



UNIVERSITY OF LEEDS

This is a repository copy of *Hybrid Deep Neural Network for Electric Vehicle State of Charge Estimation*.

White Rose Research Online URL for this paper:

<https://eprints.whiterose.ac.uk/220530/>

Version: Accepted Version

Proceedings Paper:

Kadem, O., Candan, H. and Kim, J. orcid.org/0000-0002-3456-6614 (2024) Hybrid Deep Neural Network for Electric Vehicle State of Charge Estimation. In: 2024 IEEE 3rd International Conference on Electrical Power and Energy Systems (ICEPES). 2024 IEEE 3rd International Conference on Electrical Power and Energy Systems (ICEPES), 21-22 Jun 2024, Bhopal, India. IEEE ISBN 979-8-3503-9073-5

<https://doi.org/10.1109/icepes60647.2024.10653585>

© 2024 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

Reuse

Items deposited in White Rose Research Online are protected by copyright, with all rights reserved unless indicated otherwise. They may be downloaded and/or printed for private study, or other acts as permitted by national copyright laws. The publisher or other rights holders may allow further reproduction and re-use of the full text version. This is indicated by the licence information on the White Rose Research Online record for the item.

Takedown

If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing eprints@whiterose.ac.uk including the URL of the record and the reason for the withdrawal request.



eprints@whiterose.ac.uk
<https://eprints.whiterose.ac.uk/>

Hybrid Deep Neural Network for Electric Vehicle State of Charge Estimation

Onur Kadem
School of Mechanical Engineering
Bursa Technical University
Bursa, Türkiye
Email: onur.kadem@btu.edu.tr

Hasibe Candan
School of Computer Engineering
Bursa Technical University
Bursa, Türkiye
Email: hasibe.candan@btu.edu.tr

Jongrae Kim, Senior Member, IEEE
School of Mechanical Engineering
University of Leeds
Leeds, United Kingdom
Email: menjkim@leeds.ac.uk

Abstract—In electric vehicles (EVs), a battery management system plays a critical role in ensuring the reliable and safe operation of batteries. One of their main tasks is to monitor the battery’s state of charge (SoC), reflecting the battery’s current available charge level. However, the accuracy of SoC estimation is a formidable challenge due to the intricate nature of battery modelling. To overcome this challenge, data-driven methods have recently emerged as the dominant approach for SoC estimation. Considering the SoC estimation problem as a time series problem, we propose a hybrid deep neural network (DNN) that eliminates the need for feature engineering or adaptive filtering. The proposed DNN incorporates a convolutional layer, a long short term memory layer, and a dense layer. The DNN was trained using data collected from benchmark EV driving cycles (DST, BJDST, and FUDS drive cycles) within a temperature range of 0°C to 50°C. The performance evaluation of the trained DNN has been carried out using another standard EV driving cycle (US06 drive cycle) at various operating temperatures. The results demonstrate that the trained DNN effectively captures the dynamic behaviour of the battery under various operational conditions, yielding a maximum percentage SoC estimation error of approximately 3%. Furthermore, the results indicate that the proposed DNN technique is capable of generalising the battery’s dynamic response to unseen data. Overall, our findings show that the proposed technique is promising for EV applications in which battery operating conditions would exhibit significant variability.

Index Terms—Convolutional neural network, Long short term memory, Battery management system, Electrical battery, State of charge estimation

I. INTRODUCTION

The automotive industry has experienced a significant technological shift in recent years, particularly with the increasing adoption of electric vehicles (EVs). This transition is driven by a combination of factors, including increased environmental concerns, the implementation of regulatory standards, and technological developments. Advances in battery technology have played a pivotal role, contributing to improved energy density, extended range, and reduced costs. As a result, the automotive industry is evolving towards a more eco-friendly future, marking a crucial step toward achieving environmentally conscious and sustainable transportation.

The ongoing transformation highlights the crucial role of battery management systems (BMS). As this transformation accelerates, robust and reliable BMS development gains more

prominence. Therefore, researchers are paying more and more attention to developing a robust and accurate BMS. A robust BMS is critical to ensure the optimal performance, safety, and longevity of the batteries that power EVs. One of the essential duties of the BMS is to monitor the battery state of charge (SoC), which is defined as the proportion of the current available capacity of the battery to its nominal capacity. It serves as a critical parameter informing the driver about the EV’s remaining range, thereby enhancing overall confidence and satisfaction of EV ownership. However, due to the complicated electrochemical nature of batteries and their nonlinear behaviour, which is influenced by a range of internal and external factors, accurately estimating the SoC poses a considerable challenge. Consequently, there is a notable prevalence of research inquiries focused on achieving more precise SoC estimation among BMS researchers.

Three primary approaches to SoC estimation are available, comprising model-free methods such as Coulomb counting (CC) and open circuit voltage (OCV) measurement techniques, model-based methods including electrochemical and equivalent circuit model (ECM) based approaches, and data-driven methods [1]. The CC method involves integrating the current flowing in and out of the battery over a certain period of time [2]. This requires an accurate initial guess of the SoC and precise measurement of the current sensor. However, it is not capable of capturing changes in the dynamic behaviour of the battery as a result of the change in ambient temperature and capacity degradation. Another model-free technique is the OCV measurement method [3]. This method starts by measuring the terminal voltage of the battery when it is unloaded and allowed to rest until it reaches a steady state, where the terminal voltage is presumed to be equivalent to the battery’s OCV. Following this, the corresponding SoC estimate can be determined from a predefined look-up table. However, implementing this method faces practical challenges for two primary reasons. Firstly, the necessity of a resting period makes it unfeasible in dynamic operational conditions. Secondly, the look-up table is established under controlled laboratory conditions and their values are constant, rendering it inaccurate over time as the battery undergoes usage [5].

Typically, model-based techniques rely on current and voltage measurements to compute the SoC through closed-

loop structures. These approaches involve two main battery modelling strategies to characterise battery dynamics. The first strategy employs electrochemical modelling, utilizing a set of partial differential equations to depict the electrochemical processes occurring inside the battery [4]. Although effective, this method tends to be computationally demanding and time-consuming, making it more suitable for laboratories aiming to optimise battery designs. The second strategy involves replicating the battery’s dynamic behavior using an ECM. ECM-based methods mimic the battery dynamics through circuit elements such as resistors, capacitors, and open voltage sources. ECMs are an alternative that is computationally more efficient and less time-consuming than electrochemical modelling. However, ECM-based models also rely on a predefined non-linear relationship between the SoC and the OCV. This constitutes the most challenging aspect of ECM-based methods, as the relationship is predominantly determined in laboratories, resulting in a fixed relationship. Nevertheless, this relationship is expected to vary with changes in operational conditions and the aging of the battery [5]. Consequently, the constant SoC-OCV relationship leads to an escalation in SoC estimation errors in the long run. Furthermore, model-based techniques have the following critical challenges:

- Implementing a battery model necessitates the accurate parameterisation of the proposed model. Electrochemical battery models involve a multitude of parameters. In contrast, ECMs have fewer parameters, yet their SoC-OCV relationship poses another challenge [6].
- Physical modelling of the battery ageing characteristic is challenging. The Solid Electrolyte Interphase (SEI) is a protective layer that develops on the electrodes of a battery. Estimating the growth of SEI is vital to accurately determining the battery state of health (SoH) related to the SoC calculation. Yet, predicting its growth encounters a multiscale phenomenon involving processes at various levels, posing a challenge to model comprehensively [7].

Because of the drawbacks and constraints of the aforementioned SoC estimation techniques, battery researchers are driven to discover a more dependable and effective SoC estimation method. The rapid advancements in artificial intelligence have led to the increasing popularity of data-driven SoC estimation methods in recent years. Deep learning techniques are gaining prominence in BMS design techniques due to their ability to establish end-to-end connections between input and output without requiring an exhaustive understanding of battery chemistry or dynamics. In [8], a deep neural network (DNN) with 5 layers is created to predict SoC. The initial layer of the DNN is designated as a convolutional layer, tasked with extracting features from the input data. Subsequently, the max-pooling layer and another convolutional layer are added to the DNN. The fourth layer is a gated recurrent unit (GRU) layer designed to capture sequential dependencies within the input data. Finally, a dense layer with 5 neurones is appended to the GRU layer. Each data sample includes current and voltage series as input, and the output is the corresponding

SoC determined using the CC method. In [9], the researchers introduced a long-short-term memory (LSTM) network integrated with transfer learning (TL). They approached the SoC estimation challenge as a time-series problem and opted for the LSTM architecture to address it. Transfer learning was subsequently employed to fine-tune the parameters of the pre-trained DNN, enabling the model to adapt to temperature variations and battery degradation over time. Song et al. used a combined DNN, incorporating CNN to extract interrelations within input data and LSTM to characterise the long-term dependencies inherent in SoC estimation when treated as a time series problem [10]. As input, the network takes current, voltage, and temperature measurements, along with average current and voltage calculations over a certain window size. The output corresponds to the estimated SoC.

In deep learning techniques, effective data collection is of significant importance. According to [11], the training dataset comprises voltage and current readings obtained during battery operations. The reference SoC is determined through the CC method. Furthermore, achieving a precise initial SoC involves a thorough process of fully charging and fully discharging the battery. Subsequently, the supervised training of a DNN can be facilitated using the current and voltage data, along with the reference SoC.

In this work, we calculated the SoC under various operating conditions using a trained DNN. The organisation of the paper is as follows: Section II presents the DNN architecture explaining the layers employed in DNN; Section III introduces the data extraction and preparation, and DNN training; Section IV explains the battery simulations and results; finally, conclusions and future work are discussed in Section V.

II. DEEP NEURAL NETWORK STRUCTURE

In this study, a DNN consisting of five layers shown in Fig. 1 is built. The initial layer serves as the input layer, followed by a convolutional layer with 16 filters, each with a kernel size of three. The convolution operation employs a stride of one. Due to the nature of our SoC estimation problem, a pooling layer was deemed unnecessary for downsampling spatial dimensions and reducing computational load. The primary goal of the CNN layer is to extract spatial features from the input data. Following the CNN layer, an LSTM layer with 32 nodes is included to capture the evolution of the temporal dynamics of the battery. Subsequently, a dense layer with 32 nodes is introduced for regression purposes, enhancing the DNN’s capacity to accommodate the nonlinear dynamics of the battery.

Finally, the output layer is employed, using a single node with a sigmoid activation function, to consolidate the estimation results within the range of 0 to 1, where the sigmoid function, $\sigma(\cdot)$, is given by

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (1)$$

and x is the input.

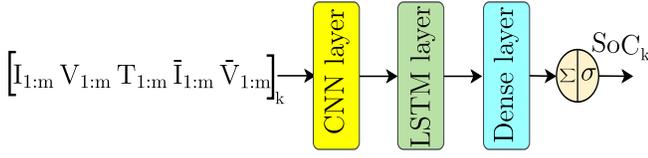


Fig. 1. Deep Neural Network Structure

A. Convolutional layer

In image processing, 2D (2-Dimensional)-CNNs are recognized as the predominant deep learning algorithms. However, for the specific task of SoC estimation, 1D (1-Dimensional)-CNNs are employed due to the temporal characteristics of the input features. In contrast to 2D-CNNs, where kernels traverse spatial dimensions both horizontally and vertically, 1D-CNNs exclusively utilise striding along a single dimension. This unidimensional striding facilitates the capture of temporally relevant features by the kernels, rendering 1D-CNNs well-suited for extracting information within the context of SoC estimation.

The standard CNN architecture typically consists of a convolutional layer, followed by a pooling layer, and then a flatten layer. On the contrary, our DNN structure is different from this convention by excluding the pooling and flattening layers, as shown in Fig. 2 a). The omission of the pooling layer is to preserve the dimensionality of the input vector. Furthermore, the flatten layer, which is traditionally employed to collapse the output dimensions, is unnecessary for our problem because the output of the CNN layer is the input to the LSTM layer which accepts a three-dimensional input.

A 1D convolutional layer utilized in this work processes an input vector, which comprises measurements represented by $[\tilde{I}_{1:m}, \tilde{V}_{1:m}, \tilde{T}_{1:m}, \bar{I}_{1:m}, \bar{V}_{1:m}]$, where I is the current, V is the terminal voltage, T is the ambient temperature and \bar{I} and \bar{V} are the average current and average voltage, respectively. The symbol (\cdot) represents the measurement of the corresponding variable. $(\cdot)_{1:m}$ denotes all values of the variable from the initial time step to the m -th time step. Training a DNN using mini-batches is used not only to improve computational and memory efficiency but also to provide advantageous stochasticity during the training process, contributing to improved generalization and faster convergence. Consequently, the arrangement of the input is formalised as a three-dimensional array, illustrated in Fig. 2 a). In this representation, the height corresponds to the batch size, the width represents the time steps in a k -th calculation step, and the depth signifies the features including sensor measurements and average values. The incorporation of the 1D convolutional layer into the DNN is achieved through the utilisation of the Keras library. The activation function employed is the Rectified Linear Unit (ReLU). ReLU returns zero for negative inputs and the inputs' value for positive inputs. It is mathematically represented as follows:

$$f_{\text{ReLU}}(x) = \max(0, x) \quad (2)$$

where f_{ReLU} is the ReLU function. ReLU accelerates the

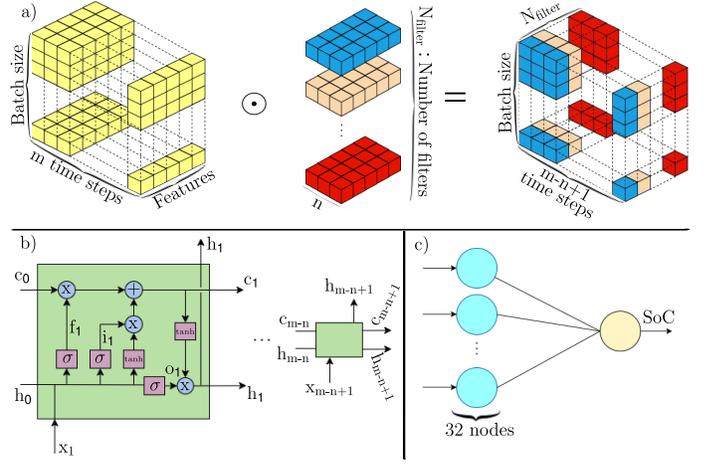


Fig. 2. Detailed visualisation of the layers used in DNN architecture: a) 1D convolutional layer, b) LSTM layer, c) Dense layer with one node output layer

learning process by preserving and computing the gradients more efficiently.

B. Long short term memory layer

When the problem involves dealing with a sequence of data over time, using a recurrent neural network (RNN) can be more advantageous. RNNs can capture information on historical elements in the data sequence that adopt a hidden state. In detail, the hidden state at the current calculation step is calculated by both the current input and the hidden state from the previous calculation time step. This recurrent calculation makes RNNs capture temporal patterns and suitable for time-series problems such as the SoC estimation problem. However, traditional RNNs suffer from a gradient vanishing problem. To overcome this, more sophisticated variants of RNNs are introduced.

LSTM network is categorised as a type of these RNNs, is introduced to address issues related to gradient explosion and gradient vanishing. The occurrence of gradient vanishing arises when weight updates become excessively small, impeding the training process due to the challenges in updating the model. On the contrary, a gradient explosion occurs when weight updates become excessively large, leading to inconsistent model training. LSTM effectively mitigates these problems through its distinctive cell structure, which includes three gates: the input gate, the forget gate, and the output gate as shown in Fig. 2 b). At time step k , the LSTM cell processes the following operations:

$$\begin{aligned} i_k &= \sigma(x_k U^i + h_{k-1} W^i + b_i) \\ f_k &= \sigma(x_k U^f + h_{k-1} W^f + b_f) \\ o_k &= \sigma(x_k U^o + h_{k-1} W^o + b_o) \\ \tilde{c}_k &= \tanh(x_k U^c + h_{k-1} W^c + b_c) \\ c_k &= f_k * c_{k-1} + i_k * \tilde{c}_k \\ h_k &= \tanh(c_k) * o_k \end{aligned} \quad (3)$$

TABLE I
BATTERY SPECIFICATIONS

Battery specification	Value
Battery type	Lithium-ion
Upper cut-off voltage	4.2 [V]
Lower cut-off voltage	2.5 [V]
Nominal battery capacity	2.28 [Ah]
Battery ageing condition	Fresh
Number of cells	1

where \odot is the element-wise product, x_k is the input to the LSTM cell, h_k is the output of the LSTM cell. c_k is the hidden state memory of the LSTM cell and \tilde{c}_k is a hidden state calculated based on the current input and previous hidden state. f_k , i_k , and o_k are the activation vectors for the forget gate, the input gate, and the output gate, respectively. σ is a sigmoid activation function whereas \tanh is a tangent hyperbolic activation function. Finally, W and U are the weight matrices and b is the bias vector.

C. Dense layer

A dense layer comprising 32 nodes is incorporated into the LSTM layer to enhance the regression capability of the suggested deep neural network to address the SoC estimation problem. ReLU is employed as an activation function. Subsequently, a single node output layer with a sigmoid activation function is introduced into the neural network architecture, as SoC estimation typically falls within the range of 0 to 1. The dense layer and the output layer are shown in Fig. 2 c).

III. DATA GENERATION AND NEURAL NETWORK TRAINING

The training data was extracted using the electrochemical battery model implemented in the PyBamm library, a Python tool designed by researchers to advance battery research [12]. The battery model is the Doyle-Fuller-Newman (DFN) lithium ion model [14], featuring a capacity of 2.28 A.h, an upper cut-off voltage of 4.2 V, and a lower cut-off voltage of 2.5 V. The battery consists of a single cell whose nominal capacity is 2.28 [Ah] and is a fresh battery with 100% SoH. The summary of simulated battery specifications is given in Table I.

In practice, a battery operating temperature below 0°C increases the internal resistance, which could potentially damage the battery while charging the battery at high voltages. Likewise, it is equally important to avoid exceeding 50°C to mitigate accelerated battery ageing and self-discharge due to accelerated chemical activities in the battery cell [15]. Therefore, the temperature is chosen in the range of 0°C to 50°C in our work.

In the simulations, a fully charged battery undergoes different standard driving cycles under various temperatures such as dynamic stress test (DST), federal urban driving schedule (FUDS), Beijing driving schedule test (BJDST), and supplemental driving cycle US06. The driving cycle profiles are given in Fig. 3. Note that the training data is generated through DST, FUDS, and BJDST driving cycles, while the US06

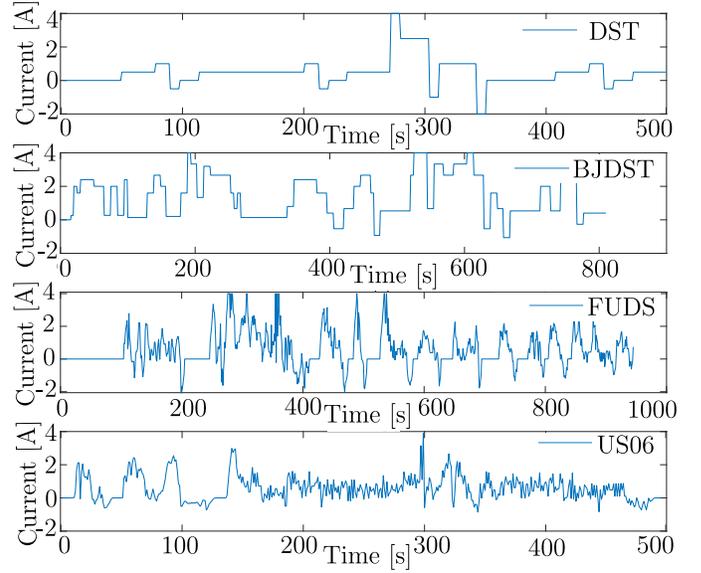


Fig. 3. Standard drive cycles for battery testing

drive cycle profile is specifically employed to test the trained DNN performance. Prior to each simulation, the battery is fully charged via a constant current-constant voltage (CCCV) charging procedure. Initially, a charging current of 1 A is applied to the battery until the terminal voltage reaches 4.2 V, after which the charging current gradually decreases to 0.05 A while the terminal voltage remains at 4.2 V. After this procedure, the battery is considered fully charged for subsequent discharge simulations. The data are recorded every 1s, the average current and average voltage are calculated every 60s.

The collected battery data are then processed to create data with sequences consisting of m successive measurements. The process of generating data, where sequences of inputs and their corresponding outputs are formed, is depicted in Fig. 4 where m corresponds to a window size employed to scan the data, N is the final time step in data. For example, m number of consecutive measurements of I , V , T , \bar{I} , and \bar{V} and the m -th SoC value form the first sequence in the processed data.

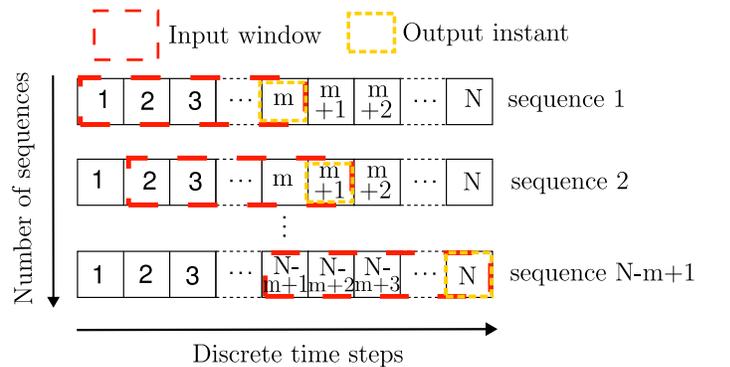


Fig. 4. Visualization of the process of generating data with sequences

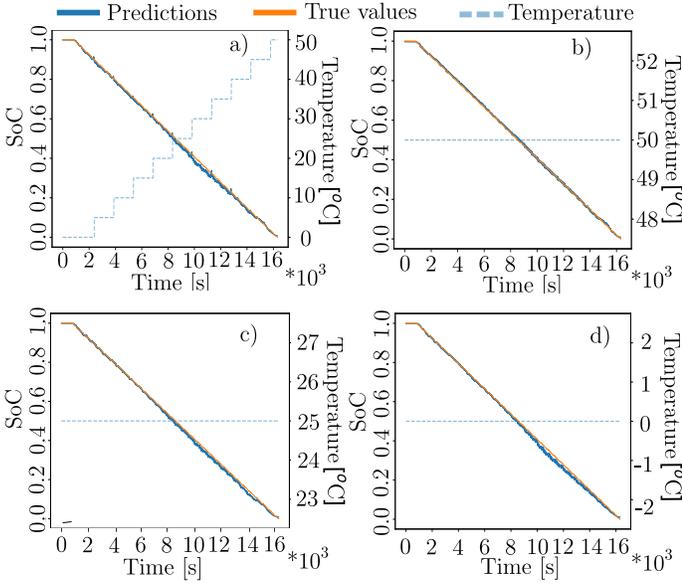


Fig. 5. SoC estimation results: a) SoC estimation result while the ambient temperatures increase over time, b) SoC estimation result at 50°C, c) SoC estimation result at 25°C, d) SoC estimation result at 0°C

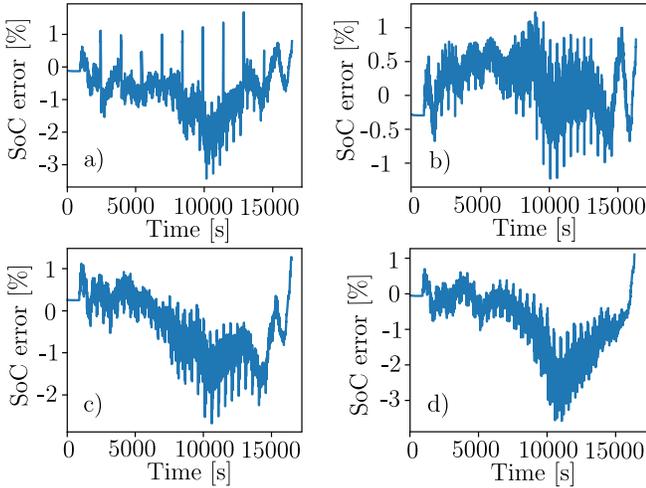


Fig. 6. SoC estimation percentage errors: a) SoC error while the ambient temperatures increase over time, b) SoC error at 50°C, c) SoC error at 25°C, d) SoC error at 0°C

After data processing, the white noise is added to the measurements. In [13], the standard deviations for current and voltage measurement errors are suggested to be 0.05 mA and 0.05 mV, respectively. Similarly, white noise with the standard deviation 1°C is added to the temperature measurements.

IV. SIMULATIONS AND RESULTS

In our work, m is set to 60. Training and testing data are then generated, consisting of the consecutive 60 data points along with their associated SoC value at the end of the data sequence. The corresponding SoC value is calculated using the Coulomb counting method. The DNN is trained with data extracted as explained above. The optimiser used in the training is the

ADAM optimiser. The loss function of MSE is adopted and the DNN is trained with 1000 epochs. The batch size is set to 2048. The initial learning rate is set to 0.0001. The decay rates are set to 0.9 and 0.999, respectively.

$$\text{MSE} = \frac{1}{K} \sum_{k=1}^K (y_k - \hat{y}_k)^2 \quad (4)$$

where y_k is the reference SoC value whereas \hat{y}_k is the estimated SoC value at calculation step k .

SoC estimation results are given in Fig. 5, where Fig. 5a) shows the SoC estimation result when the operating temperature changes during the simulation. The ambient temperature is increased from 0°C to 50°C by 5°C. Fig. 5b), Fig. 5c), and Fig. 5d) shows the SoC estimation results at constant but different temperatures, 50°C, 25°C, and 0°C, respectively. The SoC is estimated based on the data through the US06 drive cycle. Note that the US06 drive cycle is not used to extract the training data. The estimation results show that the trained DNN can capture the dynamic response of the battery at different temperatures and current profiles with high accuracy. Although the training data does not include the data at 5°C, 15°C, 25°C, 35°C, 45°C, the trained DNN performs a promising accuracy in SoC estimation results. These results indicate that the model has learned to generalise underlying patterns and features, resulting in accurate predictions on unseen data.

The percentage error is relatively greater in middle-low SoC regions, i.e., 40% and 10%, compared to the rest of the SoC regions when the temperature is 25°C and below as shown in Fig. 6. In this region, the maximum percentage error reaches around 3% at 0°C whereas it is around 1% at 50°C. This is due to the battery's SoC-OCV curve characteristic whose slope approaches zero in this region at low temperatures.

The loss evolution is plotted in Fig. 7. The loss exhibits a rapid decline and reaches a stable state after approximately 200 epochs. Nevertheless, fluctuations in the loss exist even after stabilisation, indicating that the optimisation algorithm transitions between one local optima to another.

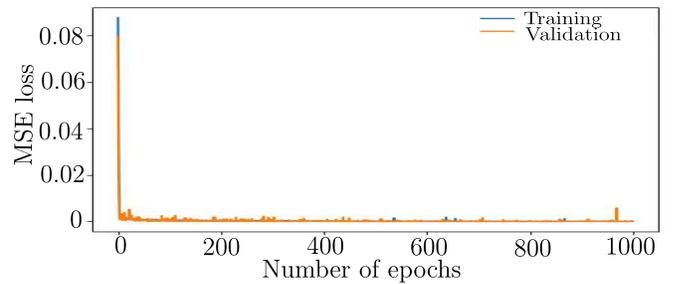


Fig. 7. MSEs of the training and validation performance

V. CONCLUSION AND FUTURE WORK

The SoC estimation problem is assumed as time series problem. Therefore, the CNN-LSTM DNN is proposed to capture the dynamic behaviour of the battery under various operating conditions. The CNN is to capture the spatial features and

LSTM is to capture the time series dependencies. A dense hidden layer is also employed to increase the ability of the DNN to better fit the non-linear behaviour of the battery. The training and testing data are generated based on the DFN electrochemical battery model. The DST, BJDST and FUDS drive cycles are used to generate the training data at various ambient temperatures ranging from 0°C to 50°C, whereas the US06 drive cycle is used to test the performance of the trained DNN at different temperatures. In the simulations, the maximum percentage SoC estimation error is around 3%. The maximum error is observed at 0°C, which is due to a significant increase in the capacity loss as the operating temperature decreases. Overall, a hybrid neural network consisting of LSTM, CNN, and dense layers shows a promising SoC estimation accuracy for batteries in practice, where the operational conditions vary with time.

Our future work will focus on real-time fine-tuning the DNN weights that were initially trained using synthetic data. This fine-tuning process will involve incorporating sensor data obtained during actual battery operation. The objective is not only to eliminate the time-consuming and labor-intensive tasks associated with laboratory work and experimental costs but also to ensure that the DNN can effectively adjust to variations in battery dynamics over time. Fine-tuning process would require another SoC estimator so we will integrate the current DNN with another SoC estimation algorithm, leading to a hybrid estimator. Accurate SoC estimates would provide information about the change in the battery SoH, which is another challenging problem to be addressed in EV applications. Finally, we will propose a SoH estimation approach that relies on the accurate estimation of the SoC.

ACKNOWLEDGMENT

The authors would like to thank to PyBamm team and all anonymous scientists who contributed to the PyBamm library for their great effort in designing a comprehensive and successful library for battery simulations.

REFERENCES

[1] R. Xiong, J. Cao, Q. Yu, H. He and F. Sun, "Critical Review on the Battery State of Charge Estimation Methods for Electric Vehicles," in *IEEE Access*, vol. 6, pp. 1832-1843, 2018, doi: 10.1109/ACCESS.2017.2780258.

[2] Y. -M. Jeong, Y. -K. Cho, J. -H. Ahn, S. -H. Ryu and B. -K. Lee, "Enhanced Coulomb counting method with adaptive SOC reset time for estimating OCV," 2014 IEEE Energy Conversion Congress and Exposition (ECCE), Pittsburgh, PA, USA, 2014, pp. 1313-1318, doi: 10.1109/ECCE.2014.6953989.

[3] Lei Pei, Tiansi Wang, Rengui Lu, Chunbo Zhu, Development of a voltage relaxation model for rapid open-circuit voltage prediction in lithium-ion batteries, *Journal of Power Sources*, Volume 253, 2014, Pages 412-418, ISSN 0378-7753, <https://doi.org/10.1016/j.jpowsour.2013.12.083>. (<https://www.sciencedirect.com/science/article/pii/S0378775313020624>)

[4] M. Doyle, T. F. Fuller, and J. Newman, "Modeling of galvanostatic charge and discharge of the lithium/polymer/insertion cell," *J. Electrochem. Soc.*, vol. 140, no. 6, pp. 1526–1533, Jun. 1993, doi: 10.1149/1.2221597.

[5] O. Kadem and J. Kim, "Real-Time State of Charge-Open Circuit Voltage Curve Construction for Battery State of Charge Estimation," in *IEEE Transactions on Vehicular Technology*, vol. 72, no. 7, pp. 8613-8622, July 2023, doi: 10.1109/TVT.2023.3244623.

[6] M.A. Hannan, M.S.H. Lipu, A. Hussain, A. Mohamed, A review of lithium-ion battery state of charge estimation and management system in electric vehicle applications: Challenges and recommendations, *Renew. Sustain. Energy Rev.* 78 (2017) 834–854, <https://doi.org/10.1016/j.rser.2017.05.001>.

[7] Satu Kristiina Heiskanen, Jongjung Kim, Brett L. Lucht, Generation and Evolution of the Solid Electrolyte Interphase of Lithium-Ion Batteries, *Joule*, Volume 3, Issue 10, 2019, Pages 2322-2333, ISSN 2542-4351, <https://doi.org/10.1016/j.joule.2019.08.018>. (<https://www.sciencedirect.com/science/article/pii/S2542435119304210>)

[8] Jinpeng Tian, Rui Xiong, Weixiang Shen, Jiahuan Lu, State-of-charge estimation of LiFePO4 batteries in electric vehicles: A deep-learning enabled approach, *Applied Energy*, Volume 291, 2021, 116812, ISSN 0306-2619, <https://doi.org/10.1016/j.apenergy.2021.116812>. (<https://www.sciencedirect.com/science/article/pii/S0306261921003147>)

[9] Yu Liu, Xing Shu, Hanzhengnan Yu, Jiangwei Shen, Yuanjian Zhang, Yonggang Liu, Zheng Chen, State of charge prediction framework for lithium-ion batteries incorporating long short-term memory network and transfer learning, *Journal of Energy Storage*, Volume 37, 2021, 102494, ISSN 2352-152X, <https://doi.org/10.1016/j.est.2021.102494>. (<https://www.sciencedirect.com/science/article/pii/S2352152X21002437>)

[10] X. Song, F. Yang, D. Wang and K. -L. Tsui, "Combined CNN-LSTM Network for State-of-Charge Estimation of Lithium-Ion Batteries," in *IEEE Access*, vol. 7, pp. 88894-88902, 2019, doi: 10.1109/ACCESS.2019.2926517.

[11] Jinpeng Tian, Cheng Chen, Weixiang Shen, Fengchun Sun, Rui Xiong, Deep Learning Framework for Lithium-ion Battery State of Charge Estimation: Recent Advances and Future Perspectives, *Energy Storage Materials*, Volume 61, 2023, 102883, ISSN 2405-8297, <https://doi.org/10.1016/j.ensm.2023.102883>. (<https://www.sciencedirect.com/science/article/pii/S2405829723002623>)

[12] Sulzer, Valentin, Scott G. Marquis, Robert Timms, Martin Robinson, and S. J. Chapman. 2020. "Python Battery Mathematical Modelling (pybamm)." *ECSarXiv*. February 7. doi:10.1149/osf.io/67ckj.

[13] Kadem O, Kim J. Mitigation of state of charge estimation error due to noisy current input measurement. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*. 2024;238(1):34-46. doi:10.1177/09596518231182308

[14] M. Doyle, T. F. Fuller, and J. Newman, "Modeling of galvanostatic charge and discharge of the lithium/polymer/insertion cell," *J. Electrochem. Soc.*, vol. 140, no. 6, pp. 1526–1533, Jun. 1993, doi: 10.1149/1.2221597.

[15] Juuso Lindgren, Peter D. Lund, Effect of extreme temperatures on battery charging and performance of electric vehicles, *Journal of Power Sources*, Volume 328, 2016, Pages 37-45, ISSN 0378-7753, <https://doi.org/10.1016/j.jpowsour.2016.07.038>. (<https://www.sciencedirect.com/science/article/pii/S0378775316308941>)