Lithium-Ion Battery Model Development and State Estimation for an Extended Single Particle Model Coupled with Thermal Modeling

by

Brianna Chevalier

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

in

Process Control

Department of Chemical and Materials Engineering
University of Alberta

Abstract

Lithium-ion batteries are the predominant battery type in portable consumer electronics and electric vehicles. Compared to experimental studies, simulation studies tend to be more cost-effective. Trustworthy simulation results depend on high-fidelity electrochemical-thermal dynamic models. To balance computational efficiency with exactness, we propose an enhanced model through an extended single-particle model with thermal dynamics to forecast the internal battery states. Experimental results from literature verify the effectiveness of the developed model. Along this line, we provide a comprehensive analysis of dynamic behaviour of a lithium-ion battery cell using the proposed model. Simulation shows promising results while maintaining computation speed. The developed model does not depend on any commercial packages and provides a high-fidelity simulation platform for battery research and development. Moreover, to enable the effective operation of portable electronics and electric vehicles, accurate and quick estimations of state of charge (SOC) and internal cell temperature are vital to battery management systems. Therefore, a long-short-term-memory (LSTM) recurrent-neural network is proposed which completes the state estimation of SOC and internal average cell temperature (Tavg) of lithium-ion batteries under varying current loads. The network is trained and evaluated using data compiled from the developed extended single particle model coupled with a thermal dynamic model. Results are promising, with root mean square error values typically under 2 % for SOC and 1.2K for Tavg, while maintaining quick training and testing times. Also examined was a comparison of a single-feature versus multi-feature network, as well as two different approaches to data partitioning.

Preface

This thesis is an original work completed by Brianna Chevalier.

Chapter 2 is in the second stage of review for publication in the Canadian Journal of Chemical Engineering, with Professor Dubljevic and Junyao Xie as contributors. Additionally, the contents of Chapter 2 were accepted and presented at the Canadian Society for Chemical Engineering conference, in a full presentation in 2023. A less comprehensive version of the study shown in Chapter 3 was accepted for a full presentation at the 2023 American Institute of Chemical Engineering conference. Meanwhile, Chapter 3 has also been submitted for publication in the Canadian Journal of Chemical Engineering, with Professor Dubljevic and Junyao Xie as contributors

I was responsible for literature review, model development and implementation, data collection, machine learning network development and coding, and the writing and editing of the manuscripts. Professor Dubljevic and Dr. Xie were supervisory authors and assisted with the conceptualization, realization approaches, and revisions.

Acknowledgements

I would like to express the utmost gratitude to my supervisor Professor Dubljevic, for giving me the opportunity to join his research group and for all of the support he provided throughout this journey. I would also like to thank Dr. Xie for his guidance and patience as he supervised and assisted me in completing this work and this program. Additionally, I extend my great appreciation to Dr. Ozzorio Cassol and Dr. Zhang, who surrendered their time and knowledge despite their incredibly busy schedules, and for their interest in my research, well-being, and scholarly advice. I am also deeply grateful to the Department of Chemical and Materials Engineering, and to the Faculty of Graduate and Postdoctoral Studies (GPS) for all of the financial support provided to me; especially the GPS Graduate Student Elevation Award and the Graduate Travel Award which enhanced my experience as a graduate student at the University of Alberta even further. Additionally, I also owe the Natural Sciences and Research Council of Canada (NSERC) as their funding enabled me to undertake this program. Furthermore, I would like to thank my family, and other group members who supported me during my program through their kindness, encouragement, and meaningful discussions. Incredibly important to the completion of this thesis was my partner Lucas, who encouraged and comforted me especially when I was struggling.

Table of Contents

List of Tables

List of Figures

List of Symbols

SPMeT Quantitative Variables

b_k	Particle surface area per unit of volume.	m^{-1}
c_k^{max}	Maximum particle concentration.	$Molm^{-3}$
C_c	Core heat capacity.	$270,000JK^{-1}kg^{-1}$
c_p	Specific heat capacity.	$11,000JK^{-1}kg^{-1}$
C_u	Surface heat capacity.	$400JK^{-1}kg^{-1}$
c_{e0}	Electrolyte initial concentration.	$1,200Molm^{-3}$
c_{k0}	Electrode initial concentration.	$Molm^{-3}$
D_e	Electrolyte lithium diffusivity in particle.	$2.010 \wedge (-14)M^2s^{-1}$
D_k	Lithium diffusivity in particle.	M^2s^{-1}
E_a	Activation energy.	$26.6Jmol^{-1}$
k^k	Reaction rate in electrode k $mol^{1+\alpha}$	$Amol^3m^{-2}$
k_i	Integral controller gain.	$0.0005 - 0.023$
k_p	Proportional controller gain.	$0.0008 - 0.01$
L_k	Electrode and separator thicknesses.	m
R_c	Conduction resistance.	$4KW^{-1}$
R_f^k	Solid-electrolyte interphase film resistance.	Ω^2
R_k	Particle k radius.	m
R_u	Convection resistance.	$100KW^{-1}$
t^+	Cation transference number.	0.4
T_0	Initial temperature.	$288.15K$
T_{amb}	Ambient temperature.	$288.15K$

Model Symbols

c_e	Lithium ion concentration in the electrolyte.	$Molm^{-3}$
c_e^l	Concentration of lithium ions in the electrolyte in domain l.	$Molm^{-3}$
c_k	Lithium concentration in the electrode particles.	$Molm^{-3}$
c_k^s	Concentration at particle surface in cathode/anode.	$Molm^{-3}$
c_k^{bulk}	Bulk concentration for electrode k.	$Molm^{-3}$
i_{app}	Applied current.	Am^{-2}
i_e	Current density in electrolyte.	Am^{-2}
i_k	Current density in electrode k.	Am^{-2}
j_k	Reaction current density.	Am^{-2}
j_{k0}	Exchange current density.	Am^{-2}
j_{k0}^{avg}	Averaged exchange current density; function of time only.	Am^{-2}
N_e	Molar flux of lithium in the electrolyte.	$Molm^{-2}s^{-1}$
N_k	Molar flux of lithium in the electrode particles.	$Molm^{-2}s^{-1}$
SOC	State of charge.	
T_{avg}	Average internal cell temperature.	K
T_c	Core temperature.	K
T_s	Surface temperature.	K
U_k	open-circuit potential (OCP).	V

Greek

α	Charge transfer coefficient.	0.5
β	Transport efficiency.	
δQ	Tolerance value to final capacity.	
η_k	Overpotential at the electrode-electrolyte interface.	V
ϕ_e	Electrolyte potential.	V
ϕ_k	Electrode potential.	V
σ	Standard deviation for normalization of data; sigmoid activation function in neural networks.	
σ_e	Electrolyte conductivity.	Sm^{-1}

σ_k	Electrode k conductivity.	Sm^{-1}
ε	Electrolyte volume fraction.	

Constants

F	Faraday constant.	$96485.33 Cmol^{-1}$
R	Universal gas constant.	$8.314472 JK^{-1}mol^{-1}$

Machine Learning Symbols

$\tilde{c_t}$	Candidate cell state values; determined by gate gate.
b_k	$k \in i, f, o, g$; set of biases, different for each associated gate.
c_t	Cell state at time step t.
f_w	Function with parameters w, where w is a set of weights
h_t	Hidden state at time step t.
i_t	Input gate; determines which info to be erased or kept.
o_t	Output gate.
w_k	$k \in i, f, o, g$; set of weight matrices, different for each associated gate.
X_i	General variable at step i.
x_t	Input variable at time step t.
X_{std}	Standardized/normalized general variable X.
Y	Testing value from input dataset.
Y_{Pred}	Predicted value from the network.

Abbreviations

BMS Battery management system.

CC-CV Constant current - constant voltage; form of charging scheme.

CNN Convolutional neural network.

DFN Doyle Fuller Neumann battery model.

DT Digital twin.

ECM Equivalent circuit model.

EV Electric vehicle.

GRU Gated recurrent unit; form of RNN.

HEV Hybrid electric vehicle.

LFP Lithium iron phosphate (Li-FePO4).

LIB Lithium-ion battery.

LSE Least squares estimation.

LSTM Long-short-term-memory; form of multi-layer RNN.

MAE Mean absolute error.

MISO Multi-input single-output.

ML Machine learning.

MSE Mean square error.

NN Neural network.

OCP Open circuit potential.

OCV Open circuit voltage.

ODE Ordinary differential equation.

PCC Pearson correlation coefficient.

PDAE Partial differential algebraic equation.

PDE Partial differential equation.

PID Proportional integral derivative controller.

PINN Physics-informed neural network.

RMSE Root mean square error.

RNN Recurrent neural network.

SISO Single-input single-output.

SOC State of charge; a measure of the difference between a fully charged battery versus a battery in use.

SPM Single particle model.

SPMeT Extended single particle model with coupled thermal model.

Chapter 1

Introduction

1.1 Background

Research into lithium-ion batteries (LIBs) is an ever-expanding field, with new advancements being made constantly. LIBs provide an alternative to traditional energy sources such as natural gas and coal, which produce high amounts of emissions and are non-renewable resources [1]. Moreover, LIBs may be used in electric vehicles (EVs), as a substitute for conventional gas and diesel vehicles which greatly contribute to climate change [2]. This is due to the high energy density of LIBs, which also makes them attractive for energy storage solutions (ESSs) [3–5]. In comparison to other familiar rechargable battery types, LIBs excel in high energy and power density, as well as environmental friendliness [2]. LIBs are also the dominant battery type in portable electronics [3–5]; where the monitoring and control of batteries in such portable electronics and EVs is handled by battery management systems (BMSs) [3–5].

Thus, it is vital to have accurate models that portray the internal characteristics of the batteries during charging and discharging, as well as precise but quick state estimation to be used in BMSs and battery design.

Commercial efforts to improve batteries involve testing the discharge/charge cycles of the batteries numerous times. These procedures produce parameters that are used in the BMSs, but are typically very time-consuming. An alternative approach is to

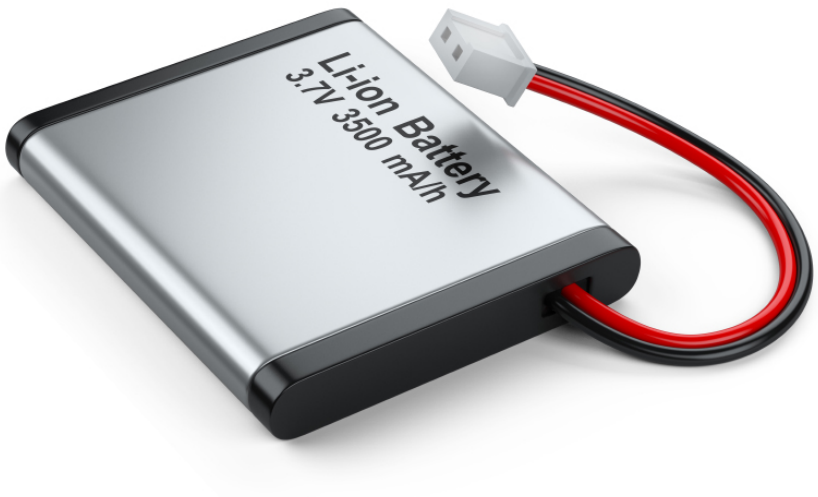


Figure 1.1: Example of a lithium-ion battery used in portable electronics [6]

model and simulate the batteries. Modern physics-based modeling efforts include 3 main model types: Doyle-Fuller-Newman/P2D (DFN), single-particle model (SPM), and equivalent circuit models (ECMs). The principal difficulty in battery modeling is the selection of the most appropriate model, as it is a balance between computational cost and model accuracy. Given the high complexity of DFN models, and various simplifying assumptions of ECMs, the SPM model was selected, with the extension to electrolyte dynamics included, for this work. Table 1 consists of a general overview of some current battery models being used in the field [7]. Moreover, the description, complexity, and classification of the models is presented [7]. Given the aim of the modelling portion of this work was for relatively comprehensive representation of the internal mechanics of the battery, the majority of the models shown in Table 1 are electrochemical models.

Also of growing interest to the academic community is the topic of machine learning, and deep-learning networks in particular. Renewed research into machine learning provides an avenue for exciting extensions into battery research. More specifically, state estimation is a widely studied topic that benefits from this alternate approach.

The usage of machine learning networks to estimate states such as state of charge (SOC) and internal average temperature is paramount in applications to BMSs and battery design, due to the flexibility and speed of some of the newly developed algorithms.

1.2 Research Motivation

The renewed interest in machine learning combined with the surge in lithium-ion battery research presents new opportunities in the alternative energy field. Lithium-ion batteries are already the predominant type in most modern portable electronics, and are an attractive avenue in powering electric vehicles. Application of battery models are vital to the operation of BMSs and in battery design research, thus there is a need for accurate models which also still represent the internal electrochemistry and mechanics of the cells. This is a non-trivial prospect by itself, as physics-based models which are more physically representative must be illustrated by PDEs and numerous parameters. Moreover, much battery modelling research is not openly available to the public in comprehensive detail due to proprietary reasons. To increase the generalizability of the developed model and gain insight into less studied trends, it was of great interest to complete state estimation via machine learning. Consequently, a specific algorithm that was best suited (based on literature) and the system identification of the problem were undertaken. Next, the research goals of this thesis will be briefly summarized.

1.3 Research Objectives

The primary objectives of this thesis are to:

- Investigate various modelling efforts for LIBs.
- Select an appropriate model type and develop an enhanced version.

Table 1.1: General Overview of LIB Models

Model Name	**Model Description**	**Complexity**	**Classification**
Doyle-Fuller Neumann (DFN/P2D)	Describes the complex behaviours via partial differential equations; microstructure problem is posed in 1D with spherical geometry.	High	White-box
Single-Particle model (SPM)	Describes complex battery behaviours via partial differential equations; refers to class of models which uses a single spherical particle to represent each electrode.	Mid	White-box
Extended Single Particle Model (SPMe)	An extension of the SPM model which includes a sub-system of 3 PDEs to represent the electrolyte dynamics.	Mid-High	White-box
Equivalent Circuit Model (ECM)	A representation of the battery via a circuit; consisting of a voltage source and internal resistance. Does not fully represent the internal mechanisms of the battery.	Low	Grey-box
Mathematical Model(s)	Uses empirical equations or stochastic models which purely evaluate the charge recovery effect while disregarding all other factors.	Mid-High	Grey-box
Artificial Neural Network (ANN)	Form of deep-learning network; seeks to mimic the brain via layers of connected "nodes". Purely data-driven type model, with no insight into the physical characteristics.	Varied	Black-box

- Implement developed model from scratch without blackbox modelling tools or commercial packages.
- Integrate a PID controller to enact the traditional CC-CV charging protocol.
- Validate the implemented model via experimental data from literature.
- Construct an LSTM network to complete state estimation of SOC and average internal cell temperature.
- Investigate the resulting trends of state estimation via LSTM for 4 different C-rates.
- Compare results of single-feature and multi-feature network.
- Examine the effect of noise on the state estimation problem.

The organization of this research is presented next.

1.4 Thesis Scope

This thesis is constructed in a two-paper format, and consists of four chapters which are summarized as follows. Chapter 1 introduces some general information regarding LIBs, reasons why they are of interest to researchers, and the objectives of this study. Chapter 2 is a completed paper containing the model development and validation of the proposed electrochemical-thermal coupled model. Meanwhile, Chapter 3 expands on the previous work via state estimation using data from the model developed in Chapter 2 by employing an LSTM network. The network architecture is presented, in addition to results of the network for SOC and average internal cell temperature estimation over varying current loads. Chapter 4 presents the main inferences from the study by summarizing Chapters 2 and 3, and delivers potential extensions to this research.

Chapter 2

Enhanced Dynamic Modeling and Analysis of a Lithium-Ion Battery: Coupling Extended Single Particle Model with Thermal Dynamics

2.1 Introduction

The prevailing battery type in portable electronics and electric vehicles are Lithium-ion batteries; they are also a promising candidate for energy storage systems [3–5, 8, 9]. Commercially, batteries are improved based on the testing of the charge-discharge cycles. This testing generates parameters which are used in battery management systems that monitor and control battery behaviours, but these procedures are time-consuming as they require multiple cycles to generate accurate and standardized parameters [3, 10]. [3]

Therefore, it is vital to develop an accurate thermal model utilizing partial differential equations (PDEs) to capture the spatio-temporal dynamics within the cell [4, 5]. In the current manuscript, a thermal model coupled with an electrochemical-based concentration model will be developed to determine ion concentrations in the Lithium-ion cell, and the model will be simulated from start to finish without the use of black box tools (as is often seen in literature). This work intends to develop an accurate thermal model and couple the concentration and thermal models to deter-

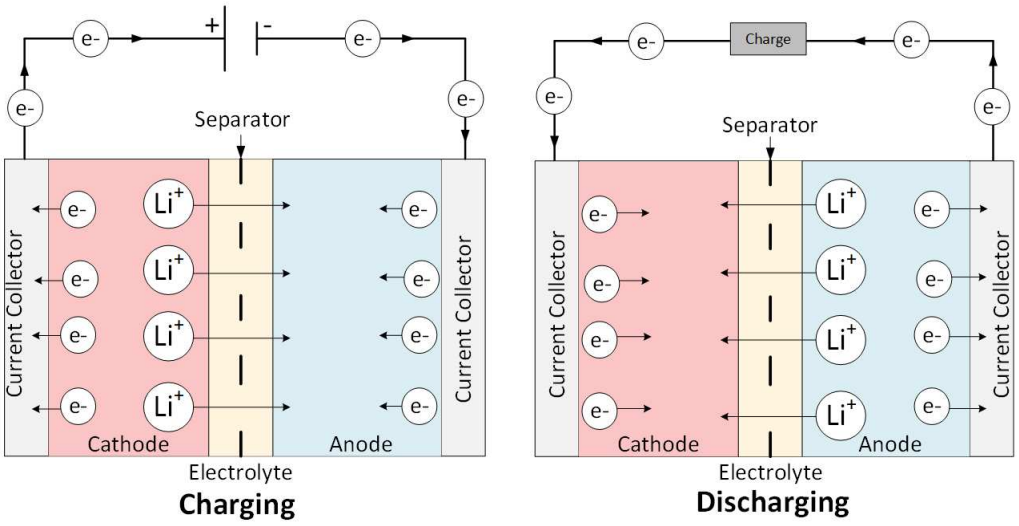


Figure 2.1: Schematic of Lithium-Ion Battery Mechanisms [11]

mine the not directly measurable concentration properties in the cell, and examine the accuracy of the coupled thermal model [5].

There are three regions in typical lithium-ion batteries; two porous electrodes made up of very small particles of varying active materials (held together by a polymer binder), and an electron-blocking separator [3]. A lithium electrolyte permeates those regions and acts to carry charge and Li-ions between the active materials [3]. On the interface between each electrode and the separator, de-intercalation takes place [3, 4]. Figure 2.1 illustrates a simple representation of a lithium-ion battery, and the described process. A common term to describe the charging and discharging of batteries is C-rate; C-rate is defined as the measurement of charge/discharge current concerning the nominal capacity [3]. The electrochemical reactions may vary depending on the type of lithium-ion battery. For example, considering a layered oxide type such as a $LiCoO_2$-graphite battery, these equations are:

$$LiCoO_2 \xrightarrow[Cathode]{\text{Charge}} yLi^+ + ye^- + Li_{1-y}CoO_2 \tag{2.1}$$

$$xLi^+ + xe^- + Li_{1-x}C_6 \xrightarrow[Anode]{\text{Charge}} LiC_6 \tag{2.2}$$

The primary difficulty in considering model type selection is the balance between computational efficiency and accuracy [3, 12–14]. Up-to-date physics-based modelling efforts utilize three major model types: the Doyle-Fuller-Neumann Model (DFN), the Single Particle Model (SPM), and Equivalent Circuit Models (ECM) [3]. While DFN models are the most accurate in portraying the electrochemical dynamics in the cell, they are traditionally the most computationally intensive due to the characterization via PDEs coupled with nonlinear algebraic equations [3, 12, 13]. Furthermore, the increased complexity yields increased difficulty in the parametrization of the model, which is an already challenging task [3, 12, 13, 15]. However, there are some examples in literature which show the implementation of reformulated P2D models that can complete simulations in under 100 ms [16]. While this is extremely promising, these reformulations hold for single-cell representations and in the case of multi-cell stacks, may have decreased computational efficiency [15]. The simplest models are ECMs, which seek to capture the battery dynamics using electrical components and circuitry [3]. While the fast computational time of these models is attractive for some applications, the downfall is the lack of representation of specific electrochemical dynamics in the cell; as such, ECMs fail to fully represent the internal physical characteristics that occur during charging/discharging cycles [3]. Due to the assumption that the electrodes may each be represented by a single spherical particle and by virtually eliminating the PDEs which deal with liquid phase diffusion, SPMs are much simpler than the DFN models, while also more accurate than ECM models and have a higher solving efficiency [3, 5, 12]. However, the cost of the simplification is in reduced accuracy; given the assumptions of the model, the electrolyte phase dynamics are not considered [3]. An extension of the SPM model class is called extended SPM (SPMe) which does take electrolyte dynamics into account. To do this the battery is portioned into three sections: the cathode, separator, and anode (more details for this domain definition are available in Appendix A.3), and then a PDE to represent the electrolyte diffusion must be solved for each domain. A graphical depiction of this model

type is shown in Figure 2.2. To accurately apply this model the (de-)intercalation of lithium-ions is assumed to be at the same rate regardless of position in the cell [3]. This assumption is valid for most types of lithium-ion batteries, with one prominent exception being $LiFePO_4$ batteries. Another class of models that is less known, and is between DFN and SPM modelling is the tanks-in-series approach [15]. This manner of battery modelling is an electrochemical-thermal model for lithium-ion batteries; it involves volume averaging in given domains of the typical PDEs (conservation laws) present in P2D models, which allows for temperature gradients to be captured in multi-cell stacks [15]. Moreover, the approach provides flexibility in the adjustment of the gradients within each domain, via tuning of the variable diffusion lengths [15].

The motives of this work are to balance computational efficiency with accuracy by using an extended SPM (SPMe) coupled with a thermal model (SPMeT). The SPMe model uses the simplifying assumptions of the SPM, but also considers the liquid phase diffusion, so that the shortcomings of the SPM are addressed with only a slight decrease in computational efficiency [3, 5]. More specifically, this work proposes an extended SPM model coupled with a thermal model consisting of two ordinary differential equations (ODEs). The novelty of the model developed in this work is that a more accurate yet computationally tractable model is developed for enhanced modelling of spatio-temporal dynamics in lithium-ion batteries, with extension to thermal behaviours. Moreover, this work utilizes a proportional-integral-derivative controller to manipulate the charging-discharging algorithm, which provides flexibility in tuning. As claimed in [3–5], currently, existing works on the coupling of SPMe and thermal models via lithium concentration are significantly under-investigated. Additionally, this work will consider a non-spatially varying exchange current density, as posed in Planella et al. [3]. The potential use of the developed model of this work in future control of monitoring efforts would allow for the consideration of thermal profiles without the time-consuming and computationally challenging efforts of DFN or other Psuedo-2D model types, while also having a more meaningful representation

of the internal mechanisms than that of the ECM type models. Moreover, the SPMeT is a promising candidate for BMS systems due to its computational efficiency and accuracy [4]. The contributions of this work are as follows:

- An enhanced LIB model through an extended single-particle model coupled with thermal dynamics is proposed.
- Compared with existing models, the proposed model is more efficient yet accurate.
- The proposed model is validated with experimental data, and shows good agreement.

2.1.1 Literature Review

Electrochemical modelling is at the forefront of lithium-ion battery research [3, 12, 13, 16]. Moreover, the compilation of battery models by Brosa et al. reviews the various battery model types, including details of construction and features [3]. Additionally, Brosa shows two comparative approaches for extended SPM modelling, Marquis et al. versus Richardson et al. [3, 12, 17]. Initially, DFN models were proposed in 1994 by Fuller et al. [18]. A significant contribution that must be noted is the work completed by Kolluri et al.; the article presents the implementation of a non-linear model predictive control algorithm onto a DFN model type [16]. The study demonstrates that through the application of model predictive control theory, P2D models may be efficiently implemented and incorporated into BMS systems; although in this instance the model is considered to be isothermal [16]. Another study of great interest that considers P2D modelling in depth is "A robust numerical treatment of solid-phase diffusion in pseudo-two-dimensional lithium-ion battery models" [19]. The work clarifies the origins of potential numerical instabilities in P2D models, as well as modifications of the Jacobian components that relate to the equations representing

the active particle surface; the result is a treatment of the P2D model class which is computationally efficient, albeit for an internally isothermal case [19].

Meanwhile, reductions of the physics-based modelling were first introduced in 1979 by Atlung et al. [20]. Since this advancement, numerous extensions of SPM models have been researched which show better agreement with DFN modelling results and only a small increase in computational load [3, 4, 12]. The work by Prada et al. considers an SPMe model, with special considerations for Li-FePO4 (LFP) batteries, which have unique characteristics; such as partial solid solution regions, and changing capacity restitution depending on discharging versus charging [4]. Additionally, a thermal model was proposed, based on the endeavour of Forgez et al. which limits computational effort via simplicity [4, 21]. However, the application of an SPM-type model to LFP batteries does not typically hold [3]. The class of SPM models relies on the assumption that during discharge all negative electrode particles delithiate at approximately the same rate and that all positive electrode particles lithiate at approximately the same rate; both independently of position in the electrode(s) [3]. This assumption proves reasonable for most materials, with LFP being an exception [3, 22]. Perez et al. also considered an SPMeT model, but instead posed the thermal model as two ODEs (one for the core temperature and one for surface temperature), as theorized by Lin et al. [5, 23]. Also, the authors present the charging protocol as an optimization problem to determine the optimal charging protocols resulting in minimum charging times, while also comparing the traditional and the determined optimal protocols [5]. There has also been significant research into porous electrode theory (PET), with numerous open-source software packages available online [24]. Packages such as DUALFOIL, and COMSOL do allow for robust modelling in P2D format, but have slow run-times which make them unsuitable for MPC, state/parameter estimation, or real-time control purposes [24]. PETLION is a more modern open-source package which has a much faster run-time and can perform complex simulations with allowances for different chemistries and charging schemes [24]. Although

this contribution and others do use spatial discretization methods to solve the systems of PDEs, many do not show the resulting spatial-temporal concentration profiles [24]. Moreover, this work intends to maintain the connection from the physics-based model through the simulations to provide insight into the internal mechanics during the charging/discharging of the cell.

In the current manuscript, the SPMe model is adopted, as proposed by Prada, Brosa, and the approach by Richardson; where the thermal model from Perez is implemented and compared. The majority of the parameters for the model were taken from the work of Prada, as it posed the most complete set, and contained validation via experimental values. Alternatively to most literature [3, 4], the exchange current density utilized in the current manuscript is both spatially and time-varying. Figure 2.3 is a flowchart depicting the model which will be described in detail in the subsequent sections. As may be seen in the figure, the input variable $i_{app}(t)$ feeds into the electrode and electrolyte dynamics, which is then used to calculate the electrode potentials U_k and voltage $V(t)$, as well as the bulk concentration in the electrodes. This represents the SPMe model. Progressing through the chart, the current $i_{app}(t)$, voltage $V(t)$, and the bulk concentrations in the electrodes c_k^{bulk} are used to determine the heat generation, which in turn is used for the calculation of the core and surface temperatures. From the core T_c and surface T_s temperatures, the average temperature is computed, updated, and fed back into the model to calculate the voltage.

While the works of Prada [4] and Perez [5] implement a form of an SPMe model coupled with a dynamic thermal model, there are some critical differences compared to this work. To elucidate, both studies make use of a simplified exchange current density, where in Prada this variable is considered as a constant [4, 5]. Additionally, Perez et al. utilize the surface concentration for the calculation of SOC, whereas the bulk concentration is used in this study [5]. Meanwhile, equation (38) in Prada et al. represents the ohmic resistance, but uses an effective conductivity and neglects the conductivity of the solid phases in the electrodes as opposed to specifying these

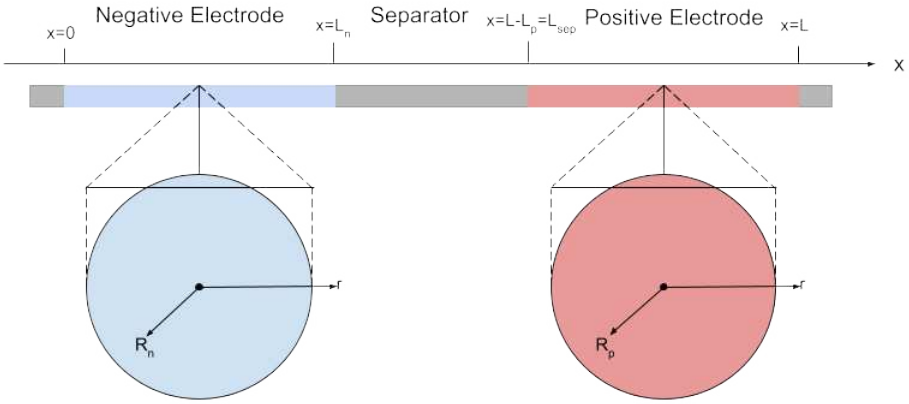


Figure 2.2: Geometry of the SPMe Model Class [3].

parameters as in this work [4]. However, research on coupled electrochemical and thermal dynamic models in general is an under-researched topic.

The remainder of this chapter will consist of four more sections. First will be a review of the methods taken for this work; the model development and implementation in particular will be revealed. Second, the results of the model will be presented, including comparisons to the literature and experimental results. Thirdly, is the discussion of these results. The final section will detail the conclusions of the work. Appendices A.1 to A.3 give the abbreviations, nomenclature, and electrolyte domain definition, accordingly.

2.2 Model Development

2.2.1 Concentration Dynamics

$$\frac{\partial c_k}{\partial t} = \frac{1}{r^2}\frac{\partial}{\partial r}(r^2 D_k(c_k)\frac{\partial c_k}{\partial r}) \tag{2.3}$$

where $0 < r < R_k k$ and $r \in \{n, p\}$.

Equation (2.3) represents mass conservation of the Lithium-ion species in the electrodes respectively and is derived from Fick's 2^{nd} law of diffusion [3]. However, it is important to note that while the most accurate representation includes diffusivity in the solid materials as a function of concentration (the so-called phase-change diffu-

sivity), the approximation of diffusivity as a constant is found to be adequate and is the most common determination in literature [3, 25].

The first boundary condition representing symmetry for the spherical particles which represent each electrode is given by equation (2.4). This condition is required for the problem to be well-posed [3]. The second boundary condition shown in (2.5) signifies that the flux of Lithium-ions at the surface of the particle depicting the electrode(s) ($r = R_k$) is proportional to the input current [3]. Equation (2.6) defines the initial condition settings.

$$\frac{\partial c_k}{\partial r} = 0, \quad \text{at} \quad r = 0 \tag{2.4}$$

$$-D_k(c_k)\frac{\partial c_k}{\partial r} = \frac{j_k}{b_k F}, \quad \text{at} \quad r = R_k \tag{2.5}$$

$$c_k = c_{k0}, \quad \text{at} \quad t = 0 \tag{2.6}$$

Current density in the electrodes is given as an average of the applied current:

$$j_n = \frac{i_{app}(t)}{L_n} \tag{2.7}$$

$$j_p = -\frac{i_{app}(t)}{L_p} \tag{2.8}$$

Similarly to equation (2.3), the governing electrolyte equation (2.9) represents the flux of lithium-ions through the electrolyte, and is derived from Fick's 2nd law [3]. The boundary conditions for the electrolyte governing equation pose that the gradient of concentration of lithium ions is zero at each end of the cell:

$$\varepsilon(x)\frac{\partial c_e}{\partial t} = \frac{\partial}{\partial x}(D_e(c_e)\beta(x)\frac{\partial c_e}{\partial x} - \frac{t^+}{F}i_e) + \frac{b(x)j_k(x,t)}{F} \tag{2.9}$$

$$\frac{\partial c_e}{\partial x} = 0, \quad \text{at} \quad x = 0 \quad \text{or} \quad x = L \tag{2.10}$$

$$c_e = c_{e0}, \quad \text{at} \quad t = 0 \tag{2.11}$$

where $x \in \{0, L\}$. The phenomena of electrophoresis, the movement of charged particles in a fluid under the influence of an electric field, is represented by the last

term in (2.9)[3]. Expressions for current density in the electrolyte in the anode, separator, and cathode domains are given by (2.12), (2.13), and (2.14), respectively; where $i_{app}(t)$ is the applied current to the cell and serves as the input to the system.

$$i_e(x,t) = \frac{i_{app}(t)}{Ln}x, \quad x \in \{n\} \tag{2.12}$$

$$i_e(x,t) = i_{app}(t), \quad x \in \{s\} \tag{2.13}$$

$$i_e(x,t) = \frac{i_{app}(t)}{Lp}(L-x), \quad x \in \{p\} \tag{2.14}$$

2.2.2 Voltage Dynamics

The contributions to terminal voltage as exhibited in (2.15) are adopted from the model presented by Richardson et al. [12]; where U_{eq} is the open circuit potential of the cell, η_r and η_c give the potential drops due to reaction and concentration overpotentials, and ϕ_e and ϕ_k are the ohmic losses in the electrolyte and electrodes. The resulting construction is a nonlinear output function, which maps the boundary values of the solid and electrolyte concentration, as well as the target current to the terminal voltage.

$$V(t) = U_{eq} - \eta_r - \eta_c - \Delta\phi_e - \Delta\phi_k \tag{2.15}$$

$$U_{eq} = U_p(c_p|_{r=R_p}) - U_n(c_n|_{r=R_n}) \tag{2.16}$$

$$\begin{aligned}\Delta\phi_e = (&\frac{1}{L_p}\int_{L-L_p}^{L}\int_0^x \frac{i_e(s,t)ds}{\sigma_e(c_e(s,t))\beta(s)}dx - \\ &\frac{1}{L_n}\int_0^{L_n}\int_0^x \frac{i_e(s,t)ds}{\sigma_e(c_e(s,t))\beta(s)}dx)\end{aligned} \tag{2.17}$$

$$\Delta\phi_k = \frac{i_{app}(t)}{3}(\frac{L_p}{\sigma_p} + \frac{L_n}{\sigma_n}) \tag{2.18}$$

$$\begin{aligned}\eta_r = \frac{2RT_{avg}}{F}(&\frac{1}{L_n}\int_0^{L_n}\text{arcsinh}(\frac{j_n(t)}{j_{n0}(x,t)})dx - \\ &\frac{1}{L_p}\int_0^{L_p}\text{arcsinh}(\frac{j_p(t)}{j_{p0}(x,t)})dx)\end{aligned} \tag{2.19}$$

$$\begin{aligned}\eta_c = (1-t^+)\frac{2RT_{avg}}{F}(&\frac{1}{L_n}\int_0^{L_n} log(c_e(x,t))dx - \\ &\frac{1}{L_p}\int_{L-L_p}^{L} log(c_e(x,t)))dx)\end{aligned} \tag{2.20}$$

While charging, the kinetic and mass transport overpotentials are all negative; thus delineating the excess potential required to overcome the internal resistances in the cell [4]. Conversely, during discharging these quantities are all positive, and demonstrate the decrease in cell potential resulting from loss mechanisms. The exchange current densities are given as:

$$j_{k0}(x,t) = FK_k\sqrt{\frac{c_e(x,t)}{c_{e0}}\frac{c_k(r,t)}{c_k^{max}}(1-\frac{c_k(r,t)}{c_k^{max}})|_{r=R_k}} \tag{2.21}$$

for $k \in \{n,p\}$. The model presented by Richardson et al. is attained via asymptotic methods based on the disparity between the magnitude of thermal potential and distinguishing change in overpotential which occurs as (de)lithiation occurs in the electrodes [3, 12]. This approach enables significant variation in the initial electrolyte concentration. Moreover, the accuracy of the model posed by Richardson et al. demonstrates similar performances with some deviations only appearing at high C-rates, where the approach of Richardson et al. establishes better accuracy than that of Marquis et al. [3, 12].

2.2.3 Thermal Dynamics

The thermal model as posed by Perez [5] considers a subsystem of two ODEs; this approach yields more accurate predictions than those obtained using the more common single lumped temperature; due to the higher core temperature in cells than on the surface under high current rates [3].

$$\frac{dT_c(t)}{dt} = \frac{T_s(t)-T_c(t)}{R_cC_c} + \frac{Q(t)}{C_c} \tag{2.22}$$

$$\frac{dT_s(t)}{dt} = \frac{T_{amb}(t)-T_s(t)}{R_uC_s} - \frac{T_s(t)-T_c(t)}{R_cC_s} \tag{2.23}$$

$$T_{avg}(t) = \frac{T_c(t)+T_s(t)}{2} \tag{2.24}$$

Where T_c and T_s are the core and surface temperature, while $Q(t)$ gives the heat generation [5]. The other parameters in this system are R_c, R_u, C_c, and C_s, which

are the heat conduction resistance, convection resistance, core heat capacity, and surface heat capacity respectively. The coolant flow rate is assumed constant (which results in a constant convection resistance, R_u), and the ambient temperature is also assumed constant. Once T_c and T_s are calculated, the average cell temperature may be found by (2.24) and is fed back into the model to re-calculate voltage. Finally, the SOC is calculated using Equation 2.25.

$$SOC(t) = 100 * \frac{\theta_n^{bulk}(t) - \theta_{n,0}^{bulk}}{\theta_{n,100}^{bulk} - \theta_{n,0}^{bulk}} \tag{2.25}$$

Where θ_n^{bulk} is the normalized concentration of the bulk material for the anode (indicated by the n subscript) as a function of time, and$\theta_{n,0}^{bulk}$ and $\theta_{n,100}^{bulk}$ represent the initial normalized bulk concentration for the anode and the final normalized bulk concentration for negative electrode respectively.

2.2.4 Model Overview

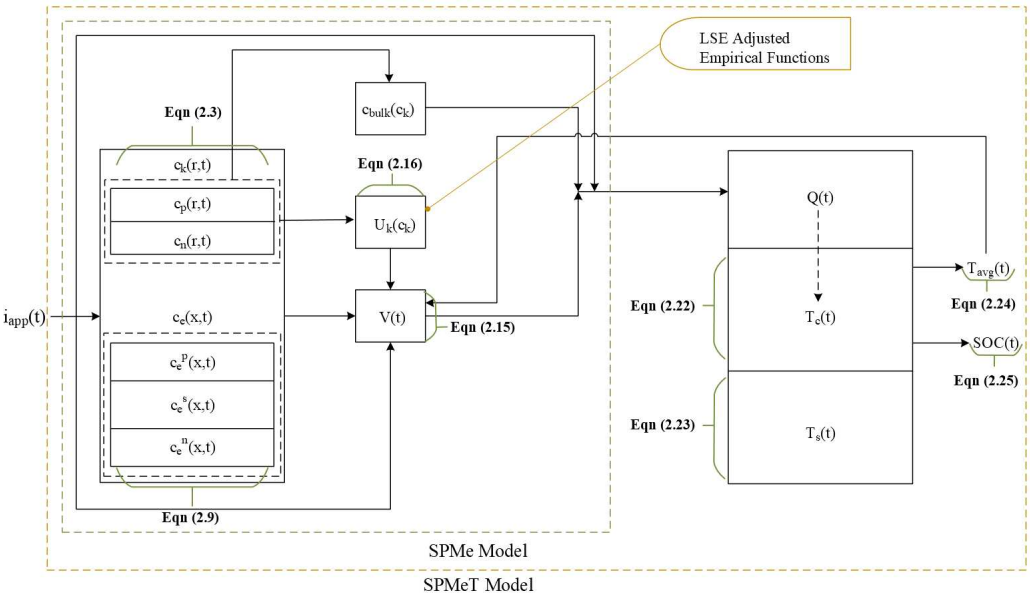


Figure 2.3: Model Flowchart with Perez et. al. [5] Thermal Approach

Table 2.1: Key Model Equations as Shown in Figure 2.3

Meaning	**Equation**	**Eq. #**
Lithium-ion flux - electrode(s)	$\frac{\partial c_k}{\partial t} = \frac{1}{r^2}\frac{\partial}{\partial r}(r^2 D_k(c_k)\frac{\partial c_k}{\partial r})$	2.3
Lithium-ion flux - electrolyte	$\varepsilon(x)\frac{\partial c_e}{\partial t} = \frac{\partial}{\partial x}(D_e(c_e)\beta(x)\frac{\partial c_e}{\partial x} - \frac{t^+}{F}i_e) + \frac{b(x)j_k(x,t)}{F}$	2.9
Terminal voltage	$V(t) = U_{eq} - \eta_r - \eta_c - \Delta\phi_e - \Delta\phi_k$	2.15
Open circuit potential	$U_{eq} = U_p(c_p\|_{r=R_p}) - U_n(c_n\|_{r=R_n})$	2.16
Surface temperature	$\frac{dT_c(t)}{dt} = \frac{T_s(t)-T_c(t)}{R_c C_c} + \frac{Q(t)}{C_c}$	2.23
Core temperature	$\frac{dT_s(t)}{dt} = \frac{T_{amb}(t)-T_s(t)}{R_u C_s} - \frac{T_s(t)-T_c(t)}{R_c C_s}$	2.22
Average cell temperature	$T_{avg}(t) = \frac{T_c(t)+T_s(t)}{2}$	2.24
State of charge	$SOC(t) = 100 * \frac{\theta_n^{bulk}(t)-\theta_{n,0}^{bulk}}{\theta_{n,100}^{bulk}-\theta_{n,0}^{bulk}}$	2.25

2.3 Implementation

To begin implementation of the model, discretization of the PDEs, and the coupled ODE subsystem is required. Second-order central finite difference discretizes the internal points of the PDEs; central finite difference is used because it yields higher accuracy results as it is classified as a second-order method (the average of the forward and backward finite differences are used to compute the derivative) [5, 26]. Meanwhile, the treatment of the boundary nodes is second-order backward and forward finite difference for the last and first points respectively. Central discretization is possible at the boundary $r = 0$ (due to the boundary condition for symmetry), but not at $r = R_k$. This is because the imaginary node required for central finite difference at the boundary cannot be assumed to be equal to the N-1 node as a result of the flux boundary condition (2.5) [27]. To construct the grid for discretization, the points were set with a spatial discretization length of $dx = 0.029$ for the anode and $dx = 0.0125$, and mapped according to the actual geometry of the battery. To elaborate, the lengths of each of the three battery sections were mapped to a proportional set of indices for the purpose of simulation.

In the current manuscript, a form of proportional-integral-derivative (PID) controller was employed to regulate current according to the CC-CV protocol. The

proportional and integral gains (k_p and k_i) determine the ratio of the output signal to the error and the contribution of the summed error over time, respectively [28]. For the charging case, these control settings were determined to be inverse gain and of a larger magnitude than for the discharging case, which required non-inverse gain for acceptable control [28].

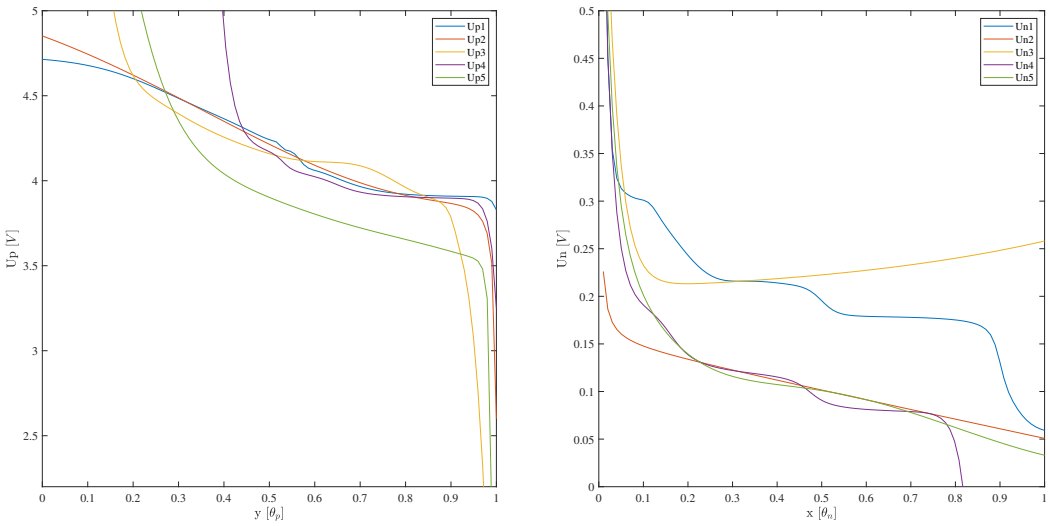


Figure 2.4: Assorted Empirical Electrode Thermodynamic Potential Functions Comparison

One of the most challenging aspects of battery modelling of this complexity is the calculation of the electrode thermodynamic potentials, which are the primary drivers behind the overall cell voltage. The most common method is the use of empirical functions which correlate the normalized concentration to the respective electrode potential. in the current manuscript, the first step was to compare such functions from various sources in literature: [29–32], and [33] denoted as U_{ki} (where $k \in p, n$ and $i \in 1, 2, 3, 4, 5$) in the current manuscript. As may be seen in Figure 2.4, the results of such functions can be vastly different; this is expected as differing parameters, composition of electrolyte, and battery type may all affect these results.

The second procedure was to choose the most suitable function from Figure 2.4 based on the desired characteristics and use Least Squares Estimation (LSE) [34] to

fit the parameters of the chosen function to match the magnitude of the primary source of literature for this work [4]. Additional terms were added to ensure the characteristics of the original functions were maintained. The chosen functions were from the work of Di Domenico for the cathode and Srinivasan for the anode [32, 33]. The desired features for the U_p function were a rapid decrease for the extreme values of θ_p, with a flatter period for the middling values. Meanwhile, the anode function features were a rapid decrease for the initial values of θ_n, with step decreases down to zero following this. A constant negative sign for the slope was wanted in both instances to limit complications with the LSE fitting. The results of the fitting for both the cathode and the anode empirical functions are shown in Figures 2.5 and 2.6 respectively.

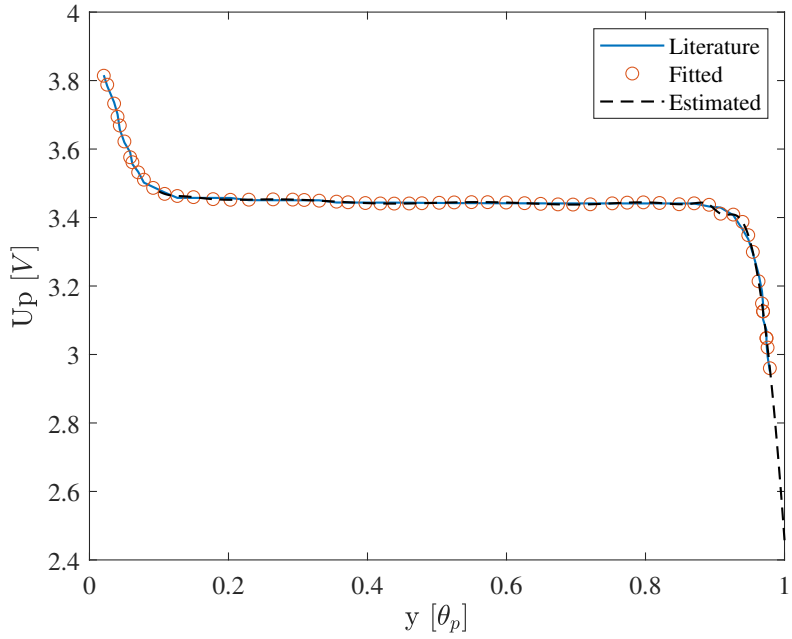


Figure 2.5: LSE Fitting Results for Cathode Empirical Potential Function

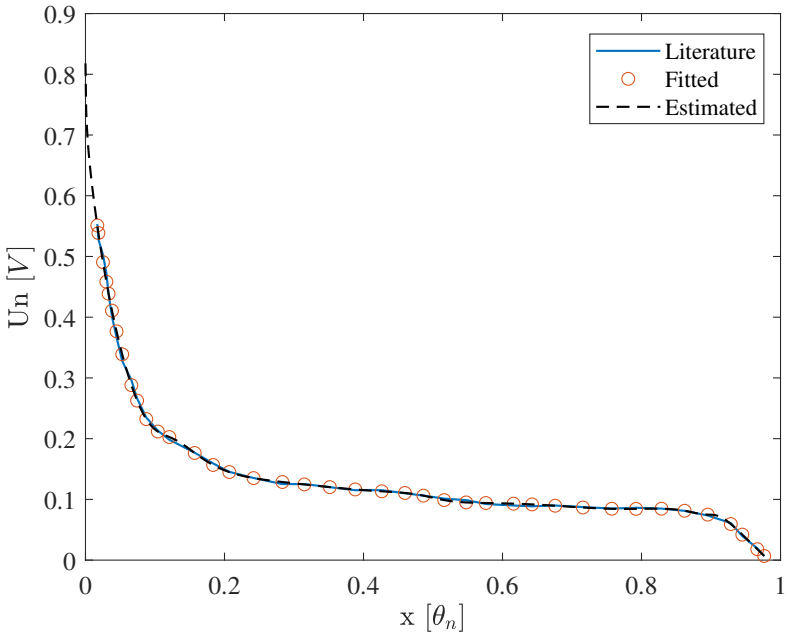


Figure 2.6: LSE Fitting Results for Anode Empirical Potential Function

The functions are in the form of summed hyperbolic tangents and power functions with the normalized concentration as the input for each respective electrode. This general formulation is found in most literature sources that do provide the empirical functions used. Un and Up are used in the calculation of voltage as in equations (2.15) and (2.16), as may be noted in Figure 2.3. To prevent overfitting, only the coefficients of the terms were estimated using LSE, and the internal arguments and values of the powers were left unchanged from the source functions. Moreover, only two terms were added to each function, so the original structure of the empirical functions was maintained. By examining the empirical functions found in various literature sources, it is clear that the form of the functions impacts computation complexity. For example, the functions from [30] are simple, and contain 5 or fewer terms. However, this is at the expense of accuracy, as it can be seen that the functions exhibit unusual behaviour in comparison with the majority of the other literature Un and Up results in Figure 2.4.

2.4 Model Validation and Discussion

Once simulations were completed (via MATLAB), model validation was performed by comparing the results to those in the literature. A combination of the results from Prada et al., and Perez et al. were used for model verification for this work.

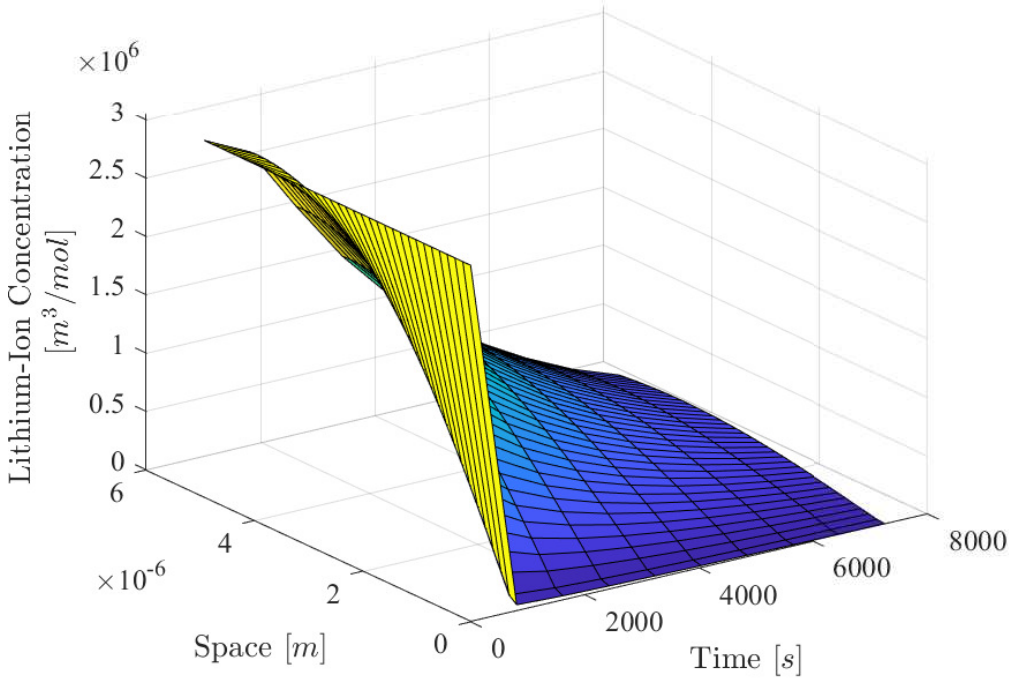


Figure 2.7: Anode Spatial-Temporal Concentration Profile

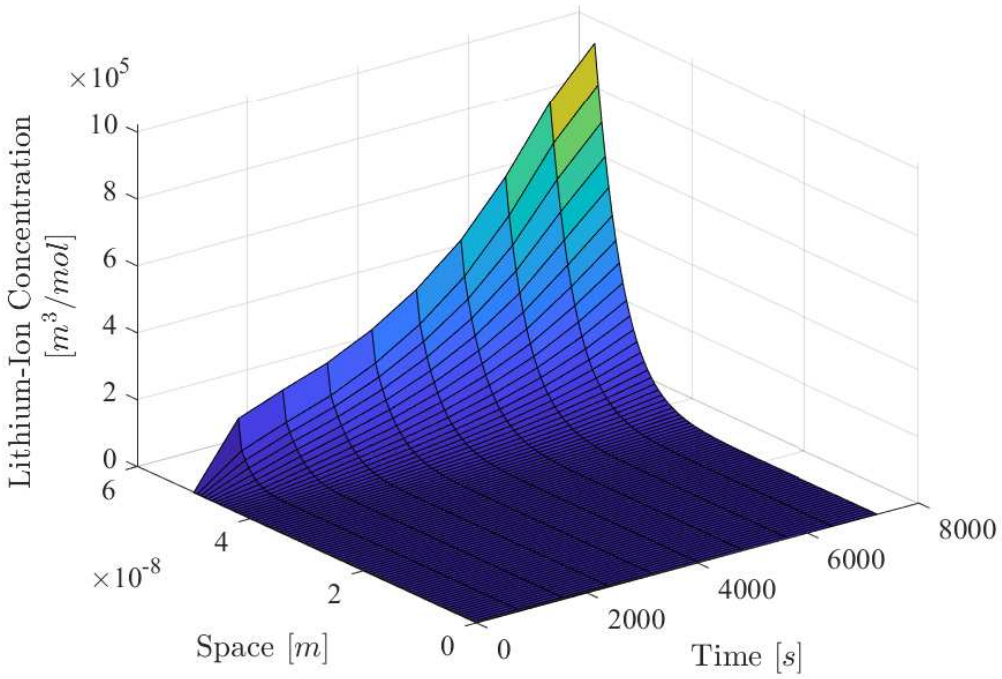


Figure 2.8: Cathode Spatial-Temporal Concentration Profile

Firstly, the spatial-temporal concentration graphs were reviewed to ensure the expected trends were occurring. For charging of the battery, it is anticipated that the anode begins at its max concentration throughout the particle; as time progresses the highest concentration for each time step should be at the centre of the particle as the ions diffuse into the electrolyte towards the cathode until eventually the concentration settles to approximately zero. Conversely, the cathode begins at a concentration of approximately zero, and as time increases the maximum concentration is along the surface of the particle for each time step and eventually reaches the design maximum concentration as charging is completed. For discharging of the battery, the opposite trends are observed.

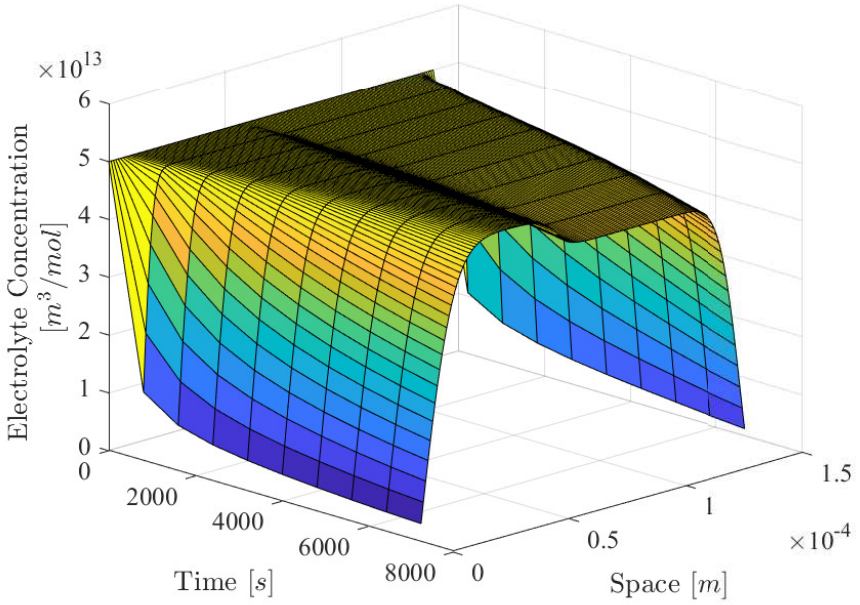


Figure 2.9: Electrolyte Spatial-Temporal Concentration Profile

The expected trends for the electrolyte are for the initial concentration to be maximum in the centre of the spatial domain, and to decrease as the domain reaches each electrode and as time progresses. Given the difference in magnitude of the diffusion coefficients, the concentration towards the anode side is expected to decrease more quickly than in the cathode domain ($D_n > D_p$). Figures 2.7, 2.8, and 2.9 show the anticipated trends.

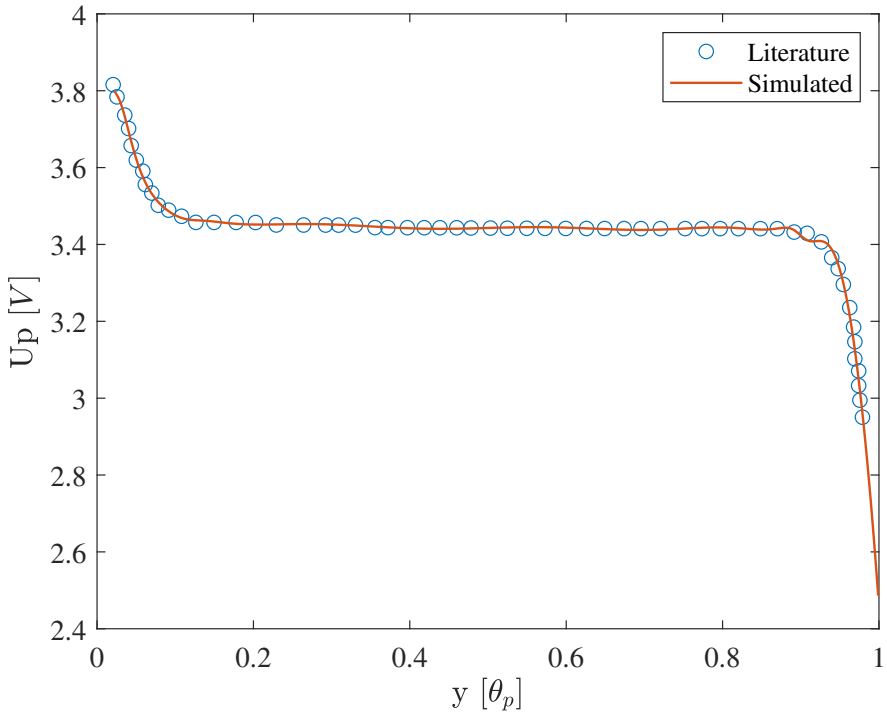


Figure 2.10: Cathode Normalized Concentration vs. Thermodynamic Potential

Secondly, the normalized concentration θ_k for each electrode is plotted as an input to the thermodynamic potential U_k functions. The normalized concentrations are found as:

$$\theta_k = \frac{c_k^s}{c_{k,max}^s} \tag{2.26}$$

by most literature; although [4] utilizes the bulk concentration in the electrodes to compute this ratio instead.

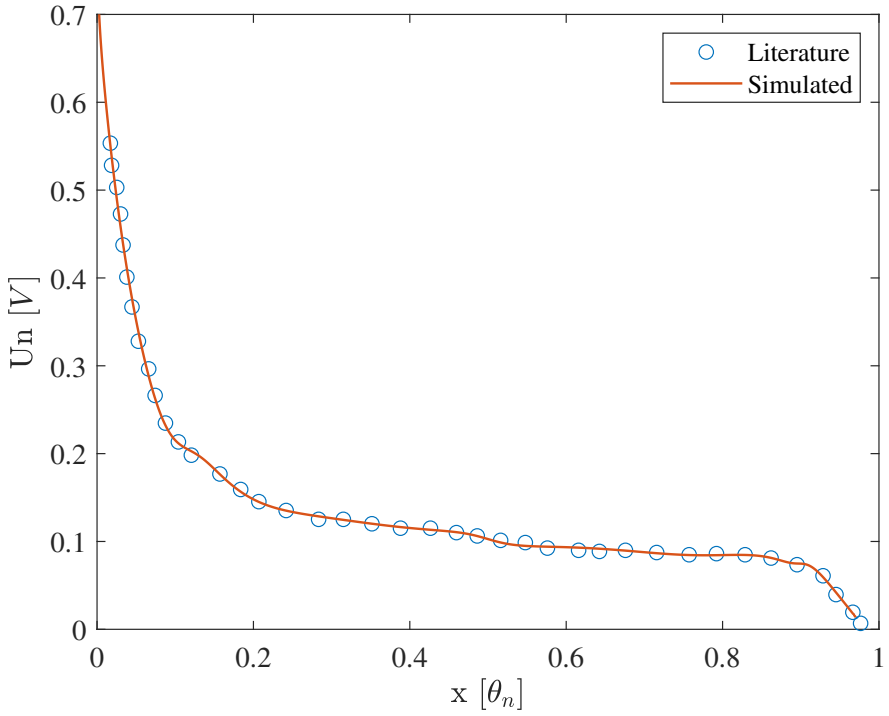


Figure 2.11: Anode Normalized Concentration vs. Thermodynamic Potential

Figures 2.10 and 2.11 show the comparison between literature and simulation for the normalized concentration versus electrode potential for the cathode and anode respectively. The simulation results are consistent with the corresponding outcomes in literature [4].

As aforementioned, functions for thermodynamic potential are experimentally determined; for this work, the functions determined in [33], and [32] were adopted, LSE fitting applied to adjust function coefficients, and then the resulting functions used in the final simulations. The presumed tendencies are that both curves are decreasing as θ_k increases, with the magnitude for the cathode being larger than the anode (approximately 3-4V vs. 0-1V) and the cathode, Figure 2.10, demonstrating more prominent behaviour of the CC-CV charging protocol. Note that the curves for charging and discharging are essentially the same, with simply the direction the plot proceeds along the x-axis changing. The results for both the cathode and anode potentials show excellent agreement with the primary literature source [4]. As evident

Table 2.2: Minimum and Maximum Error Between Simulation vs Literature for Capacity-Voltage Curve in Charging Case

C-Rate	**Min Error [%]**	**Max Error [%]**
0.5	0.146	14.538
1	0.0591	15.011
2	0.086	15.309
4	0.1614	16.523

in Figures 2.10 and 2.11, the fitting of the electrode functions is successful in shape and magnitude as compared with literature [4].

Thirdly, the trends for voltage versus capacity for multiple C-rates and both charging and discharging were examined. It is expected for charging, that the curves each begin at 0 for capacity, show rapid increase in voltage which represents the CC period of charging, flattening out as the process proceeds to the CV period, and finally rapid increase again for the rest period. It is noticed that the cutoff voltage V_{cutoff} triggering the CV period increases with the C-rate. The results in Figure 2.12 show good agreement with experimental values from the work of Prada et al. [4]. Also of note is that the disparity between each curve also increases with the C-rate. This is because at high currents the rates of (de-)intercalation are much higher; thus the resulting transfer of charge (which in turn generates the potential difference between the anode and cathode) is higher [3]. This observation is reflected by the results in Table 2.2; where the maximum error is for the 4C-rate simulation. Furthermore, the majority of the percent error values are under 5 % for all C-rates, with the largest values being at the beginning of the simulation when capacity is smallest.

Table 2.3: Minimum and Maximum Error Between Simulation vs Literature for Capacity-Voltage Curve in Discharging Case

C-Rate	**Min Error [%]**	**Max Error [%]**
0.5	0.173	28.499
1	0.128	29.287
2	0.040	25.888
4	0.308	24.024

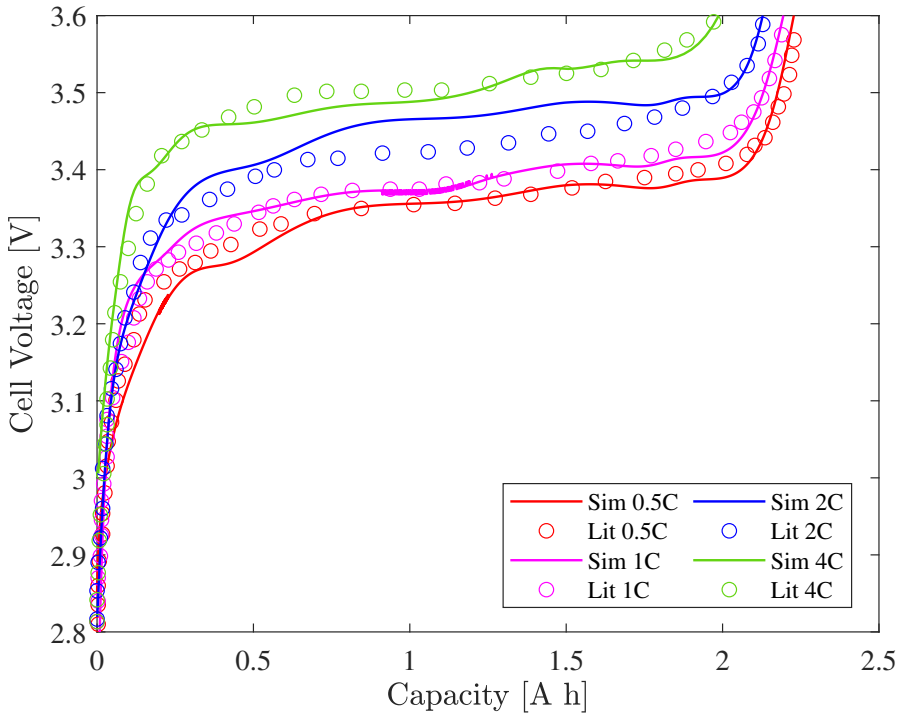


Figure 2.12: Capacity vs. Cell Potential for Charging

For discharging as seen in Figure 2.13, the curves begin with a rapid decrease representing the CC period, then give way to the CV period (flattened curve), and again the rapid decrease indicating the rest period. The V_{cutoff} for the CV period, in this case, decreases with increasing C-rate. The inverse trend from Table 2.2 may be observed in Table 2.3, whereas the C-rate decreases, the overall error increases. Moreover, the majority of the simulations show a percent error under 5%, with the greatest

error being at the end of the simulation as the nominal capacity is approached. Also, the charging results show better agreement with lower maximum errors for all C-rates.

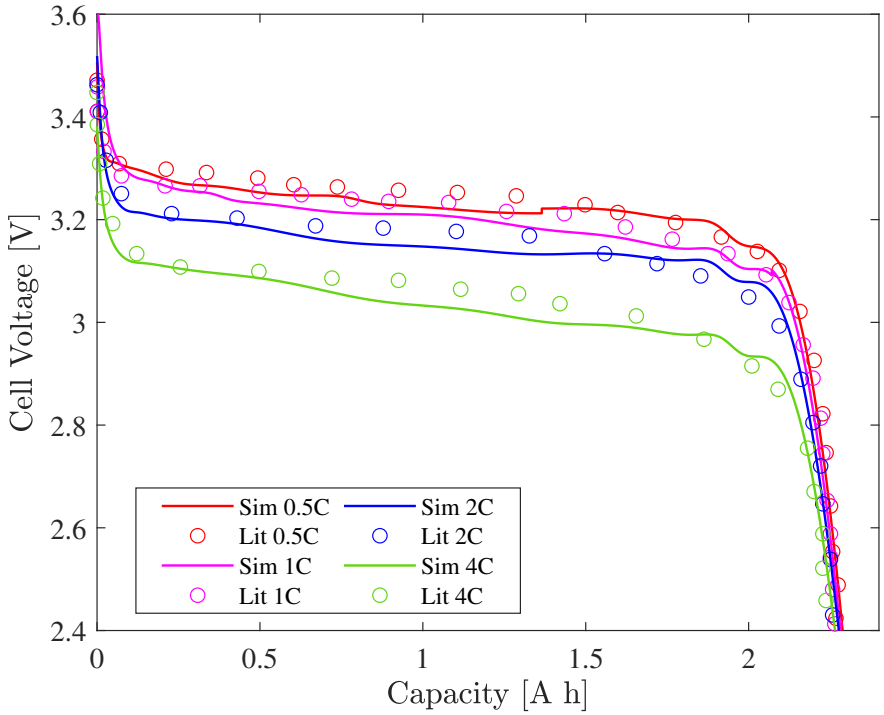


Figure 2.13: Capacity vs. Cell Potential for Discharging

Overall, the simulations slightly underestimate the cell potential for both the charging and discharging scenarios. This is likely a result of the control schemes utilized. For this work, a PID controller was implemented to realize the CC-CV-charging protocol. The PID controller manipulates current as the input variable into the simulation; a separate file was created that designates current as a function (denoted as i_{app} in Figure 2.3) that is controlled via the PID controller. The process/control variables for this work are the voltage and the capacity in the CC and CV periods respectively [35]. Specifically, the control algorithm for charging began with the CC period, then the first "switch" (in the current) dictated by proximity to pre-set V_{cutoff} (CV period), finally the switch to rest period dictated by capacity value δQ (δQ, V_{cutoff}, k_i, and k_p were manually altered for each C-rate). In Figures 2.12 and 2.13, it is observed that the CV period does not require strict control (held to a single value). Also of note is

that the CV period concerning the charging case is steeper than for discharging; this is reflected by a lower magnitude of k_i and k_p for charging than discharging, which indicates less aggressive control required for this scenario. The control scheme for discharging was very similar, but key differences include a sign-swapped nominal current and the initialization of the voltage at 5V. Meanwhile, Prada et al. do apply a PID controller, but it is unclear how the CC-CV protocol was implemented with regards to the controller [4]. Also considered when undertaking this work was the use of a derivative condition for the trigger into the CV and rest period. However, this proved difficult to control; the complexity of the voltage component equations led to large variances in the numerical computation of the derivatives. In turn, changes to the control settings or any other parameters resulted in dramatic changes and numerical errors.

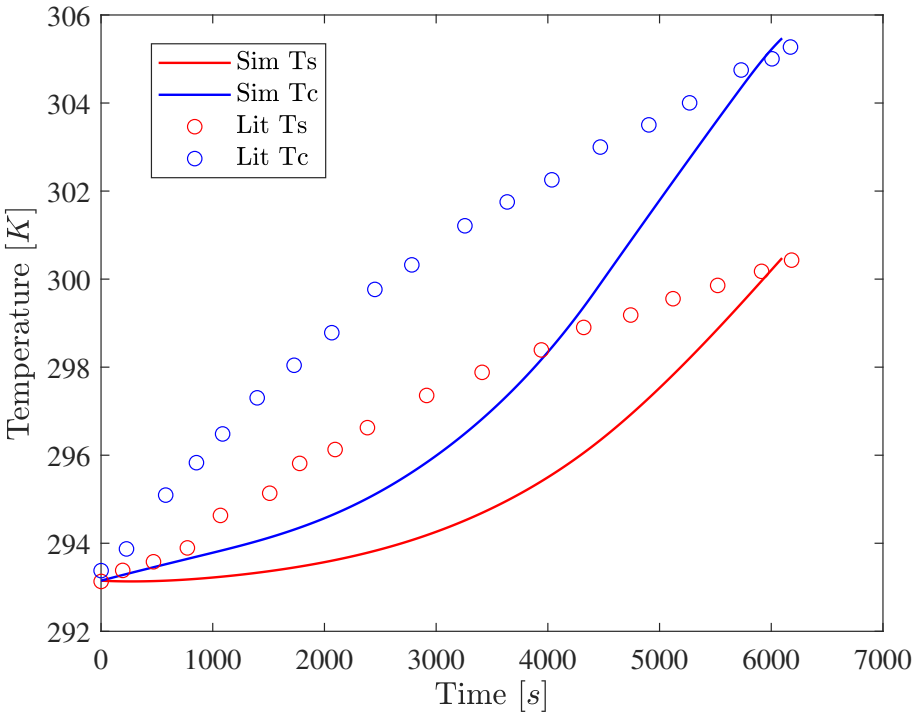


Figure 2.14: Core and Surface Cell Temperature Literature Comparison

Finally, the core and surface temperatures as a function of time are compared to simulation results obtained by Perez et al. in Figure 2.14. The core temperature is

predicted to be greater than the surface temperature, and to have a quicker increase with time. The results were obtained at a 6 C-rate in literature, and recreated via simulation in the current manuscript. There is good agreement in terms of magnitude and for the trend of $T_c > T_s$. As may be seen in Figure 2.14, the simulations show a convex curve as opposed to the concave shape observed in [5]. This presumably is due to the differing charging protocols; the Perez work utilized an optimization scheme for the simulation of their proposed model, which implemented an "optimal charge protocol" as opposed to the traditional CC-CV scheme.

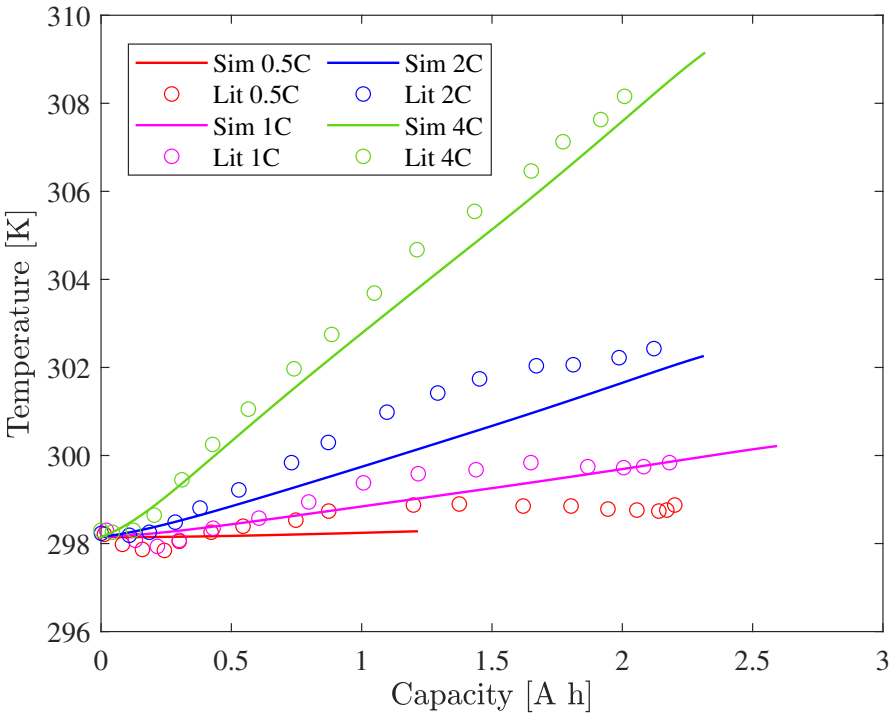


Figure 2.15: Simulation of Capacity vs. Surface Temperature Comparative to Experimental Values for Various C-Rates

Figure 2.15 shows a comparison between the simulations and literature for temperature versus capacity. It is observed that the change in temperature increases significantly with growing C-rates; as mentioned previously, the rate of (de)intercalation is accelerated at higher currents. The ensuing considerable change in the overpotential generates a larger amount of heat, and consequently a greater change in temperature

in the battery [13]. There is acceptable agreement with the literature; the overall trends are the same but there is variance in the shapes of the curves. Once again, a contributing factor to this is the control settings which dictate the current profile. Moreover, the thermal model considered in the current manuscript is different than that implemented in [4]. The model realized by Prada et al. is a simplified model adopted from the work of [21]. It considers surface and internal temperature as functions of resistances (which are expressed as functions of the thermal conductivity of the cell and the Newton Convective coefficient in relation to the geometric parameters of the cell). However, in the current manuscript, the thermal model was based on the work of Perez et al. [5]. A key difference is that the model takes into account a heat generation term, which provides an indirect connection to the concentrations in the cell (through U_k). Thus, the produced temperature is fed back into the model and continuously updated for the most realistic implementation.

Software specifications of an Intel(R) Core(TM) i7-7700HQ, with a CPU @ 2.80GHz, were used to complete the simulations via MATLAB. The use of a newer computer (as this work was completed on a 7-year-old laptop) would also decrease the run-time of the code; for reference, a CPU of 3.5 - 4.0GHz on a desktop is appropriate for current gaming and more demanding tasks, and above 4.0GHz is considered excellent for high-performance tasks [36]. Furthermore, the Intel i7 processor core has been discontinued as of May 24th, 2024. The approximate running time varied depending on the C-rate, with larger current loads resulting in exponentially quicker simulations. Moreover, the amount of data is approximately 25,000 points and could be scaled to increase the speed of the simulations. For the 1C-rate charging simulation, the total run-time (including the plotting of over 30 figures) is approximately 2 minutes with these settings. Meanwhile, for a 4C rate, the total simulation time is approximately 90 seconds. This decrease shows the high variability of simulation time due to the current load.

A key factor that directly impacts the computational scalability is the discretiza-

tion lengths; given that these are based on the geometry of the battery, it is likely that significant changes to the parameters used would be required before altering these values. Increasing the discretization lengths would decrease the resolution of the simulation data by increasing the size of the resulting interval,s [26]. Moreover, the approach to the voltage calculations also greatly influences the data density. The approach of Marquis utilizes more simplifications than that of Richardson; particularly, the use of a spatially averaged exchange current density. Thus, the Richardson approach has an additional spatial-temporal component in the calculation of reaction overpotential which increases the computational run-time of the simulation but yields higher accuracy at higher C-rates [3].

2.5 Conclusion

An extended single-particle model coupled with a thermal model was developed by adopting and combining the approaches of Brosa et al.[3], Richardson et al.[12], Forgez et al.[21] (as shown in [4]), and Perez et al.[5]. Validation of the model was completed via the comparison of literature and experimental values to that of the simulations. Anticipated trends are observed for key states such as electrode potential, voltage, and temperature. Moreover, there is excellent agreement for the electrode potentials as a function of normalized concentrations when compared to the literature. Acceptable concurrence is noted for the relation between experimental data and the simulation temperature predictions; this could be improved by the inclusion of a term to represent the reversible contribution of heat transfer between the battery and the environment (and vice versa). The next steps for this work are to complete parameter estimation (geometric, and the coefficients for the thermodynamic potential functions) utilizing machine learning techniques [37]. After which a controller and observer system may be designed and implemented. The work presented in this thesis has potential applications in battery management systems, lithium-ion battery design, and for use in future studies of lithium-ion battery behaviours. Possible im-

provements to this work include the implementation of a current-dependent radius for the electrode "particles", a concentration-dependent diffusivity for the electrodes, a further automated controller for varying C-Rates (eliminate manual manipulation of k_i, k_p, V_{cutoff}, and δQ), and consideration for degradation mechanisms (which are of significant importance at higher C-rates). Another possible avenue for future study is the application of this model to multi-cell stacks, with a particular examination of computation efficiency.

Chapter 3

State of Charge and Average Temperature State Estimation of SPMeT Lithium Ion Battery Model via LSTM

3.1 Introduction

As a result of the ongoing climate crisis, alternatives to traditional forms of energy such as coal and gas are becoming more and more popular [3, 38, 39]. Lithium-ion batteries are a promising candidate for energy storage systems, and are already being used in many portable applications and electric vehicles [3, 38–40]. A battery management system (BMS) is what controls the battery during operation, and relies on accurate data to operate effectively [38, 40, 41]. One key parameter to BMS operations is state of charge (SOC); SOC quantifies the difference between a battery in use and its nominal/fully charged capacity [38, 40–44]. Another very important characteristic when it comes to battery control and design is the internal temperature of the cell [39, 45, 46]. Several safety concerns arise when operating batteries if the internal temperature is outside the appropriate limit, in addition to accelerated degradation and poor performance [39, 45, 46]. Therefore it is vital to have an accurate estimation of the internal temperature of a cell during charging/discharging under varying current loads. Moreover, the speed of obtaining these estimations (SOC

and temperature) must be fast enough to be utilized in real-time applications [38–46].

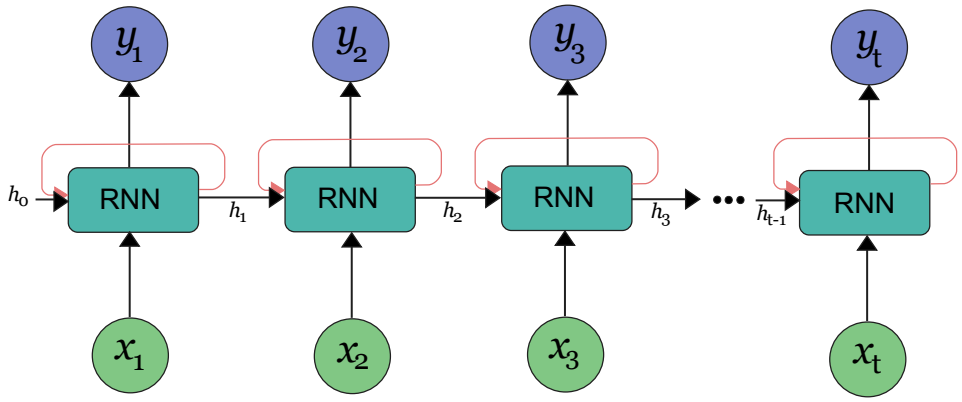


Figure 3.1: General RNN Configuration - Unrolled [47]

The estimation of SOC has been extensively studied, and may be separated into 4 main categories: direct measurement methods, book-keeping methods, adaptive methods, and hybrid methods [44]. Given the recent advancements in machine learning (ML) and studies of deep learning networks, there is increasing interest in utilizing data-driven and algorithm approaches instead of more traditional methods. The general idea behind machine learning is that an algorithm is utilized to learn a pattern between a set of inputs and outputs [47]. There are two primary categories of machine learning: statistic-based learning, and neural networks. While statistic-based learning has its merits, the more suitable approach in this case is the use of neural networks; neural networks typically show better RMSE values than that of conventional regression methods, and thus are the preferred method of choice for battery state estimation [48]. Some forms of neural networks include convolutional NNs (CNN) and recurrent NNs (RNNs). While CNNs are efficient at extracting positional invariant features, they are also hierarchical in nature [47, 48]. Meanwhile, RNNs provide much more

flexibility, and are better suited for sequential modeling as they take into account previous state information [47]. The hidden state which is calculated via equation (3.1) is what enables this [47].

$$h_t = f_w(h_{t-1}, x_t) \tag{3.1}$$

Where h_t is the new hidden state, h_{t-1} is the old hidden state, x_t is the input vector at some discrete time step t, f_w is a function with parameters w, and w refers to a set of weights [47]. In "vanilla" RNN's f_w is often a hyperbolic tangent function to provide nonlinearity [47].

However, traditional RNNs struggle with the so-called vanishing gradients problem [38, 47, 49]. To handle this issue, the long-short-term-memory (LSTM) network architecture was developed [38, 47, 49]. LSTM networks are a form of multi-layer RNN, which is composed of four different gates, and has two hidden states (hidden state and cell state) which must be maintained instead of one [38, 47, 49]. As there are four interacting internal states as opposed to one, LSTM networks can add or remove information to the cell state which preserves the older cell memory after multiple time steps [38, 47, 49]. A diagram illustrating the internal structure of an LSTM network may be seen in Figure 3.2, while the main steps for RNNs in general are outlined in Figure 3.1. While studies such as Yi et al and Jiang et al have completed SOC estimation for varying ambient temperatures using LSTM, typically the data is obtained from a database or using an equivalent circuit model, which eliminates the reflection of the internal characteristics of the cell during charging [3, 39, 50].

Motivated by the above work and findings, the contribution of this work is to combine "model" and "data-driven" estimation methods using data from previous work, which completed simulation and validation of an extended single particle model coupled with a thermal model, and completing supervised learning using an LSTM algorithm of both state of charge and average cell temperature for a lithium-ion battery for four different C-rates. Thus the generalizability of the previous modelling work is

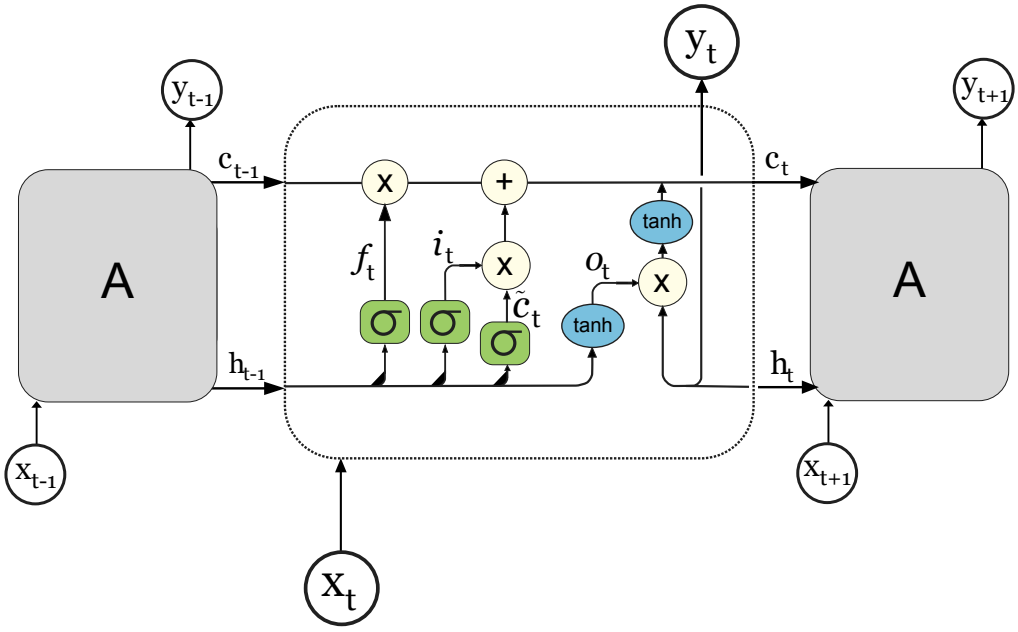


Figure 3.2: LSTM Internal Diagram [40, 47]

extended, and the internal characteristics of the cell dynamics have some reflection in the estimation of the cell states [44]. The remainder of this paper is organized as follows: problem formulation which contains a comprehensive literature review and network architecture, discussion, results, and finally conclusions. Additional information such as abbreviations, nomenclature, and hyperparameter definitions can be found in Appendix B.1 and B.2 respectively.

3.2 Problem Formulation

To approach this daunting task, first a literature review considering state estimation of SOC and internal cell temperature is completed, and the network theory and architecture are reviewed.

3.2.1 Literature Review

Data-driven methods such as neural networks, support-vector machines, regression techniques, and more, have been utilized to predict various measures of battery life; namely, the SOC, state of health, and remaining-useful-life. Roman et al designed a

machine learning pipeline to estimate battery capacity fade, and then evaluated the model on 179 cells which were cycled under different operating conditions [51]. The process involved refining and substantial testing of ML algorithms for application to capacity fade estimation [51]. Meanwhile, the work of Danko et al reviewed commonly used SOC estimation methods of all types, with advantages and disadvantages for each method described [52]. Neural networks in this instance/context are classified as an adaptive method; they use mathematical algorithms to process data and solve relations between numerous initial complex components [52].

More specifically, Zhao et al contributes an extensive review of the three primary steps for completing different types of SOC estimation methods via ML [44]. The advantages and disadvantages of traditional NNs versus deep-learning networks are also described [44]. Particularly of note is that traditional methods of SOC determination have difficulty resolving the complex battery models' responsiveness to model parameters, poor flexibility, and high computational complexity [44]. Such lacking efficiencies which are consequent of the uncertainty and complexity inherent to most battery models highlights the necessity for further work into real-time capabilities, accuracy, robustness, and flexibility of SOC estimations [44]. One path to improving the accuracy of SOC estimation, generalizability of the models, and overall model performance is the use of high-quality datasets [44]. Such datasets contribute greatly to this field of research by expanding the variability of data, and facilitating more precise predictions and optimizations concerning SOC determination models [44]. Possible evaluation indicators for these models include mean average error (MAE), mean squared error (MSE) and the root mean square error (RMSE) [44]. Typically most literature uses RMSE, and some combination of MSE, MAE, or another form of error value [38, 44, 52]. Equations (3.2), (3.3), (3.4), and (3.5) were used to calculate the general standard deviation and standardization, RMSE, and MSE respectively. Where X_i is a general variable X at instance i, μ is the mean of variable X, n is the size of the population, and σ is the standard deviation of X. In this instance, Y_{pred} is

the predicted value from the network and Y_i is the testing value from that particular time step.

Moreover, the comparison of hyperparameters - which are manually manipulated and directly affect model performance and generalizability - is of great interest when reviewing studies pertaining to this topic [44]. Tables 3.1 and 3.2 compare some of the hyperparameters used in this work with others in the literature.

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(X_i - \mu)^2} \tag{3.2}$$

$$X_{std} = \frac{(X - \mu)}{\sigma} \tag{3.3}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Y_{pred,i} - Y_i)^2} \tag{3.4}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_{pred,i} - Y_i)^2 \tag{3.5}$$

As aforementioned in the introduction, the use of LSTM algorithms in this application has been shown to be effective and accurate, with benefits over traditional NNs and some other RNN configurations [44, 47, 48]. Yang et al extended a previous work which completed SOC estimation using a gated RNN by proposing an LSTM network to describe the complex behaviours of lithium-ion batteries under diverse

Table 3.1: Hyperparameters for SOC

Hyperparameters (SOC Estimation)					
Parameter	**Value 1**	**Value 2[44]**	**Value 3[38]**	**Value 4[40]**	**Value 5[41]**
Epochs	150	3000		150	2000
Initial Learning Rate	0.01	0.01	0.01	0.01	0.01
Learning Rate Drop Period	25				
Minibatches	32	89	60	64	60
Loss Function Optimizer	ADAM	ADAM	ADAM	ADAM	

Table 3.2: Hyperparameters for Temperature Estimation

Hyperparameters (Temperature Estimation)					
Parameter	**Value 1**	**Value 2[50]**	**Value 3[46]**	**Value 4[53]**	**Value 5[45]**
Epochs	300	8	1000	5000	
Initial Learning Rate	0.01	5		0.01	0.00001
Learning Rate Drop Factor	0.5	0.9999	0.2	0.1	0.1
Learning Rate Drop Period	25		200	1000	
Minibatches	32		256		
Loss Function Optimizer	ADAM		ADAM		ADAM

ambient temperatures [38, 41]. Furthermore, an unscented Kalman filter (UKF) is integrated to filter out noise from the data and provide an even more accurate estimation [38]. The method presented is model-free and entirely data-driven, and provides satisfying SOC estimation under multiple operating temperatures [38]. One particular reason LSTM is more suited for state estimation as opposed to classic RNN configurations is that instead of gradient backpropagation occurring exponentially, the LSTM allows gradient flow to be unchanged by using a cell state [38, 47]. Moreover, the four different gates as shown in Figure 3.2 decide what data is retained vs "forgotten" and thus able to address long-term dependencies [38, 47]. Two major roadblocks with regards to SOC determination from complex battery systems are the inability to cope directly with varying ambient temperatures, and the flat regions of the OCV-SOC curves (especially present in LFP batteries and studies of porous electrode theory) can cause large fluctuations in the SOC due to small errors in voltage measurement [38]. The benefit of using ML methods, and LSTM networks in particular is that these deficiencies may be addressed without increasing the run time of the simulations [38, 40]. A trade-off must be made however between testing accuracy and training cost; the work of Yang determined an epoch number of approximately 8000, with a 1.7-hour training time [38].

There have also been efforts, although more limited than that of SOC estimation, into the state estimation of cell temperatures via ML. While traditional methods

have been used for temperature estimation such as the measurement-based method by Richardson et al or a simplified thermal model combined with a Kalman filter, a common drawback of these approaches is the assumptions required to implement such models [43, 50, 54]. These assumptions can limit the applicability and reduce the estimation accuracy of the models [50]. Consequently, there have been increased investigations into the use of NN's for the determination of temperature [46, 50]. Two types of RNNs, an LSTM and a gated recurrent unit (GRU) algorithm, are proposed for the estimation of the surface temperature of LIBs during discharging under varying ambient temperatures by Jiang et al [50]. Datasets from the Prognostics Center of Excellence were used to train, validate, and test the two different networks [50]. While in previous studies the temperature was fixed as the output, Jiang et al elected instead to adopt the temperature difference along the time axis as the output [50]. The results are promising, with both RNN types demonstrating accurate real-time temperature estimation [50].

Additionally, the LSTM network shows better performance in trend tracking of the temperature variance when compared to the GRU NN, although a slightly longer training time is required [50]. Similarly, Cho et al proposed a hybrid LSTM-Physics-informed neural network (PINN) method for estimating LIB pack temperature [45]. In this instance, the algorithm makes use of an exponential function and shows more accurate results for a direct current fast charge protocol [45]. The PINN is a NN type approach which conducts learning by incorporating physics laws into the loss function; while this method has recently attracted attention in many applications, it does not include representation of the electrochemical mechanics present during discharging/charging of LIBs [45]. Another example of the use of LSTM for temperature prediction is that of Yi et al [39]. The work demonstrates a digital twin (DT) technology and LSTM-based method for real-time temperature prediction and degradation analysis of lithium-ion batteries [39]. The DT model is formulated based on lumped thermal equivalent circuit models (ECM) to describe the dynamic thermal

behaviour of LIB cells, and the results from two different C-rates are compared [39]. This proposed approach provides acceptable accuracy for real-time temperature prediction for the charging process, and uses the Pearson correlation coefficient (PCC) to extract parameters displaying high correlation with the DT model parameters from the charging curve [39]. However, ECMs lack representation of the electrochemical dynamics within the cell and thus do not fully describe the internal physical characteristics which occur during charging/discharging cycles [3, 39]. The next section of this thesis will discuss LSTM theory in more detail and why it is the chosen algorithm for this problem, as well as the architecture and settings of the developed network.

3.2.2 LSTM Architecture

LSTM is a form of multi-layer RNN which was developed by Hochreiter and Schmidhuber to help deal with the vanishing gradients problem [38, 47, 49]. The vanishing gradients problem refers to error signals vanishing during conventional backpropagation; the tendency to vanish arises from overly long learning time, which yields time lags between relevant inputs [38, 49]. Moreover, there is the issue of older inputs being "forgotten" by the network as time steps increase with previous RNN configurations [38, 49]. One of the key features of LSTM networks is the uninterrupted gradient flow, which can be thought of as a gradient highway, as opposed to the backpropagation of gradients which is present in classic RNNs [47]. LSTM achieves this through the implementation of a cell state c_t, which is an internal state that provides (in combination with the four internal gates) the ability to remove or add information and address long-lasting reliances [38, 47]. The four aforementioned gates are: the input, forget, output, and gate gate.

$$\begin{pmatrix} i \\ f \\ o \\ \tilde{c} \end{pmatrix} = \begin{bmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{bmatrix} w^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix} \tag{3.6}$$

Where i, f, o, and $\tilde{c}$ are the input, forget, output, and gate gate respectively [47]. Note that each gate has a different associated non-linearity, such as a sigmoid function (σ) or hyperbolic tangent (tanh) [47]. b_k and w_k where $k \in i, f, o, g$ is the bias and weight matrices respectively, and correspond to the associated gate [40, 47].

$$\begin{pmatrix} i_t \\ f_t \\ o_t \\ \tilde{c}_t \end{pmatrix} = \begin{bmatrix} \sigma(w_i[h_{t-1}, x_t] + b_i) \\ \sigma(w_f[h_{t-1}, x_t] + b_f) \\ \sigma(w_o[h_{t-1}, x_t] + b_o) \\ \tanh(w_c[h_{t-1}, x_t] + b_c) \end{bmatrix} \tag{3.7}$$

To begin the forward pass, the forget gate f is the first, and determines which information is kept; the previous hidden state h_{t-1} and current input x_t are concatenated and then go through this gate which is calculated by equation (3.7) [47]. The sigmoid function provides an output between 0 and 1 for each element in c_t, where 1 indicates keep completely and 0 is erase completely [47]. Next, the input gate determines which value to update; once again h_{t-1} and x_t and inputs to calculate the gate as in equation 3.7, with 1 meaning update completely, and 0 ignore completely [47]. Subsequently, the gate gate creates candidate values for the cell state via equation (3.7) [47]. Due to the hyperbolic tangent, the output from this gate is a number between -1 and 1 for each element to be added to c_t [47]. The cell state is updated by erasing data from it as determined by the forget gate, and then the new values are added as determined by the gate gate; the mathematical formulation is shown as [47]:

$$c_t = f \bullet c_{t-1} + i \bullet \tilde{c}_t \tag{3.8}$$

Where $\bullet$ refers to element-wise multiplication. Next is the output gate which produces the output y_t from the cell state [47]. This is determined by using the h_{t-1} and x_t once again to output a number between 0 and 1 for each element which must be revealed from the cell state to the output y_t, as shown in equation (3.7) [47].

Finally, the hidden state is updated as dictated by the output gate and updated cell state and passed through a hyperbolic tangent to limit the values between -1 and 1, which is calculated as [47]:

$$h_t = o \bullet \tanh(c_t) \tag{3.9}$$

As previously stated, Figure 3.2 is a representation of the internal structures outlined [47]. Considering the backwards pass, the backpropagation from c_t to c_{t-1} is only elementwise multiplication by the forget gate, and does not involve multiplication by the weight matrix as in traditional RNNs [47].

The presented walkthrough of the forward pass and mechanisms allow the LSTM network to remember previous inputs from multiple time steps prior - which is vital for a sequential/time-series problem [38, 47, 48]. Moreover, the hyperbolic tangent function can provide quicker convergence than an analogous process with a non-symmetric activation function (such as a sigmoid function) [38]. The backwards pass uses a loss optimization function which is set by the user; most often in literature for LSTM networks this is designated as the adaptive moment estimation (ADAM) [38, 40, 44]. The ADAM algorithm minimizes total loss by updating the weights and biases of the network as indicated by the gradient of the loss function [38, 44]. Note that during training, a single epoch refers to one or more batches, with each batch consisting of a forward and a backward pass [38]. The training process continues through batches while the forward and backward passes continually update the network until convergence or validation criteria are met [38, 47].

A diagram overview of the steps to complete the state of estimation via ML in this work is shown in Figure 3.3. Choosing the inputs/features and outputs/responses

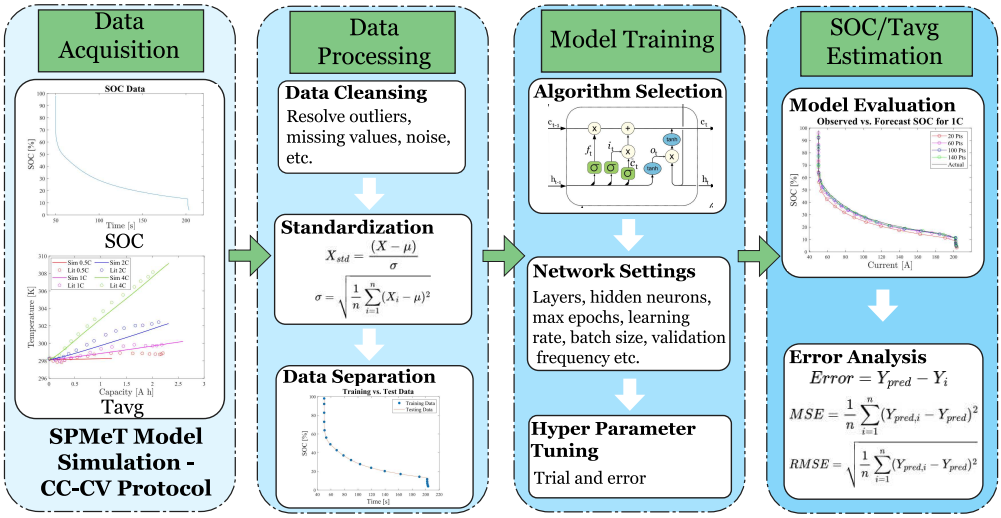


Figure 3.3: State Estimation Overview

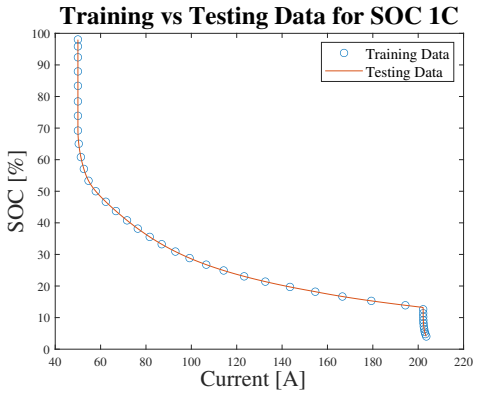
for an ML problem is one of the most important and challenging issues [47]. For this thesis, the SOC and average cell temperature were chosen as the responses to the network. SOC was selected as it is one of the most significant battery health indicators and is of utmost interest to manufacturers, BMS usage, and more [38, 40, 41, 44]. Meanwhile, average cell temperature was chosen due to its significance in contributing to degradation, safety implications, and ensuring high efficiency in charging/discharging procedures [11, 45, 50, 53]. The input into the algorithm in this work was the applied current, with 4 different C-rates considered (0.5C, 1C, 2C, and 4C). This choice is the most logical as it is what is applied to charge a battery in practice, and is the input into the electrochemical-thermodynamic model developed in the previous work [3]. However, an examination of the implementation of a multi-input single-output (MISO) network was completed, where the current, voltage, and capacity were taken as inputs. The resulting network is much more complex to ensure acceptable results; cross-validation checks were implemented with 5 folds total, as well as dropout layers for regularization and to prevent overtraining, and two layers of hidden units as opposed to one (containing 100 hidden units each). The dropout

layers have a rate of 20 % [38]. To elaborate, the purpose of the cross-validation was to determine the optimal amount of training data versus testing data; in this instance, the validation data used was a portion of the training data to ensure the testing data remained unseen by the algorithm before testing [44]. In contrast, the single-feature designation was evaluated at varying amounts of training data to enable a more in-depth investigation. The multi-feature configuration performs sufficiently, but it has a much longer run time (approximately a minute longer in total) when compared to the single-input single-output configuration. Nonetheless, it was of great interest to examine this option, and does provide greater flexibility for included features and improved generalizability of the model [44].

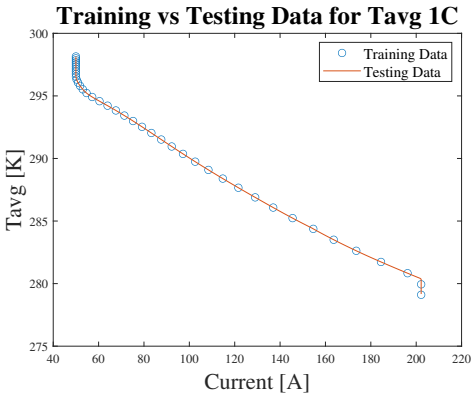
Also integral to network specification is the hyperparameter specifications. Tables 3.1 and 3.2 contain the hyperparameters utilized in this work and others found in similar studies in the literature. For estimation of SOC, an initial learning rate of 0.01 and loss function optimizer as ADAM is the default, but the other variables seem to have much more differentiation. The maximum epochs settings have the greatest deviations, with a range of 150 to 3000 [38, 40, 41, 44]. Meanwhile for internal temperature studies the only commonality is the loss function optimizer [45, 46, 50, 53]. As for SOC, the greatest departures are seen in the epoch setting, with a range of 8 to 5000 [45, 46, 50, 53]. The maximum epoch settings must be carefully considered to balance run-time and prevent over-training [38, 44].

3.3 Results

This section contains the state estimation results from the LSTM network and is organized into 5 subsections: data separation, SOC estimation, temperature estimation, estimation using multiple inputs, and estimation using noisy data.



(a) 3.4a



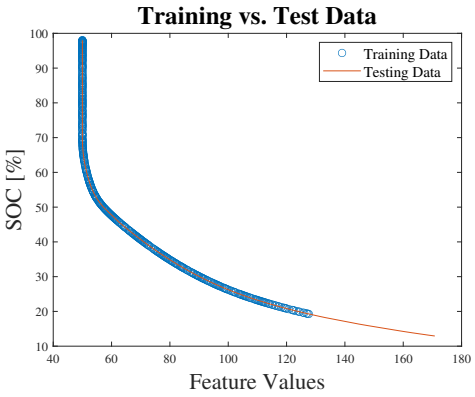
(b) 3.4b

Figure 3.4: Data Separation - Evenly Spaced for 1C; a) Training vs. Testing Data for SOC b) Training vs. Testing Data for T_{avg}

3.3.1 Data Separation

Two figures showing the two different methods for data partitioning examined in this work are presented; the evenly spaced training data with 40 points, and the first 70 % of data for training (as is traditional) for the multiple input configuration are depicted for both SOC and T_{avg} in Figures 3.4 and 3.5 respectively.

(a) 3.5a



(b) 3.5b

Figure 3.5: Data Separation - Traditional Format for 1C; a) Training vs. Testing Data for SOC b) Training vs. Testing Data for T_{avg}

To ensure the validity of the model and provide a comprehensive analysis, both separation methods were used, where the primary format shown in Figure 3.4, and the method in Figure 3.5 used as verification.

3.3.2 SOC Estimation

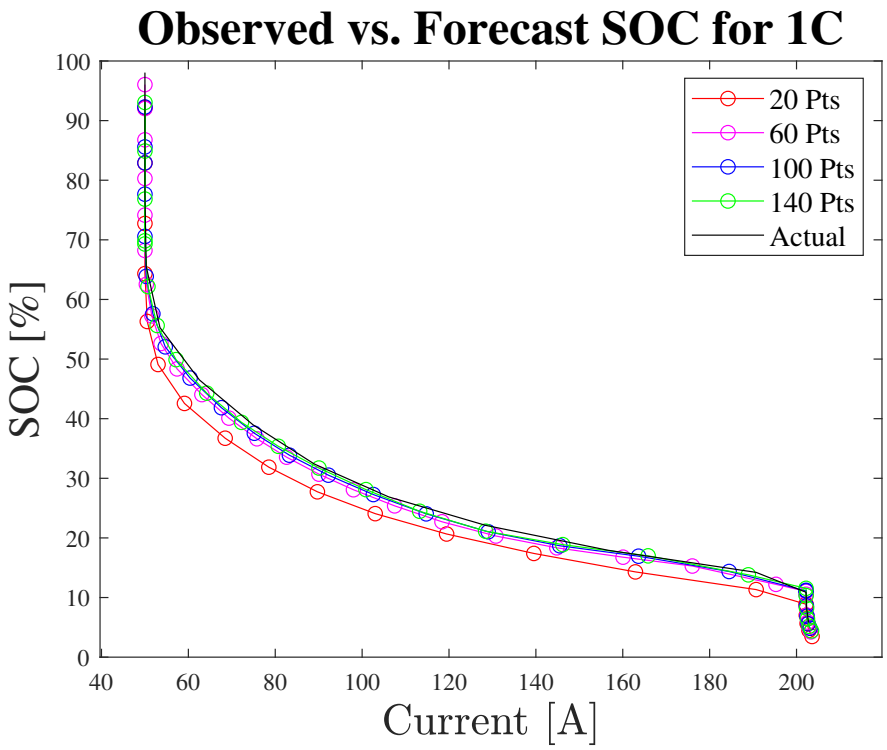


Figure 3.6: Observed vs Forecast 1C

Figure 3.6 shows the observed versus forecast values of SOC for varying amounts of training data at 1C. Meanwhile, a similar figure shows these results for multiple C-rates instead in Figure 3.7. As can be seen in Figure 3.6, as the number of training data points increases, the algorithm predicts closer to the actual data; this is as expected. Moreover, the LSTM results in Figure 3.7 are promising, with the forecast values increasing in accuracy as C-Rate decreases.

Table 3.3: Evaluation Metrics for SOC Estimation

Evaluation Metrics - SOC					
C-Rate	**Training Points**	**RMSE** [%]	**MSE** [%]	**MAE** [%]	**PCC**
0.5	20	4.166	17.356	3.732	0.9991
	60	1.372	1.881	1.226	0.9999
	100	0.835	0.696	0.735	0.9999
	140	0.894	0.800	0.561	0.9996
1	20	5.461	29.817	4.728	0.9991
	60	1.826	3.334	1.581	0.9999
	100	1.913	3.658	1.050	0.9986
	140	2.564	6.576	0.857	0.9965
2	20	3.206	10.276	3.136	0.9995
	60	1.054	1.110	1.022	0.9999
	100	0.669	0.447	0.617	0.9999
	140	0.950	0.903	0.480	0.9991
4	20	2.473	6.117	2.435	0.9997
	60	1.118	1.251	0.854	0.9987
	100	1.017	1.035	0.544	0.9982
	140	0.995	0.989	0.405	0.9980

Comparing the error values of the separation methods, there is a slight increase in error when applying traditional data partitioning but is negligible. To be specific, for the 1C case, the customary approach yielded an RMSE of 2.504%, MSE of 6.270%, MAE of 0.454%, and PCC of 0.9934. Meanwhile, the corresponding values for the primary form of data separation are shown in Table 3.3; where the RMSE of the standard data separation is actually lower than the training results using 20 and 140 training points. Therefore, we have verified that the method of evenly spaced data

partitioning across the entire data set is valid for training.

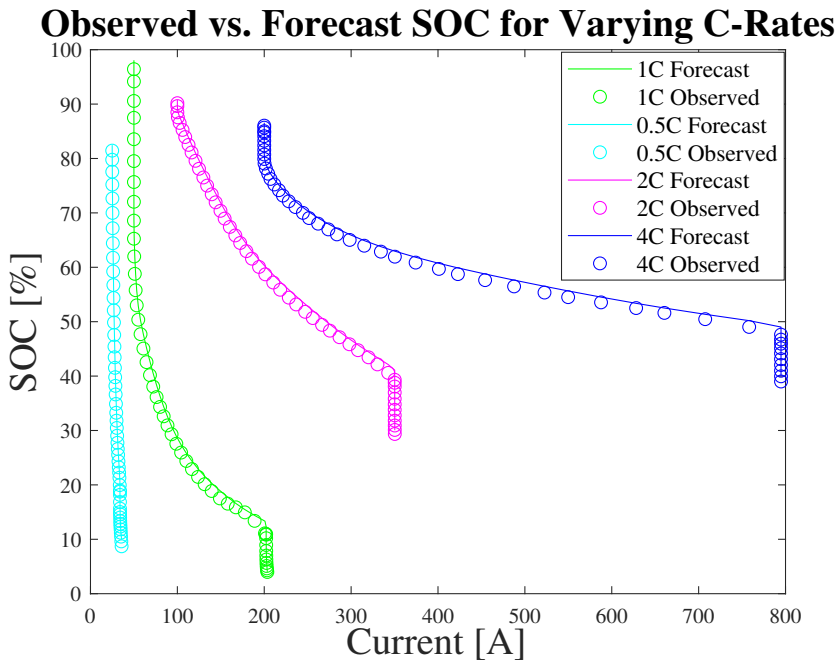


Figure 3.7: Observed vs Forecast All C-Rates

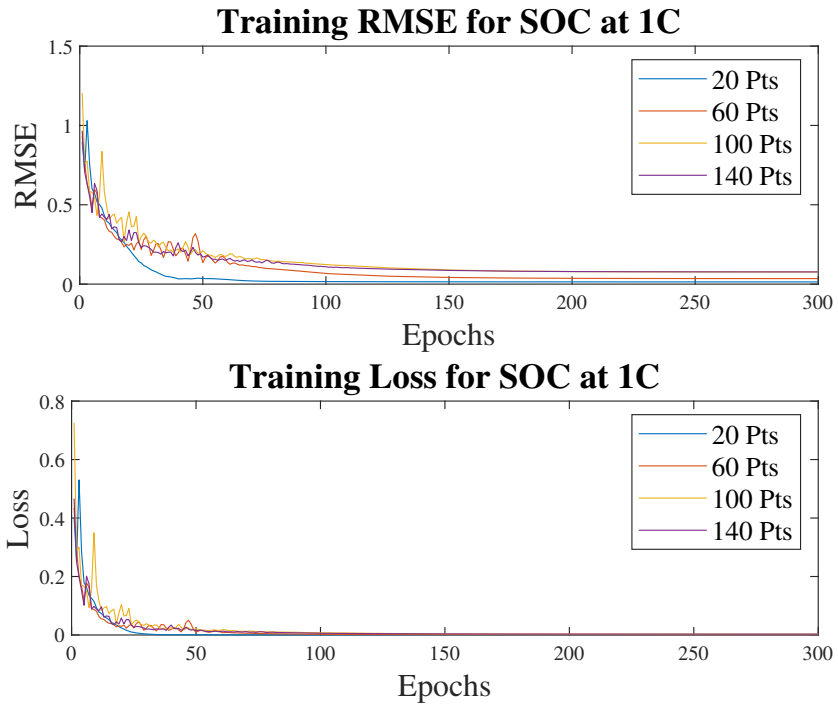


Figure 3.8: Training Progress

The training progress is illustrated in Figure 3.8, and features the RMSE and loss over the course of training. While lower amounts of training results exhibit smoother curves for training progress, the overall values are typically higher. This is supported by the evaluation metrics shown in Table 3.3.

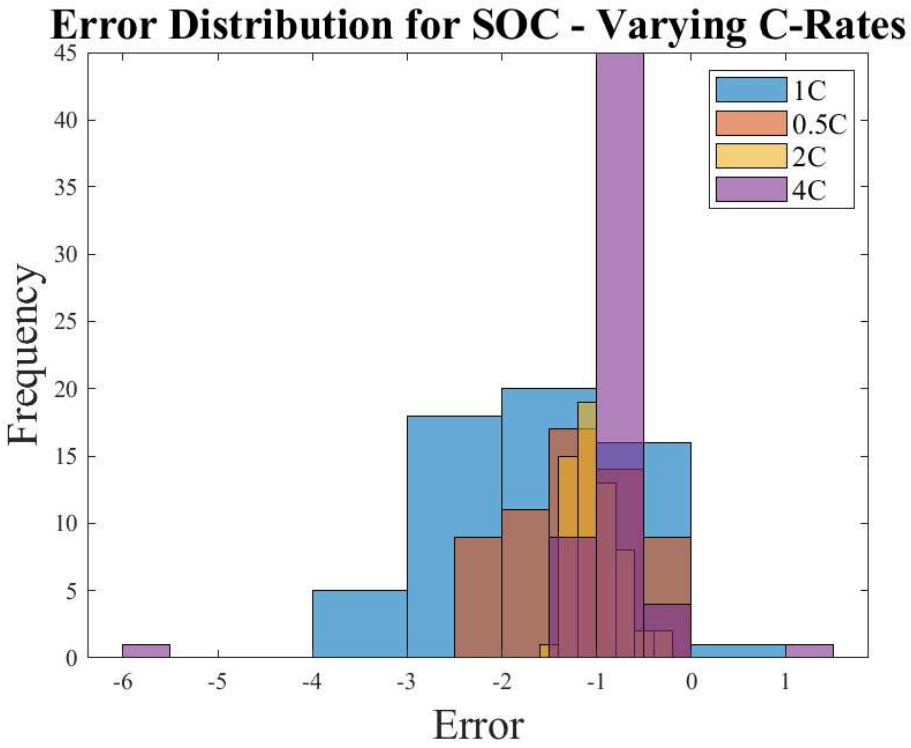


Figure 3.9: Error Distribution

Figure 3.9 displays the error distribution for SOC for varying C-rates. While the higher C-Rates show lower error values, they exhibit higher frequencies of such errors; consequently, this generally yields a tighter distribution of errors as the C-Rate increases. For predictions with a 4C-rate, the most frequent error range is between 0 and -1%.

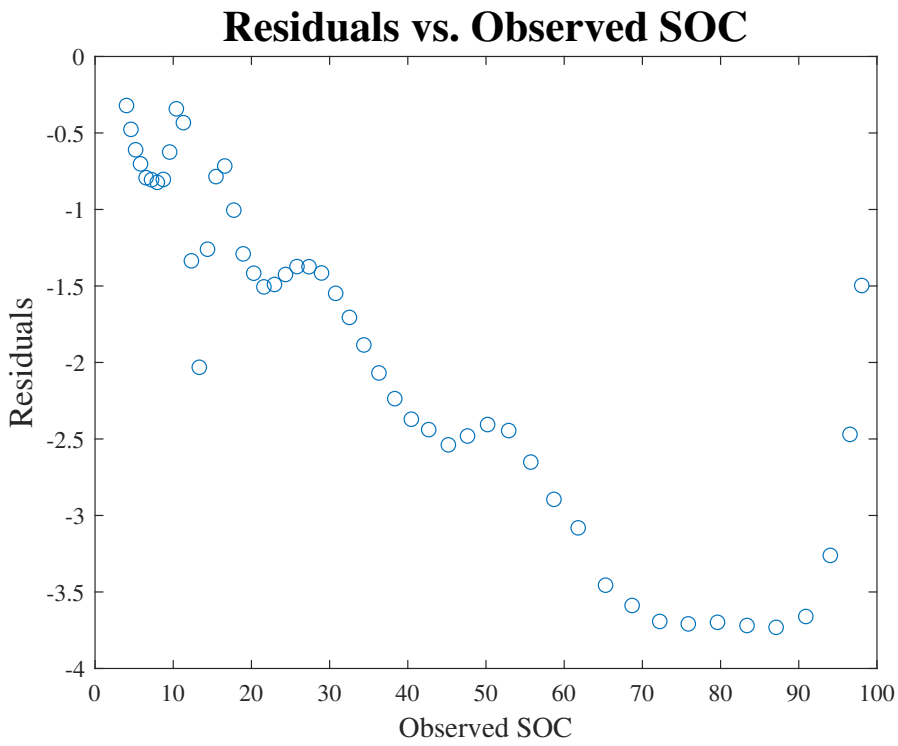


Figure 3.10: Residuals vs. Observed SOC for 1C

Lastly, Figure 3.10 demonstrates the relationship between the observed SOC and the associated residuals; where residuals in this case show the variance of the predicted data from the observed. The maximum residual is around -4 at an observed SOC of approximately 95%. This is validated by Figure 3.9 which shows the maximum error of -4 with a current setting of 1C.

3.3.3 Temperature Estimation

Once again, Figure 3.11 illustrates the observed versus forecast values of T_{avg} for varying amounts of training data for 1C conditions. Similarly to the trends in Figure 3.6, as the number of training points increases the algorithm yields predictions closer to the actual values.

Concurrently to Figure 3.7, Figure 3.12 features the observed versus predicted values for varying C-rates. Once again it is discerned that as C-rate increases, the resulting forecast values stray further from the associated observed data.

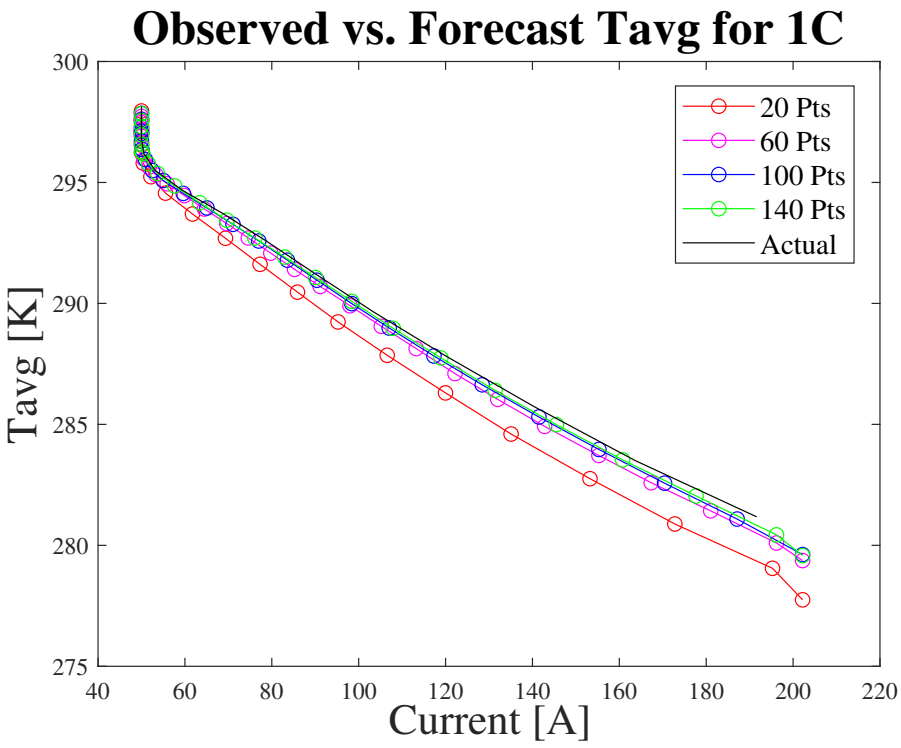


Figure 3.11: Observed vs Forecast 1C

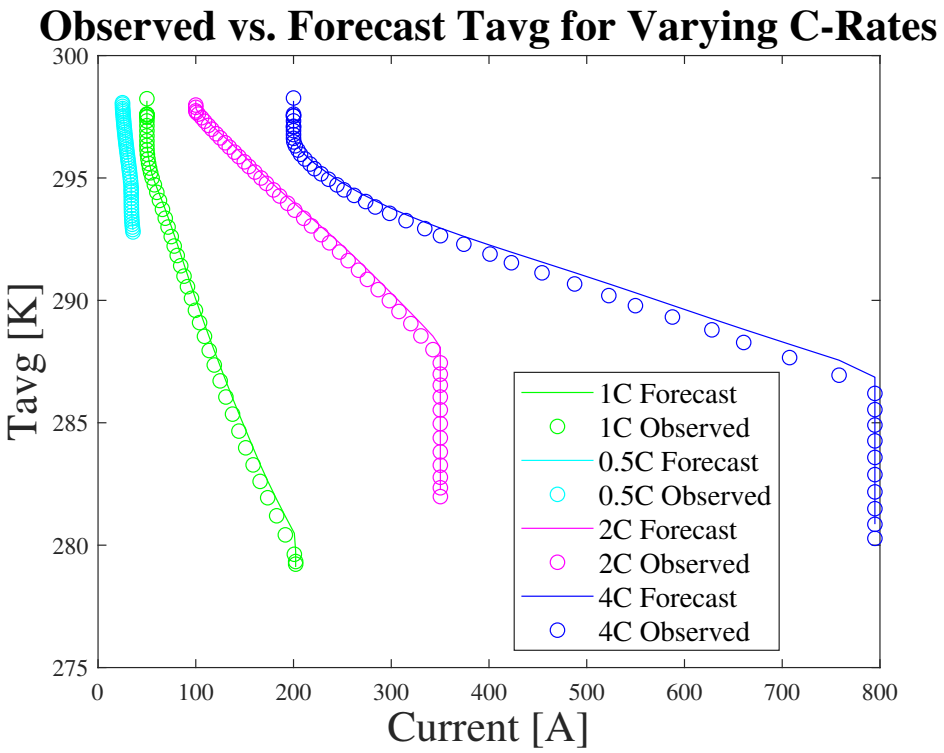


Figure 3.12: Observed vs Forecast All C-Rates

The training progress, namely the RMSE and loss development is shown in Figure 3.13 as a function of epochs. While decreasing the amount of training data yields a smoother curve, again the overall errors are larger. This is supported by Table 3.4 which shows the evaluation metrics for varying C-rates and training points for T_{avg}.

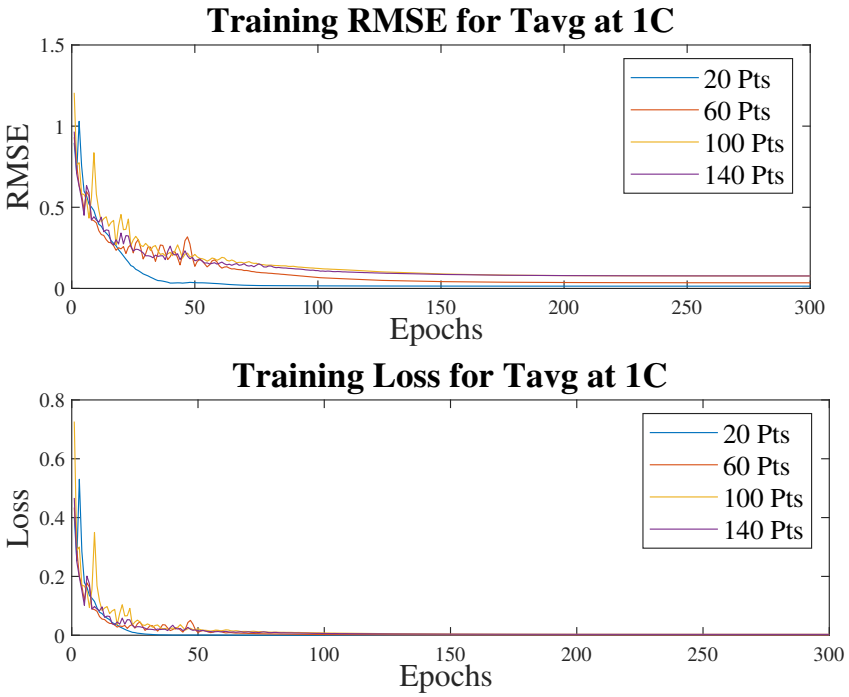


Figure 3.13: Training Progress

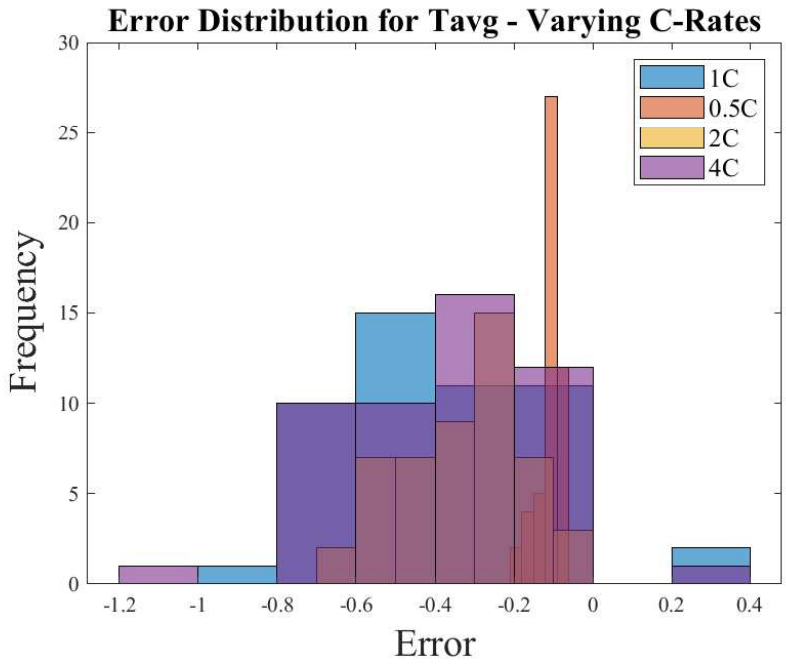


Figure 3.14: Error Distribution

Figure 3.14 shows the error distribution for the average cell temperature training for varying C-rates. Interestingly, the figure shows an opposite trend to that in Figure 3.9; instead, as the C-rate decreases so does the spread of errors, while the frequency increases. This is likely a result of the data with which the algorithm is trained - given that the settings for each C-rate remained the same and the plot contains data using runs from the same amount of training data. The results in Table 3.4 also mirror this observation. For 0.5 C-rate settings, the most frequent error ranged from 0K to -0.2K.

Table 3.4: Evaluation Metrics for T_{avg} Estimation

Evaluation Metrics - T_{avg}					
C-Rate	**Training Points**	**RMSE** [K]	**MSE** [K]	**MAE** [%]	**PCC**
0.5	20	0.311	0.097	0.296	0.9993
	60	0.092	0.009	0.090	0.9999
	100	0.059	0.004	0.055	0.9999
	140	0.063	0.004	0.041	0.9995
1	20	1.174	1.377	1.038	0.9990
	60	0.383	0.147	0.315	0.9995
	100	0.257	0.066	0.192	0.9996
	140	0.238	0.057	0.143	0.9995
2	20	0.963	0.927	0.872	0.9998
	60	0.301	0.091	0.269	0.9999
	100	0.259	0.067	0.170	0.9992
	140	0.239	0.057	0.119	0.9990
4	20	1.136	1.292	0.982	0.9995
	60	0.380	0.144	0.307	0.9994
	100	0.317	0.100	0.195	0.9989
	140	0.298	0.089	0.144	0.9987

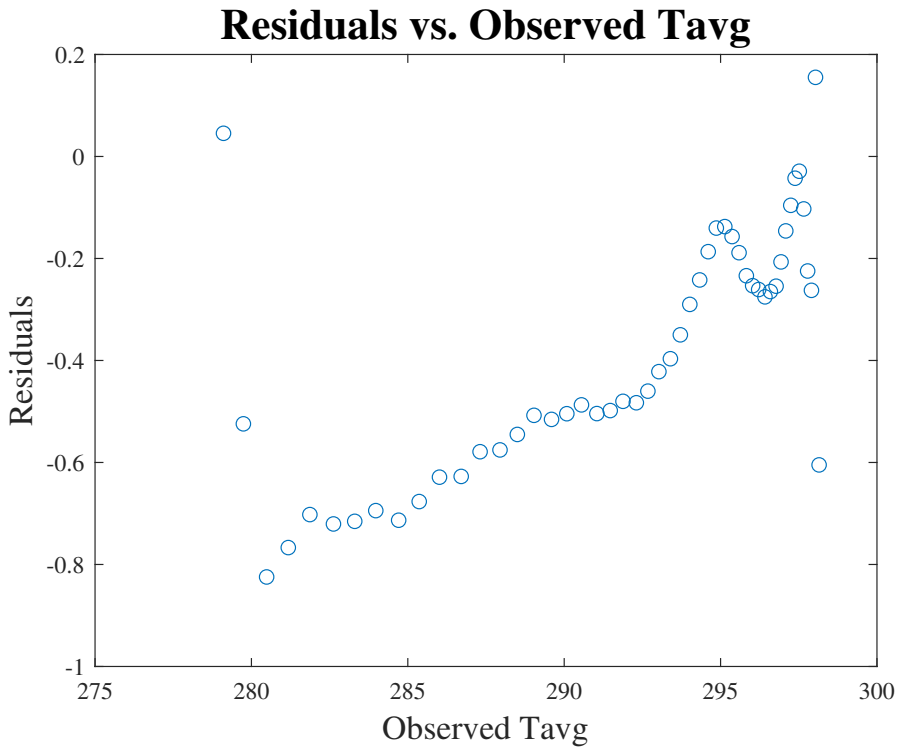


Figure 3.15: Residuals vs. Observed T_{avg} for 1C

Meanwhile, Figure 3.15 consists of the residuals versus the observed data for the average internal cell temperatures. The greatest magnitude of residuals is approximately -0.8 at an observed internal cell temperature of 280K. This result is confirmed by Figure 3.14, which shows a maximum error between -0.8 and -1 for 1C-rate simulations.

3.3.4 Estimation Using Multiple Inputs

Figure 3.16 depicts the observed versus forecasted values for the SOC estimation with multiple features as the input. Additionally, the evolution of prediction errors is also displayed alongside this figure. As the settings of the network were altered to complete these simulations there are some differences in the results when compared to Figure 3.6. The increased resolution of data (decreasing downsampling factor) yields a less smooth curve for the observed values. Moreover, the distribution of the prediction errors is more widely spread, with the highest errors occurring at the beginning of

the simulation.

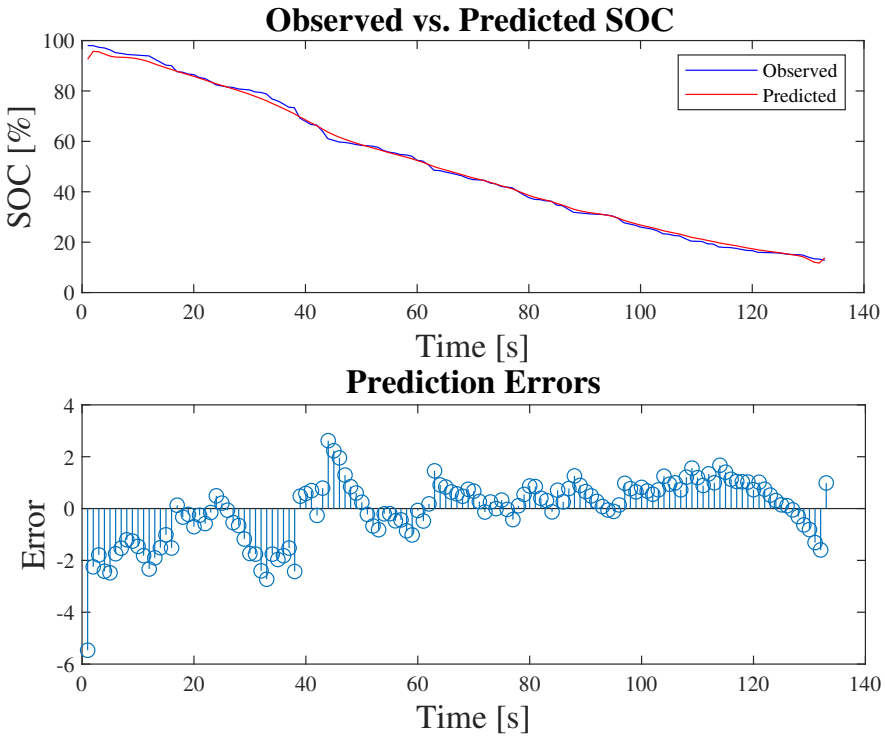


Figure 3.16: MISO Configuration Results for 1C - SOC

The evaluation metrics are as follows:

- RMSE=1.54
- MSE=2.370
- MAE=0.257
- PCC=0.994

These metrics show increased accuracy compared to the results depicted for 1C-rate simulations of SOC in Table 3.3; the lowest RMSE value being 1.826% for the SISO configuration. Although, overall the PCC shows stronger correlations for the SISO system than the MISO; this is expected as the associations between multiple features and one output are much more varied than that of a single feature and single output.

Meanwhile, the same set of figures is also depicted for the average cell temperature in Figure 3.17. The observed curve is once more less smooth due to the increased resolution of data (reduced downsampling factor). In contrast to Figure 3.16, the highest errors are at the end of the simulation. The evaluation of the MISO configuration for T_{avg} delivered the errors and PCC as:

- RMSE=0.346
- MSE=0.120
- MAE=0.148
- PCC=0.9984

In parallel to the comparison of the multi-feature to single-feature formats of SOC, the T_{avg} juxtaposition also shows increased accuracy with the MISO format. However, this holds only for approximately 60 training points or less. The RMSE for 60 training points is 0.383K, which is slightly greater than the 0.346K found from the altered network. Although, the MAE is approximately the same or less for all amounts of training data. This indicates stronger performance in terms of accuracy for the multi-input algorithms when referenced with the single-input networks for both SOC and T_{avg}. The PCC of 0.9984 is also less than the corresponding values of PCC for all training point values, which is expected due to the aforementioned reasoning of increased complexity in relationships between multiple features to one response versus one feature to one response.

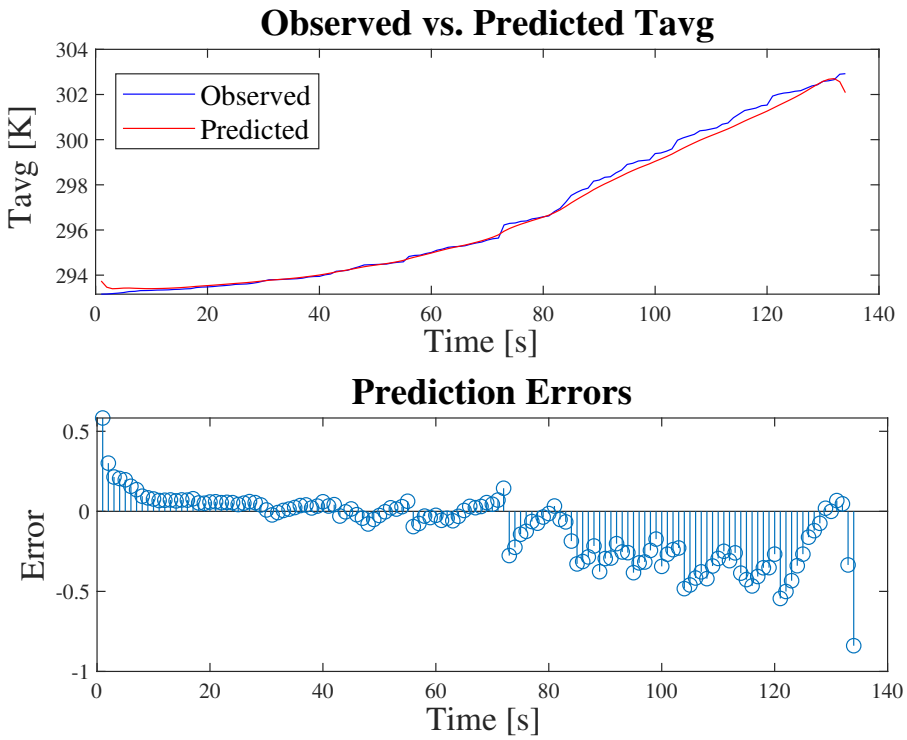


Figure 3.17: MISO Configuration Results for 1C - T_{avg}

While there is increased accuracy overall with multiple features included in the network, the tradeoff is model complexity and run-time. Depending on the application of the network, either configuration could be appropriate as both show promising results in short run-time.

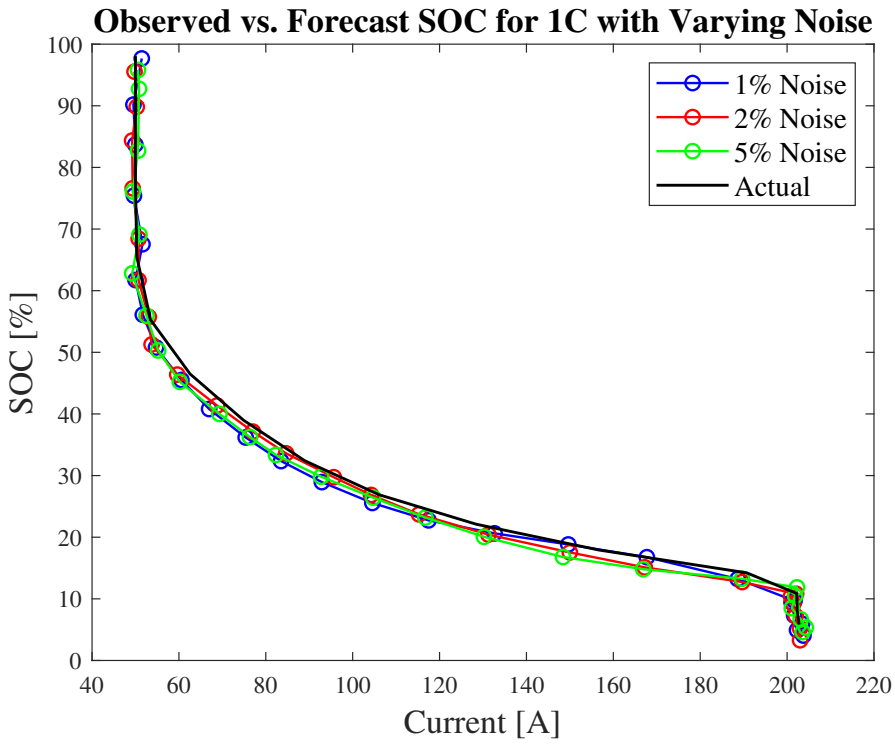


Figure 3.18: SOC Observed vs Forecast with Varying Noise Levels

3.3.5 Estimation Using Noisy Data

Figure 3.18 shows the observed versus forecasted values for SOC for varying levels of noise at 1C. Additional figures showing the effect of differing amounts of training data for SOC and T_{avg} are in Appendix B.3. Similarly to figure 3.6, as the number of data points increases, the forecast values lie closer to the actual. As can be noted from Figure 3.18, noise seems to have very minute effects on the efficiency of the network at these levels, with all trends holding and predicted curves lying relatively close to the observed curve.

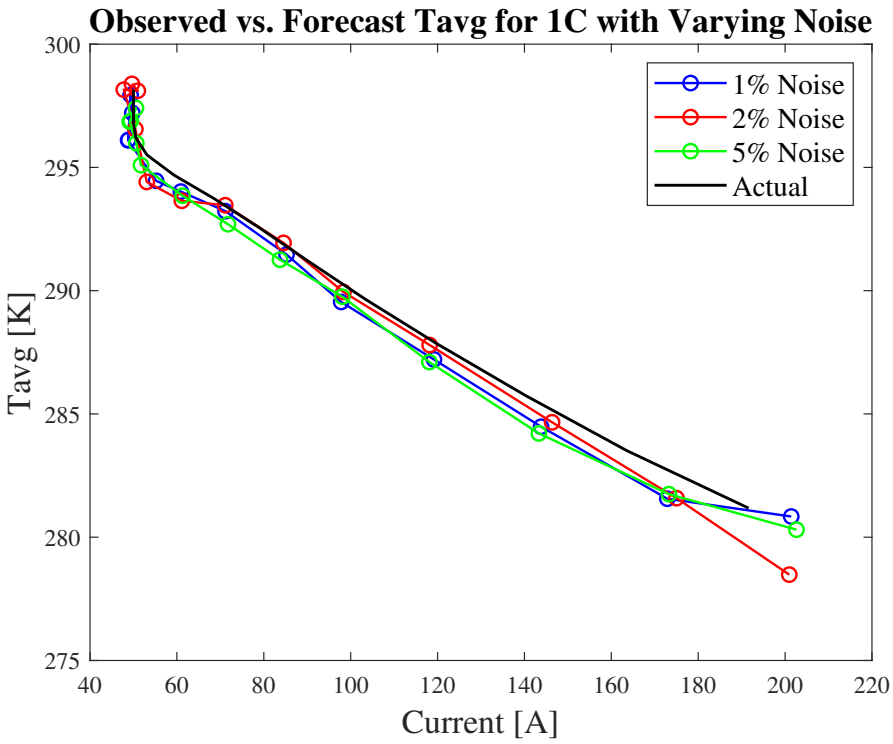


Figure 3.19: T_{avg} Observed vs Forecast with Varying Noise Levels

Moreover, the observed versus forecast T_{avg} data is shown for deviating noise quantities in Figure 3.19. The effect of noise is much more obvious in Figure 3.19 than in 3.18. The simulations with 1 and 2 % noise show some correctional behaviours at lower amounts of training data, where much larger deviations are noted for the 5 % noise setting. Nonetheless, when 110 training points or more are used, the inclusion of noise appears to have very little influence on the accuracy of the algorithm.

Another foremost assessment of machine learning networks is the speed of convergence, which can be considered by the epoch at which the validation criteria are met [38, 47]. In the simulations of the SOC at varying C-rates, this occurs at approximately epoch 150 or prior, with decreased amounts of training data resulting in quicker convergence (as seen in Figure 3.8). A similar result is observed in examining the results of the T_{avg} simulations.

Overall, the estimation of SOC using the single-feature configuration produces satisfactory results with RMSE values for all C-rates using 60 training points or more

ranging between 0.669% - 1.826%, and MAE between 1.581% - 0.405%. Additionally, all correlation values are above 0.99. The results of the average internal cell temperature are also sufficient. Values of RMSE for all C-rates and all training points are between 0.059K-1.174K, MSE between 0.004K-1.377K, and MAE ranging from 0.041%-1.038%. The PCC values also indicate a strong correlation with values above 0.99 for all cases. The issue of increased error at higher SOC values is also present in literature, as shown in [38],

3.4 Conclusion

An LSTM algorithm was constructed to complete the state estimation of SOC and average cell temperature for a lithium-ion battery, for 4 different current loads, using data collected from the SPMeT model developed in a previous work. This approach extends the generalizability of the previously developed model, and allows for some reflection of the internal mechanics to remain. The performance of the model was evaluated based on RMSE, MSE, MAE and the PCC. The expected trends for varying C-rates and varying amounts of training data are observed for both SOC and T_{avg} state estimation. The values of RMSE and MAE for SOC are typically below 1.5% for all C-rates. Even stronger performance is shown by the network when estimating average internal cell temperature with RMSE and MAE values below 1%. These indicators demonstrate that the network produces very accurate SOC and temperature estimations in suitable run-time for real-time applications. Also investigated was the effect of different data partitioning versus the traditional approach; this demonstrated negligible differences, indicating the validity of the proposed method. The influence of using multiple features in the network versus a single feature was examined as well. The MISO format produced overall more accurate predictions, however the network required to yield such results was much more complex, and had a longer run-time. Some possible extensions to this work would be to apply the network to estimate the concentration of Li-ions as a spatial-temporal state by using the structure of the

SPMeT model. Moreover, the network could be tested using other available datasets and charging schemes.

Chapter 4

Conclusions and Potential Extensions for Future Work

4.1 Summary of Research

The contribution to knowledge demonstrated by this thesis includes the development, implementation, and validation of an enhanced SPMeT LIB model. The developed model utilizes a spatially varying current density as suggested by Richardson et al. In addition, the model combines a traditional SPMe type model, with a dynamic thermal model to allow the prediction of not easily measurable qualities of batteries during discharging/charging processes. Moreover, a PID controller was integrated to implement the traditional CC-CV charging/discharging protocols. The comparison of various empirical thermodynamic potential functions was completed, and then LSE was utilized to fit the function exhibiting the desired characteristics to the constructed model. The model was validated via experimental data from literature. This model may be used as a platform for battery development and research.

To extend the modelling work, the data collected via the model simulation was used for state estimation of SOC and average internal cell temperature by employing an LSTM network. The architecture and construction of the network are presented as well. Additionally, the comparison of a single-feature versus multi-feature network configuration was fulfilled, as well as an investigation of the effect of noise on the state estimation problem. The machine learning results show promising results, with root

mean square errors below 2% and 1.2K for SOC and average internal temperature respectively, with quick training and testing times. A key contribution is the analysis of the consequence of varying current loads on the machine learning network effectiveness. Moreover, the estimation of internal average cell temperature has not been as comprehensively studied as that of other battery health indicators. Contributions to battery simulation datasets are also of great importance, as they supplement the variability of available data and enable more accurate and adaptable predictions of assorted states. Thus, this research addresses multiple gaps in the current literature regarding this topic.

4.2 Potential Extensions for Future Work

With regards to Chapter 2, some potential extensions of this research could include an investigation into the class of Tanks-in-Series modelling, and integration of a current dependent radii and concentration dependent diffusivity for the "electrode" particles. Moreover, the realization of a more automated controller that could apply varying control schemes would be of interest. Additionally, the existing SPMeT model could be altered via the inclusion of terms to the thermal model which take into account reversible heat contributions; furthermore, the consideration of degradation mechanisms which are of particular interest at higher C-rates.

For the study in Chapter 3, some recommendations for additions would be the application of the network to estimate the concentration of Li-ions as a spatial-temporal state by using the structure of the SPMeT model. Additionally, the network could be tested using other available datasets and charging/discharging schemes. Investigation of hybrid methods, such as an LSTM-GRU algorithm or LSTM combined with a Kalman filter, could yield attractive conclusions.

Bibliography

[1] M. A. Dmitrienko, G. S. Nyashina, and P. A. Strizhak, "Major gas emissions from combustion of slurry fuels based on coal, coal waste, and coal derivatives," *Journal of Cleaner Production*, vol. 177, pp. 284–301, Mar. 10, 2018, ISSN: 0959-6526. DOI: 10.1016/j.jclepro.2017.12.254. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0959652617332481.

[2] W. Chen, J. Liang, Z. Yang, and G. Li, "A review of lithium-ion battery for electric vehicle applications and beyond," *Energy Procedia*, Innovative Solutions for Energy Transitions, vol. 158, pp. 4363–4368, Feb. 1, 2019. DOI: 10.1016/j.egypro.2019.01.783. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1876610219308215.

[3] F. B. Planella *et al.*, "A continuum of physics-based lithium-ion battery models reviewed," *Progress in Energy*, vol. 4, p. 42 003, 2022.

[4] E. Prada, D. Di Domenico, Y. Creff, J. Bernard, V. Sauvant-Moynot, and F. Huet, "Simplified electrochemical and thermal model of lifepo $_4$ -graphite li-ion batteries for fast charge applications," *J. Electrochem. Soc.*, vol. 159, no. 9, A1508, 2012, ISSN: 0013-4651, 1945-7111.

[5] H. E. Perez, S. Dey, X. Hu, and S. J. Moura, "Optimal charging of li-ion batteries via a single particle model with electrolyte and thermal dynamics," *J. Electrochem. Soc.*, vol. 164, no. 7, A1679, 2017.

[6] C. D. Millholland. "Electrolyte materials in lithium-ion batteries," Advancing Materials. (Nov. 27, 2018), [Online]. Available: https://www.thermofisher.com/blog/materials/electrolyte-materials-in-lithium-ion-batteries/.

[7] S. Tamilselvi *et al.*, "A review on battery modelling techniques," *Sustainability*, vol. 13, no. 18, p. 10 042, Jan. 2021, Number: 18 Publisher: Multidisciplinary Digital Publishing Institute. DOI: 10.3390/su131810042. [Online]. Available: https://www.mdpi.com/2071-1050/13/18/10042.

[8] G. Xie, X. Peng, X. Li, X. Hei, and S. Hu, "Remaining useful life prediction of lithium-ion battery based on an improved particle filter algorithm," *The Canadian Journal of Chemical Engineering*, vol. 98, no. 6, pp. 1365–1376, 2020. DOI: 10.1002/cjce.23675. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/cjce.23675.

[9] M. Guo, M. Ren, W. Zhang, Z. Yang, and J. Chen, "Two-layer dynamic economic nonlinear model predictive control for a lithium-ion battery charge process with random disturbances," *The Canadian Journal of Chemical Engineering*, vol. n/a, 2024. DOI: 10.1002/cjce.25263. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/cjce.25263.

[10] A. Pozzi, G. Ciaramella, S. Volkwein, and D. M Raimondo, "Optimal design of experiments for a lithium-ion cell: Parameters identification of an isothermal single particle model with electrolyte dynamics," *Ind. Eng. Chem. Res.*, vol. 58, no. 3, p. 1286, 2018.

[11] Y. Chen *et al.*, "A review of lithium-ion battery safety concerns: The issues, strategies, and testing standards," *J. of Energy Chemistry*, vol. 59, p. 83, Aug. 1, 2021. DOI: 10.1016/j.jechem.2020.10.017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2095495620307075.

[12] G. Richardson, I. Korotkin, R. Ranom, M. Castle, and J. Foster, "Generalised single particle models for high-rate operation of graded lithium-ion electrodes: Systematic derivation and validation," *Electrochimica Acta*, vol. 339, p. 135 862, 2020, ISSN: 0013-4686.

[13] G. Hwang, N. Sitapure, J. Moon, H. Lee, S. Hwang, and J. Sang-Il Kwon, "Model predictive control of Lithium-ion batteries: Development of optimal charging profile for reduced intracycle capacity fade using an enhanced single particle model (SPM) with first-principled chemical/mechanical degradation mechanisms," *J. Chem. Eng.*, vol. 435, p. 134 768, May 2022, ISSN: 1385-8947.

[14] C. Delacourt and M. Safari, "Analysis of lithium deinsertion/insertion in LiyFePO4 with a simple mathematical model," *Electrochimica Acta*, vol. 56, no. 14, p. 5222, May 2011, ISSN: 0013-4686.

[15] A. Subramaniam, S. Kolluri, S. Santhanagopalan, and V. R. Subramanian, "An efficient electrochemical-thermal tanks-in-series model for lithium-ion batteries," *Journal of The Electrochemical Society*, vol. 167, no. 11, p. 113 506, Jul. 2020, Publisher: IOP Publishing. DOI: 10.1149/1945-7111/aba700. [Online]. Available: https://dx.doi.org/10.1149/1945-7111/aba700.

[16] S. Kolluri, S. V. Aduru, M. Pathak, R. D. Braatz, and V. R. Subramanian, "Real-time nonlinear model predictive control (NMPC) strategies using physics-based models for advanced lithium-ion battery management system (BMS)," *Journal of The Electrochemical Society*, vol. 167, no. 6, p. 63 505, Jan. 4, 2020. DOI: 10.1149/1945-7111/ab7bd7. [Online]. Available: https://iopscience.iop.org/article/10.1149/1945-7111/ab7bd7.

[17] S. G. Marquis, V. Sulzer, R. Timms, C. P. Please, and S. J. Chapman, "An Asymptotic Derivation of a Single Particle Model with Electrolyte," *J. Electrochem. Soc.*, vol. 166, no. 15, A3693, Nov. 2019, ISSN: 1945-7111.

[18] T. F. Fuller, M. Doyle, and J. Newman, "Simulation and Optimization of the Dual Lithium Ion Insertion Cell," *J. Electrochem. Soc.*, vol. 141, no. 1, p. 1, Jan. 1994, ISSN: 0013-4651, 1945-7111.

[19] J. Kim, A. Mallarapu, S. Santhanagopalan, and J. Newman, "A robust numerical treatment of solid-phase diffusion in pseudo two-dimensional lithium-ion battery models," *Journal of Power Sources*, vol. 556, p. 232 413, Feb. 1, 2023, ISSN: 0378-7753. DOI: 10.1016/j.jpowsour.2022.232413. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378775322013908.

[20] S. Atlung, K. West, and T. Jacobsen, "Dynamic Aspects of Solid Solution Cathodes for Electrochemical Power Sources," *J. Electrochem. Soc.*, vol. 126, no. 8, p. 1311, Aug. 1979, ISSN: 0013-4651, 1945-7111.

[21] C. Forgez, D. Vinh Do, G. Friedrich, M. Morcrette, and C. Delacourt, "Thermal modeling of a cylindrical LiFePO4/graphite lithium-ion battery," *J. of Power Sources*, vol. 195, no. 9, p. 2961, May 2010, ISSN: 0378-7753.

[22] M. Castle, G. Richardson, and J. M. Foster, "Understanding rapid charge and discharge in nano-structured lithium iron phosphate cathodes," *Eur. J. of Applied Mathematics*, vol. 33, no. 2, p. 328, Apr. 2022, ISSN: 0956-7925, 1469-4425.

[23] X. Lin *et al.*, "A lumped-parameter electro-thermal model for cylindrical batteries," *J. of Power Sources*, vol. 257, p. 1, Jul. 2014, ISSN: 0378-7753.

[24] M. D. Berliner, D. A. Cogswell, M. Z. Bazant, and R. D. Braatz, "Methods—PETLION: Open-source software for millisecond-scale porous electrode theory-based lithium-ion battery simulations," *J. Electrochem. Soc.*, vol. 168, no. 9, p. 090 504, Sep. 1, 2021. DOI: 10.1149/1945-7111/ac201c. [Online]. Available: https://iopscience.iop.org/article/10.1149/1945-7111/ac201c.

[25] S. Santhanagopalan, Q. Guo, and R. White, "Parameter Estimation and Model Discrimination for a Lithium-Ion Cell," *J. Electrochem. Soc.*, vol. 154, no. 3, A198, Jan. 2007.

[26] A Yew, *Numerical differentiation: Finite differences*, 2011. [Online]. Available: https://www.dam.brown.edu/people/alcyew/handouts/numdiff.pdf.

[27] B. J. Szekeres and F. Izsák, "A finite difference method for fractional diffusion equations with Neumann boundary conditions," en, *Open Mathematics*, vol. 13, no. 1, Sep. 2015, Publisher: De Gruyter Open Access, ISSN: 2391-5455.

[28] K. J. Åström and R. M. Murray, *Feedback systems: an introduction for scientists and engineers*, 2nd ed. Princeton: Princeton University Press, 2008, ISBN: 978-1-4008-2873-9.

[29] W. Luo, C. Lyu, L. Wang, and L. Zhang, "A new extension of physics-based single particle model for higher charge–discharge rates," *J. of Power Sources*, vol. 241, p. 295, Nov. 2013, ISSN: 0378-7753.

[30] R. Masoudi, T. Uchida, and J. McPhee, "Parameter estimation of an electrochemistry-based lithium-ion battery model," *J. of Power Sources*, vol. 291, p. 215, Sep. 2015, ISSN: 0378-7753.

[31] W. Li *et al.*, "Parameter sensitivity analysis of electrochemical model-based battery management systems for lithium-ion batteries," *J. of Applied Energy*, vol. 269, p. 115 104, Jul. 2020, ISSN: 0306-2619.

[32] V. Srinivasan and J. Newman, "Discharge Model for the Lithium Iron-Phosphate Electrode," *J. Electrochem. Soc.*, vol. 151, no. 10, A1517, Sep. 2004, Publisher: IOP Publishing, ISSN: 1945-7111.

[33] D. Di Domenico, G. Fiengo, and A. Stefanopoulou, "Lithium-ion battery state of charge estimation with a Kalman Filter based on a electrochemical model," in *2008 IEEE International Conference on Control Applications*, ISSN: 1085-1992, Sep. 2008, p. 702.

[34] S. Torsten and P. Stoica, "System identification," *Hertfordshire, UK*, 1989.

[35] J. Verret, R. Qiao, and R. A. Barghout, "Process control," *Foundations of Chemical and Biological Engineering I*, Aug. 16, 2020. [Online]. Available: https://pressbooks.bccampus.ca/chbe220/chapter/process-control/.

[36] S. Sirois. "CPU speed explained: What's a good processor speed? | HP® tech takes." (Aug. 2, 2024), [Online]. Available: https://www.hp.com/us-en/shop/tech-takes/what-is-processor-speed.

[37] C. Zhang, S. Wang, C. Yu, Y. Xie, and C. Fernandez, "Novel improved particle swarm optimization-extreme learning machine algorithm for state of charge estimation of lithium-ion batteries," *Ind. Eng. Chem. Res.*, vol. 61, no. 46, p. 17 209, 2022.

[38] F. Yang, S. Zhang, W. Li, and Q. Miao, "State-of-charge estimation of lithium-ion batteries using LSTM and UKF," *Energy*, vol. 201, p. 117 664, Jun. 15, 2020. DOI: 10.1016/j.energy.2020.117664. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360544220307714.

[39] Y. Yi *et al.*, "Digital twin-long short-term memory (LSTM) neural network based real-time temperature prediction and degradation model analysis for lithium-ion battery," *Journal of Energy Storage*, vol. 64, p. 107 203, Aug. 1, 2023. DOI: 10.1016/j.est.2023.107203. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2352152X2300600X.

[40] J. Chen, Y. Zhang, J. Wu, W. Cheng, and Q. Zhu, "SOC estimation for lithium-ion battery using the LSTM-RNN with extended input and constrained output," *Energy*, vol. 262, p. 125 375, Jan. 1, 2023. DOI: 10.1016/j.energy.2022.125375. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360544222022575.

[41] 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 15, 2019. DOI: 10.1016/j.energy.2019.03.059. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360544219304633.

[42] D.-W. Chung, J.-H. Ko, and K.-Y. Yoon, "State-of-charge estimation of lithium-ion batteries using LSTM deep learning method," *J. Electr. Eng. Technol.*, vol. 17, no. 3, pp. 1931–1945, May 1, 2022. DOI: 10.1007/s42835-021-00954-8. [Online]. Available: https://doi.org/10.1007/s42835-021-00954-8.

[43] C. Zhang, K. Li, and J. Deng, "Real-time estimation of battery internal temperature based on a simplified thermoelectric model," *Journal of Power Sources*, vol. 302, pp. 146–154, Jan. 20, 2016. DOI: 10.1016/j.jpowsour.2015.10.052.

[44] F. Zhao, Y. Guo, and B. Chen, "A review of lithium-ion battery state of charge estimation methods based on machine learning," *World Electric Vehicle Journal*, vol. 15, no. 4, p. 131, Apr. 2024, Number: 4 Publisher: Multidisciplinary Digital Publishing Institute. DOI: 10.3390/wevj15040131. [Online]. Available: https://www.mdpi.com/2032-6653/15/4/131.

[45] G. Cho, D. Zhu, J. J. Campbell, and M. Wang, "An LSTM-PINN hybrid method to estimate lithium-ion battery pack temperature," *IEEE Access*, vol. 10, pp. 100 594–100 604, 2022, Conference Name: IEEE Access. DOI: 10.1109/ACCESS.2022.3208103. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9895422.

[46] Q. Yao, D. D.-C. Lu, and G. Lei, "A surface temperature estimation method for lithium-ion battery using enhanced GRU-RNN," *IEEE Transactions on Transportation Electrification*, vol. 9, no. 1, pp. 1103–1112, Mar. 2023. DOI: 10.1109/TTE.2022.3197927. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9853612.

[47] F.-F. Li, R. Krishna, and D. Xu, *Lecture 10: Recurrent neural networks*, Stanford University - School of Engineering, Apr. 29, 2021.

[48] W. Yin, K. Kann, M. Yu, and H. Schütze, *Comparative study of CNN and RNN for natural language processing*, Feb. 7, 2017. DOI: 10.48550/arXiv.1702.01923. [Online]. Available: http://arxiv.org/abs/1702.01923.

[49] S. Hochreiter, "The vanishing gradient problem during learning recurrent neural nets and problem solutions," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 6, pp. 107–116, Apr. 1, 1998. DOI: 10.1142/S0218488598000094.

[50] Y. Jiang, Y. Yu, J. Huang, W. Cai, and J. Marco, "Li-ion battery temperature estimation based on recurrent neural networks," *Sci. China Technol. Sci.*, vol. 64, no. 6, pp. 1335–1344, Jun. 1, 2021. DOI: 10.1007/s11431-020-1736-5. [Online]. Available: https://doi.org/10.1007/s11431-020-1736-5.

[51] D. Roman, S. Saxena, V. Robu, M. Pecht, and D. Flynn, "Machine learning pipeline for battery state-of-health estimation," *Nat Mach Intell*, vol. 3, no. 5, pp. 447–456, May 2021. DOI: 10.1038/s42256-021-00312-3. [Online]. Available: https://www.nature.com/articles/s42256-021-00312-3.

[52] M. Danko, J. Adamec, M. Taraba, and P. Drgona, "Overview of batteries state of charge estimation methods," *Transportation Research Procedia*, TRANSCOM 2019 13th International Scientific Conference on Sustainable, Modern and Safe Transport, vol. 40, pp. 186–192, Jan. 1, 2019. DOI: 10.1016/j.trpro.2019.07.029. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2352146519301905.

[53] M. Naguib, P. Kollmeyer, C. Vidal, and A. Emadi, "Accurate surface temperature estimation of lithium-ion batteries using feedforward and recurrent artificial neural networks," in *2021 IEEE Transportation Electrification Conference and Expo (ITEC)*, Jun. 2021, pp. 52–57. DOI: 10.1109/ITEC51675.2021.9490043. [Online]. Available: https://ieeexplore.ieee.org/document/9490043/?arnumber=9490043.

[54] R. Richardson, P. Ireland, and D. Howey, "Battery internal temperature estimation by combined impedance and surface temperature measurement," *Journal of Power Sources*, vol. 265, 2014, Publisher: Elsevier, ISSN: 0378-7753. [Online]. Available: https://ora.ox.ac.uk/objects/uuid:fe731bf8-547b-4d95-b040-22751a1981f8.

[55] T. M. Inc., *Statistics and machine learning toolbox*, Natick, Massachusetts, United States, 2022. [Online]. Available: https://www.mathworks.com/help/stats/index.html.

Appendix A: Modelling Supplementary Information

A.1 Abbreviations

Table A.1: Abbreviations and associated meanings

Abbreviation	**Meaning**
BMS	Battery management system
CC-CV	Constant current - constant voltage; form of charging scheme
DFN	Doyle Fuller Neumann battery model
ECM	Equivalent circuit model
EV	Electric vehicle
HEV	Hybrid electric vehicle
LFP	Lithium iron phosphate (Li-FePO4)
LIB	Lithium-ion battery
LSE	Least squares estimation
OCP	Open circuit potential
OCV	Open circuit voltage
ODE	Ordinary differential equation
PDAE	Partial differential algebraic equation
PDE	Partial differential equation
PID	Proportional integral derivative controller
SOC	State of charge
SPM	Single particle model
SPMeT	Extended single particle model with coupled thermal model

A.2 Nomenclature

Table A.2: Parameters and functions, values where available

Parameters and Functions					
Symbol	**Meaning**	**Units**	**Value [k=p]**	**Value [k=s]**	**Value [k=n]**
b_k	Particle surface area per unit of volume	m^{-1}	22440000	-	348000
c_k^{max}	Maximum particle concentration	$Molm^{-3}$	22806	-	30555
c_{k0}	Electrode initial concentration	$Molm^{-3}$	200046	-	19624
c_{e0}	Electrolyte initial concentration	$Molm^{-3}$	-	1200	-
c_k^s	Concentration at particle surface in cathode/anode	$Molm^{-3}$	-	-	-
c_k^{bulk}	Bulk concentration for electrode k	$Molm^{-3}$	-	-	-
C_c	Core heat capacity	$JK^{-1}kg^{-1}$		270000	
C_u	Surface heat capacity	$JK^{-1}kg^{-1}$		400	
cp	specific heat capacity	$JK^{-1}kg^{-1}$		11000	
D_k	lithium diffusivity in particle	M^2s^{-1}	5.9×10ˆ(-20)	-	3.0×10ˆ(-15)
D_e	electrolyte lithium ion diffusivity	M^2s^{-1}	-	2.0×10ˆ(-14)	-
E_a	Activation Energy	$Jmol^{-1}$		26.6	
i_{app}	applied current density	Am^{-2}		-	
j_{k0}^{avg}	Averaged exchange current density; function of time only	Am^{-2}	-	-	-
k_i	Integral controller gain	-		0.0005 - 0.023	
k^k	Reaction rate in electrode k	$\frac{Amol^3m^{-2}}{mol^{1+\alpha}}$	3×10ˆ(-11)	-	1×10ˆ(-10)
k_p	Proportional controller gain	-		0.0008 - 0.01	
L_k	Electrode and separator thicknesses	m	8×10ˆ(-5)	2.5×10ˆ(-5)	3.4×10ˆ(-5)
L^l	Electrolyte region length	m	8×10ˆ(-5)	2.5×10ˆ(-5)	3.4×10ˆ(-5)
Q_{irr}	irreversible heat source	Wm^{-3}		-	
Q_{rev}	reversible heat source	Wm^{-3}		-	
R_k	particle radius	m	5×10ˆ(-8)	-	5×10ˆ(-6)
R_f^k	Solid-electrolyte interphase film resistance	Ω^2	1×10ˆ(-3)	-	5×10ˆ(-3)
R_c	Conduction resistance	KW^{-1}		4	
R_u	Convection resistance	KW^{-1}		100	
t^+	cation transference number	-		0.4	
T_0	initial temperature	K		288.15	
T_{amb}	ambient temperature	K		288.15	
U_k	open-circuit potential (OCP)	V	-	-	-
α	Charge transfer coefficient	-		0.5	
β	transport efficiency	-	0.085542	0.224340	0.256195
δQ	Tolerance value to final capacity	-		-	
ε	electrolyte volume fraction	-	0.374	0.55	0.58
σ_e	conductivity (electrolyte)	Sm^{-1}			
σ_k	conductivity (electrode)	Sm^{-1}	10	-	100

Table A.3: Variables and associated symbols with units

Variables		
Symbol	**Meaning**	**Units**
c_k	lithium concentration in the electrode particles	$Molm^{-3}$
c_e	lithium ion concentration in the electrolyte	$Molm^{-3}$
c_e^l	Concentration of lithium ions in the electrolyte in domain l	$Molm^{-3}$
c_{e0}	initial/rest lithium ion concentration in the electrolyte	$Molm^{-3}$
N_k	molar flux of lithium in the electrode particles	$Molm^{-2}s^{-1}$
N_e	molar flux of ions in the electrolyte	$Molm^{-2}s^{-1}$
i_k	current density in the electrodes	Am^{-2}
i_e	current density in the electrolyte	Am^{-2}
j_k	reaction current density	Am^{-2}
j_{k0}	exchange current density	Am^{-2}
ϕ_k	electrode potential	V
ϕ_e	electrolyte potential	V
T_{avg}	Average cell temperature	K
T_c	Cell core temperature	K
T_s	Cell surface temperature	K
SOC	State of charge (bulk)	-
η_k	overpotential at the electrode-electrolyte interface	V

Table A.4: Universal constant symbols with units and values

Constants			
Symbol	**Meaning**	**Units**	**Value**
F	Faraday constant	$Cmol^{-1}$	96485.33
R	Gas constant	$JK^{-1}mol^{-1}$	8.314472

Table A.5: Subscript and superscript meaning and symbols

Subscripts	
Symbol	**Meaning**
e	in electrolyte
n	in negative electrode/particle (anode)
sep	in separator
p	in positive electrode/particle (cathode)
k	in domain $k \in \{n, p\}$
l	In domain $l \in \{n, sep, p\}$
Superscripts	
$bulk$	in bulk
max	maximum

A.3 Electrolyte Domain Definition

Table A.6: Electrolyte domain definition with description

Symbol	Description	Domain Definition
n	Negative electrode domain; x is between 0 (beginning of the cell) and the end of the anode thickness (anode-separator interface).	$0 \leq x < L_n$
s	Separator domain; x is between the end of the anode thickness (anode-separator interface) and the end of the separator thickness (separator-cathode interface).	$L_n \leq x \leq L - L_p$
p	Positive electrode domain; x is between the end of the separator thickness (anode-separator interface) and the L (the end of the cathode thickness/end of the cell).	$L - L_p < x \leq L$

Appendix B: State Estimation Supplementary Information

B.1 Abbreviations and Nomenclature

Table B.1: Abbreviations

Abbreviations			
Symbol	**Meaning**	**Symbol**	**Meaning**
BMS	Battery management system	ML	Machine learning
CNN	Convolutional neural network	MSE	Mean square error
DT	Digital twin	NN	Neural network
ECM	Equivalent circuit model	OCV	Open-circuit voltage
GRU	Gated recurrent; form of RNN	PCC	Pearson correlation coefficient
LFP	Lithium-iron-phosphate	PINN	Physics-informed neural network
LIB	Lithium-ion battery	RMSE	Root mean square error
LSTM	Long-short-term-memory; form of multi-layer RNN	RNN	Recurrent neural network
MAE	Mean absolute error	SISO	Single-input single-output
MISO	Multi-input single-output	SOC	State of charge; measure of difference between a fully charged battery versus a battery in use.

Table B.2: Nomenclature

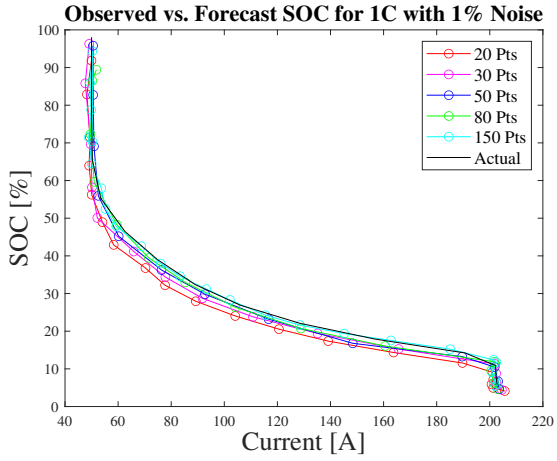
Nomenclature	
Symbol	**Meaning**
b_k	$k \in i, f, o, g$; set of biases, different for each associated gate
c_t	Cell state at time step t
$\tilde{c_t}$	Candidate cell state values; dictated by the gate gate
f_w	Function with parameters w, where w is a set of weights
h_t	Hidden state at time step t
i_t	Input gate
o_t	Output gate
σ	Standard deviation for normalization of data; sigmoid activation function in neural networks
w_k	$k \in i, f, o, g$; set of weight matrices, different for each associated gate
X_i	General variable at step i
X_{std}	Standardized/normalized general variable X
x_t	Input variable at time step t
Y	Testing value from input dataset
Y_{Pred}	Predicted value from the network

B.2 Hyperparameter Definitions

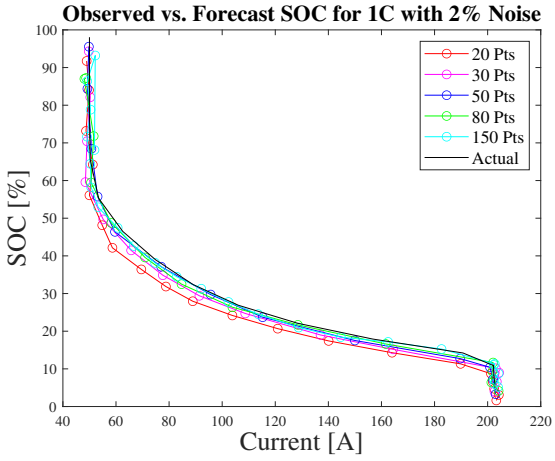
Table B.3: Hyperparameter Definitions

Hyperparameter Definitions [55]	
Name	**Definition**
Max Epochs	Maximum number of epochs (full passes of the data) to use for training, specified as a positive integer.
Initial Learning Rate	Initial learning rate used for training, specified as a positive scalar. If the learning rate is too low, then training can take a long time. If the learning rate is too high, then training might reach a suboptimal result or diverge.
Learning Rate Drop Period	Number of epochs for dropping the learning rate, specified as a positive integer. This option is valid only when the LearnRateSchedule training option is "piecewise".
Learning Rate Drop Factor	Factor for dropping the learning rate, specified as a scalar from 0 to 1. This option is valid only when the LearnRateSchedule training option is "piecewise".
Minibatches	Size of the mini-batch to use for each training iteration, specified as a positive integer. A mini-batch is a subset of the training set that is used to evaluate the gradient of the loss function and update the weights.
Gradient Threshold	Gradient threshold, specified as Inf or a positive scalar. If the gradient exceeds the value of GradientThreshold, then the gradient is clipped according to the GradientThresholdMethod training option.
Validation Frequency	Frequency of neural network validation in number of iterations, specified as a positive integer. The validation frequency value is the number of iterations between evaluations of validation metrics
Validation Patience	Patience of validation stopping of neural network training, specified as a positive integer or Inf. Specifies the number of times the objective metric on the validation set can be worse or equal to the previous best value before training stops.
ADAM	Adaptive moment estimation (Adam). Adam is a stochastic solver.

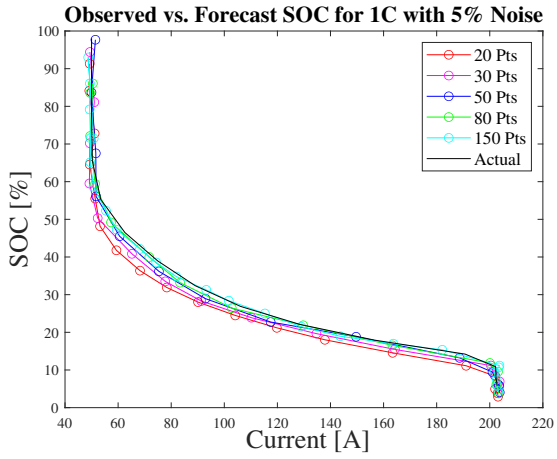
B.3 Additional Noise Figures



18a)

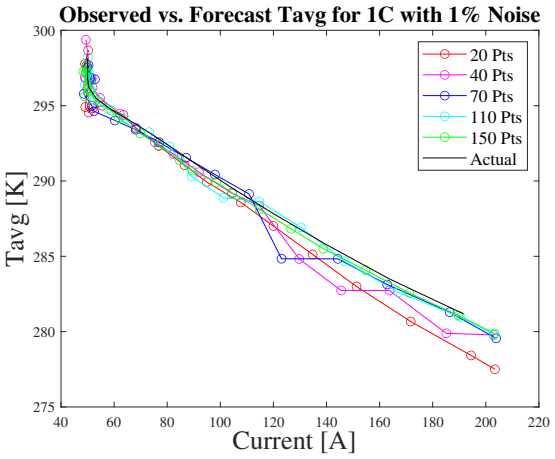


18b)

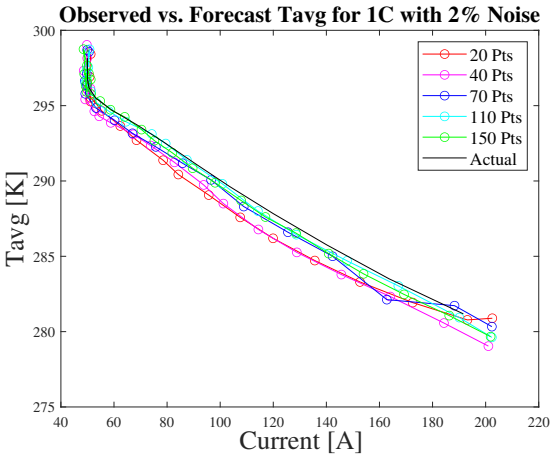


18c)

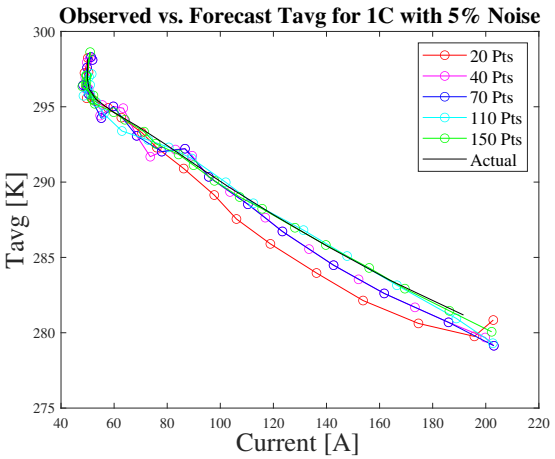
Figure B.1: SOC Estimation Using Noisy Data For Varying Training Data Points



19a)



19b)



19c)

Figure B.2: Tavg Estimation Using Noisy Data For Varying Training Data Points