establish an advanced battery management system (BMS) to accurately and timely monitor the battery status [1]. The state-of-charge (SOC), as one of the key states of the BMS, quantifies the remaining energy of the battery in the current cycle and indicates the remaining time before charge is needed [2]. In analogy to the fuel tank of a car, SOC can be regarded as the ''gas gauge'' or ''fuel gauge'' of a battery, which is formally defined as the ratio of the remaining capacity to the nominal capacity of a battery. Mathematically, SOC can be evaluated as a function of time and current:

$$\mathrm{SOC}(t) = \mathrm{SOC}(0) - \frac{1}{C_n}\int_0^t I(t)dt, \tag{1}$$

where C_n is the cell nominal capacity and $I(t)$ is the cell current at time t.

Accurate SOC estimation is essential for EV mileage estimation and trip planning. Moreover, it can ensure the battery working within a safe operating window to extend the battery cycle life. However, SOC cannot be directly measured, and it is generally inferred from other measurable variables such as voltage, current, and temperature. Accurate SOC estimation is still a challenging task due to the complex dynamics inside the battery, especially for the lithium iron phosphate (LFP) battery, which features a typical flat open circuit voltage (OCV) - SOC curve [3].

Common methods used for SOC estimation of lithium-ion batteries mainly include OCV method, Ampere-Hour integral method, model-based filtering method, and machine learning method.

The OCV method estimates the SOC by a one-to-one correspondence between SOC and OCV, which requires establishing a corresponding OCV-SOC lookup table in advance [4]. This method is simple, but it cannot be applied

The associate editor coordinating the review of this manuscript and approving it for publication was Dong Wang.

on-board because the precise OCV value needs to be measured at a stable open-circuit condition, which usually requires the battery to be disconnected from the load for a sufficiently long time. In addition, differences between battery samples and battery degradation levels also influence the relationship between OCV and SOC. Moreover, when the OCV-SOC curve is very flat, the SOC estimate becomes very sensitive, in which case small OCV errors account for large SOC estimation errors.

The Ampere-Hour integral method is an open-loop estimation method for estimating the SOC of a battery by time-integrating the discharge current, which is simple to operate and can be adopted for real-time application [5]. Nonetheless, the Ampere-Hour integral method requires the initial SOC as a priori and is totally dependent on the accuracy of the current sensor. In case low precision sensor is used or a biased noise exists in the measurements, the estimation error would accumulate over time and there is simply no way to correct the estimation.

The model-based filtering method relates the hidden SOC state to the measured variable such as voltage and current by constructing a state-space model, and then uses filtering technique such as extended Kalman filter and unscented Kalman filter (UKF) to obtain the optimal estimate of battery SOC, which is essentially to correct estimated SOC from Ampere-Hour integral method with measured voltage value [6]. This method is robust to unknown initial SOC values and measurement noise, is suitable for real-time applications, and shows a good performance for on-board battery SOC estimation. But its performance is heavily dependent on the accuracy of battery model [7]. Moreover, to account for other factors, such as degradation level, a new model must be established. Finally, model parameters, such as measurement noise, must be fine-tuned to obtain a satisfying performance.

The machine learning method treats the battery as a black box, and directly learns its internal dynamics through massive charge-discharge data to establish the nonlinear relationship between battery SOC and measured variables [8]. The commonly used machine learning methods for SOC estimation include fuzzy logic [9], support vector machine [10] and neural network (NN) [11], [12] and so on. This kind of method does not require a specific mathematical model and can be easily extended to account for other influencing factors. However, its accuracy depends heavily on the quantity and quality of the training data, and the training time is quite long in the case of a large amount of data.

In recent years, with the rapid development of graphics processing units (GPUs), the computing power has increased steadily, the training time of NNs is greatly shortened. Deep learning neural network-based methods have attracted a lot attention from the research world. For on-board batteries, massive complex field data can be obtained via online BMS and then continuously upload to offline data server. The battery data can also be collected by laboratory simulation under dynamic driving profiles. He *et al.* [13] combined an artificial NN and UKF to estimate the battery SOC. Sahinoglu *et al.* [14] proposed a recurrent neural network (RNN) to estimate the SOC of lithium-ion batteries. Chaoui *et al.* [15] then used the RNN to estimate the SOC and state-of-health of lithium-ion batteries and assessed the estimation performance on two different batteries. Compared with the traditional feedforward neural network, the RNN correlates the past charge-discharge information of the battery and associates the current SOC state with the previous state and measured values, thus showing an excellent estimation performance.

However, due to the gradient vanishing or explosion phenomenon in the training process, RNN generally fails to capture long-term dependency [16]. Long short-term memory (LSTM), as the most well-known extension of RNN, is proposed by [17] to address this problem, which can store information for much longer temporal steps. The LSTM has achieved great success in various scenarios including natural language processing [18], machine translation [19], and bearing degradation assessment [20].

In this paper, a stacked LSTM network with multiple hidden LSTM layers is proposed to estimate the SOC of LFP batteries. With the current, voltage, and temperature measurements as input and the SOC as output, the proposed network is trained offline to model the complex battery dynamics, using data collected from dynamic stress test, US06 test, and federal urban driven schedule. Its performance on SOC estimation is then evaluated and compared with the UKF method, with regard to tracking accuracy, computation time, and robustness against unknown initial SOCs. Moreover, the performance of proposed LSTM network with different training and testing data sets is also studied.

The rest of the paper is organized as follows. Section II illustrates the experiment design and data collection. A stacked LSTM network is proposed for battery SOC estimation in Section III. Results and discussions are presented in Section IV. Finally, Section V concludes this paper.

II. EXPERIMENTS

The test platform is shown in Fig. 1. It includes an Arbin BT2000 tester for testing batteries, a Votch temperature chamber for controlling operating temperatures, and a PC with Arbins' Mits Pro Software for battery charging/discharging control and data collection. Collected data are then transferred to a PC with MATLAB R2018a and Python 3.7.2 for post-analysis. The cylindrical A123 18650 battery samples with LFP cathode and graphite anode were used in the experiments, the specifications of which are listed in Table 1.

To simulate real-world EV battery load behaviors, the DST, US06, and FUDS profiles were employed to discharge the battery samples under room temperature. While all the profiles are designed by the US Advanced Battery Consortium [21], they simulate different battery usages. Specifically, the DST profile, which consists of a series of current steps with different lengths and amplitudes, considers the battery capacity regeneration, while the FUDS profile simulates

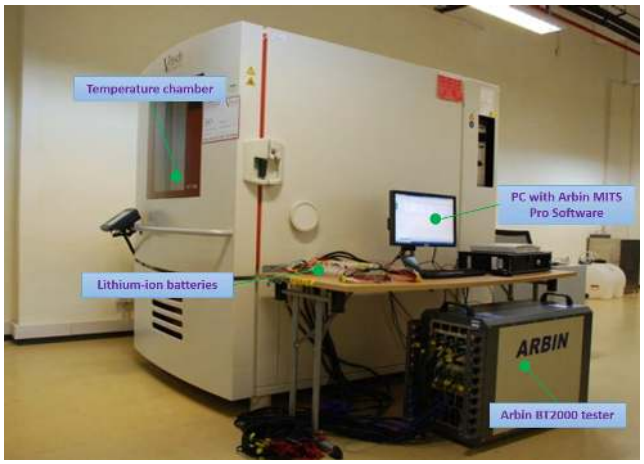

FIGURE 1. Battery charging and discharging platform.

TABLE 1. Specifications of the batteries used in the experiment.

Type	A123 18650
Cathode	LFP
Anode	Graphite
Nominal capacity	1.1Ah
Charge voltage	3.6V
Discharge cut-off voltage	2V
End-of-charge current	0.01C

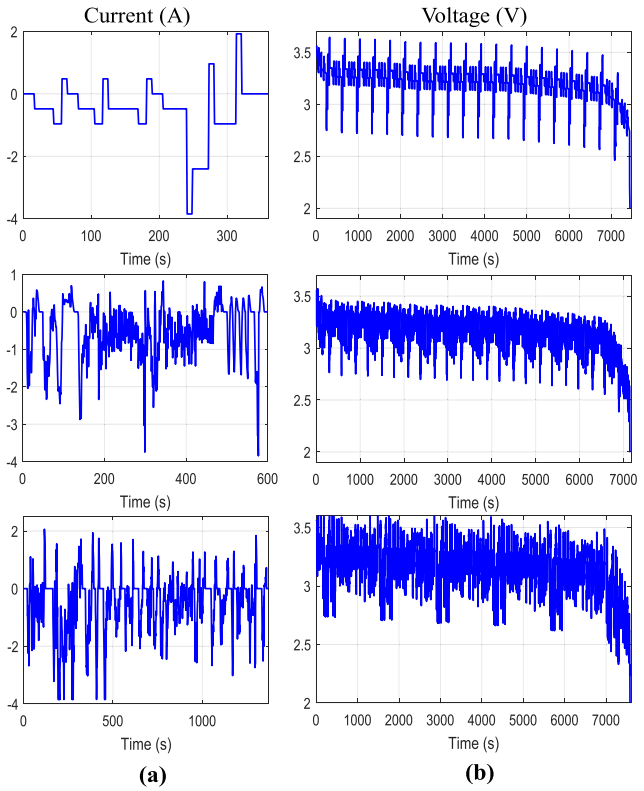

FIGURE 2. Current profile and measured voltage in one discharge cycle: (a) current profiles of DST (top), US06 (middle), and FUDS (bottom); (b) measured voltages during the DST test (top), US06 test (middle), and FUDS test (bottom).

a city driving profile with fast speed fluctuations and the US06 simulates the highway driving with high acceleration and rapid speed fluctuations. Fig. 2(a) plots their current profiles, respectively. In one test, the battery was first fully charged under a constant-current/constant-voltage charge mode. Then during discharge, one of the above profiles were adopted repeatedly until fully discharge. The current, voltage, and temperature information were recorded every 1 second.

Fig. 2(b) plots measured voltage data during the discharge process of DST, US06, and FUDS tests, respectively. The DST and US06 data set are used to train the model, while the FUDS data set is used to test the performance of SOC estimation.

III. STATE-OF-CHRAGE ESTIMATION BASED ON STACKED LONG SHORT-TERM MEMORY NETWORK

A. LONG SHORT-TERM MEMORY

The LSTM network, proposed by Hochreiter *et al.* [17], was developed on the basis of classical RNNs, which uses memory units instead of ordinary hidden nodes to avoid gradient vanishing or explosion after passing many time steps, thus overcoming the difficulties encountered in traditional RNN training. An illustrative structure of LSTM unit is shown in Fig. 3. The calculation process at time k for the forward pass of an LSTM unit is summarized as followings:

$$f_k = \sigma_g\left(W_f x_k + U_f h_{k-1} + b_f\right)$$

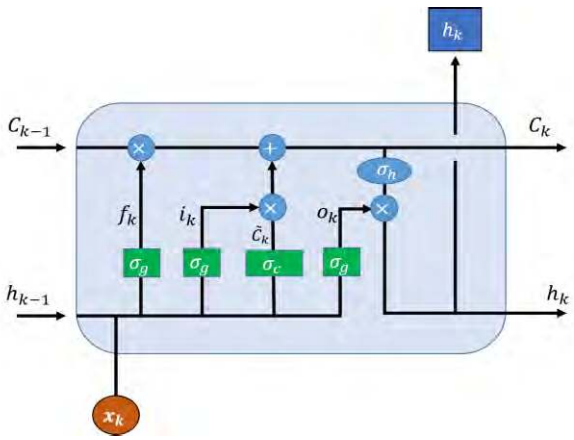

FIGURE 3. Structure of the LSTM unit.

$$\begin{aligned} i_k &= \sigma_g\left(W_i x_k + U_i h_{k-1} + b_i\right) \\ o_k &= \sigma_g\left(W_o x_k + U_o h_{k-1} + b_o\right) \\ c_k &= f_k \circ c_{k-1} + i_k \circ \sigma_c\left(W_c x_k + U_c h_{k-1} + b_c\right) \\ h_k &= o_k \circ \sigma_h(c_k), \end{aligned} \tag{2}$$

where x_k is the input of the LSTM unit at time k; h_k is the output of the LSTM unit as well as the network hidden state; c_k is the unit memory; i_k, f_k, and o_k are the activation vectors of input gate, forget gate, and output gate, respectively;

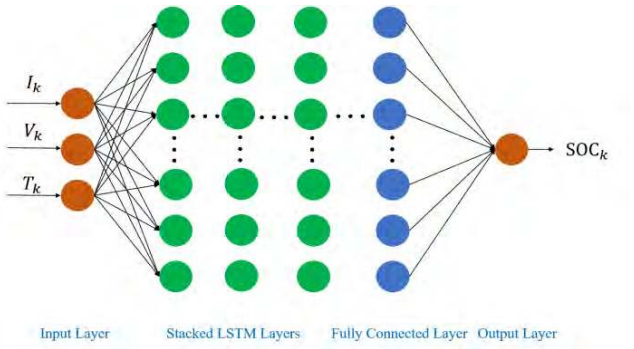

FIGURE 4. Architecture of the proposed stacked LSTM network.

W, U, and b are weight matrices and bias parameter which need to be learned during training; σ_g, σ_c, and σ_h are smooth, bounded activation functions including a logistic sigmoid function, a hyperbolic tangent function, and a hyperbolic tangent function, respectively.

The output of a sigmoid function is bounded between 0 and 1, which in the LSTM framework can interpreted as a forgetting factor. For a trained network, when the value of the forget gate is close to 0, it means there is no need to store old memory anymore, hence the past memory is eliminated. The same interpretation applies to the input gate and the output gate analogously. For instance, if then value of the input gate is close to 0, it means that the LSTM unit decides that the input is trivial and need not to be memorized.

B. THE PROPOSED STACKED LSTM NETWORK

Fig. 4 shows the proposed stacked LSTM network. The network starts with a sequence input layer, where the battery variables including measured current, voltage and temperature are the input vectors. The following three layers are hidden LSTM layers each with 50 nodes. The stacked LSTM layers can recombine the learned representation from prior layers and create new representations at high levels of abstraction. They perform nonlinear transformation on the input data, generate a memory state for the past information, and establish a dependency relationship between SOCs of different time periods. A fully connected layer is added later to further scale and transform the LSTM outputs to improve the ability of the model in processing nonlinear data. Finally, an output layer gives the SOC estimation. The size of the memory unit (50 nodes) is selected by taking the best compromise between the training workload and the testing performance. We trained another network with 100 nodes per layer, but the performance was not significantly improved.

The model was trained using the back propagation through time algorithm [22]. Considering possible over-training during the training phase, the dropout algorithm with a dropout percentage of 50% is used in each of the hidden layers [23]. The learning rate is initialized to 0.01. With a well-trained network, the SOC estimation can then be performed on the online testing process.

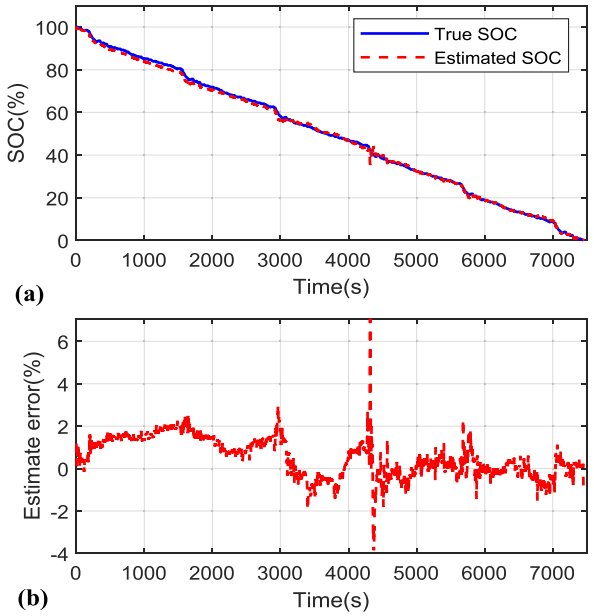

FIGURE 5. Results of SOC estimation with an initial SOC = 100%: (a) SOC tracking; (b) estimation error.

C. PERFORMANCE EVALUATION

The performance of the stacked LSTM network is evaluated using root mean square error (RMSE) and mean absolute error (MAE) criteria:

$$\text{RMSE} = \sqrt{\frac{1}{K}\sum_{k=1}^{K}\left(y_k - \hat{y}_k\right)^2},$$
$$\text{MAE} = \frac{1}{K}\sum_{k=1}^{K}\left|y_k - \hat{y}_k\right| \tag{3}$$

where y is the true value while $\hat{y}$ is the estimated value. The MAE measures how close estimates are to the corresponding outcomes without considering the sign. The RMSE is more sensitive to large errors than the MAE. It characterizes the variation in errors.

IV. RESULTS AND DISCUSSIONS

In this section, a stacked LSTM network with three hidden LSTM layers is trained to approximate the nonlinear battery dynamics. Data collected from the DST test (7460 samples) and the US06 test (7013 samples) are used to train the network parameters. The input of the network is $\boldsymbol{x}_k = [V_k, I_k, T_k]$ while the output is the estimated SOC at the same time step, namely $y_k = [\text{SOC}_k]$. The performance of the network is evaluated with data from the FUDS test (7451 samples). All the training processes are carried out on our lab server, equipped with two GeForce GTX 1080 Ti GPUs.

A. EXPERIMENTS ON A FULLY CHARGED BATTERY

Fig. 5 shows the estimation results when the battery is fully charged, namely the initial SOC is 100%. The tracking results

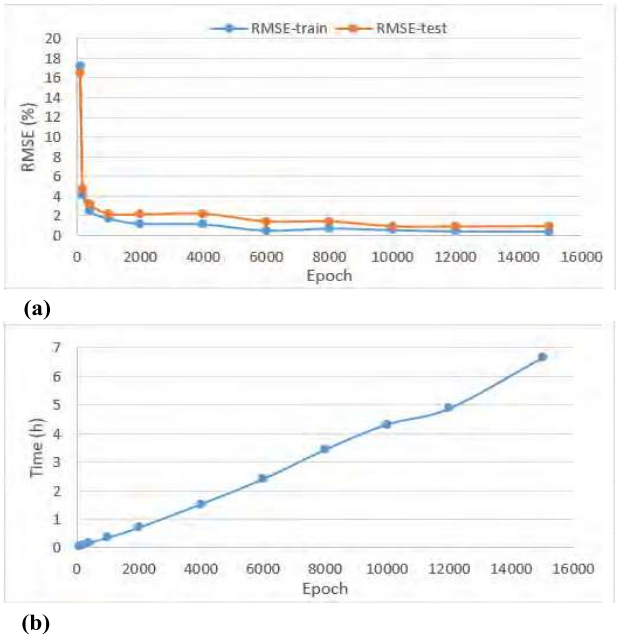

FIGURE 6. Performance of the LSTM network versus epoch number: (a) RMSE; (b) training time.

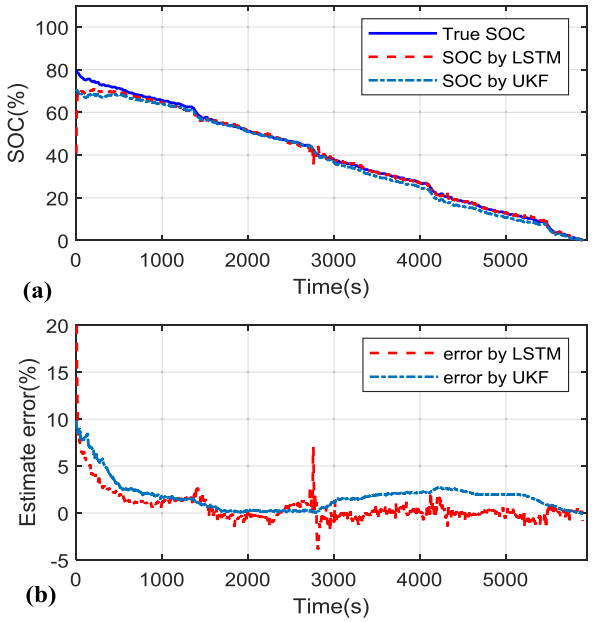

FIGURE 7. Results of SOC estimation with an initial SOC = 80%: (a) SOC tracking; (b) estimation error.

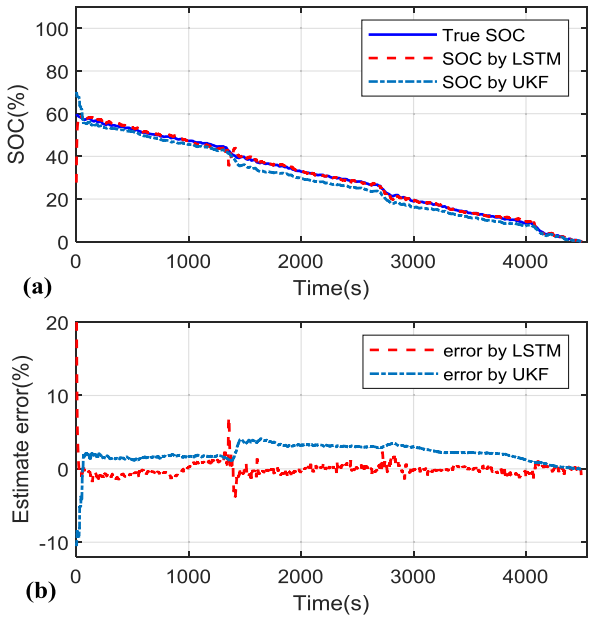

FIGURE 8. Results of SOC estimation with an initial SOC = 60%: (a) SOC tracking; (b) estimation error.

are showed in Fig. 5(a). The estimation error (evaluated as true value minus estimated value) is plotted in Fig. 5(b). The network tracks the true SOC during the whole discharge process, with overall maximum absolute error around 2%. The RMSE is 1.07% while the MAE is 0.84%. Under a GPU environment, the training time of the network is about 260 minutes, with 10000 epochs. For the testing, the average computation time at each time step is 0.058ms on our lab computer (OS: Win10 64-bit, CPU: Intel i7-7500U @2.70GHz, Memory: 8 GB), which is apparently fast enough for real-time on-board applications.

Fig. 6 shows the training and testing performance versus epoch numbers. The RMSEs under different training epochs are plotted in Fig. 6(a), which drop sharply in the first few hundred cycles, and then decrease slowly until around 10000 epochs, after which the RMSEs keep steady. While in general the increase of epoch number reduces the training and testing errors, the training time grows accordingly, which is demonstrated in Fig. 6(b), where a strong linear correlation is observed between training time and epoch number. Trading off between testing performance and training time, 10000 is selected as an optimal choice in this work, with a local optimal testing accuracy and a reasonable training time.

B. EXPERIMENTS WITH UNKNOWN INITIAL STATES

In real applications, it is unrealistic to know the precise initial SOC, such as when recover from hardware failure, reboot after battery replacement, or use after charge for a short time. It's thus important to test the robustness of the proposed network against unknown initial states. Fig. 7 shows the experimental results when the initial SOC starts from 80%. A large error is observed at the beginning for the estimation of proposed LSTM network. Then it converges to within 10% after several seconds. After about 500s, the estimated SOC approaches the true value. The overall RMSE and MAE are 1.76% and 0.96%, respectively. Fig. 8 shows the result with an initial SOC at 60%, where the estimation of proposed LSTM network converges to the true values in several seconds. The RMSE is 1.07% and the MAE is 0.53%. In both cases, the proposed network provides satisfying estimation results and presents a fast convergence ability against unknown initial states.

TABLE 2. Results of SOC estimation when initial SOC starts from various values.

Initial SOC (%)	RMSE (%)		MAE (%)	
	LSTM	UKF	LSTM	UKF
100	1.07	0.42	0.84	0.36
80	1.76	2.41	0.96	1.78
60	1.07	2.54	0.53	2.28
40	0.61	2.87	0.38	2.15
20	0.57	7.20	0.39	1.89

TABLE 3. Results of SOC estimation when the network is trained using two data sets.

Training	Testing	RMSE (%)	MAE (%)
DST, US06	FUDS	1.07	0.84
DST, FUDS	US06	1.29	1.08
US06, FUDS	DST	1.39	2.02

TABLE 4. Results of SOC estimation when the network is trained using one data set.

Training	Testing	RMSE (%)	MAE (%)
DST	US06	2.79	2.40
	FUDS	2.53	2.23
FUDS	DST	3.41	2.45
	US06	2.15	1.62
US06	DST	2.55	1.63
	FUDS	1.58	0.94

More experimental results are shown in Table 2, with initial SOC values among 100%, 80%, 60%, 40%, and 20%. The performance of the proposed network is also compared with the UKF method, which is experimentally shown to have a more stable, robust, and efficient estimation performance compared with other state-of-art model-based filtering methods such as extended Kalman filter and particle filter [6]. In this paper, the UKF filter for SOC estimation is constructed following basically the same work as in [6]. A combined model is adopted to construct the state-space model, and the variances of state function and measurement function are fine-tuned as 1e-8 and 1e-2, respectively.

When the initial SOC value is known (100%), the RMSEs of LSTM and UKF are 1.07% and 0.42% respectively, while the MAEs are 0.84% and 0.36%, respectively. The UKF outperforms the LSTM. The total computation time over 7451 timestamps is 0.436s for the LSTM and 0.235s for the UKF. The LSTM takes about two times as much computation time compared with the UKF.

When the initial SOC value varies from 80% to 20% (but keep the initial guess at 70%), the RMSE and MAE values of UKF increase, as in Table 2. The overall RMSE and MAE values are larger than those of LSTM. The blue dash-dot lines in Figs. 7 and 8 plot the estimation results when the initial SOC is 80% and 60%, respectively. In both cases, it takes more time for the UKF to track the true SOC. After approaching the true SOC, the UKF presents relatively larger tracking errors as well. For the proposed LSTM network, in all cases, the RMSE values vary within 2% and the MAE values vary within 1%. Compared with the UKF, the proposed LSTM network has stronger robustness against unknown initial SOC values and provides better tracking performance.

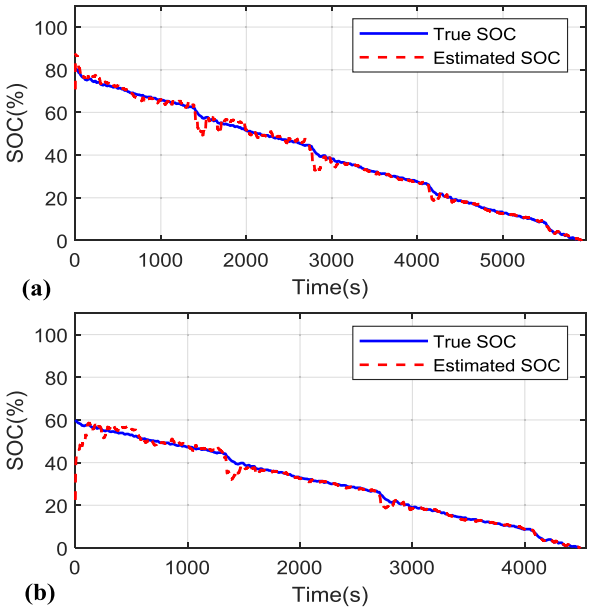

FIGURE 9. Results of SOC estimation on FUDS data set when the network is trained using US06 data set: (a) SOC starts from 80%; (b) SOC starts from 60%.

C. EXPERIMENTS WITH DIFFERENT TRAINING DATA SETS

The quantity and quality of training data sets have a significant influence on the estimation results. In this section, we train the network by using different training data sets.

In the first simulation, the network is trained using two different data sets and is tested on the remaining data set. The training epoch is still 10000, which carefully balances between the training time and testing error. The average training time is about 250 minutes. Table 3 summarizes the results of all cases. The RMSEs range from 1.07% to 1.39%, while the MAEs range from 0.84% to 2.02%.

In the second simulation, the network is trained using only one data set and then tested on the other two data sets. The training epoch is set to 3000 and the average training time is about 100 minutes. Table 4 tabulates the RMSE and MAE results of SOC estimation of all cases. The RMSEs range from 1.58% to 3.41%, while the MAEs range from 0.94% to 2.45%. The RMSEs and MAEs are almost double of those obtained in the first simulation. Fig. 9 shows the estimation results on FUDS data set when the network is trained using US06 data set. Fig. 9(a) plots the SOC tracking results when the initial SOC starts from 80% while Fig. 9(b) plots the results when initial SOC starts from 60%. Fig. 10 provides the

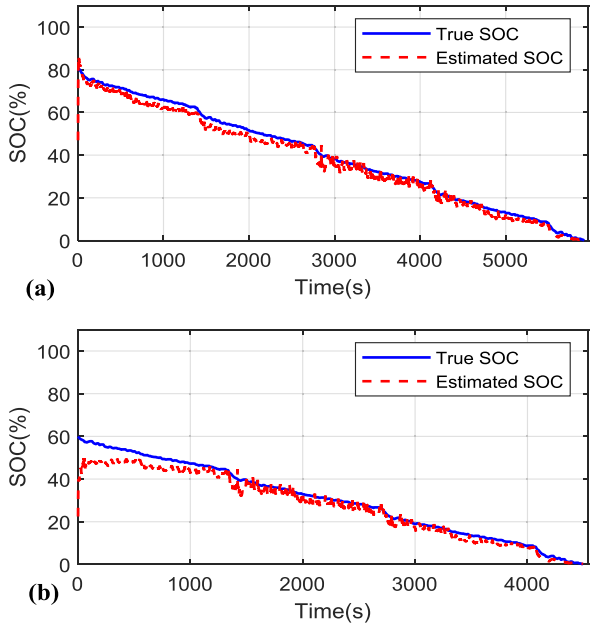

FIGURE 10. Results of SOC estimation on FUDS data set when the network is trained using DST data set: a) SOC starts from 80%; b) SOC starts from 60%.

estimation results when the network is trained using DST data set. Compared with results using two data sets for training (shown in Section IV-B), the estimation results show a comparative convergence ability against unknown initial states, however, presents more fluctuate estimation results. It is obvious that as more diversified data is provided, the estimation performance of proposed LSTM network increases.

V. CONCLUSIONS

In this paper, a stacked long short-term memory network was proposed to estimate the SOC of lithium iron phosphate batteries. Data were collected using different charge-discharge profiles, including DST, US06, and FUDS. These data were utilized for off-line training of the proposed network. Its performance on SOC estimation was then evaluated on the FUDS data set. Experimental results showed that the proposed network can successfully capture the non-linear correlation between SOC and measurable signals, namely current, voltage, and temperature, with maximum absolute error within 2%. In case of inaccurate initial SOCs, the network presented quick convergence to the true SOC, with slight increases of RMSE and MAE. As a comparison, UKF, the well-known model-based filtering method, showed worse tracking accuracy. Moreover, the average evaluation time on our lab computer was sub-millisecond, meeting the requirements of real-time applications such as modern EV BMSs. As a supplement, the performance of proposed LSTM network with different training and testing data sets was also studied. The estimation performance enhanced with more diversified data involved in the training process.

The proposed method is also a powerful framework for SOC estimation of lithium-ion batteries. It can be easily extended to consider more factors, such as humidity, degradation, and battery materials.

REFERENCES

[1] F. Yang, X. Song, G. Dong, and K.-L. Tsui, "A coulombic efficiency-based model for prognostics and health estimation of lithium-ion batteries," *Energy*, vol. 171, pp. 1173–1182, Mar. 2019.
[2] C. Huang, Z. Wang, Z. Zhao, L. Wang, C. S. Lai, and D. Wang, "Robustness evaluation of extended and unscented Kalman filter for battery state of charge estimation," *IEEE Access*, vol. 6, pp. 27617–27628, 2018.
[3] Y. Xing, W. He, M. Pecht, and K. L. Tsui, "State of charge estimation of lithium-ion batteries using the open-circuit voltage at various ambient temperatures," *Appl. Energy*, vol. 113. pp. 106–115, Jan. 2014.
[4] H. He, X. Zhang, R. Xiong, Y. Xu, and H. Guo, "Online model-based estimation of state-of-charge and open-circuit voltage of lithium-ion batteries in electric vehicles," *Energy*, vol. 39, no. 1, pp. 310–318, 2012.
[5] K. S. Ng, C.-S. Moo, Y.-P. Chen, and Y.-C. Hsieh, "Enhanced coulomb counting method for estimating state-of-charge and state-of-health of lithium-ion batteries," *Appl. Energy*, vol. 86, no. 9, pp. 1506–1511, 2009.
[6] F. Yang, Y. Xing, D. Wang, and K.-L. Tsui, "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*, vol. 164, pp. 387–399, Feb. 2016.
[7] F. Yang, D. Wang, Y. Xing, and K.-L. Tsui, "Prognostics of $Li(NiMnCo)O_2$-based lithium-ion batteries using a novel battery degradation model," *Microelectron. Rel.*, vol. 70, pp. 70–78, Mar. 2017.
[8] L. Lu, X. Han, J. Li, J. Hua, and M. Ouyang, "A review on the key issues for lithium-ion battery management in electric vehicles," *J. Power Sources*, vol. 226, pp. 272–288, Mar. 2013.
[9] A. J. Salkind, C. Fennie, P. Singh, T. Atwater, and D. E. Reisner, "Determination of state-of-charge and state-of-health of batteries by fuzzy logic methodology," *J. Power sources*, vol. 80, nos. 1–2, pp. 293–300, 1999.
[10] J. c. A. Antón, P. J. G. Nieto, C. B. Viejo, and J. A. V. Vilán, "Support vector machines used to estimate the battery state of charge," *IEEE Trans. Power Electron.*, vol. 28, no. 12, pp. 5919–5926, Dec. 2013.
[11] L. Kang, X. Zhao, and J. Ma, "A new neural network model for the state-of-charge estimation in the battery degradation process," *Appl. Energy*, vol. 121, pp. 20–27, May 2014.
[12] F. Yang, W. Li, C. Li, and Q. Miao, "State-of-charge estimation of lithium-ion batteries based on gated recurrent neural network," *Energy*, vol. 175, pp. 66–75, May 2019.
[13] W. He, N. Williard, C. Chen, and M. Pecht, "State of charge estimation for Li-ion batteries using neural network modeling and unscented Kalman filter-based error cancellation," *Int. J. Elect. Power Energy Syst.*, vol. 62, pp. 783–791, Nov. 2014.
[14] G. O. Sahinoglu, M. Pajovic, Z. Sahinoglu, Y. Wang, P. V. Orlik, and T. Wada, "Battery state-of-charge estimation based on regular/recurrent Gaussian process regression," *IEEE Trans. Ind. Electron.*, vol. 65, no. 5, pp. 4311–4321, May 2018.
[15] H. Chaoui and C. C. Ibe-Ekeocha, "State of charge and state of health estimation for lithium batteries using recurrent neural networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 10, pp. 8773–8783, Oct. 2017.
[16] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, Mar. 1994.
[17] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[18] S. Wang and J. Jiang. (2015). "Learning natural language inference with LSTM." [Online]. Available: https://arxiv.org/abs/1512.08849
[19] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo, "Image captioning with semantic attention," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 4651–4659.
[20] B. Zhang, S. Zhang, and W. Li, "Bearing performance degradation assessment using long short-term memory recurrent network," *Comput. Ind.*, vol. 106, pp. 14–29, Apr. 2019.
[21] G. Hunt, "USABC electric vehicle battery test procedures manual," United States Dept. Energy, Washington, DC, USA, Tech. Rep., 1996. [Online]. Available: https://avt.inl.gov/sites/default/files/pdf/battery/usabc_manual_rev2.pdf and https://scholar.google.com.hk/scholar?hl=zh-CN&as_sdt=0%2C5&q=USABC+electric+vehicle+battery+test+procedures+manual&btnG=

[22] F. J. Pineda, "Generalization of back-propagation to recurrent neural networks," *Phys. Rev. Lett.*, vol. 59, no. 19, p. 2229, 1987.
[23] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

FANGFANG YANG received the Ph.D. degree in systems engineering and engineering management from the City University of Hong Kong, in 2017, where she is currently a Postdoctoral Fellow with the School of Data Science. Her research interests include system health estimation, remaining useful life prediction, and deep learning.

XIANGBAO SONG received the M.Phil. degree in electronic and computer engineering from the Hong Kong University of Science and Technology (HKUST), Hong Kong, in 2016. He is currently a Researcher with Googol Technology (Shenzhen) Ltd., HKUST Shenzhen Research Institute. His research interests include robotics, motion planning, signal processing, prognostics and health management, and deep learning.

FAN XU received the Ph.D. degree in mechanical engineering from Wuhan University, Wuhan, China, in 2017. He is currently a Postdoctoral Fellow with the School of Data Science, City University of Hong Kong. His research interests include machinery condition monitoring, intelligent fault diagnostics, and deep learning.

KWOK-LEUNG TSUI was a Professor with the School of Industrial and Systems Engineering, Georgia Institute of Technology. He is currently the Chair Professor of industrial engineering with the School of Data Science, City University of Hong Kong, and the Founder and the Director of the Center for Systems Informatics Engineering. His current research interests include data mining, surveillance in healthcare and public health, prognostics and systems health management, calibration and validation of computer models, process control and monitoring, and robust design and Taguchi methods. He is a Fellow of the American Statistical Association, the American Society for Quality, and the International Society of Engineering Asset Management, and a U.S. representative to the ISO Technical Committee on Statistical Methods. He was a recipient of the National Science Foundation Young Investigator Award. He was the Chair of the INFORMS Section on Quality, Statistics, and Reliability and the Founding Chair of the INFORMS Section on Data Mining.