State of Charge and Range Estimation of Lithium-ion Batteries in Electric Vehicles

State of Charge and Range Estimation of Lithium-ion Batteries in Electric Vehicles

By Fauzia Khanum,

A Thesis Submitted to the School of Graduate Studies in the Partial Fulfillment of the Requirements for the Degree Master of Applied Science

McMaster University

Master of Applied Science (2021)

Hamilton, Ontario (Department of Electrical and Computer Engineering)

TITLE: State of Charge and Range Estimation of Lithium-ion Batteries in Electric Vehicles

AUTHOR: Fauzia Khanum B.Eng & Mgmt, Co-op (McMaster University)

SUPERVISOR: Dr. Ali Emadi

NUMBER OF PAGES: xix, 118

Abstract

Switching from fossil-fuel-powered vehicles to electric vehicles has become an international focus in the pursuit of combatting climate change. Regardless, the adoption of electric vehicles has been slow, in part, due to range anxiety. One solution to mitigating range anxiety is to provide a more accurate state of charge (SOC) and range estimation. SOC estimation of lithium-ion batteries for electric vehicle application is a well-researched topic, yet minimal tools and code exist online for researchers and students alike. To that end, a publicly available Kalman filter-based SOC estimation function is presented. The MATLAB function utilizes a second-order resistor-capacitor equivalent circuit model. It requires the SOC-OCV (open circuit voltage) curve, internal resistance, and equivalent circuit model battery parameters. Users can use an extended Kalman filter (EKF) or adaptive extended Kalman filter (AEKF) algorithm and temperature-dependent battery data. A practical example is illustrated using the LA92 driving cycle of a Turnigy battery at multiple temperatures ranging from -10°C to 40°C.

Current range estimation methods suffer from inaccuracy as factors including temperature, wind, driver behaviour, battery voltage, current, SOC, route/terrain, and much more make it difficult to model accurately. One of the most critical factors in range estimation is the battery. However, most models thus far are represented using equivalent circuit models as they are more widely researched. Another limitation is that any machine learning-based range estimation is typically based on historical driving data that require odometer readings for training.

A range estimation algorithm using a machine learning-based voltage estimation model is presented. Specifically, the long short-term memory cell in a recurrent neural network is used for the battery model. The model is trained with two datasets, classic and whole, from the experimental data of four Tesla/Panasonic 2170 battery cells. All network training is completed on SHARCNET, a resource provided by Canada

Compute to researchers. The classically trained network achieved an average root mean squared error (RMSE) of 44 mV compared to 34 mV achieved by the network trained on the whole dataset. Based on the whole dataset, all test cases achieve an end range estimation of less than 5 km with an average of 0.29 km.

Acknowledgements

First and foremost, I would like to thank my academic supervisor Dr. Ali Emadi for his patience and kindness through the many hurdles I faced in this journey. I would also like to thank Dr. Emadi for this fantastic opportunity to learn at one of the leading automotive research facilities.

I want to thank Dr. Phillip Kollmeyer for his continued mentorship, guidance, and Teams meetings, without whom this work would not have come to fruition. Dr. Kollmeyer's breadth and depth of knowledge on electric vehicles and batteries astounds me to this day.

The Battery Laboratory at the McMaster Automotive Resource Center, where all the testing was performed, is all in thanks to Dr. Saeid Habibi and Cam Fisher. This research was supported, in part, by the Natural Sciences and Engineering Research Council of Canada (NSERC) and Canada Research Chair in Transportation Electrification and Smart Mobility.

Finally, I would like to thank Dr. Ryan Ahmed, my colleagues on the EMS team, and my friends for all the support and guidance they provided. They were always available to lend a hand and encouraged me to continue in my work.

Contents

Abstract **iii**

Acknowledgements **v**

Abbreviations **xvi**

1 Introduction **1**

1.1 Background and Motivation 1

1.2 Thesis Contributions 5

1.3 Thesis Organization 7

2 State-of-the-Art Review on Lithium-Ion Batteries in Electric Vehicles **8**

2.1 Introduction 8

2.2 Lithium-ion Battery Basics and Models 9

2.2.1 Battery Basics 10

2.2.2 Battery Capacity 10

2.2.3 Battery Power and Energy 11

2.2.4 Battery Life 11

2.3 Modelling Batteries 12

2.3.1 Equivalent Circuit Models 12

2.3.2 Electrochemical Models 14

2.3.3 Data Driven Models 15

2.4 State-of-Charge Estimation Methods 18
2.4.1 Coulomb Counting 18
2.4.2 SOC-OCV 19
2.4.3 Model-Based Methods 20
2.4.4 Data Driven Methods 21
2.5 Range Estimation 22
2.5.1 Range Estimation Influencing Factors 22
2.5.1.1 Vehicle Design and Simulation 23
2.5.1.2 Driver Behaviour 24
2.5.1.3 Environmental Factors 25
2.5.2 Range Estimation Methods 26
2.5.3 Current Problems with Range Estimation 30
2.6 Summary 31

3 A Kalman Filter Based Battery State of Charge Estimation 32
3.1 Introduction 32
3.2 State of Charge Estimation Algorithm 33
3.2.1 Battery Modelling 34
3.2.2 Extended Kalman Filter 36
3.3 Extended Kalman Filter State-of-Charge Estimation Function 38
3.3.1 Battery Parameters 38
3.4 Simulation Example 43
3.5 Summary 47

4 Battery Cell Tests: Vehicle Model and Design of Experiment 48
4.1 Introduction 48
4.2 Electric Vehicle Model 50
4.2.1 Model of Tesla Model 3 Drive Train Power 50

4.2.2 Energy Storage System 55
4.3 Experimental Measurement and Preparation of Data 57
4.3.1 Cell Characterization 60
4.3.2 Drive Cycles 63
4.3.3 Cut-off Capacity Calculation 66
4.3.4 Data Preparation 69
4.4 Summary 71

5 Range Estimation of Electric Vehicles using a Recurrent Neural Network-based Voltage Estimation 72

5.1 Introduction 72
5.2 Recurrent Neural Network Voltage Estimation Model 73
5.3 Driving Range Estimation 77
5.4 LSMT-RNN Training on SHARCNET vs. Personal Computer 78
5.5 Voltage Estimation Results 80
5.5.1 Learnable Parameters and Hyper-Parameters 80
5.5.2 Whole vs. Classic Dataset Comparison 81
5.6 Range Estimation based on LSTM-RNN Battery model 84
5.6.1 Model Performance: Best vs. Worst Result for Standard Four Drive Cycles 84
5.6.2 Model Performance Over Temperature 85
5.6.3 Model Performance in Custom Cycles 87
5.6.4 Voltage Estimation Comparison to Literature 90
5.7 Range Estimation Results 93
5.7.1 Range Estimation Comparison to Literature 96

6 Conclusions and Future Work 98

6.1 Concluding Remarks 98

6.2 Future Work 99
6.3 Publications 101

Appendix **102**

References **105**

List of Figures

1.1 New motor vehicles registration by vehicle fuel type from 2011 to 2020 in Canada. 2

1.2 Overview of the proposed range estimation algorithm usage in a BEV. . . 4

1.3 Nissan Leaf infotainment screen, driving range estimation shown in the white region. 5

2.1 First order resistor capacitor equivalent circuit model of a battery cell. . . 13

2.2 Reduced-Order electrochemical model from [19]. 15

2.3 (a) Perceptron. (b) FNN Architecture. 17

2.4 SOC-OCV relationship of a 4.5Ah Panasonic/Telsa battery cell. 19

2.5 Forces acting on a vehicle. 23

2.6 Summary of range estimation types found in literature. 30

3.1 State of charge estimation process using a second order equivalent circuit model and extended Kalman filter. 33

3.2 Second order RC equivalent circuit battery model diagram. 34

3.3 OCV vs. SOC of a Turnigy Graphene 5000 mAh Li-ion battery at different temperatures. 39

3.4 Discharge resistance vs SOC of a Turnigy graphene 5000 mAh Li-ion battery at different temperatures. 40

3.5 RMS Error of SOC at different temperatures for LA92 drive cycle. 44

3.6 RMS Error of V_t at different temperatures for LA92 drive cycle. 44

3.7 (a). Measured vs. Estimated Vt at 40°Cfor LA92 drive cycle. (b). SOC Coloumb Counting vs. SOC EKF Estimation at 40°Cfor LA92 drive cycle. 45
3.8 SOC RMS Error and V_t RMS Error with no error, initial 10% SOC offset, and +/- 0.1A current offset at 40°Cfor the LA92 drive cycle. 46
3.9 (a). Measured vs. Estimated Vt with 10% initial SOC offset at 40°Cfor LA92 drive cycle. (b). SOC Coloumb Counting vs. SOC EKF Estimation with 10% initial SOC offset at 40°Cfor LA92 drive cycle. 46
3.10 (a). Measured vs. Estimated Vt with +/-0.1A current offset at 40°Cfor LA92 drive cycle. (b). SOC Coloumb Counting vs. SOC EKF Estimation with +/-0.1A current offset at 40°Cfor LA92 drive cycle. 47

4.1 Overall procedure taken to range estimation in this thesis. 49
4.2 Electrical accessory power (W) versus temperature (°C) based on equation 4.15 and a heater power of 80 W/°C. 54
4.3 (a) Tesla/Panasonic 2170 4.5 Ah cell in Arbin holder with T-Type temperature sensor. (b) Four Tesla/Panasonic cells in Arbin holders placed inside Envirotronics thermal chamber. 58
4.4 Test bench schematic of testing Tesla/Panasonic batteries. 59
4.5 Arbin cell cycles and Environtronics thermal chamber in battery lab at the McMaster Automotive Resource Center. 59
4.6 Open circuit voltage (OCV) vs. discharge capacity for various temperatures and cell 1 (m80). 61
4.7 Discharge resistance vs. SOC at various temperatures and 1/2C for cell 1 (m80). 62
4.8 Capacity vs. temperature at various C-rates for cell 1 (m80). 62
4.9 Speed and grade profiles of select drive cycles. 64
4.10 Power profile of LA92 drive cycle from 0 to 150 seconds at 40°C. 65
4.11 Power profiles of select drive cycles for 40°C and m80. 67

4.12 Discharge power capability and discharge power cut-off point for m80 cell and temperatures 40°C to 0°C. 69
4.13 Cut-off SOC of for all cells and temperature. The SOC is calculated using the C/20, 40°C, discharge ageing test at each temperature set. 69

5.1 Architecture of RNN (left) and architecture of RNN unfolded in time (right). 73
5.2 RNN network layers with LSTM cell. The network layers consist an input layer taking sequence data, hidden layer (LSTM), an output layer consisting of a fully connected layer, a clipped ReLU function, and finally the output, voltage. 74
5.3 Range estimation method at end of drive cycle and cut-off capacity. . . . 78
5.4 Whole dataset voltage estimation training timeline of SHARCNET Graham Cluster (GPU) vs. PC. 10 trainings were completed on SHARCNET between Oct. 29th and Nov. 5th. The equivalent of this training on a regular PC would take until Dec. 4th. 80
5.5 Lowest RMSE of 4 trainings per learnable parameter. Each RMSE is based on the average of UDDS, HWFET, LA92, and US06 drive cycles for m80 kg. 82
5.6 Average RMSE per training for each dataset. The orange bar is the best trained network for each dataset. 83
5.7 Voltage estimation, comparison of whole dataset and classic dataset trained networks. The bar graphs represent the average RMSE of test drive cycles UDDS, HWFET, LA92, US06, HWCUST2, and HWGRADE2 which is representative of testing on the whole dataset. 84
5.8 LSTM voltage prediction vs. observed (top) and error between predicted and observed (bottom). Close-up of the first 5 minutes of the LA92 and US06 drive cycles. 86

5.9 Range estimation vs. actual (top) and close-up at cut-off capacity of 4.514 Ah for 40°C and 3.627 Ah for 0°C. 86
5.10 LSTM voltage prediction vs. observed (top), error between predicted and observed (middle), and battery surface temperature (bottom) for m80, US06 drive cycle, and all test temperatures. 87
5.11 LSTM voltage prediction vs. observed (top), error between predicted and observed (middle), and battery surface temperature (botton) at 0°C for the full cycles (left). LSTM voltage prediction vs. observed (top), and error between predicted and observed (botton) at 0°C for the first 12 minutes (right). 88
5.12 Range estimation vs. actual (top) and close-up at cut-off capacity of 3.659 Ah for m80 and 3.627 for m1000 (bottom). 89
5.13 LSTM voltage prediction vs. observed (top), error between predicted and observed (middle), and battery surface temperature (botton) at 0°C for the full cycles (left). LSTM voltage prediction vs. observed (top), and error between predicted and observed (botton) at 0°C for the first 12 minutes (right). 89
5.14 Range estimation vs. actual (top) and close-up at cut-off capacity of 3.659 Ah for m80 and 3.627 for m1000 (bottom). 90
5.15 Voltage estimation RMSE for m80, all temperatures, and all drive cycles. 91
5.16 Absolute error between estimated range and actual range. Estimated range is based on network trained on whole dataset for all drive cycles, all temperatures, and m80 kg. 93
5.17 Actual range vs. simple C/2 range estimation vs. LSTM-based range estimation and efficiency for m448-N battery and select drive cycles. SOC_{min} indicates the cut-off capacity in SOC at the respective temperature. . . . 95

5.18 Range estimation vs. actual (top) and close-up at cut-off capacity of 4.259Ah at 25°C, for the HWGRADE2 drive cycle and m448. 96

6.1 Tesla Model X infotainment screen: directions and range estimation from Hamilton, ON to the CN Tower in Toronto, ON based on the current SOC. 101

List of Tables

2.1 Basic technical requirements of a li-ion cell in a BEV from [11]. 10

2.2 Comparison of different types of battery modelling. 18

2.3 Comparison of range estimation methods, data source, inputs, and notable point found in literature. 29

4.1 Target Coefficients. 52

4.2 Sample speed and power profile scaled down to one cell from the pack. . . 55

4.3 Tesla/Panasonic 2170 4.5 Ah li-ion battery cell measured parameters. . . 58

4.4 Envirotronics thermal chamber specifications utilized for battery testing. . 58

4.5 Arbin cell cycler specifications utilized for battery testing. 60

4.6 Cell parameters and limits utilized in power profile generation. 63

4.7 Battery/Cell testing parameters and referenced names for differentiation. 65

4.8 Cell parameters and limits utilized in cut-off capacity calculations. 66

4.9 Size and division of drive cycles of whole dataset. Cells 1 through 4 refer to m80, m488, m448-N, and m1000, respectively. 70

4.10 Size and division of drive cycles of classical dataset. Cell 1 refers to m80. A classical dataset contains various temperatures but not mass, HVAC, aggressive drive cycles, or grade parameters. 71

5.1 LSTM Parameters. 81

5.2 Comparison of voltage estimation method results in literature. 92

5.3 Comparison of Range Estimation Results from Literature 97

Abbreviations

AEKF	Adapative-Extended Kalman Filter
Ah	Ampere-hour
ANN	Artificial Neural Network
API	Application Programming Interface
BEV	Battery Electric Vehicle
BiLSTM	Bidirectional Long Short-Term Memory
BN	Bayesian Networks
BOL	Beginning of Life
CAN	Control Area Network
CPU	Central Processsing Unit
ECM	Equivalent Circuit Model
EKF	Extended Kalman Filter
EOL	End of Life
EPA	Environmental Protection Agency
FNN	Feedforward Neural Network
GA	Genetic Algorithm
GHG	Greenhouse Gas
GPS	Global Positioning System
GPU	Graphics Processing Unit
GRU	Gated Recurrent Unit

GUI	Graphical User Interface
GVWR	Gross Vehicle Weight Rating
HEV	Hybrid Electric Vehicle
HM	Hamming Networks
HPPC	Hybrid Pulse Power Characterization
HVAC	Heating, Ventilation, and Air Conditioning
HWCUST	Custom higway driving schedule
HWFET	Highway Fuel Economy Test
HWGRADE	Custom highway with varying grade driving schedule
ICEV	Internal Combustion Engine Vehicle
KF	Kalman Filter
LA92	California Light-Duty Unified Cycle
LightGBM	Light Gradient Boosting Regression Tree
LSTM	Long Short-Term Memory
MAE	Mean Absolute Error
MLP	Multilayer Perceptrons
NCA	Nickel Cobalt Aluminum Oxide
NiMH	Nickel Metal Hydride
NN	Neural Network
OCV	Open Circuit Voltage (V)
P2D	Pseudo-two-dimensional model
PF	Particle Filter
PI	Proportional Integral

PHEV	Plug-in Hybrid Electric Vehicle
PSO	Particle Swarm Optimization
PTC	Positive Temperature Coefficient
RBF	Radial Basis Function
RC	Resistor-Capacitor
ReLU	Rectified Linear Unit
RLS	Recursive least square
RMSE	Root Mean Squared Error
RNN	Recurrent Neural Network
SA	Simulated Annealing
SHARCNET	Shared Hierarchical Academic Research Computing Network
SVSF	Smooth Variable Structure Filter
SP	Single Particle Model
STDDEV	Standard Deviation
SVM	Support Vector Machines
SOC	State-of-Charge (%)
SOE	State-of-Energy
SOH	State-of-Health
SOP	State-of-Power
UDDS	Urban Dynamometer Driving Schedule
US06	Supplemental Federal Test Procedure
Wh	Watt-hour
WLTC	Worldwide Harmonized Light Vehicle Test Cycles
XGBoost	Extreme Gradient Boosting Regression Tree

This work is dedicated to my parents, Adalat Khan and Nahid Bibi, and my siblings.

Chapter 1

Introduction

1.1 Background and Motivation

Climate change can no longer be denied. The increase in adverse effects, both intensity and number, can be directly linked to the increase in greenhouse gas (GHG) emissions, of which fossil fuels are the main source. The transportation industry is the second most prominent user of fossil fuels [1]. As part of an effort to drastically reduce CO2 emission and curb climate change, the European Union is proposing to stop sales of solely internal combustion engine vehicles (ICEVs) by 2035 [2].

Canada does not have a reliance on fossil fuel for power generation [3], indicating that an increase in electric vehicle (EVs) adoption can drastically reduce the country's emissions. Despite the current climate crisis and the usage of personal vehicles, Canadians are slow to adopt EVs. The new motor vehicle registration data from Statistics Canada show that EVs, which include BEVs, hybrid electric vehicles (HEVs), and plug-in hybrid electric vehicles (PHEVs), only made up 7% of 2020 registrations as shown in Fig. 1.1.

According to a survey done by J.D. Power [4], the most critical factor in both purchasing and driving an EV is the battery and driving range. 20% of the owners' overall

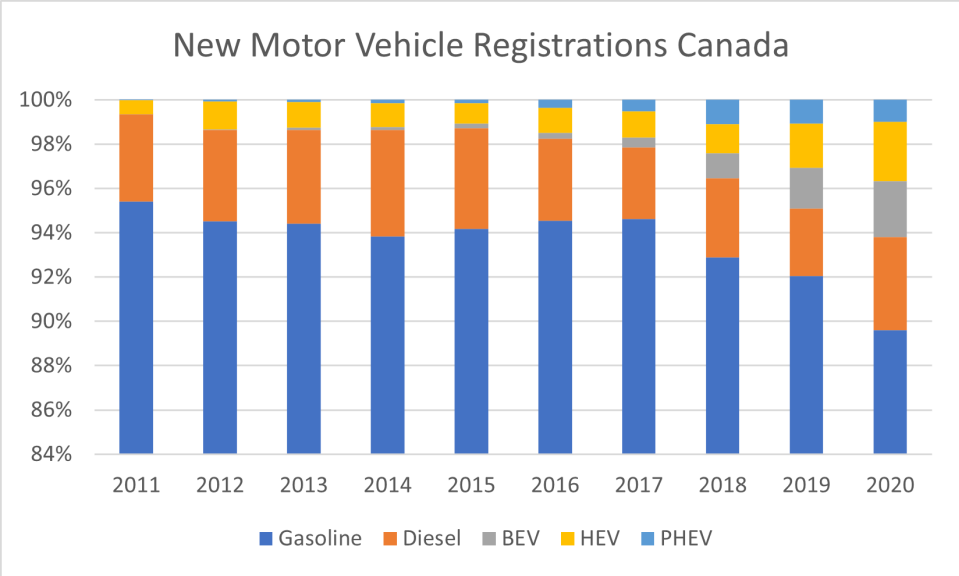


FIGURE 1.1: New motor vehicles registration by vehicle fuel type from 2011 to 2020 in Canada.

satisfaction is based on the accuracy of the stated range versus the experienced range in both premium and mass markets [4].

Contrary to popular belief, range anxiety is one of the biggest hindrances to purchasing BEVs rather than the actual range a BEV can provide. Range anxiety is the fear that one's EV battery will run out of charge before their next destination or charging station. In order to increase the use of BEVs, range anxiety mitigation solutions must be explored. One method to reduce range anxiety is to provide a more transparent and accurate estimate of the remaining distance a BEV can drive at any moment, otherwise known as remaining driving range or range estimation. Traditionally, for fossil-fuel powered vehicles, drivers look to the fuel gauge to decide how much further they can drive until they need to refuel. In BEVs, the equivalent of the ICEVs fuel gauge is the state of charge (SOC). The SOC provides information on how much charge remains in the BEV battery compared to its total capacity. Unlike ICEV's, there is no convenient sensor to measure SOC; instead, it must be estimated based on other measured data such as current and voltage. The accuracy of SOC estimation is dependent on various

factors such as temperature, battery chemistry, sensor error, etc. There are numerous SOC estimation methods, with the most prominent method being the Kalman filter (KF). Despite its popularity and commonality, there is a lack of publicly available tools available for learning and research purposes.

Pevec et. al. [5] found that people preferred the remaining driving range in kilometres or miles over SOC to decide where to charge. Having a SOC level of 30% or less is enough to make people uncomfortable, leading to the onset of range anxiety. While SOC is a good gauge on the current battery status, range estimation takes it a step further by providing distance either in miles or kilometers that the vehicle can safely drive [6].

There is no standard in current BEVs in the market on how to calculate range estimation. For example, in the Tesla Model 3, the range is estimated based on the remaining energy in the battery and the vehicle's energy consumption rating. The consumption rating for BEVs is typically given in Wh/km or Wh/mile. Simple methods such as this do not consider the weather (temperature, wind, etc.), route, recent driving history, etc., and therefore do not provide an accurate estimation. Other models that consider multiple factors and provide a more dynamic range estimation often utilize an equivalent circuit battery model. No machine learning-based battery model has been utilized for range estimation to the author's knowledge. Inaccuracies in battery modelling or battery data can lead to misleading range estimation.

The following thesis will explore two topics related to BEV's. First, a KF-based SOC estimation tool is developed in MATLAB for public use. The goal of this work is to provide students and researchers with a means to quickly learn SOC estimation through a standard method and a basis for further exploration. Second, a solution to create a driving range estimation focused on the battery is proposed. An overview for the driving range estimation algorithm is provided in Fig. 1.2. Creating a machine learning-based battery model is a necessary step in reporting an accurate and straightforward range

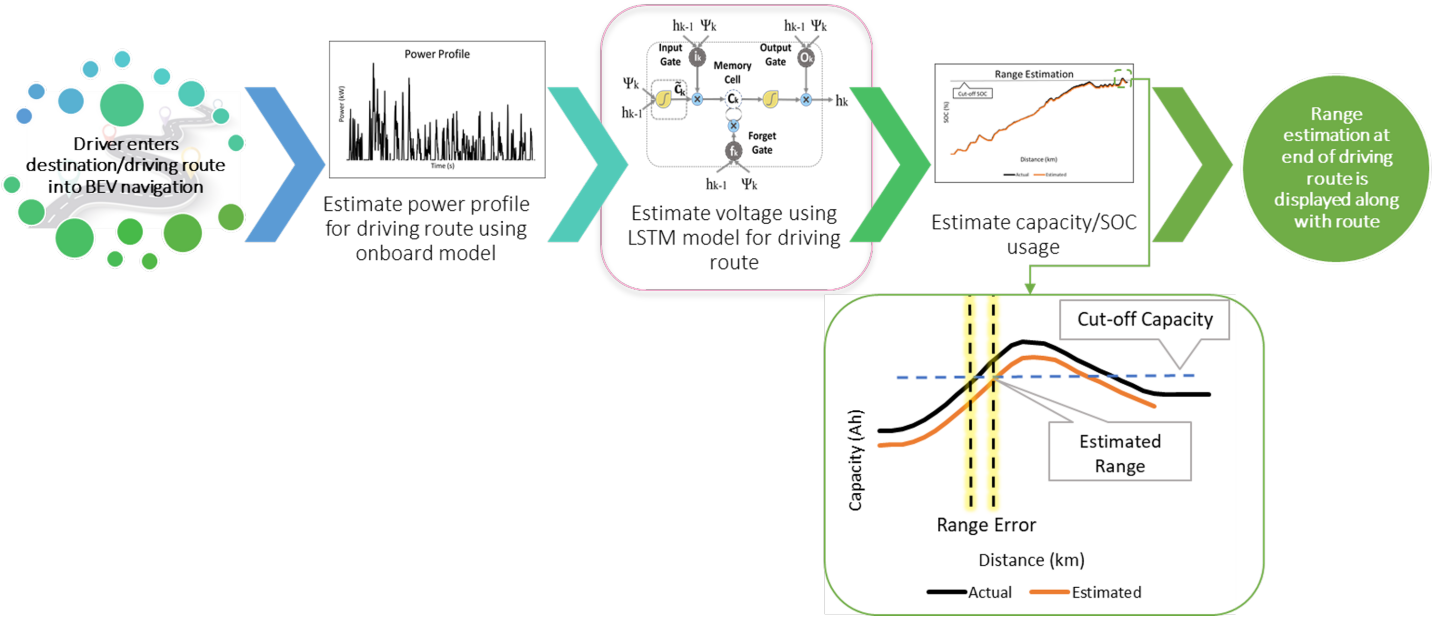


FIGURE 1.2: Overview of the proposed range estimation algorithm usage in a BEV.

estimation. The range estimation will consider factors such as temperature, payload mass, and HVAC (heating, ventilation, and air conditioning) usage, assuming that the driving route and destination are known.

A real world example of the proposed range estimation exists within the Nissan Leaf. Fig. 1.3 is a picture of a Nissan Leaf infotainment system displaying the possible range it could travel given its current location at McMaster University in Hamilton, ON and the charge in the battery pack. The white region represents the distance the vehicle could travel with current charge in the battery pack. The gray region is outside the driving range. However, this range estimation can be unreliable due to changing factors un HVAC usage, weather, and driving route.

FIGURE 1.3: Nissan Leaf infotainment screen, driving range estimation shown in the white region.

1.2 Thesis Contributions

This thesis provides two main contributions to the research of BEVs. The first contribution of this research project is the development of a SOC estimation tool for public use. The tool is an extended Kalman filter-based SOC estimator using a second-order equivalent circuit model function in MATLAB. The objective is to provide a starting point for

researchers, students, and industry learning and testing purposes. The second contribution of this research project is developing a BEV driving range estimation method based on a recurrent neural network battery model. To the author's knowledge, this is the first range estimation model centered around a machine learning battery model. This research aims to provide the first steps in developing a more accurate range estimation method easily implementable by BEV manufacturers. This model comes with a couple of assumptions:

- The purpose is not to improve the vehicle powertrain or software, merely to use it as a testbed for algorithm development.
- The driving route from point A to B is given or known.

This work aims to achieve accuracy within 5 km at the end of the drive.

Secondary contributions of this thesis include:

- creating a battery dataset varying in temperatures, payload mass, and HVAC options in battery cell testing,
- a recurrent neural network (RNN) cell-based voltage estimation model with power instead of current as input,
- utilizing SHARCNET (Shared Hierarchical Academic Research Computing Network) for faster offline battery model training.
- and, a comparison of a battery model trained with only temperature varying data versus one trained with data varying in temperature, payload mass, HVAC, grade, and aggressiveness.

1.3 Thesis Organization

The remainder of this dissertation is organized as follows. **Chapter 2** provides a state-of-the-art review on lithium-ion battery models, state of charge estimation methods, and driving range estimation in BEVs. In **chapter 3**, a Kalman filter-based battery state of charge estimation MATLAB function is given. The function uses a second-order equivalent circuit model with an extended Kalman filter. A simulation example is shown using publicly available Turnigy battery data.

Next, in **chapter 4**, four Tesla/Panasonic lithium-ion cells are tested using the Arbin cell cycler and data generated from a Tesla Model 3 model. The data represents drive cycles varying in temperature, payload mass, HVAC, grade, and aggressiveness. In **Chapter 5**, a range estimation method for electric vehicles is provided using a RNN-based battery model. A comparison of training on a standard computer CPU versus SHARCNET is provided, and a comparison of voltage estimation from two datasets. Lastly, **chapter 6** will conclude this dissertation and provide some options for future work.

Chapter 2

State-of-the-Art Review on Lithium-Ion Batteries in Electric Vehicles

2.1 Introduction

Batteries and the battery management system (BMS) are a major area of interest within battery electric vehicles (BEVs). BEV battery packs consist of individual battery cells in various series-parallel configurations. Often battery testing is completed at the cell level and then stepped up based on pack configuration. The BMS is responsible for data monitoring, battery modelling, state estimation such as state of charge (SOC), range estimation, and more [7]. Data monitoring includes taking current and temperature sensor data and using the information as part of battery models, state estimations, fault diagnosis, and so on. Battery models, in turn, are then used for other state estimations such as SOC and state-of-health (SOH). Battery models can be divided into equivalent circuit models, electrochemical models, and data-driven models. SOC, one of the more essential battery states, is the ratio of a battery's capacity over its total capacity [8].

SOC is comparable to a fuel gauge in a conventional gas-powered vehicle. SOC is a critical parameter in computing range estimation, in order to provide the user with a prediction of how much further the BEV can drive before the battery is completely depleted. This metric is much more challenging to estimate and is not directly measurable.

The following sections give an overview of battery models, SOC estimation, and range estimation found in the current literature. Section 2 provides an overview of battery basics and standard battery models. Section 3 outlines the state of charge estimation techniques. Range estimation factors, estimation methods, and common problems are reviewed in section 4. Finally, a summary of the chapter is provided in section 5.

2.2 Lithium-ion Battery Basics and Models

A battery is a device that converts stored chemical energy to electrical energy through an oxidation-reduction reaction. There are three types of batteries: primary, standby, and secondary. Primary batteries are disposable and have an irreversible chemical reaction meaning they cannot be charged. In a standby battery, the active chemical is isolated to minimize self-discharge and are typically utilized in emergencies. Secondary batteries can be recharged and will be the focus of this thesis. The base form of a battery is called a cell and is available in many shapes such as cylindrical, coin, prismatic, and pouch (thin and flat) and chemistries including lead-acid, nickel-metal-hydride (NiMH), lithium-ion (li-ion), and sodium nickel chloride (Na-Ni-Cl) [8][9]. Battery cells are connected in a series and parallel configuration and contained within a case to create a module. Combining a group of battery modules in a series/parallel configuration creates the final battery pack seen in commercial BEVs. In this chapter, a battery cell will be referred to as a battery.

2.2.1 Battery Basics

There are many voltage terms used to describe a battery. A battery's voltage will vary from fully charged (maximum voltage) to discharged (cut-off voltage). The nominal voltage is the average of this curve and is usually the battery's rated voltage [9]. Alternatively, the open-circuit voltage (OCV) describes the voltage between the battery terminals with no load applied, while the terminal voltage is the voltage with load applied [9]. All batteries have an internal resistance dependent on their chemistry, charging/discharging, temperature, SOC, and SOH [9] [10]. The standard requirements of li-ion batteries can be seen in Table 2.1.

TABLE 2.1: Basic technical requirements of a li-ion cell in a BEV from [11].

Specification	**Unit**	**BEV Requirements**
Nominal cell voltage	V	∼3.75
Cycle life	cycles	1500 – 3000
Specific power (10 s, 50% SOC, 25°C)	W/kg	1000 – 3000
Power density (10 s, 50% SOC, 25°C)	W/L	2000 – 4000
Specific energy (1C rate at 25°C)	Wh/kg	150 – 230
Energy density (1C rate at 25°C)	Wh/L	250 – 550
Self-discharge rate (50% SOC, at 25°C)		<3%/month

2.2.2 Battery Capacity

The battery capacity is the total amount of electrical charge a cell can hold, usually measured in ampere-hour (Ah) [8]. The capacity is dependent on the chemistry, temperature, and current rate [10]. The current rate or C-rate is the rate at which the battery is charged or discharged. For example, a battery with a 1C rate means that the battery will fully discharge in 1-hour [9]. SOC is the ratio of a battery's capacity over its total capacity and is comparable to a fuel gauge in a conventional gas-powered vehicle. It is a unitless number between 0 and 1, with 1 representing 100% or fully charged. 0% SOC differs between electrochemical and reported SOC. Electrochemical SOC refers to

when the battery has reached the cut-off voltage, or minimum allowable voltage [9]. At this point, any more charge extracted from the battery can lead to irreversible damage and ageing. However, in a BEV, the reported 0% SOC does not align with the electrochemical SOC. BEV manufacturers reserve some battery capacity to conceal the battery ageing from the consumer. SOC estimation methods are reviewed in section 3 [12].

2.2.3 Battery Power and Energy

The watt-hour (Wh), which represents the battery energy, can be calculated by multiplying the capacity by the nominal voltage [12]. A battery has multiple power and energy terms that are used to judge the battery. Specific power, in W/kg, is the maximum available power per unit mass and is based on the chemistry and packaging. Power density in W/L is the maximum available power per unit volume. Specific power and power density determine the battery weight and size required to achieve a particular performance target such as acceleration. Specific energy in Wh/kg is the nominal energy per unit mass. Energy density in Wh/L is the nominal energy per unit volume. The specific energy and energy density determine the battery weight and size required to achieve a particular driving range [9]. A related metric is the state of power or SOP, which refers to the ratio of peak power to nominal power for a specific time, usually a few seconds [13]. In BEVs, SOP can determine if there is enough power for starting, accelerating, or climbing (driving uphill) [14].

2.2.4 Battery Life

Batteries degrade over time due to current rates, temperature, number of charge/discharge cycles and storage conditions. A fresh battery has 100% capacity usage is at the beginning of life (BOL), while the end of life (EOL) is when it reaches a certain percentage of BOL, typically 70% - 80%, depending on the manufacturer[10]. The metric used to evaluate battery life and ageing is the state of health (SOH). Batteries aging can

be caused by capacity fade or power fade. Like SOC, SOH cannot be directly measured and therefore needs to be estimated through other means [15].

2.3 Modelling Batteries

Batteries have complex dynamics that vary with age, temperature, chemistry, and current rate. However, determining the varying battery states (SOC, SOP, SOH) can sometimes require a battery model. This section provides an overview of commonly used battery models found in literature.

2.3.1 Equivalent Circuit Models

The most widely used battery model type in BEVs is the equivalent circuit model (ECM). This model type uses electrical components such as resistors and capacitors to describe the dynamics of the battery [12]. ECM's can easily be implemented online in real-time in a BMS since they are computationally efficient and straightforward to implement [16]. However, ECMs only represent a subset of the total parameters affecting battery dynamics. Further parameter identification and implementation to the model can be computationally expensive [17]. Typical ECM's include linear and non-linear electric models, impedance-based models, and n-RC models, where n is the order of resistor-capacitor pairings in the circuit. Commonly, first-, second- or third-order models are used as parameter identification becomes more complex with higher order models.

For example, a first-order RC ECM is shown in Fig. 2.1. The output, terminal voltage (V_t), is calculated using the battery's current (i), OCV (V_{OC}), internal resistance (R_0), estimated resistor and capacitor parameters R_1 and C_1. Model parameters are estimated using battery test data, such as the hybrid pulse power characterization (HPPC) test. The parameters can differ by various factors such as temperature, SOC, and current rate. Due to this, ECMs are not interchangeable between different types of batteries.

Due to variances in battery manufacturing, ECMs can sometimes differ among the same batteries.

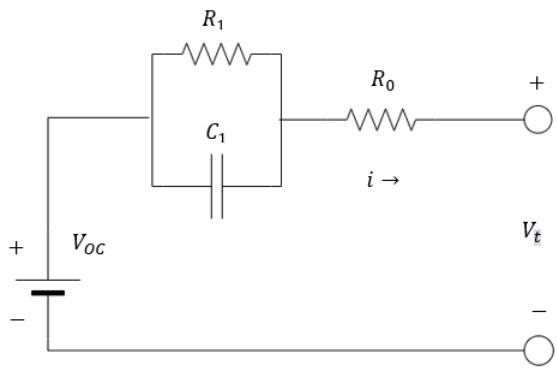


FIGURE 2.1: First order resistor capacitor equivalent circuit model of a battery cell.

The terminal voltage is calculated as:

$$V_t(k) = V_{OC}(k) - V_1(k) - i(k)R_0 \tag{2.1}$$

where V_1 is the voltage across the RC network and calculated as:

$$V_1(k+1) = \exp^{\frac{-\Delta t}{R_1 C_1}} V_1(k) + R_1(1 - \exp^{\frac{-\Delta t}{R_1 C_1}})i(k) \tag{2.2}$$

The input to the model is the current $i(k)$ at time step k. Δt is the sample time in seconds, R_1 and C_1 are the model parameters. V_{OC} in (2.1) is calculated in (2.3) as a function of SOC. The OCV can also be a function of temperature given sufficient data.

$$V_{OC} = f(SOC) \tag{2.3}$$

Model parameters can be estimated either online or offline. Online refers to computation occurring on-board the vehicle BMS while offline refers to parameter estimation on a separate computer before being implemented in a BMS. One example of an online method is recursive least square (RLS). Offline methods typically consist of computationally heavy optimization algorithms such as genetic algorithm (GA).

2.3.2 Electrochemical Models

Electrochemical models are based on the physics of the battery, specifically, the electrochemical processes through a set of partial differential equations. They model the macroscopic properties of batteries such as current and voltage and microscopic properties such as diffusion or reaction kinetics [18]. Since electrochemical models can more accurately estimate battery states, they are preferred over ECM. However, electrochemical models are computationally expensive due to a large number of parameters [18]. Therefore, they cannot be implemented online in real-time.

Example models include the pseudo-two-dimensional (P2D) model, single-particle (SP) model, extended SP model, full electrochemical model, and reduced-order electrochemical model (Fig. 2.2) [19]. While simplified, this model is still too complex for online real-time estimation without further simplification [19].

A typical li-ion battery consists of a negative electrode (anode) made of carbon, a positive electrode (cathode) made of a metal oxide, and a separator made of a lithium salt electrolyte. The negative electrode reaction, positive electrode reaction, and overall battery reaction chemical equations are given by (2.4), (2.5), and (2.6), respectively [19].

$$Li_xC_6C_6 + xLi^+ + xe^- \tag{2.4}$$

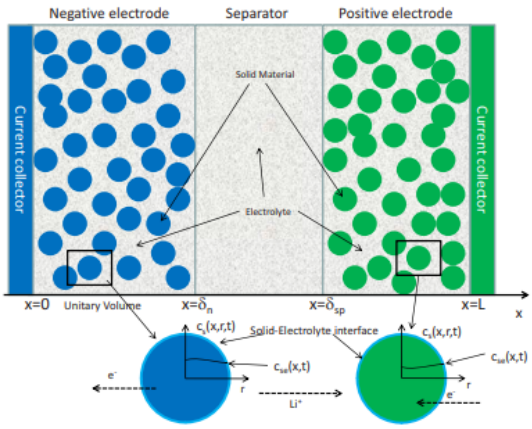


FIGURE 2.2: Reduced-Order electrochemical model from [19].

$$Li_{y-x}Mn_2O_4 + xLi^+ + xe^- Li_yMn_2O_4 \tag{2.5}$$

$$Li_{y-x}Mn_2O_4 + Li_xC_6Li_yMn_2O_4 + C_6 \tag{2.6}$$

2.3.3 Data Driven Models

Data-driven models are an abstract representation of the battery and hold no physical meaning. They require a large number of battery testing data to build and are largely machine learning-based (ML) [20]. ML, a subsection of artificial intelligence, are algorithms that learn and improve from data, and they comprise of supervised and unsupervised algorithms [10]. Supervised learning requires a dataset with the relationship between the input and output known; in other words, the output is fed into the system for training. On the other hand, unsupervised learning requires finding structure in the dataset without the output being given. ML algorithms require immense data and computational time, and power for training. However, ML algorithms can provide more accurate results than ECM models once properly trained [17].

Artificial neural networks (ANN) are ML algorithms that are based on the human brain. ANNs map inputs to outputs by approximating an unknown function between the input x and output y. There are both supervised (classification) and unsupervised (clustering) ANNs. Various ANN algorithms have been utilized for battery estimation, such as voltage [21], SOC [22], and SOH [22]. Common supervised ML algorithms used for estimation include [22]:

- Feedforward neural networks (FNN)
- Recurrent neural networks (RNN)
- Support vector machines (SVM)
- Radial basis function (RBF)
- Extreme learning machine (ELM)
- Hamming networks (HM)
- Bayesian networks (BN)

FNNs, sometimes referred to as multilayer perceptrons (MLPs) are made up of multiple perceptrons, also termed layers or neurons, as shown in Fig. 2.3a. The first or leftmost layer is the input layer and the last or rightmost layer is the output layer as shown in Fig. 2.3b. Any layer(s) in between are referred to as the hidden layers. Information in an FNN flows in one direction from input to output and trains using back propagation. An architecture whereby the output information is fed back as an input for model improvement is termed RNN, and will be the focus of chapter 5.

The goal of an FNN is to approximate a function f(x) by mapping an input x to an output y via learning parameters θ, where $y = f(x; \theta)$. Each neuron in each layer l after the input uses an activation function such as the sigmoid function, the weights $W^l_{(i,j)}$,

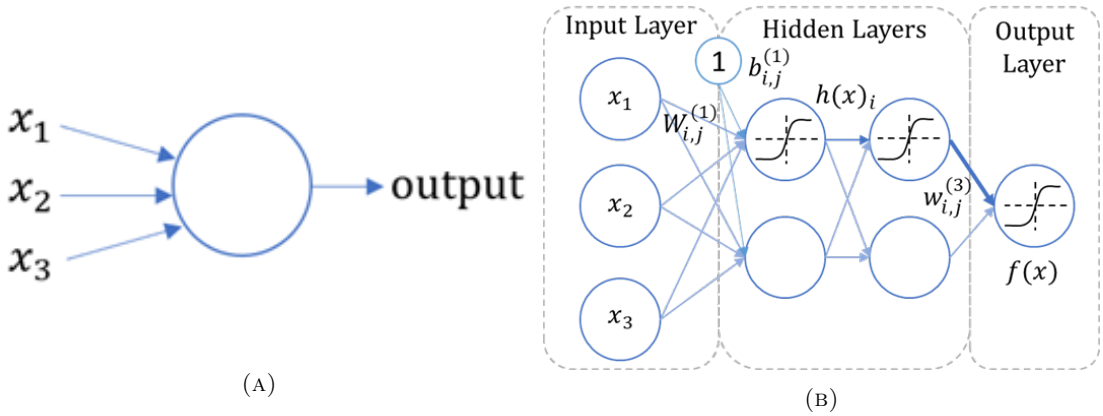


FIGURE 2.3: (a) Perceptron. (b) FNN Architecture.

and biases $b_{(i,j)}$ to calculate the output, $h(x)_i$. This process is referred to as the forward pass. A gradient-based optimizer is used to minimize a cost function to ensure that small changes in weights and biases create an improvement. The error between the calculated output and actual output is used during a backward pass to update the weights and biases of the network during training. The network training is deemed complete once a threshold of error or when the maximum number of epochs (a full pass) has been reached. After training, only the forward pass of the network is used, resulting in a method that is implementable on-board a BMS.

In [17], the authors compared an ECM and an RNN model for battery voltage estimation. They utilized a long short-term memory (LSTM), a type of RNN, with current, temperature, and capacity in Ah as inputs. The LSTM model outperformed the ECM model in 92% of the test cases. The authors in [21] utilize an RNN with a gated recurrent unit (GRU) to estimate battery voltage. The same model was trained with two different inputs: current and power. Both models can provide accurate predictions at various temperatures. ML algorithms have also been used in conjunction with other estimation methods to predict battery voltage. For example, [23] combined a single-layer FNN with an ECM and thermal model. In [24], a particle filter was implemented to predict model parameters for an RBF algorithm.

The coverage and range of data-driven models are highly dependent on the data used to train the model. The more parameters, such as temperature, are included in the training, the more coverage the model will have [20].

In summary, ECM, electrochemical models, and data-drive models are a subset of the vast amounts of methods used for battery modelling. A comparison of the three methods is given in Table 2.2.

TABLE 2.2: Comparison of different types of battery modelling.

Battery Model Type	**Model Parameterization (Computational)**	**Complexity**	**Accuracy**
Equivalent Circuit Model	Medium	Low	Medium
Electrochemical Model	High	High	High
Data-Driven Models	Medium	Medium to High	Medium

2.4 State-of-Charge Estimation Methods

State of charge is the ratio of a battery's current capacity over its total capacity. This section will cover SOC estimations found in literature including Coloumb counting, SOC-OCV, model-based estimation, and data-driven estimation.

2.4.1 Coulomb Counting

Coulomb counting or ampere counting is a widely used method in small devices that employ batteries such as medical equipment. This method works by utilizing the measured battery current (i) with a known initial or previous SOC ($SOC(k)$), Coulombic efficiency (η), and nominal capacity (C_n). The following equation can calculate the SOC at time k:

$$SOC(k+1) = SOC(k) - \frac{\eta \Delta t \, i(k)}{C_n} \tag{2.7}$$

The advantage of this method is that it can be utilized online, meaning in a vehicle or device; it is accurate and straightforward. The disadvantages of coulomb counting is that it does not consider irregularities between different battery types, ageing, or temperature, and inaccurate measurements of current can lead to large errors [20]. As noted in the following sections, this method is best suited in combination with other methods.

2.4.2 SOC-OCV

The SOC has a direct relationship with OCV, and this method is sometimes referred to as a 'look-up table' method. The SOC-OCV relationship is determined by bringing the battery to a known state, such as fully charged, fully cycling the battery at very low C-rates, and then letting the battery rest [20]. This relationship can vary depending on the battery chemistry, temperature, age, and C-rate [25]. Fig. 2.4 provides an example charging and discharging SOC-OCV curve of a 4.5 Ah Tesla Model 3 battery cell tested at a rate of C/20. Due to hysteresis, the charging and discharging curves are not identical and are usually averaged to obtain one curve.

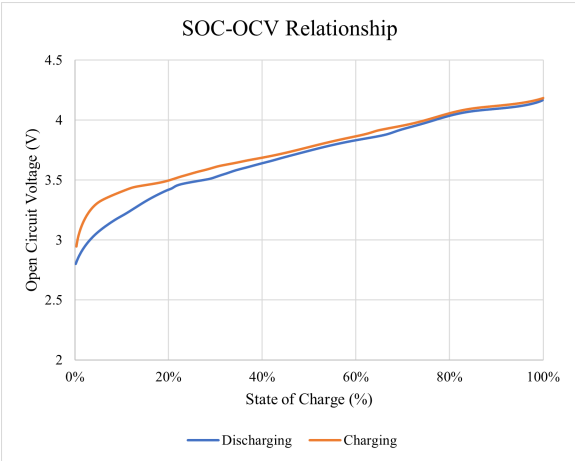


FIGURE 2.4: SOC-OCV relationship of a 4.5Ah Panasonic/Telsa battery cell.

This method cannot be implemented online in an BEV since it requires the battery

to rest for long periods. It is prone to error in some batteries due to a linear trending curve between 10% to 90% SOC [20].

2.4.3 Model-Based Methods

Model-based SOC estimation techniques utilize either ECM or electrochemical models. There are state observer methods such as proportional-integral (PI) observer, H∞ observer, Luenberger observer, sliding mode observer, and non-linear observer [20]. Filter-based methods include Kalman filter (KF) and variations, particle filter (PF), and smooth variable structure filter (SVSF). They are accurate, have low complexity, and are implementable in real-time and thus preferred over state observers [26].

Filter-based methods for SOC estimation follow the same steps: identify the battery model, parameter identification, and finally, apply the filter algorithm. The parameter identification portion can be complex, time-consuming, and computationally expensive. It can be completed online via RLS variations or offline through optimization algorithms. GA, particle swarm optimization (PSO) algorithm, simulated annealing (SA) algorithm are just some of the example offline optimization algorithms [26]. An extended Kalman filter (EKF) based SOC estimation using a second order resistor capacitor (2RC) model is presented in chapter 3.

A variety of papers have been published with different Kalman filtering techniques, battery testing conditions, and battery models. In [27], the KF was compared against an unscented KF, while in [28] an extended KF (EKF) was compared to the central difference KF. In [29], an EKF was used with a PNGV battery model using MATLAB/Simulink. In [30], several battery parameter estimation methods are compared using a Thevenin battery model and EKF. These are a handful of the hundreds of published results on SOC estimation using KF. Despite these findings, very few publicly

available tools, functions, or scripts are available for researchers. One example available online is [31]. It is a MATLAB/Simulink based SOC estimation using EKF and unscented KF.

The disadvantage of model-based SOC estimation methods is that they require extensive parameter identification based on temperature, SOH, and chemistry. The problem of increasing accuracy of SOC while decreasing the number of parameters required can be solved with machine learning methods.

2.4.4 Data Driven Methods

As mentioned in section 2.2.3, data-driven methods, especially machine learning algorithms, have been used for various battery modelling and estimation, including SOC. Data-driven approaches are typically represented as black-box models where a global mathematical model of the controlled system is unknown. With proper training and data, these models can solve non-linear problems with high accuracy [20].

Chemali et al. [32] utilized a deep feedforward neural network or DNN to estimate SOC at five temperatures ranging from -20°C to 25°C. A traditional FNN has one to two layers, and a DNN typically has three or more layers, including the input and output layers. They achieved a mean absolute error (MAE) of 1.10% at 25°C while an an MAE of 2.17% at -20°C.

An LSTM-RNN model was also utilized by Chemali et al. [33] without any battery model or filters to estimate SOC at ambient/fixed temperatures and varying temperatures. LSTMs are best used for time-series-based applications due to their feedback loop achieved through the various gates. The input to the model is measured battery voltage, current, and temperature at the current time-step. The model was trained with a Panasonic 18650 PF cell and achieved as low at 0.573% MAE in one testing scenario but can be trained for other batteries without changing the architecture.

Other ML algorithms utilized include random forest regression with a gaussian filter [34], DNN [35], LSTM with transfer learning [36] [37], and SVMs [38][39] to estimate SOC. Between RBF, FNN, and RNN, RNN has been found to have a lower root mean squared error (RMSE) of 0.5% on average [22].

2.5 Range Estimation

Range estimation is a user-friendly report of the distance in either in miles or kilometres that the vehicle can safely drive [6]. The authors in [5] found that people preferred the remaining driving range over SOC to decide where to charge. Having a SOC level of 30% or less is enough to make people uncomfortable, leading to the onset of range anxiety. To put this into perspective, the 2019 Tesla Model 3 Standard EPA range is 220 miles or 354 km, and 30% of this range is 66 miles or 106 km [40]. However, some studies suggest that a "range buffer" of 20 miles or 32 km was sufficient to reduce drivers' range anxiety and boost confidence [6]. Range estimation is also referred to as remaining driving range, range prediction, and distance to empty in BEVs.

The following section will review range estimation influencing factors, methods and techniques, and finally, current problems.

2.5.1 Range Estimation Influencing Factors

To provide an accurate range estimation, the factors that influence range must be reviewed. These factors can be divided into vehicle design and simulation, driver behaviour, and environmental factors. Each type depends on both direct and indirect parameters as well as constant and variable parameters [41].

2.5.1.1 Vehicle Design and Simulation

Range estimation is a problem based on the whole vehicle. It is dependent on the overall vehicle design/characteristics (ex. auxiliary power requirements, vehicle mass, speed, etc.), and the battery management system (ex. SOC, SOH, thermal management etc.) [20]. The basis of range estimation is to calculate the power and energy required to move the vehicle at specific speed(s) and then comparing to the energy remaining in the battery. First, from the vehicle perspective, to calculate the power, all forces acting on the vehicle must be considered. The net force (F_{net}) acting on a moving vehicle include the force at the wheel account for rolling resistance (F_{wh}), the gravitational force when on an incline (F_g), and the aerodynamic drag force (F_{drag}) as shown in Fig. 2.5, where F_a is the accelerating force, CG is the center of gravity and θ is the angle of incline.

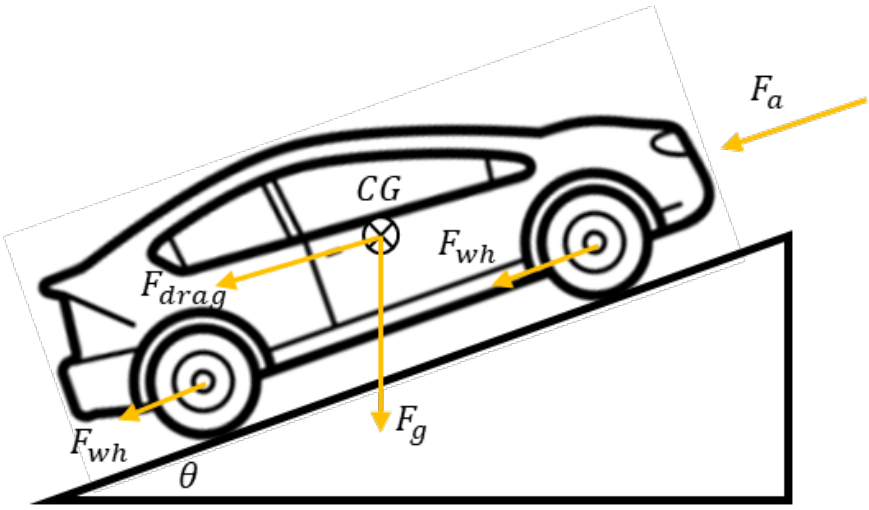


FIGURE 2.5: Forces acting on a vehicle.

Vehicle data can be generated using a model/simulation or gathered from real-world driving. Models and simulations do not provide real world driving scenarios but are flexible in testing. Historical data gathered from vehicles on the road, reflects real-world driving scenarios but are limited to the vehicle manufacturer's design. A quasi-static vehicle model is used in [42] to determine the power demand with the assumption that

the vehicle will follow the exact speed input. An advantage of this model type is that it does not require complex differential equations. Historical or real vehicle data is used in [43] and [44] as part of a hybrid ML and regression analysis range estimation model, respectively.

Next, the energy remaining in the battery must be considered. As stated in the previous sections, battery modelling and state estimation for BEVs is quite complex. For information on battery modelling and SOC estimation, please see sections 2.3 and 2.4. SOC accounts for approximately 54% of the factors that influence range prediction [41]. Since SOC is the primary value in current BEVs providing information on energy remaining in the battery, the SOC influence can be extended to the battery management system overall and its accuracy in predicting the battery states.

A combined battery model using the hydrodynamic model and 2RC ECM is used in [42] to predict SOC and voltage, respectively. [45] uses the Thevenin ECM battery model in MATLAB/Simulink to estimate energy consumption.

2.5.1.2 Driver Behaviour

Driving style and behaviour have a significant impact on the range of an EV [46]. According to [41], a driver's behaviour accounts for 10% of the factors influencing range prediction. Driving style is typically divided into three categories: aggressive, moderate, and conservative [47]. Aggressive driving style is characterized by frequent short-duration braking, rapid acceleration, and high speed [46], leading to decreased energy efficiency [47]. However, driving style is not easily quantified. Studies have used velocity, acceleration, jerk, power, throttle position, and a combination of these metrics to quantify driving styles [48]. For example, [48] estimates driver profile via periodogram of jerk trace where anything >0.1 Hz is considered part of the driver's behaviour and the remainder is credited to traffic. This factor is then used as input into an ANN range

estimation model. In reference [42], Markov chains with inputs speed and acceleration for the first chain and slope for the second chain are utilized to predict a driving profile for remaining driving range estimation.

2.5.1.3 Environmental Factors

Environmental factors that influence range estimation include the weather (temperature, wind speed, wind direction, precipitation, etc.), the road (road type, segment length, elevation, etc.), and traffic [47]. The weather captures several factors that influence range estimation. For example, in [49], the effects of thermal management consumption on the remaining range estimation of BEVs were analyzed. The study found that at -15°C, a PTC-heater will consume between 3.7kWh to 4.8kWh in energy. Assuming an average energy consumption of 15kWh, this equates to 25km to 32km. Minimizing auxiliary load, the vehicle range can still decrease by 20% at -7°C [10]. When accounting for strong headwind, energy consumption estimation error reduced from 30% to a few percent [50].

According to [46], route and traffic are estimated to have a combined 11% influencing factor on range estimation. For example, a route with lots of hills will require more power to overcome the incline, while urban routes and more traffic will lead to more accelerating/braking cycles. These scenarios and actions will lead to the vehicle operating less efficiently. In [51], the driving distance, terrain, speed limit, and traffic are all considered using Google Maps API for range estimation using a big-data framework. In [48], mobile data provides traffic information while a decision tree is used to determine the road type for range estimation in BEVs. The road network topology, road grade, and road link travel speed are all considered in [52].

2.5.2 Range Estimation Methods

With advances in data collection and computing power, data-driven techniques have become a popular method for range estimation. In [51], a big-data framework is utilized with a graphical user interface (GUI) in MATLAB to provide users with range estimation. The users enter the origin and destination into the system and are presented with the current SOC, range estimation, and weather. [52] uses a telematics system to provide a two-tiered estimation, a rough range estimation and a precise range estimation based on a preset battery level. Telematics is a data center used to track vehicles using GPS and onboard diagnostics.

[51] and [52] provide feasible methods that range can be estimated but do not provide validation. However, the authors in [44] used regression analysis to provide an online prediction of the remaining range with under 5km error. In [53], a hybrid ML model consisting of regression trees and self-organizing maps estimated power consumption which is then used to estimate range. The study achieved an RMSE of 1.5 kWh. Another ML-based research [43] uses a blend of extreme gradient boosting regression tree (XGBoost) and light gradient boosting regression tree (LightGBM) to estimate the remaining driving range. Over 2000 trips from five BEVs from a cloud-based database are utilized for training and testing the model. The model's input includes a K-means clustering algorithm to predict driving patterns. They are able to achieve an RMSE of 0.75 km.

Studies show non-data-driven techniques as well. For example, in [54], range estimation is done via a KF. The KF, with speed and acceleration as inputs, estimates energy consumed. The remaining energy is calculated and subsequently the remaining range as:

$$RemainingRange = \frac{remaining\ battery\ energy\ capacity}{fuel\ efficiency} \tag{2.8}$$

Another study [42] predicts range using a particle filter and Markov chains. The Markov chains are used to predict a driving profile which is then used with the battery state estimation to predict range. A summary of the range estimations methods found in literature is provided in Table 2.1. Fig. 2.6 illustrates different range estimation method types such as filters which would include the KF [54] and hybrid which would include Kernel Principle Component, fuzzy C clustering, Markov chains, and back propagation neural networks [57].

Method	**Data Source/Inputs**	**Notable Points**
Big Data [51]	Google API for map and route information (distance, terrain, location), Wunderground.com for weather information (temperature, wind), driver history (speed), Tesla Roadster model (vehicle parameters, battery power, current)	Utilizes a MATLAB/Simulink vehicle model to input power consumption into a 2RC ECM battery model with internal resistance based on temperature
Telematics [52]	Unspecified	Includes minimum cost route searching and zero-energy point determination.

Regression Tree and Self-Organizing Maps [53]	ChargeCar project of the CREATE Lab at Carnegie Mellon University (421 EV trips). Inputs include 12 features of time-series and static data such as mean acceleration and distance.	Predicts power consumption of EV trip which will then be used for range estimation. Does not elaborate/extend to range in km or miles.
XGBoost-LightGBM blend [43]	NDANEV 2000 trips from 5 EV's. Inputs include speed, motor and battery voltage and current, SOC, cell temperature, mileage from odometer.	Blended model includes the output of XGBoost and LightGBM (using the same features) into a secondary XGBoost
Kalman Filter [54]	Electric bus logged data. Inputs include speed, acceleration, and variations of the two.	KF estimates energy consumption used to calculate the fuel efficiency and then estimates range as a function of SOC.
Particle Filter and Markov Chains [42]	UDDS, Artemis rural, and Artemis motorway drive cycles for a Nissan Leaf vehicle with the battery scaled down to 2.15 Ah Li-ion cell. Inputs include velocity, acceleration, and slope of road to calculate power demand. Battery model input includes current based on power demand.	Utilizes a combined kinetic (well-based) and 2RC ECM to estimate battery voltage. The particle filter is used to calculate the battery states and the Markov chains are used to estimate the driving profile.

Regression Analysis [44]	2000 kms of a Tazzari EM1 with max 12.288 kWh energy. Inputs include energy consumption, elevation, and speed profile.	Compares three regression models: conventional, linear, and SVR. Battery energy, odometer, and velocity data logged from CAN network of vehicle.
Kernel Principal Component, fuzzy C clustering, Markov chains, and back propagation neural networks [57]	Unnamed EV with a 78 Ah battery pack using various urban driving routes (max speed of 32 km/h).	Validation using ECE 15 condition rotary drum test bench
Battery equivalent circuit model and SOC estimation [58]	2 unspecified vehicles with prototype device installed. Inputs include battery voltage, current, air temperature	Lead acid battery using 2RC ECM. Prototype device installed onboard test vehicle.
Artificial Neural Network [48]	Commercial EV with 35.06 kWh battery (LiFePO4) with WLTC drive cycle on a dynamometer. Inputs include but are not limited to SOC/SOH, temperature, traffic, aggressiveness.	Periodogram used to calculate aggressiveness, decision tree used for road type. Output of ANN is range compared to vehicle odometer.

TABLE 2.3: Comparison of range estimation methods, data source, inputs, and notable point found in literature.

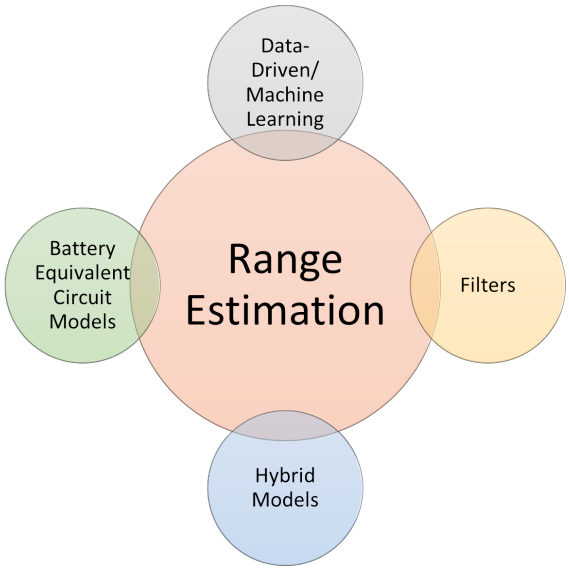


FIGURE 2.6: Summary of range estimation types found in literature.

2.5.3 Current Problems with Range Estimation

Range estimation provides essential information to the end-user on the state of the battery more than SOC. However, range estimation is a complex problem to solve. First, increasing the number of factors considered in the range estimation also increases the error [41]. A review in [47] found 56 different factors in literature. Furthermore, some factors such as SOC require an independent estimation model, which comes with errors that must be propagated to the range estimation error [41]. Studies have found more accurate battery models in ML [17], but range estimation studies using battery models still use less accurate ECMs [42],[45]. Recently, studies have shown that state of energy (SOE) to be a more accurate indicator of residual energy in batteries than SOC [55]. SOE is the ratio of the remaining energy in a battery over it's total energy [56]. Other factors such as driving style and preferences (i.e., HVAC) are challenging to quantify, standardize, and predict [47]. Finally, few studies have explored an online implementable solution for BEV usage.

2.6 Summary

This chapter reviewed published literature on batteries used in BEVs, specifically modelling, state of charge estimation, and range or remaining driving range estimation. Batteries have several crucial states, including state of charge, state of health, and state of power. They can be modelled using equivalent circuit models, electrochemical models, data-driven models. Equivalent circuit models are widely used for their low complexity and ease of use in online real-time applications. Electrochemical models provide more accuracy but require more complex computation and knowledge and are unsuitable for online applications. Data-driven or, more specifically, machine learning models are a good mix of the former. While they require immense computational power, once trained, machine learning-based battery models are accurate, robust, and can be utilized for online applications.

Four SOC estimation methods were reviewed: Coulomb counting, SOC-OCV, model-based, and data-driven. Coulomb counting and SOC-OCV are simple but inaccurate methods best suited for low-level applications such as cell phones. Model-based is the primary method in real-world BEV application. However, with the advances in computing power and machine learning techniques, data-driven strategies are becoming as accurate, if not more, as model-based methods in SOC estimation.

Accurate range estimation can improve range anxiety and therefore increase the adoption of BEVs into the current market. However, the complexity of range estimation due to the influence of factors such as the driver vehicle model and design, the driver behaviour, and the environment, means improved methods must be developed to accurately report range. Range estimation methods span from filters such as the KF and particle filters to machine learning such as a hybrid XGBoost and LightGBM. As of yet, no estimation approach is favoured.

Chapter 3

A Kalman Filter Based Battery State of Charge Estimation

3.1 Introduction

Electric vehicles require a complex battery management system to monitor the SOC, SOH, and cell temperature [12]. Battery parameters cannot always be directly measured, instead models are used as an indirect estimation of current battery state. The SOC is commonly estimated through Coulomb counting, however this method is often inaccurate due to errors in the current sensor and determination of initial SOC [60]. Combining Coulomb counting and SOC-OCV mapping approaches with equivalent circuit models (ECM) and Kalman filter (KF) can produce more accurate SOC estimations [61][62]. [27]-[30] are a handful of the hundreds of published results on SOC estimation using KF. Despite these findings, very few publicly available tools, functions, or scripts are available for researchers. One example available online is [31]. It is a MATLAB/Simulink based SOC estimation using EKF and unscented KF.

Battery parameters are significantly impacted by temperature, in cold temperatures, for example, the capacity decreases and resistance increases [10][63]. However, there are

not many SOC estimation studies that consider this impact in more detail. For example, in [64], the authors estimated the state of charge only at three different temperatures. Although the temperatures covered a wide range of 36℃, the paper did not capture the dynamics of the entire temperature range. More information on SOC estimation is provided in chapter 2.

In this chapter, a MATLAB function with the objective to provide a public tool that estimates the battery SOC and terminal voltage at different temperatures using a second-order resistor-capacitor (2RC) ECM along with an extended Kalman filter (EKF) is given. A flowchart of this work is given in Fig. 3.1. The goal of this work is to have a MATLAB function to serve as a basis for other researchers to utilize.

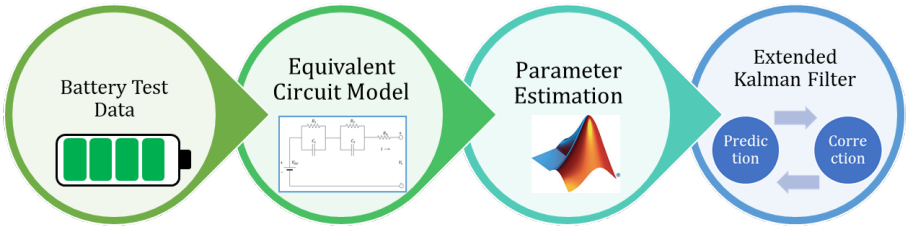


FIGURE 3.1: State of charge estimation process using a second order equivalent circuit model and extended Kalman filter.

This chapter is organized as follows: the state of charge estimation algorithm is presented in section 3.2, while the required battery data and how to use the function are explained in section 3.3. In section 3.4, an example use of the function is shown, and a summary and future work are discussed in section 3.5.

3.2 State of Charge Estimation Algorithm

One of the most common battery models seen in literature is the 2RC ECM (Fig. 3.2). It consists of the battery OCV, internal resistance, and two parallel RC pairs. Once these parameters have been optimized, the discrete-time state-space form of the battery

model is used in the EKF algorithm. Given battery measurements (i.e. current, voltage, temperature) over time, the EKF will estimate the unknown variables in a dynamic system.

3.2.1 Battery Modelling

In Fig. 3.2, the OCV is represented by V_{OC}, the output terminal voltage by V_t, and the internal resistance of the battery by R_0. The voltage across the first RC network is V_1 and V_2 across the second network. (3.1) to (3.3) describe the ECM dynamics in state-space [8].

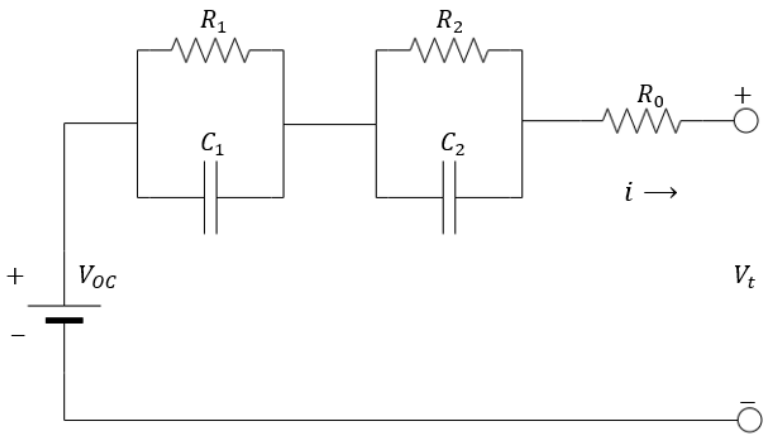


FIGURE 3.2: Second order RC equivalent circuit battery model diagram.

$$SOC(k+1) = SOC(k) - \frac{\eta \Delta t \, i(k)}{C_n} \tag{3.1}$$

$$V_t(k) = V_{OC}(k) - V_1(k) - V_2(k) - i(k)R_0 \tag{3.2}$$

$$\begin{aligned} V_1(k+1) &= \exp^{\frac{-\Delta t}{R_1 C_1}} V_1(k) + R_1(1 - \exp^{\frac{-\Delta t}{R_1 C_1}})i(k) \\ V_2(k+1) &= \exp^{\frac{-\Delta t}{R_2 C_2}} V_2(k) + R_2(1 - \exp^{\frac{-\Delta t}{R_2 C_2}})i(k) \end{aligned} \tag{3.3}$$

The input to the model is the current $i(k)$ at time step k. Δt is the sample time in seconds, R_1, C_1, R_2 and C_2 are the RC model parameters. V_{OC} in (3.2) is calculated in (3.4) as a function of SOC and battery surface temperature. Each SOC-OCV curve is calculated using the HPPC test results obtained at 40℃, 25℃, 10℃, 0℃, and -10℃ [65].

$$V_{OC} = f(SOC, Temperature) \tag{3.4}$$

The state and measurement equations can be calculated in (3.5) and (3.6) as follows:

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}_k \boldsymbol{x}_k + \boldsymbol{B}_k u_k \tag{3.5}$$

$$z_k = \boldsymbol{C}_k \boldsymbol{x}_k + D_k u_k \tag{3.6}$$

Where $\boldsymbol{x}_{k+1}$ is the system state vector at time $k+1$, the state variables are $\boldsymbol{x} = [SOC, V_1, V_2]$, system input is $\boldsymbol{u}_k = \boldsymbol{i}_k$, and system output is $z_k = V_t$. The $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, and D matrices and variables are given by (3.7) to (3.10) using (3.1) to (3.3):

$$\boldsymbol{A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \exp^{\frac{-\Delta t}{R_1 C_1}} & 0 \\ 0 & 0 & \exp^{\frac{-\Delta t}{R_2 C_2}} \end{bmatrix} \tag{3.7}$$

$$\boldsymbol{B} = \begin{bmatrix} -\frac{\Delta t}{Q}\eta[k] \\ R_1(1-\exp^{\frac{-\Delta t}{R_1 C_1}}) \\ R_2(1-\exp^{\frac{-\Delta t}{R_2 C_2}}) \end{bmatrix} \tag{3.8}$$

$$\boldsymbol{C} = \begin{bmatrix} \frac{\partial V_{OC}}{\partial SOC} & \frac{\partial V}{\partial V_1} & \frac{\partial V}{\partial V_2} \end{bmatrix} = \begin{bmatrix} \frac{\partial V_{OC}}{\partial SOC} & -1 & -1 \end{bmatrix} \tag{3.9}$$

$$D = -R_0 \tag{3.10}$$

3.2.2 Extended Kalman Filter

The EKF, a version of the regular KF, is used to estimate the states for a non-linear system. EKF uses a two-step prediction-correction algorithm as described in (3.11) to (3.15), adapted from [66], where k denotes a discrete point in time, K is the Kalman gain, P is the covariance of the measurement error, Q is the covariance of the process, and R is the covariance of the output. First, a prediction or time update is done and then the correction or measurement update. This cycle repeats until the end of the data. Note, the hat symbol, $\hat{}$, represents an estimate of a variable, $|k$ denotes predicted or a-priori estimate, and $|k+1$ denotes updated or a-posteriori estimate.

Prediction (Time Update)

1. Project the states ahead (a-priori):

$$\hat{\boldsymbol{x}}_{k+1|k} = \boldsymbol{A}\hat{\boldsymbol{x}}_{k|k} + \boldsymbol{B}u_k \tag{3.11}$$

2. Project the error covariance ahead:

$$\boldsymbol{P}_{k+1|k} = \boldsymbol{A}\boldsymbol{P}_{k|k}\boldsymbol{A}^T + \boldsymbol{Q}_k \tag{3.12}$$

Correction (Measurement Update)

1. Compute the Kalman gain:

$$\boldsymbol{K}_{k+1} = \boldsymbol{P}_{k+1|k}\boldsymbol{C}^T(\boldsymbol{C}\boldsymbol{P}_{k+1|k}\boldsymbol{C}^T + R_{k+1})^{-1} \tag{3.13}$$

2. Update the estimate with measurement z_k (a-posteriori):

$$\hat{\boldsymbol{x}}_{k+1|k+1} = \hat{\boldsymbol{x}}_{k+1|k} + \boldsymbol{K}_{k+1}(z_{k+1} - \boldsymbol{C}\hat{\boldsymbol{x}}_{k+1|k}) \tag{3.14}$$

3. Update the error covariance:

$$\boldsymbol{P}_{k+1|k+1} = (1 - \boldsymbol{K}_{k+1}\boldsymbol{C})\boldsymbol{P}_{k+1|k} \tag{3.15}$$

Since the the KF assumes the data is in the form of a Gaussian distribution and the functions applied on it linear, the states described in (3.7) to (3.10) need to be linearized as well as utilizing the extended or EKF version for the algorithm to work properly. Matrix $\boldsymbol{C}$ as seen in (3.9) is the only matrix that requires linearization as the battery's SOC-OCV relationship is non-linear.

A fourth optional step was added to the EKF equations based on [67] in which the matrix Q is updated with each iteration using (3.16) resulting in an AEKF, allowing for noise covariance to be updated online

$$\boldsymbol{Q}_k = \boldsymbol{K}_k * Error_k * \boldsymbol{K}_k^t \tag{3.16}$$

3.3 Extended Kalman Filter State-of-Charge Estimation Function

This section will review the required battery data as well as the syntax and commands in order to utilize the EKF SOC estimation function. One can find publicly available battery test data at [65] [68] [69]. The MATLAB function, based on [66], as well as the example illustrated below can be found at https://www.mathworks.com/matlabcentral/fileexchange/90381-state-of-charge-estimation-function-based-on-kalman-filter.

3.3.1 Battery Parameters

Before using the `EKF_SOC_Estimation` function, users will need the SOC-OCV curve, R_0, and the 2RC ECM battery parameters for the specific battery that the SOC is being estimated for. This data is loaded within the function and are not passed in as function parameters.

Appropriate battery testing should be done to obtain data for OCV as a function of SOC for a desired range of battery temperatures (i.e., multiple datasets, each for a different battery temperature as shown in Fig. 3.3). This testing typically involves charging and discharging the battery at low currents (0.05C), however, the complete details of this testing may vary. Once finalized datasets are obtained, it is important to ensure that there are no repeated SOC points (especially if these points have differing OCV values) as this will cause an interpolation error when the function is running. This does not apply to repeated OCV values.

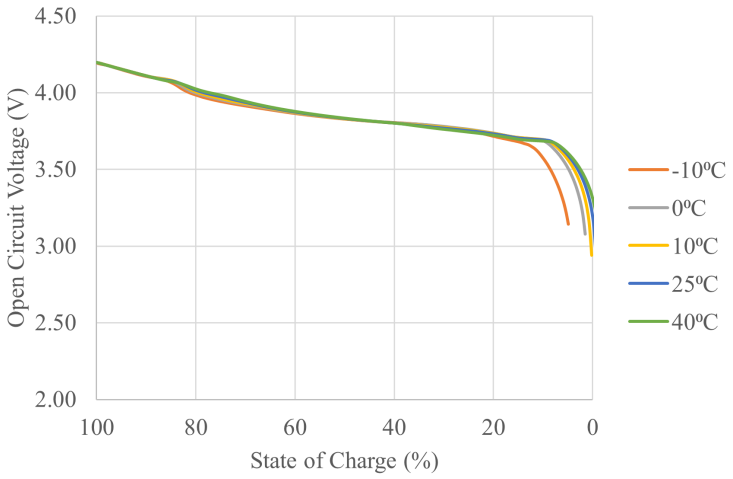


FIGURE 3.3: OCV vs. SOC of a Turnigy Graphene 5000 mAh Li-ion battery at different temperatures.

Typical 2RC ECM parameters are R_0, R_1, C_1, R_2 and C_2, where R_0 represents the internal resistance of the cell, and the rest of the parameters represent non-physical characteristics of the cell, which when grouped into RC branches, as shown in Fig. 3.4, describe the cell's dynamics. To obtain these parameters at different temperatures as functions of SOC, a model-based parameter optimization approach within MATLAB [71] was used. As described by [72], HPPC test data at different temperatures [65] was evaluated within the optimization algorithm to ensure the ECM accurately described the behaviour of the battery.

The function can be called using:

```
function [SOC_Estimated, Vt_Estimated, Vt_Error] = ...
    EKF_SOC_Estimation(Current, Vt_Actual, Temperature)
```

The function takes a drive cycle current measured in amps (`Current`), V_t measured in volts (`Vt_Actual`), and battery temperature measured in °C (`Temperature`), all as vectors of type double as input. The length of all vectors must be the same. The function

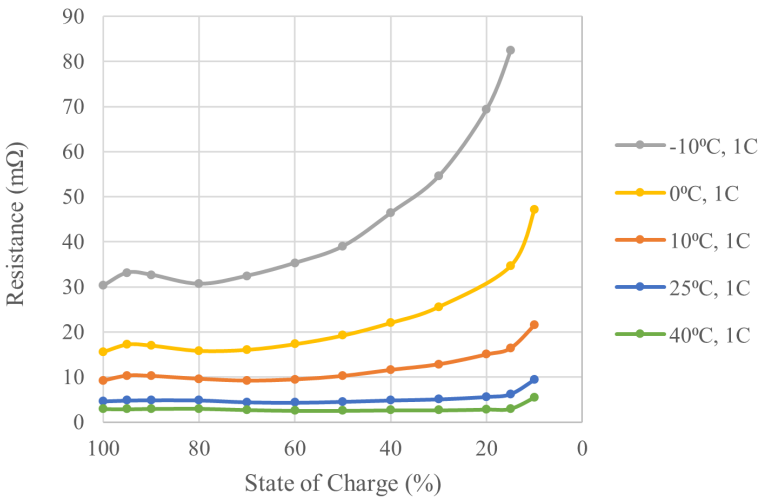


FIGURE 3.4: Discharge resistance vs SOC of a Turnigy graphene 5000 mAh Li-ion battery at different temperatures.

outputs a vector of the estimated SOC (`SOC_Estimated`), estimated V_t (`Vt_Estimated`), and the error between V_t measured and V_t estimated (`Vt_Error`).

The function by default loads the provided `'BatteryModel.mat'` and `'SOC-OCV.mat'` files which are labeled tables. The BatteryModel table contains the SOC, R_0, R_1, C_1, R_2, C_2, and T data in columns 1 to 7, respectively. The SOC ranges from 0% to 100% by intervals of 10%, however, users can vary the intervals as needed. The SOC-OCV table contains the SOC, OCV, and T data in columns 1 to 3, respectively.

```
load 'BatteryModel.mat'; % Load the battery parameters
load 'SOC-OCV.mat'; % Load the SOC-OCV curve
```

The initial SOC is set between 0 and 1 where 0 is 0% and 1 is 100%. This can either be set to the initial SOC of the drive cycle or with a bias to test convergence and robustness. `DeltaT` is the time difference in seconds between each value in the current and X is the initial input and subsequently the system state vector from (3.5).

```
SOC_Init     = 1; % intial SOC
X            = [SOC_Init; 0; 0]; % state space x parameter intializations
DeltaT       = 1; % sample time in seconds
Qn_rated     = 4.81 * 3600; % Ah to Amp-seconds
```

The KF has three tunable parameters: R_x, P_x and Q_x. These will need to be adjusted for each battery either manually or through an optimization algorithm. The AEKF algorithm requires substantially different tuning from the EKF, and may be unstable when the same parameters are used.

```
R_x    = 2.5e-5;
P_x    = [0.025 0 0;
0 0.01 0;
0 0 0.01];
Q_x    = [1.0e-6 0 0;
0 1.0e-5 0;
0 0 1.0e-5];
```

The function allows for temperature dependent data and by default utilizes five different temperatures: 40℃, 25℃, 10℃, 0℃, and -10°. The function interpolates the battery parameters between temperatures at each iteration of the cycle using the output function from `scatteredInterpolant`.

```
F_R0 = scatteredInterpolant(param.T,...
param.SOC,param.R0);
```

```
R0      = F_R0(T,SOC);
```

To use this feature, users will require internal resistance data, and battery parameters for each temperature. If the datasets available to the user is not temperature dependent or the temperature is unknown, the `scatteredInterpolant` function should be replaced with `pchip` or `interp1`.

The SOC-OCV line is curve fitted using the `polyfit` function in a least squares sense. Polynomial differentiation is then completed on the curve to be used to calculate the matrix C. Both the regular curve `SOCOCV` and the differentiated one `dSOCOCV` are then evaluated within the KF loop using `polyval`.

```
SOCOCV  = polyfit(param.SOC,param.OCV,11); % calculate 11th order ...
    polynomial for the SOC-OCV curve
dSOCOCV = polyder(SOCOCV); % derivative of SOc-OCV curve for matrix C
```

```
OCV = polyval(SOCOCV,SOC); % calculate the values of OCV at the given ...
    SOC, using the polynomial SOCOCV
```

```
dOCV = polyval(dSOCOCV, SOC);
C_x    = [dOCV -1 -1];
```

To use the function as an AEKF instead of an EKF, uncomment out the following line in the code.

```
% Q_x = KalmanGain_x * Error_x * KalmanGain_x';
```

3.4 Simulation Example

To illustrate the use of the function, it was evaluated at 40℃, 25℃, 10℃, 0℃, and -10℃ using EKF and the battery data available at [65].

A new Turnigy Graphene 5000mAh 65C cell was tested extensively in a thermal chamber by [65]. SOC-OCV mapping and HPPC tests were performed at 40℃, 25℃, 10℃, 0℃, and -10℃. The tests cover SOC range from 100% to 5% with four different charging and discharging currents at 1, 2, 5 and 10 C-rates. After the characterization, the battery was subjected to driving cycles UDDS, HWFET, LA92, US06 as well as combinations of these cycles. The drive cycles were sampled every 0.1 seconds, and other tests were sampled at a slower or variable rate. [65].

The SOC-OCV curve and battery model parameters were optimized for each temperature and SOC level from 100% to 0% at 10% increments using [71]. The battery data was then re-sampled on a per second rate. The convergence and estimation of the SOC and V_t are highly dependent on the battery model parameters. The initial P, Q and R values in the EKF were manually tuned so the average root mean squared error (RMSE) of all temperatures SOC was less than 5% and V_t was less than 100mV. The results of this tuning can be seen in Fig. 3.5 and Fig. 3.6. The RMSE at 40℃ was 1.78% for SOC and 1mV for V_t for the LA92 drive cycle. Fig. 3.7 illustrates the measured versus estimated V_t and SOC of the LA92 drive cycle at 40°C. Since there is only one set of KF parameters for all the temperatures, the lower temperatures have higher error. One solution to this is to tune different parameters for each temperature similar to the battery parameters.

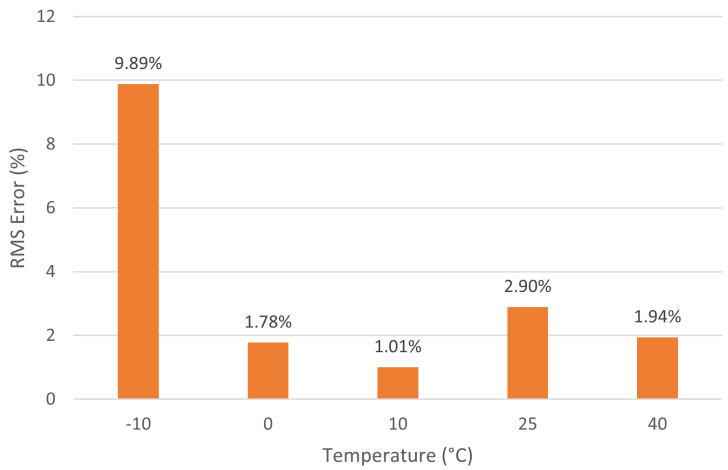


FIGURE 3.5: RMS Error of SOC at different temperatures for LA92 drive cycle.

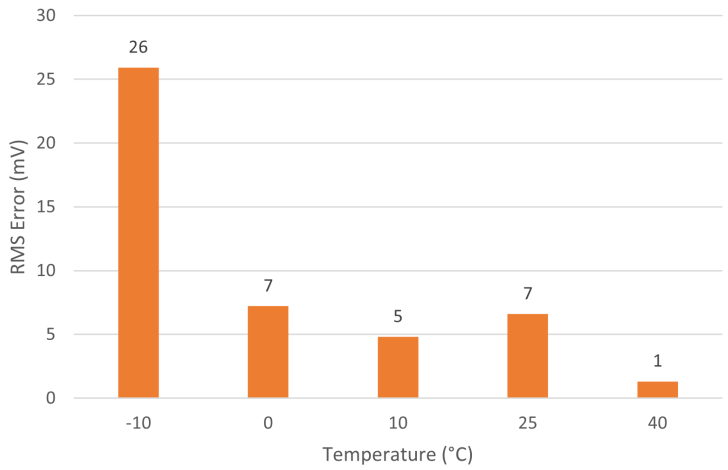


FIGURE 3.6: RMS Error of V_t at different temperatures for LA92 drive cycle.

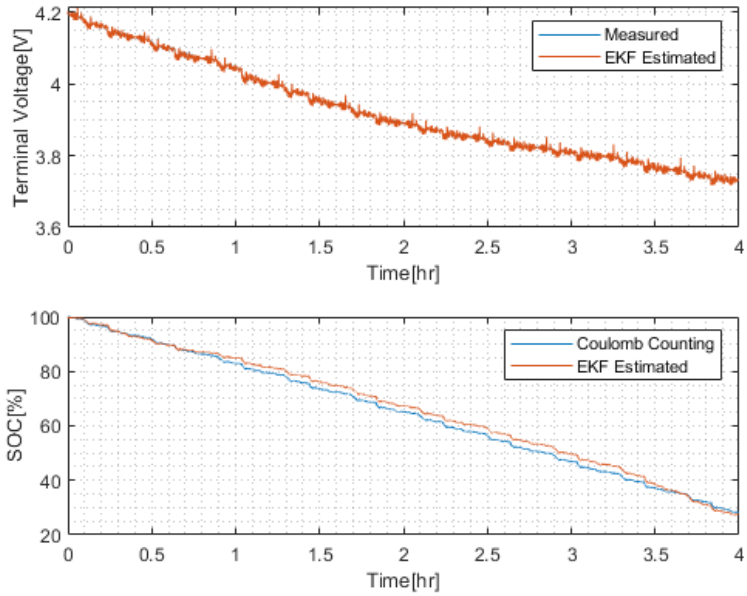


FIGURE 3.7: (a). Measured vs. Estimated Vt at 40℃ for LA92 drive cycle. (b). SOC Coloumb Counting vs. SOC EKF Estimation at 40℃ for LA92 drive cycle.

To test the robustness of the function, the initial SOC was offset by 10%, and the current was offset by +/-0.1A. Fig. 3.8 provides details of the SOC and V_t RMSE given these conditions. The system proves to be robust at 40°C as the RMSE values do not differ from each other drastically. With the 10% initial SOC offset, the system reaches within 5% of the Coulomb counted SOC within 3 minutes (Fig. 3.9). A graph of the +/-0.1A current offset can be seen in Fig. 3.10. The convergence rate and RMSE values of error input can be improved upon with more accurate KF tuning values P, Q, and R.

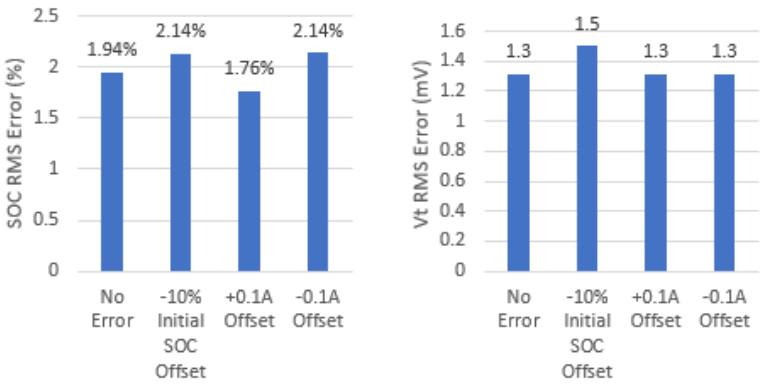


FIGURE 3.8: SOC RMS Error and V_t RMS Error with no error, initial 10% SOC offset, and +/- 0.1A current offset at 40°C for the LA92 drive cycle.

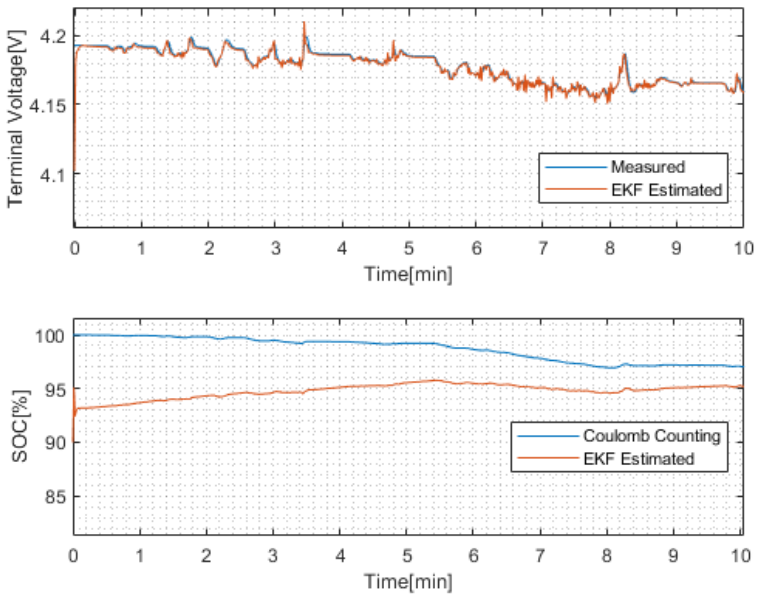


FIGURE 3.9: (a). Measured vs. Estimated Vt with 10% initial SOC offset at 40°C for LA92 drive cycle. (b). SOC Coloumb Counting vs. SOC EKF Estimation with 10% initial SOC offset at 40°C for LA92 drive cycle.

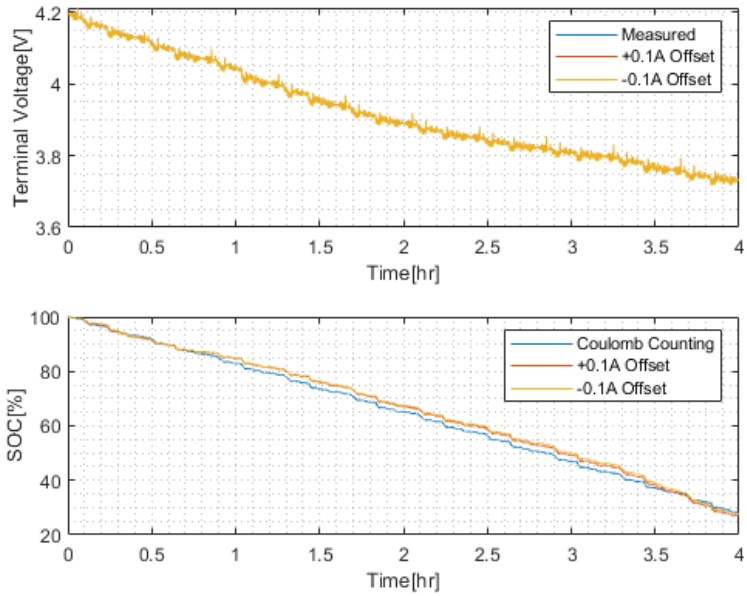


FIGURE 3.10: (a). Measured vs. Estimated Vt with +/-0.1A current offset at 40℃ for LA92 drive cycle. (b). SOC Coloumb Counting vs. SOC EKF Estimation with +/-0.1A current offset at 40℃ for LA92 drive cycle.

3.5 Summary

In this chapter a Kalman filter based SOC estimation MATLAB function was developed for public use. The function is based on a second-order ECM. It allows users to load their specific battery data including the SOC-OCV curves, internal resistance, and the battery model parameters. The function has the flexibility to be used as an EKF or AEKF as well as using battery temperature based data. An example is illustrated with publicly available Turnigy Graphene battery data where less than 2% average RMSE is achieved across the various temperatures. The users of the function have the ability to build on this function or integrate it into a more complex model.

Chapter 4

Battery Cell Tests: Vehicle Model and Design of Experiment

4.1 Introduction

Range estimation provides essential information to the end user on battery state, and is more meaningful compared to reporting SOC alone. This chapter explores a battery centered approach as the first of many steps towards a more accurate range estimation model. A diagram of the range estimation process is given in Fig. 4.1.

A model based on the BEV that the battery cells are extracted from is utilized to generate a power profile with various temperatures, payload mass, HVAC, and driving cycles as input. The power profile is then used as input to test four battery cells in the Arbin cell cycler. All drive cycles tests are run from fully charged or 100% SOC to cut-off capacity. The cut-off capacity is calculated by using a 60 kW power cut-off based on testing of a large PHEV and scaled down. The processed battery test data is then used to train, test, and validate the RNN-based battery model in which the output is used to estimate the distance achieved by the initial driving cycle. The range estimation is taken at the cut-off capacity. The cut-off capacity is equivalent to 0% SOC based on

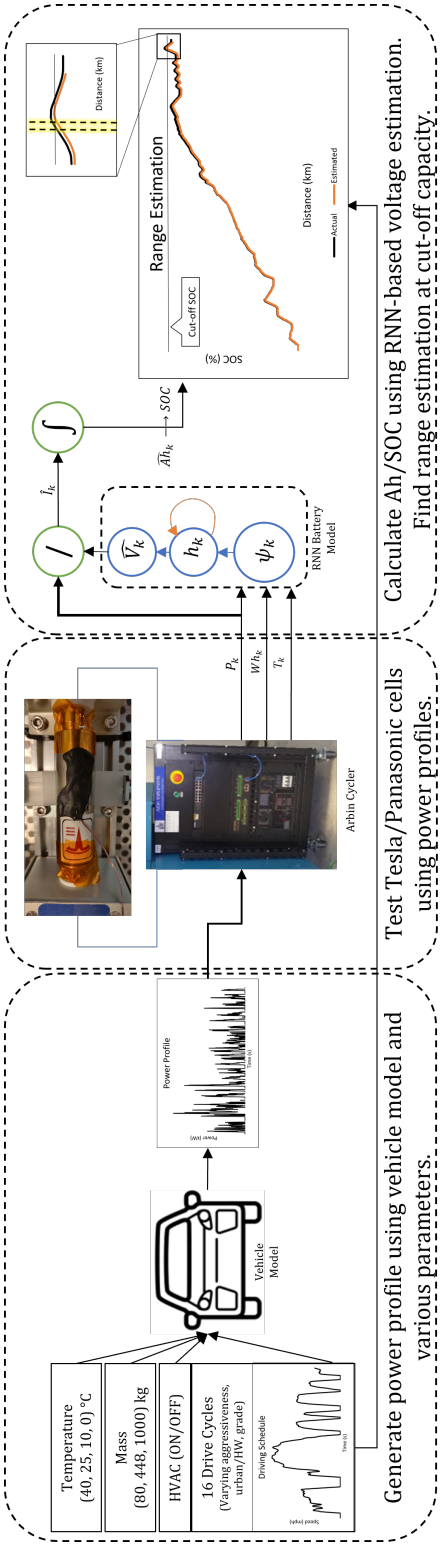


FIGURE 4.1: Overall procedure taken to range estimation in this thesis.

the electrochemical SOC and not the reported SOC to the user. The following chapter will will be an in-depth study of the vehicle model, the setup of the battery tests, and the overall experimental data collection and preparation.

4.2 Electric Vehicle Model

The electric vehicle model is based on the Tesla Model 3 Standard Range using a backward looking BEV model type. The vehicle has a 50 kWh battery pack made up of 2976 li-ion 2170 cells in s96p31 configuration. The EPA combined energy efficiency is listed at 26 kWh/100mi and a range of 220 miles or 354 km [40]. It has a curb weight of 1611 kg and a gross vehicle weight rating (GVWR) of 2060 kg. The GVWR is the maximum operating weight of the vehicle including passengers and cargo [74].

A backward looking model assumes that the vehicle can always meet the drive cycle input, however, does not provide an opportunity to optimize control strategies. The goal of the vehicle model is to produce power profiles for the battery cell testing based on static and dynamic parameters. Static parameters are those that are constant throughout the testing such as the target coefficients and motor efficiencies. Dynamic parameters are those that change per profile: temperature, payload mass, HVAC, and drive cycle. The input to the model is a standard drive cycle given as speed over time.

4.2.1 Model of Tesla Model 3 Drive Train Power

The following section will review the forces acting on the vehicle, utilized by the vehicle model, based on [75], to determine how a speed request calculates to the power requested from the battery pack in a BEV. First, the net forces, F_{net}, acting on a vehicle is given by:

$$F_{net} = F_{wh} - F_g - F_{drag} \tag{4.1}$$

where F_{wh} is the force at the wheels, F_g is the force of gravity, and F_{drag} is the drag force. The total force, F_{acc}, and power, P_{acc}, required to accelerate the vehicle using an equivalent mass, m_{equiv}, given in (4.4), of the vehicle taking into account linear and rotational forces are (4.3) and (4.4).

$$m_{equiv} = m_{payload} + m_{veh} * 1.02 \tag{4.2}$$

$$F_{acc} = m_{equiv} * a_{veh} \tag{4.3}$$

$$P_{acc} = m_{equiv} * a_{veh} * v_{veh} \tag{4.4}$$

where $m_{payload}$ is the total mass of the cargo and passengers, m_{veh} is the mass of the vehicle, r_{wh} is the radius of the wheel, a_{veh} is the acceleration and v_{veh} is the velocity of the vehicle. Since no inertial information is available on the Tesla Model 3, m_{equiv} is calculated based on 2% of the total mass of the vehicle. The force required to move the vehicle up an incline is the gravitational incline force given by (4.5) and the subsequent gravitational power in terms of vehicle speed is (4.6), where g is the gravitational constant.

$$F_g = m_{veh} * g * \sin\theta \tag{4.5}$$

$$P_g = m_{veh} * g * \sin\theta * v_{veh} \tag{4.6}$$

The total drag force acting on a vehicle is a combination of aerodynamic drag, rolling resistance, motor friction, and other small sources. The aerodynamic drag force, F_{aero}, acting on a vehicle can be calculated using (4.7) where A is the rectangular area covered by the front in m^2, ρ is the air density at 1.23 kg/m^3, and C_d is the coefficient of drag. The power required to overcome this force, P_{aero}, is shown in (4.8).

$$F_{aero} = \frac{1}{2}\rho * C_d * A * v_{veh}^2 \tag{4.7}$$

$$P_{aero} = \frac{1}{2}\rho * C_d * A * v_{veh}^3 \tag{4.8}$$

However, to account for all the drag forces, (4.9) based on a third-order polynomial is calculated using coast-down test. The parameters for various vehicles can be found on the EPA website [76]. The parameters used for this model are given in Table 4.1.

TABLE 4.1: Target Coefficients.

Coefficient	**Value**	**Value, metric**
A	41.72 lbf	185.58 N (fully loaded)
B	0.05150 lbf/mph	$0.51N/(m/s)$
C	0.01498 lbf/mph^2	$0.35N/(m/s)^2$

The power required to overcome the total drag, P_{drag}, is given by:

$$P_{drag} = A * v_{veh} + B * v_{veh}^2 + C * v_{veh}^3 \tag{4.9}$$

The total mechanical power, $P_{mech-tot}$, required by the vehicle motor is given by:

$$P_{mech-tot} = P_{acc} + P_g + P_{drag} \tag{4.10}$$

Starting with a speed profile, the model will calculate the acceleration, the model starts by calculating the acceleration (4.11), where Δt is change in time.

$$a_{veh} = \frac{\Delta v_{veh}}{\Delta t} \tag{4.11}$$

In some tests, the grade of the road has also been taken into consideration. A grade ($grade$) pattern, in percent, is created for every point in the speed profile and the power loss or gain, P_{grade}, from such conditions is calculated in (4.12), where $m_{total} = m_{veh} + m_{pass}$ is the total mass of the vehicle with passenger and cargo (m_{pass}). the mechanical power P_{mech} is calculated using the gearbox efficiency, $\eta_{gearbox}$ as (4.13).

$$P_{grade} = \frac{m_{total} * g * grade}{(grade^2 + 1) * v_{veh}} \tag{4.12}$$

$$P_{mech} = \begin{Bmatrix} \frac{P_{acc}+P_{drag}+P_{grade}}{\eta_{gearbox}}, & (P_{acc} + P_{drag}) > 0 \\ (P_{acc} + P_{drag} + P_{grade}) * \eta_{gearbox}, & (P_{acc} + P_{drag}) < 0 \end{Bmatrix} \tag{4.13}$$

The power required simply to move or drive the vehicle, $P_{drive,input}$, taking into account the inverter (η_{inv}) and motor efficiencies (η_{motor}), is given by:

$$P_{drive,input} = \begin{Bmatrix} \frac{P_{mech}}{\eta_{inv}*\eta_{motor}}, & P_{mech} > 0 \\ P_{mech} * \eta_{inv} * \eta_{motor}, & P_{mech} < 0 \end{Bmatrix} \tag{4.14}$$

The peripherals of a BEV include systems such as the lights, windshield wipers, heating, ventilation, and air conditioning (HVAC), etc. The total power required by the peripherals of the vehicle, $P_{accessory}$, is given by (4.15), where T is the temperature in °C. A diagram of the electrical accessory power versus temeprature is given in Fig. 4.2. Similar to the cut-off power, $P_{accessory}$ is scaled down from a series of tests from large PHEV with a resistive heater.

$$P_{accessory} = 250 + \begin{Bmatrix} (20-T)*80, & T <= 20 \\ (20-T)^2*1.4 + (20-T)*5 + 700, & T > 25 \end{Bmatrix} \tag{4.15}$$

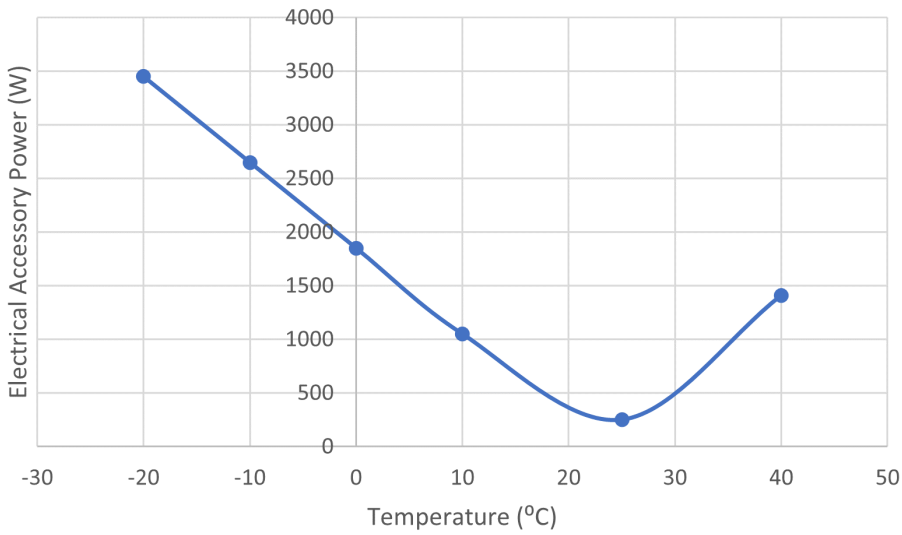


FIGURE 4.2: Electrical accessory power (W) versus temperature (°C) based on equation 4.15 and a heater power of 80 W/°C.

The total power required from the battery is then calculated as (4.16), per second, and stepped down to the cellular level from the pack, where $N_{cells,series}$ is the number of cells in series, and $N_{cells,parallel}$ is the number of cells in parallel in the battery pack. Where $N_{cells,series} = 96$ and $N_{cells,parallel} = 31$. This is the power used to create the

power profile used in the next phase.

$$P_{batt,cell} = \frac{P_{drive,input} + P_{accessory}}{N_{cells,parallel} * N_{cells,series}} \tag{4.16}$$

A sample driving schedule and subsequent power profile is shown in Table 4.2. More details on the driving schedules are available in section 4.3.

TABLE 4.2: Sample speed and power profile scaled down to one cell from the pack.

Time (s)	**Vehicle Speed (km/h)**	**Power (W)**
0	0	0.474
1	0	0.474
2	0	0.474
3	3.2	0.777
4	7.9	1.842
5	13.0	3.264
6	18.2	4.629
7	23.3	6.005
8	27.8	6.514
9	31.5	6.351
10	35.1	6.832
...	...	...

4.2.2 Energy Storage System

The following energy storage system model adapted from what was is presented in [75]. The nominal voltage the battery pack ($V_{nom,pack}$) can provide is calculated using the nominal voltage of one cell ($V_{nom,cell}$) in (4.17). The battery current, I_{batt}, is given in (4.18), where R_{nom} is the nominal resistance.

$$V_{nom,pack} = N_{cells,series} * V_{nom,cell} \tag{4.17}$$

$$I_{batt} = \frac{V_{nom,pack} - \sqrt{V_{nom,pack}^2 - 4 * R_{nom} * (P_{drive,input} + P_{accessory})}}{2 * R_{nom}} \tag{4.18}$$

Equations (4.19) to (4.26) are the theoretical calculation of the driving range each drive cycle will achieve based on the vehicle model outlined above. This model takes into account the power loss at the battery, $P_{batt,loss}$, due to R_{nom} given as:

$$P_{batt,loss} = I_{batt}^2 * R_{nom} \tag{4.19}$$

Therefore, the total power required by the battery is:

$$P_{batt} = (P_{drive,input} + P_{batt,loss} + P_{accessory}) \tag{4.20}$$

The total energy consumption of the drive cycle from 100% SOC to 0% SOC in kWh is given by (4.21). (4.22) then provides the max distance, calculated at each point (per second), as a cumulative number and (4.23) is the energy per kilometer ($E_{use,km}$), where d_{t-1} is the previous time-step distance.

$$E_{total} = \frac{\sum(P_{batt})}{3600 * \Delta t} \tag{4.21}$$

$$d_{max} = v_{veh}\Delta t + d_{t-1} \tag{4.22}$$

$$E_{use,km} = \frac{E_{total}}{d_{max}} \tag{4.23}$$

Next, the max capacity, $C_{max,pack}$, and nominal energy, $E_{nom,pack}$, of the battery pack are given by:

$$C_{max,pack} = N_{cells,parallel} * C_{nom,cell} \tag{4.24}$$

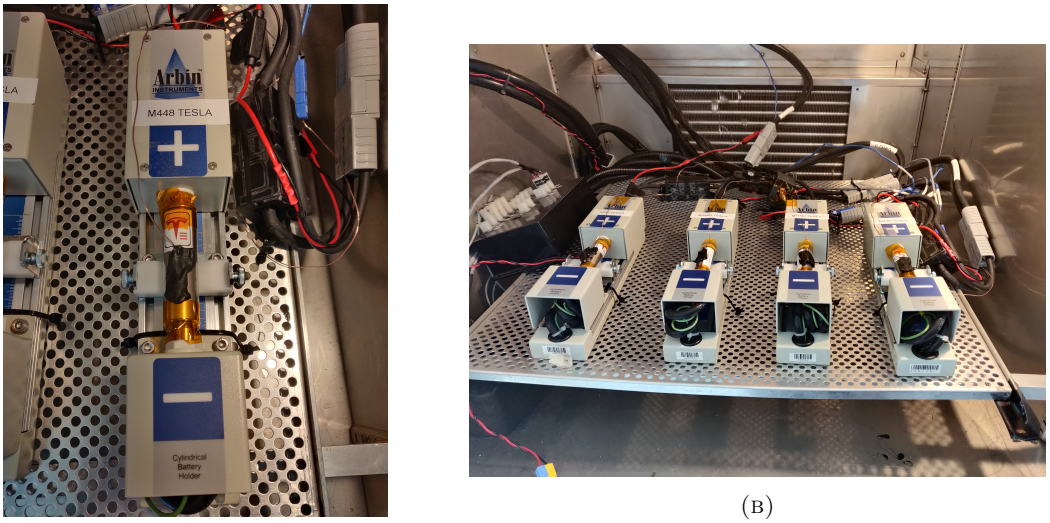
$$E_{nom,pack} = V_{nom,pack} * C_{max,pack} \tag{4.25}$$

where $C_{nom,cell}$ is the nominal capacity per cell. Finally, predicted range based on the nominal pack energy, usable range of SOC ($SOC_{usuable,range}$), and energy consumption per km.

$$Range = \frac{E_{nom,pack} * SOC_{usuable,range}}{E_{use,km}} \tag{4.26}$$

4.3 Experimental Measurement and Preparation of Data

The battery cells tested were four Tesla/Panasonic 2170, nickel cobalt aluminium (NCA) chemistry li-ion batteries with a nominal capacity of 4.5 Ah (Fig. 4.3a and Fig. 4.3b) and other measured parameters listed in Table 4.3. Since there is no available datasheet for the tested cells, all parameters available were taken from experimental measurements. The tests were conducted in an Envirotronics SH16C thermal chamber controlled by the Arbin Cycler battery tester, pictured in Fig. 4.5 along with a test bench schematic in Fig. 4.4. The specifications for the thermal chamber are in Table 4.4 and the specifications for the battery tester are in Table 4.5 [77].



(A) (B)

FIGURE 4.3: (a) Tesla/Panasonic 2170 4.5 Ah cell in Arbin holder with T-Type temperature sensor. (b) Four Tesla/Panasonic cells in Arbin holders placed inside Envirotronics thermal chamber.

TABLE 4.3: Tesla/Panasonic 2170 4.5 Ah li-ion battery cell measured parameters.

Manufacturer, Model	**Panasonic/Tesla 2170**
Chemistry	NCA
Size	21 mm x 70 mm
Measured Capacity	4.5 Ah
Min and Max Voltage	2.8 V & 4.2 V
Mass	60 g

TABLE 4.4: Envirotronics thermal chamber specifications utilized for battery testing.

Manufacturer/Model	**Envirotronics SH16C Thermal Chamber**
Size	16 ft^3
Temperature Range	-30°C to 177°C
Includes Humidity Control	Yes

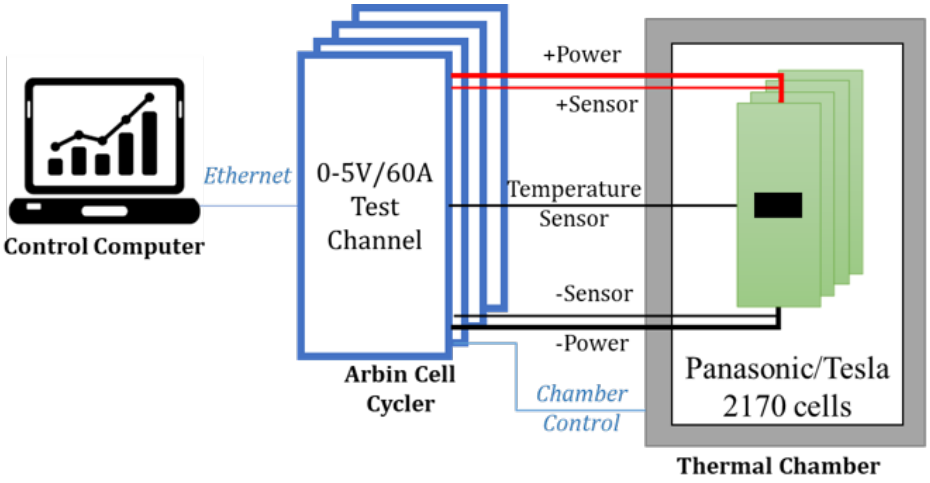


FIGURE 4.4: Test bench schematic of testing Tesla/Panasonic batteries.

FIGURE 4.5: Arbin cell cycles and Environtronics thermal chamber in battery lab at the McMaster Automotive Resource Center.

TABLE 4.5: Arbin cell cycler specifications utilized for battery testing.

Manufacturer/Model	**Arbin Cell Cycler, LBT21084-0∼5V-60/5/0.5/0.02A-8CH-208V3P**
Voltage	0V to 5V
Current	60A per channel
# of Channels	8
Parallel Operation	2 - 8 sequential channels can be operated in parallel
Input Impedance	50 mΩ
Current Range	+/- 60A, 5A, 500mA, 20mA
Control Accuracy	+/- 24mA, 2mA, 200µA, 8µA & +/- 2mV
Max Command Rate	5ms
Max System log rate	2000 samples per second
Temperature Sensing	16 channels, type T thermocouples
Control Software	Arbin MITS 8.0

4.3.1 Cell Characterization

Each cell was subjected to characterization tests and drive cycles tests at four temperatures: 40°C, 25°C, 10°C, and 0°C. To prevent rapid aging and provide accurate characterization data, the tests were completed in order of highest temperature to lowest and the characterization testing before the drive cycles.

The testing at each temperature was split into two categories: characterization and drive cycles. Characterization tests consisted of a 40°C, C/20 discharge test at the beginning of each temperature set to track ageing of the battery cells through the whole testing life cycle. The remainder of the test included C/3 discharge, HPPC, C/20 discharge and charge, C/2 discharge, 1C discharge, and 5C discharge. The HPPC tests results are then used to determine the cut-off capacity used for the drive cycles at the specific temperature.

Fig. 4.6 to 4.8 display notable characterization results for cell 1, while Fig. 4.6 illustrates the effects of temperature on the capacity of the battery. The change in internal resistance of the battery can be seen Fig. 4.7. The internal resistance is typically

high from 100% - 80% SOC and 20% to 0% SOC. The 50% SOC discharge resistance changes from 20 $m\Omega$ at 40°C to 115 $m\Omega$ at -10°C. The capacity of the cell is also effected by the C-rate as seen in Fig. 4.8. The higher the C-rate, the lower the capacity.

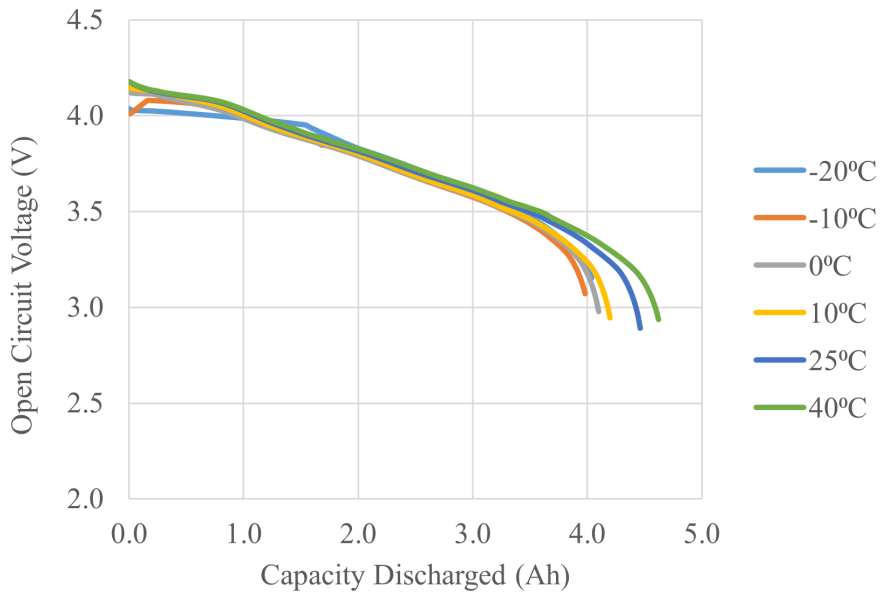


FIGURE 4.6: Open circuit voltage (OCV) vs. discharge capacity for various temperatures and cell 1 (m80).

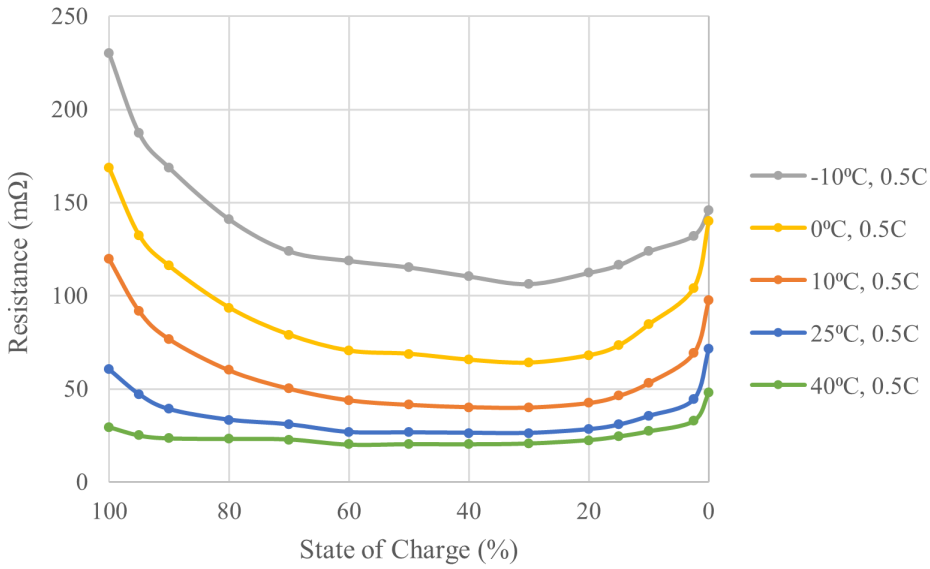


FIGURE 4.7: Discharge resistance vs. SOC at various temperatures and 1/2C for cell 1 (m80).

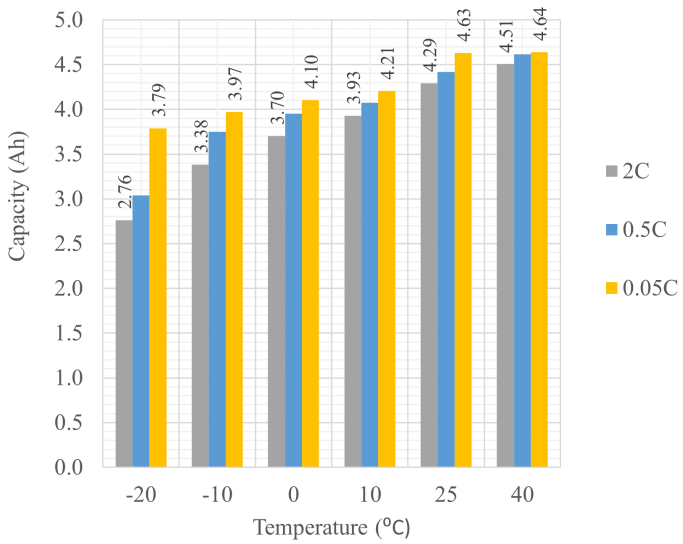


FIGURE 4.8: Capacity vs. temperature at various C-rates for cell 1 (m80).

4.3.2 Drive Cycles

Sixteen drive cycles are tested on each cell at differing payload mass and HVAC parameters. The first four drive cycles are the standard Urban Dynamics Drive Cycle (UDDS), Highway Fuel Economy Cycle (HWFET), California Light-Duty Unified Cycle (LA92), and the Supplemental Federal Test Procedure (US06). The next eight are the first four cycles split into segments where speed is 0 and the segments shuffled to create Reordered 1 through 8. The last four cycles are HWCUST1, HWCUST2, HWGRADE1, and HWGRADE2. These are custom drive cycles where the speed is randomized between a minimum and maximum throughout the cycle. The HWGRADE cycles also increase and decrease in grade and final cycles simulate driving through a mountain pass. Each drive cycle is repeated until the cell reaches the cut-off capacity. A speed versus time graph for the UDDS, HWFET, LA92, US06, HWCUST1, HWCUST2, HWGRADE1, and HWGRADE2 as well as grade versus time for HWGRADE1, and HWGRADE2 is presented in Fig. 4.9. Between each discharge test, the temperature is changed to 25°C for charging to ensure the cells do not age rapidly.

A list of cell parameters and limits utilized in the vehicle model for power profile generation from the drive cycles are listed in Table 4.6.

TABLE 4.6: Cell parameters and limits utilized in power profile generation.

Parameter	**Value/Limit**
Energy	16 Wh
Minimum voltage (V_{min})	2.5 V
Maximum voltage (V_{max})	4.2 V
Discharge current limit	40 A
Charge current	40 A

Each batteries power profile is set with various payload mass and HVAC conditions as given in Table (4.7) to generate four unique datasets. Each payload mass is the

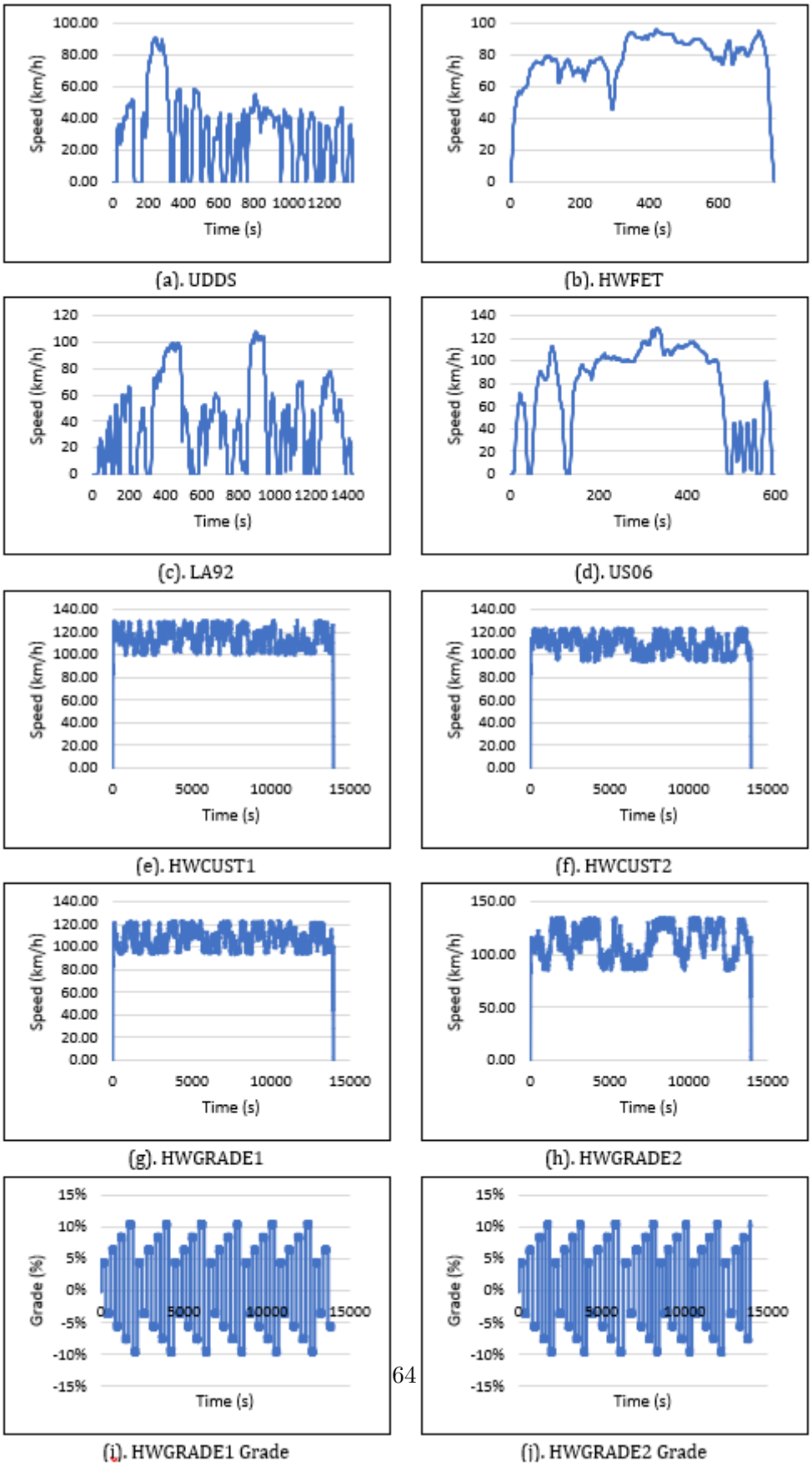


FIGURE 4.9: Speed and grade profiles of select drive cycles.

additional amount to the curb mass of the vehicle. One person driving is represented by 80 kg, 448 kg for max GVWR, and 1000 kg for towing.

TABLE 4.7: Battery/Cell testing parameters and referenced names for differentiation.

Cell	Payload Mass (kg)	HVAC Included	Referenced Name
Cell 1	80 kg	Yes	m80
Cell 2	448 kg	Yes	m448
Cell 3	448 kg	No	m448-N
Cell 4	1000 kg	Yes	m1000

To illustrate the difference in power requirements for the four battery testing conditions listed in Table 4.7, an overlay of the LA92 drive cycle for the first 150 seconds at 40°C for all four batteries is given in Fig. 4.10.

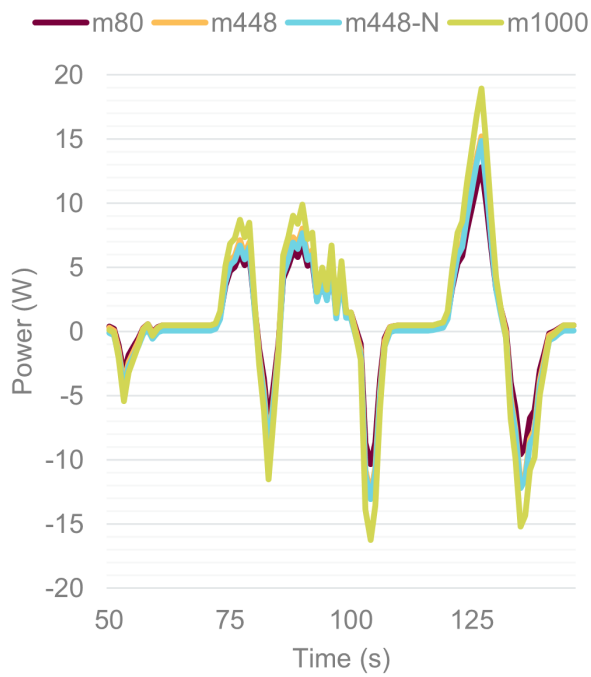


FIGURE 4.10: Power profile of LA92 drive cycle from 0 to 150 seconds at 40°C.

A sample of power profiles from 40°C and m80 is available in Fig. 4.11 used by the

Arbin cell cycler as input. The standard four cycles (UDDS, HWFET, LA92, and US06), HWCUST1, and HWGRADE1, pictured, are the full cycle repeated by the tester until cut-off capacity, while Reordered 1 and 2, pictured, are a portion of the full cycle that is repeated until cut-off capacity.

4.3.3 Cut-off Capacity Calculation

For each temperature, a cut-off capacity was calculated using the HPPC test data and a discharge power cut-off of 60 kW based on [72]. The power cut-off simulates the battery pack having insufficient power to continue powering the vehicle. The calculated capacity is then used as the stopping point in each temperatures drive cycles test. Since each cell ages slightly differently, the cut-off capacity is calculated separately for each cell. The HPPC test is conducted at SOC 100%, 95%, 90% to 10% at 10% interval and finally at 5% using discharge C-rates of 0.5C, 1C, 2C, and 3C. The Ah discharge (A_{dis}), OCV (V_{OC}), discharge pulse voltage, discharge pulse resistance (R_{dis}) are calculated at each SOC interval and used to calculate the cut-off capacity ($Ah_{cut-off}$).

A list of cell parameters and limits utilized in calculating the cut-off capacity are listed in Table 4.8.

TABLE 4.8: Cell parameters and limits utilized in cut-off capacity calculations.

Parameter	**Value/Limit**
Number of cells in pack	2976 (s96p31)
Minimum voltage (V_{min})	2.8 V
Maximum voltage (V_{max})	4.2 V
1C rate definition	4.45 A
Discharge power cut off	60 kW

First, the discharge power capability, $P_{dis,cell}$, is calculated

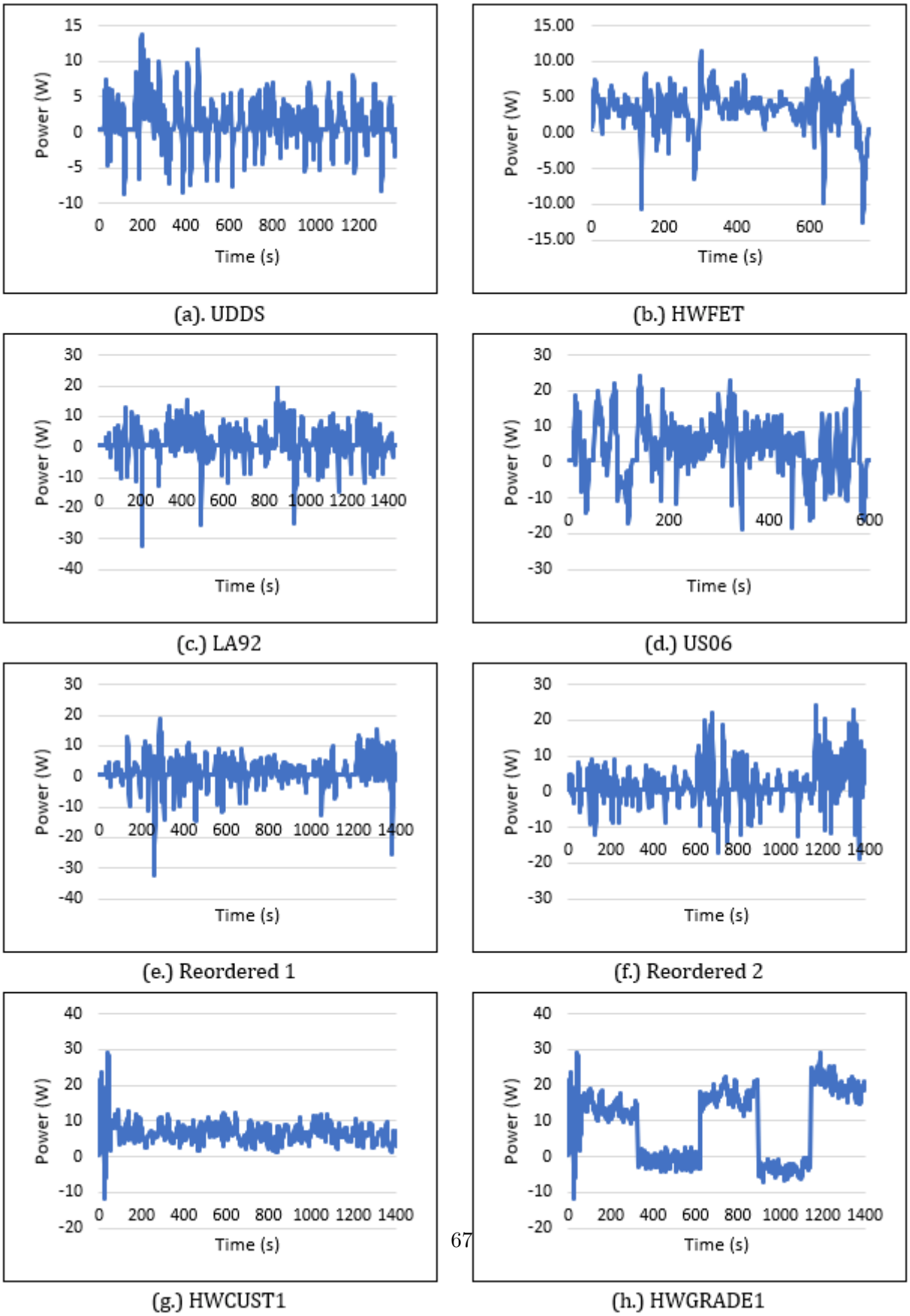


FIGURE 4.11: Power profiles of select drive cycles for 40°C and m80.

$$P_{dis,cell}(i) = \frac{V_{OC}(i) - V_{min}}{min(R_{dis}(i))} * V_{min} \tag{4.27}$$

where V_{min} is the minimum voltage, and i is the current SOC interval. Next using the number of cells in a pack, $N_{cells,pack}$, given in (4.28), the power discharge is scaled up to the pack level in (4.29), given by $P_{dis,pack}$.

$$N_{cells,pack} = N_{parallel} * N_{series} \tag{4.28}$$

$$P_{dis,pack}(i) = P_{dis,cell}(i) * N_{cells,pack} \tag{4.29}$$

If $P_{dis,pack}$ at the previous SOC interval, $i-1$, is greater than $P_{cut-off}$ but the $P_{dis,pack}$ at the current SOC interval. i is less than $P_{cut-off}$, then Ah_{dis} is calculated as follows:

$$Ah_{dis} = \frac{(P_{cut-off} - P_{dis,pack}(i)) * (Ah_{dis}(i-1) - Ah_{dis}(i))}{P_{dis,pack}(i-1) - P_{dis,pack}(i)} + Ah_{dis}(i) \tag{4.30}$$

otherwise, it is 0. Finally the max Ah_{dis} is the cut-off Ah.

$$Ah_{cut-off} = max(Ah_{dis}, 0) \tag{4.31}$$

The pack discharge power capability for temperatures 40°C to 0°C is given in Fig. 4.12. The power capability shown is calculated from the m80 cell HPPC data at each temperature and scaled up to the pack.

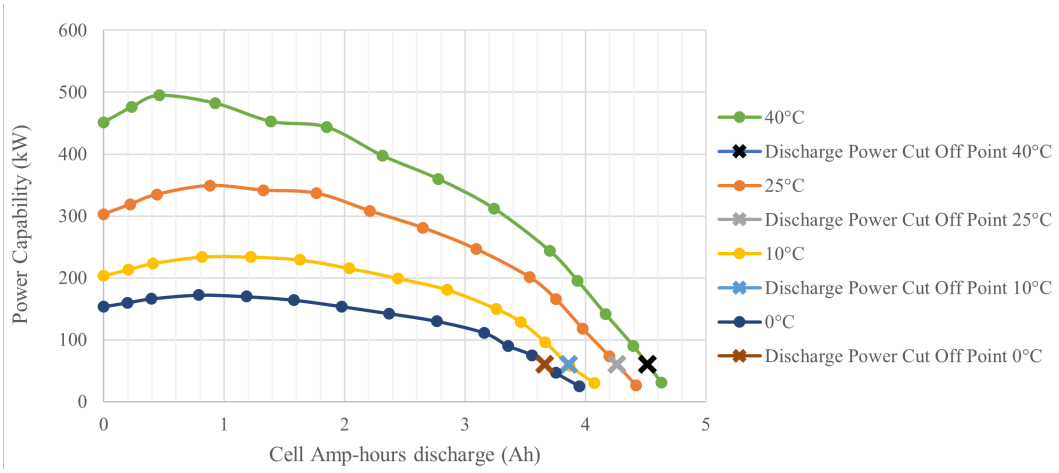


FIGURE 4.12: Discharge power capability and discharge power cut-off point for m80 cell and temperatures 40°C to 0°C.

The cut-off capacity can also be expressed in terms of the more well-known SOC (Fig. 4.13) using the C/20, 40°C, discharge ageing test at each temperature set.

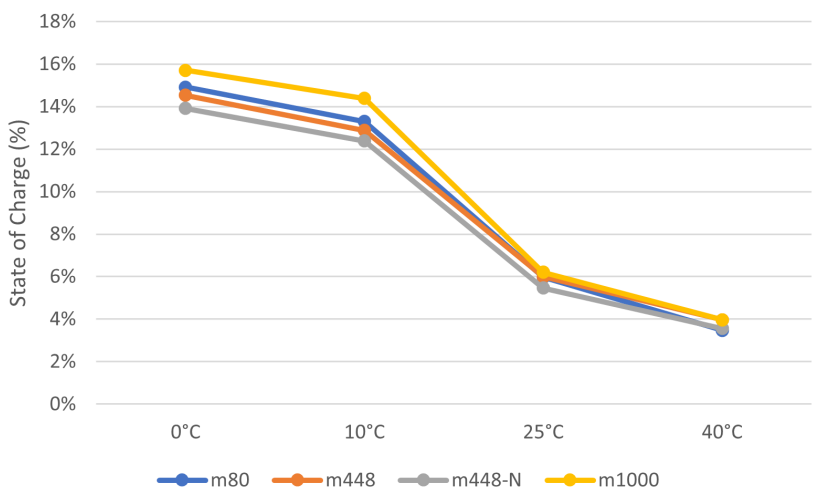


FIGURE 4.13: Cut-off SOC of for all cells and temperature. The SOC is calculated using the C/20, 40°C, discharge ageing test at each temperature set.

4.3.4 Data Preparation

The exported data from the Arbin cell cycler went through two main steps in order to train and test the network. The first step consisted of preparing the data for general

use as part of a larger dataset by splitting the data into charge and discharge cycles per channel and temperature. The second set was taking the drive cycle data and converting it into training, testing, and validation data. All drive cycle data was normalized using (4.32) as part of standard practice for ANNs to improve training speed and accuracy [35], where $Min = [0, -65, -5, 2.5]$ and $Max = [18, 55, 55, 4.5]$ for energy, power, battery surface temperature, and voltage ($\mathbf{\Psi} = [Wh,\ P,\ T],\ V$), respectively. The dataset used for this chapter is part of a larger dataset ranging from 40°C to -20°C.

$$Normalized\ Data = \frac{Data - Min}{Max - Min} \tag{4.32}$$

The training data was then concatenated together into one file and divided into files of equal lengths to avoid extra padding of data from MATLAB. The testing data was divided by drive cycle for easy metric evaluation. The full details of each dataset are in Tables 4.9 and Table 4.10. Standard practice for splitting of dataset is typically 2/3 training, 1/3 testing. The purpose of the classic dataset is to validate whether a large dataset with multiple options in mass, HVAC, grade, and aggressive highway cycles are necessary for a robust estimation.

TABLE 4.9: Size and division of drive cycles of whole dataset. Cells 1 through 4 refer to m80, m488, m448-N, and m1000, respectively.

Use	**Drive Cycles (Cells 1 - 4)**	**Size (pts)**	**Size(days)**
Training	REORDERED 1 - 8, HWCUST1, HWGRADE1	3,753,078	43.4
Testing	UDDS, HWFET, LA92, US06 HWCUST2, HWGRADE2	1,954,754	22.6
Validation	REORDERED1	470,789	5.5

Table 4.10: Size and division of drive cycles of classical dataset. Cell 1 refers to m80. A classical dataset contains various temperatures but not mass, HVAC, aggressive drive cycles, or grade parameters.

Use	**Drive Cycles (Cells 1)**	**Size (pts)**	**Size (days)**
Training	REORDERED 1 - 8	899,388	10.4
Testing	UDDDS, HWFET, LA92, US06	419,414	4.9
Validation	REORDERED1	137,228	1.6

4.4 Summary

In this chapter, an electric vehicle model and battery experiment process for testing four Tesla/Panasonic 2170, 4.5 Ah battery cells are provided. The backward-type vehicle model, based on the Tesla Model 3 Standard Range, generates the power profile utilized by the Arbin cell cycler for varying drive cycle tests. The battery cells are subjected to 7 standard characterizations, and 16 drive cycle tests are varying temperatures. Each cell is tested with unique payload mass and HVAC parameters to create four different datasets.

Chapter 5

Range Estimation of Electric Vehicles using a Recurrent Neural Network-based Voltage Estimation

5.1 Introduction

Battery modelling research has made several strides in the past few year. The exploration of machine learning, specifically ANNs, as a viable option for battery terminal voltage estimation is now possible. Voltage estimation has several applications not only in BEVs but drones, electric aircrafts, and more. One such application is estimating the range at the end of a drive. ANNs require a high volume of data for training and typically have a high computational requirement. The following section will cover a voltage estimation model using a recurrent neural network based on the script provided in [69]. A comparison of training on a large computing resource versus a personal computer (PC) is provided along with a comparison of networks trained on single battery cell dataset

versus a multi-battery cell dataset. Standard procedure at the Battery Lab in MARC is to train and test ANNs on data from a single battery without the HWCUST/HW-GRADE cycles and thus this dataset is referred to as the classic dataset. One novelty of the work presented in chapter 4 is the outcome of a multi-battery dataset which will be referred to as the whole dataset. The voltage estimation network will be the basis for determining the range at the end of the drive, when the battery pack can no longer power the vehicle at the requested speed. Results of end drive range estimation is provided and compared with other range estimation methods in literature.

5.2 Recurrent Neural Network Voltage Estimation Model

Recurrent neural networks (RNNs) are a type of ANN appropriate for time-dependent or sequential datasets applications. Prominent examples of RNNs include speech recognition [78], language understanding [79], and acoustic modelling [80]. RNNs have a very similar structure to FNNs except the loop or recurrence of data in the hidden layer as shown by the orange arrow in Fig. 5.1.

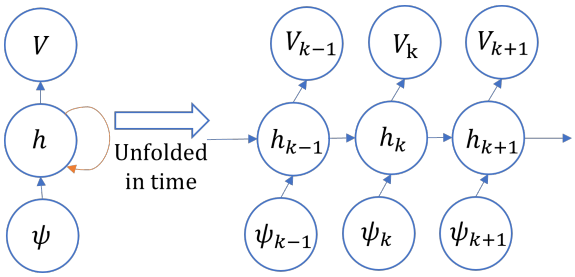


FIGURE 5.1: Architecture of RNN (left) and architecture of RNN unfolded in time (right).

RNNs compute an output vector $\boldsymbol{y}_k$ (5.1) given an input vector $\boldsymbol{\Psi}_k$ through a hidden vector sequence $\boldsymbol{h}_k$ (5.2), where H is the hidden layer function and, $\boldsymbol{W}$ and b are the weights and bias matrices for each gate ($\boldsymbol{W}_{hi}$ is between the hidden-input gate) [82].

$$\boldsymbol{h}_k = H(\boldsymbol{W}_{\Psi h}\boldsymbol{\Psi}_k + \boldsymbol{W}_{hh}\boldsymbol{h}_{k-1} + b_h) \tag{5.1}$$

$$\boldsymbol{y}_k = \boldsymbol{W}_{hy}\boldsymbol{h}_k + b_y \tag{5.2}$$

However, RNNs are subject to the exploding and vanishing gradient problems. An exploding gradient is an exponential growth of the long term components more than the short ones and a vanishing gradient is the opposing [81]. These issues can be resolved with specialized cells such as gated-recurrent unit (GRU), long short-term memory (LSTM), and bi-directional long short-term memory (BiLSTM).

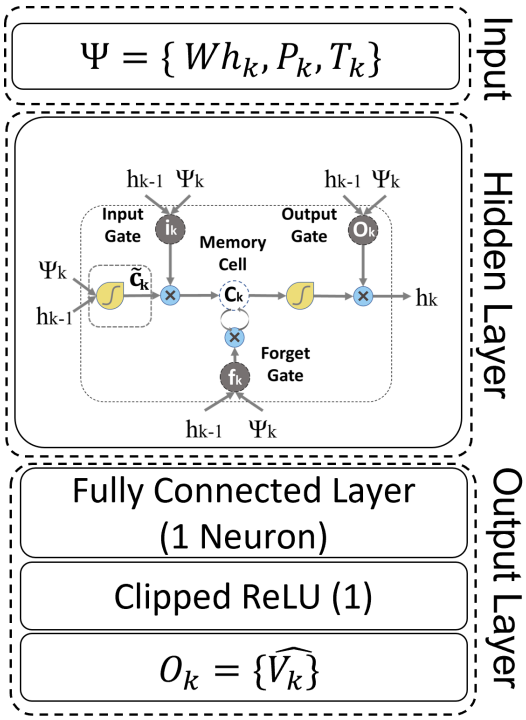


FIGURE 5.2: RNN network layers with LSTM cell. The network layers consist an input layer taking sequence data, hidden layer (LSTM), an output layer consisting of a fully connected layer, a clipped ReLU function, and finally the output, voltage.

LSTM is a type of ANN that is an RNN with an LSTM cell typically used for time-series forecasting. The LSTM-RNN architecture show in Fig 5.2 consists of an input layer, N hidden layers, and an output layer. When applying the LSTM model towards battery voltage estimation, the input vector is given by $\boldsymbol{\Psi}_k = [Wh(k), P(k), T(k)]$ where $Wh(k)$ is the battery's energy in watt-hour, $P(k)$ is the power in watts, and $T(k)$ is the battery surface temperature in °C at training time-step k. For voltage estimation, the output is given by $\hat{V}_k = \boldsymbol{W}_h \boldsymbol{y} \boldsymbol{h}_k + \boldsymbol{b}_y$. An LSTM unit consists of a cell (c_k), an input gate (i_k), forget gate (f_k), and output gate (o_k) [82]. The hidden layer function in an LSTM is given as:

$$i_k = \eta(\boldsymbol{W}_{\Psi i}\boldsymbol{\Psi}_k + \boldsymbol{W}_{hi}\boldsymbol{h}_{k-1} + b_i) \tag{5.3}$$

$$f_k = \eta(\boldsymbol{W}_{\Psi f}\boldsymbol{\Psi}_k + \boldsymbol{W}_{hf}\boldsymbol{h}_{k-1} + b_f) \tag{5.4}$$

$$c_k = f_k c_{k-1} + i_k \tan h(\boldsymbol{W}_{\Psi c}\boldsymbol{\Psi}_k + \boldsymbol{W}_{hc}h_{k-1} + b_c) \tag{5.5}$$

$$o_k = \eta(\boldsymbol{W}_{\Psi o}\boldsymbol{\Psi}_k + \boldsymbol{W}_{ho}\boldsymbol{h}_{k-1} + b_o) \tag{5.6}$$

$$\boldsymbol{h}_k = o_k clippedReLu(c_k) \tag{5.7}$$

where η is the activation function which in this case is a sigmoid function [82]. The non-linearity of the system is captured using the Clipped Rectified Linear Units (Clipped ReLU), given by:

$$f(x) = \begin{cases} 0 & x < 0 \\ x & 0 \leq x < ceiling \\ ceiling & x \geq ceiling \end{cases} \tag{5.8}$$

At each time-step, the ideal voltage is compared to the estimated voltage and the error signal is calculated by:

$$e(k) = V_k - \hat{V}_k \tag{5.9}$$

where V_k is the ideal or measured voltage. This error is then used to compute the loss function (5.10), where N is the length of the sequence, which provides a good understanding of the network.

$$L = \sum_{k=0}^{N} \frac{1}{2}(e(k))^2 \tag{5.10}$$

This process of the training data fed into the network to the voltage estimation and loss calculated at each time step is the forward pass. A full epoch, ϵ, consists of a forward and backward pass for training. The backward pass consists of sending the overall loss backward through the network to update the weights. An optimization method called *Adam* [83] is used for this update based on the gradient of the loss function. The learning rate, a parameter of the *Adam* optimizer, determines the step size of each iteration. Training continues until a convergence criteria has been met and the loss function minimized. During validation, only the forward pass is computed since the weights and biases of the network have been learned.

The size of the network is determined by the number of hidden layers and hidden units. The number of learnable parameters in an LSTM-RNN can be calculated using (5.11) given one layer.

$$LP = 4(i * HU + HU * HU + HU) + HU + o \tag{5.11}$$

where i is the input, o is the output, and HU is the number of hidden units.

To evaluate the voltage estimation performance, a few different performance metrics are calculated. These include the RMSE, mean absolute error (MAE), max error (MAX), and standard deviation of errors (STDDEV).

5.3 Driving Range Estimation

After the voltage estimation network has been trained, it is used to predict the voltage for an upcoming driving route. The current is estimated using the measured power and estimated voltage in (5.12). The current is then integrated to calculate the estimated Ah, $\hat{A}h_k$, in (5.13).

$$\hat{I}_k = \frac{P_k}{\hat{V}_k} \tag{5.12}$$

$$\hat{A}h_k = \int_0^N \hat{I}_k \tag{5.13}$$

The estimated Ah is graphed against the distance of the drive cycle and compared to the actual Ah. The end of the driving route is given by the $Ah_{cut-off}$ where the

difference in distance is calculated. The drive cycle end range estimation is illustrated in Fig. 5.3.

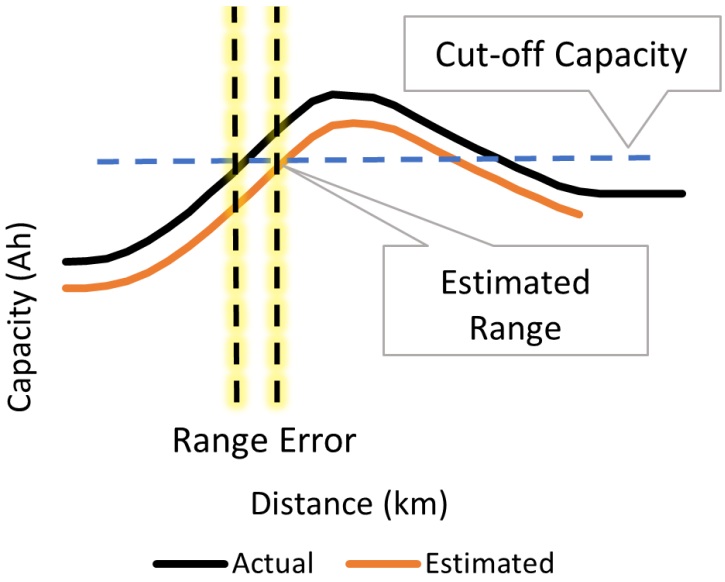


FIGURE 5.3: Range estimation method at end of drive cycle and cut-off capacity.

5.4 LSMT-RNN Training on SHARCNET vs. Personal Computer

All LSTM-RNN network training was completed on SHARCNET (Shared Hierarchical Academic Research Computing NETwork), a member of the Compute Canada network. It is a network of 19 Ontario institutes providing advanced computing resources and access to over 50,000 CPU cores and more than 200 GPUs to researchers. There are several clusters available for use, however, the main ones used for this work were Graham, Cedar, and Narval clusters. Listed are specifications for the main GPUs/CPUs used on each cluster as well as the authors personal computer (assigned MARC computer, PC) used in testing/training.

- **PC**: Intel i7-8700 CPU @3.20 GHz, 32 GB RAM, and 6 cores.

- **Graham cluster**: NVIDIA T4 Turing GPU with 16 GB of memory and Intel Zeon Gold 6238 Cascade Lake @2.10 GHz, and 44 cores.

- **Cedar cluster**: NVIDIA V100 Volta with 32 G HBM2 memory, and Intel Silver 4216 Cascade Lake @2.1 GHz, and 32 cores.

- **Narval cluster**: AMD Rome 7432 @2.4 GHZ, and 64 cores.

SHARCNET allowed the training of multiple networks simultaneously while also providing speed-up of the training with the use of GPUs. Fig. 5.4 provides a timeline of the 10 trainings of the whole dataset. The orange bars indicate the starting and end of each training. The blue bar indicates the time taken to run all 10 trainings on the PC back to back. This number was calculated by estimating the time per epochs based on a sample of 10 epochs run on the PC and multiplying it by the total number of epochs run for all 10 trainings. If the 10 trainings were started on Oct 29th, the trainings would not be completed till Dec 4th.

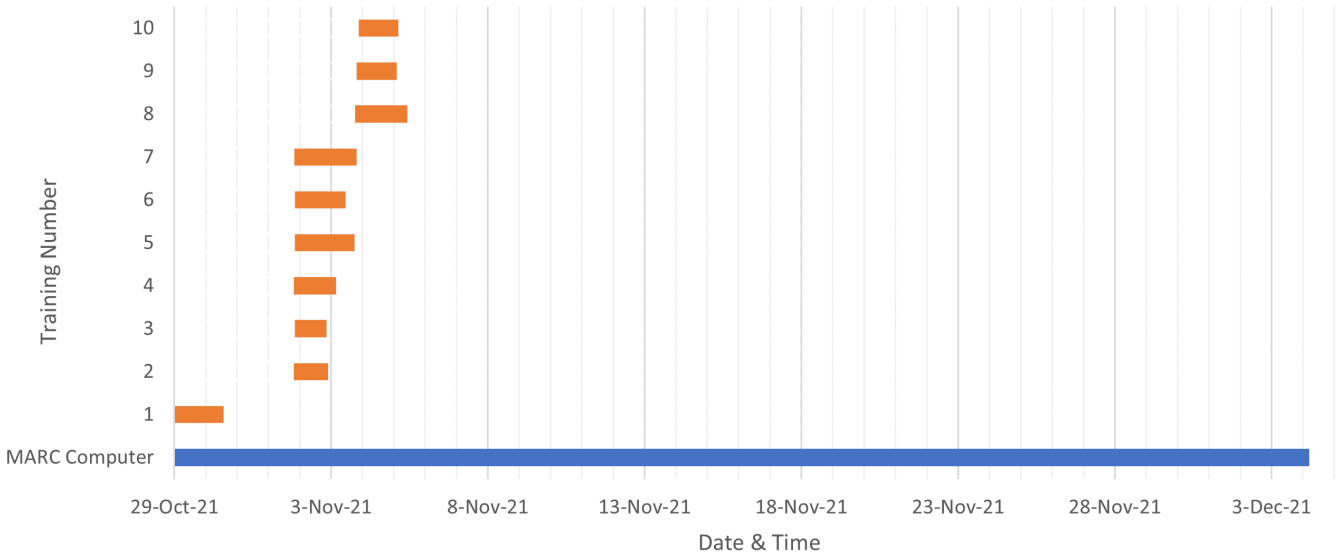


FIGURE 5.4: Whole dataset voltage estimation training timeline of SHARCNET Graham Cluster (GPU) vs. PC. 10 trainings were completed on SHARCNET between Oct. 29th and Nov. 5th. The equivalent of this training on a regular PC would take until Dec. 4th.

5.5 Voltage Estimation Results

5.5.1 Learnable Parameters and Hyper-Parameters

As mentioned earlier, the inputs to the LSTM-RNN are Wh, P, and T, and V is the output. After some testing, the network hyper-parameters of the LSTM-RNN listed in Table 5.1 were chosen. The default values given by MATLAB were used for all other hyper-parameters. Before training to find the best network, the network was tested at six different learnable parameters (LP), given in Table 5.1, to see if there was a network size that was better suited than others. Each network is trained until either the max number of epochs, 100,000, is reached or the validation patience 4,000 is reached. The validation patience is the number of epochs that the algorithm will run trying to improve the performance of the network before the training is considered complete. The performance of the network is based on the loss function previously outlined. The learning rate is not

a static parameter, instead an initial learn rate is set that is then updated based on the learn rate drop period and learn rate drop factor. The drop period specifies the number of epochs passes before the learn rate is updated and the drop factor is the amount by which the previous learn rate is multiplied. In Matlab, the size of the memory for the LSTM is based on the number of hidden units value, which in this instance is 10.

Table 5.1: LSTM Parameters.

Hyper-parameters	**Values**	**HU**	**LP**
Number of Features	3	1	22
Max Epochs	100,000	3	88
Patience	4,000	5	186
Initial Learn Rate	0.01	7	316
Learn Rate Drop Period	2,000	10	571
Learn Rate Drop Factor	0.85	25	2926

Each network training is initialized with random weights and biases. Due to this randomization and the chance of falling into a local minima, multiple trainings are required. Four trainings for each learnable parameter were run on SHARCNET using the classic dataset for faster results due to its size. The classic dataset consists of only m80 (cell 1) data at temperatures 40°C, 25°C, 10°C, and 0°C data. Drive cycles reordered 1 through 8 are used for training, drive cycles UDDS, HWFET, LA92, and US06 are used for testing, and drive cycle reordered 1 for validation. As indicated in Fig. 5.5, there was no network size that was more well suited than the others. LP #24 has the highest RMSE of 30 mV, proving to be too small of a network size. The remainder LPs are within 1-2 mV RMSE of each other. LP #571 or 10 HU was chosen for the remainder of the testing as it is the largest size tested that can fit on a microcontroller in a BMS.

5.5.2 Whole vs. Classic Dataset Comparison

Ten trainings of the classic and whole datasets were run to find the best network for voltage estimation. In comparison to the classic dataset, the whole dataset utilizes data

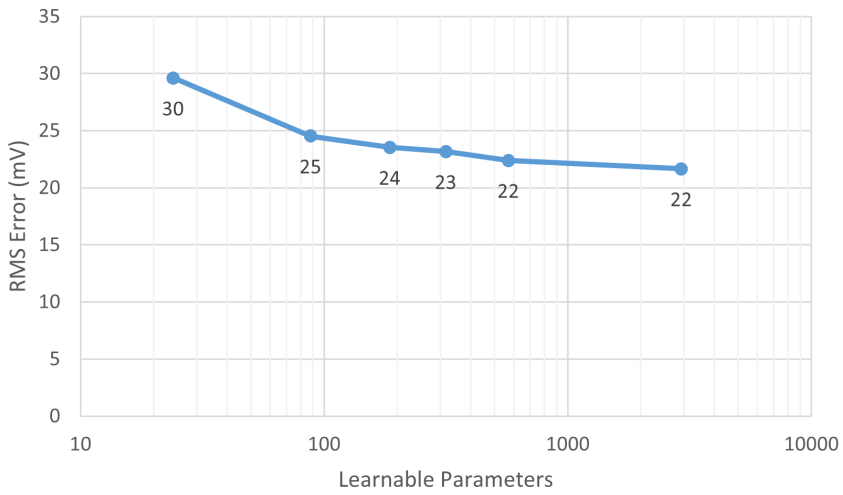
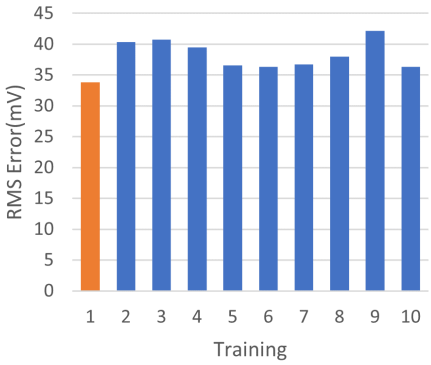


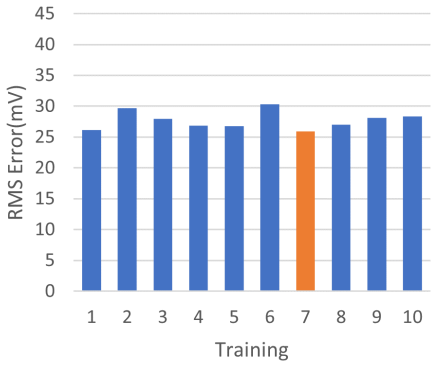
FIGURE 5.5: Lowest RMSE of 4 trainings per learnable parameter. Each RMSE is based on the average of UDDS, HWFET, LA92, and US06 drive cycles for m80 kg.

from all four cells as well as HWCUST1 and HWGRADE1 for training. HWCUST2 and HWGRADE2 drive cycles are also utilized for testing. Fig. 5.6 illustrates the average RMSE for each training based on testing criteria for each dataset. Training 1 with an average RMSE of 34 mV is the best trained network for the whole dataset while training 7 with an average RMSE of 26 mV is the best for the classic dataset. To ensure that the HWCUST and HWGRADE cycles were not skewing the results, the average RMSE without these drive cycles were calculated for the whole dataset and training 1 was still found to be the best. Training 1 ran for a total of 37,437 epochs and took 37 hrs and 53 min. Training 7 ran for a total of 28,038 epochs and took 35 hrs and 42 min.

Based on Fig. 5.6, the networks trained on the classic dataset provide a lower RMSE and therefore better voltage estimation. However, when the classically trained network was tested with the 94 test cases (drive cycles that vary in mass, HVAC, highway, and grade), the network based on the whole dataset provided better results in every category, as can be seen in Fig. 5.7. The full testing criteria consists of the UDDS, HWFET, LA92, US06, HWCUST2, HWGRADE2 drive cycles for each temperature (40°C, 25°C, 10°C,



(A) Average RMSE of all test cases for and trained on whole dataset.



(B) Average RMSE of all test cases for and trained on classic dataset.

FIGURE 5.6: Average RMSE per training for each dataset. The orange bar is the best trained network for each dataset.

0°C) and cell (m80, m448, m448-N, m1000); a total of 94 cases. At 25°C, m1000 there is the largest difference of 48% between the networks while at 40°C, m1000, there is the smallest difference of 11%. Only varying temperature in the training dataset does not provide sufficient coverage of testing scenarios for BEV battery modelling.

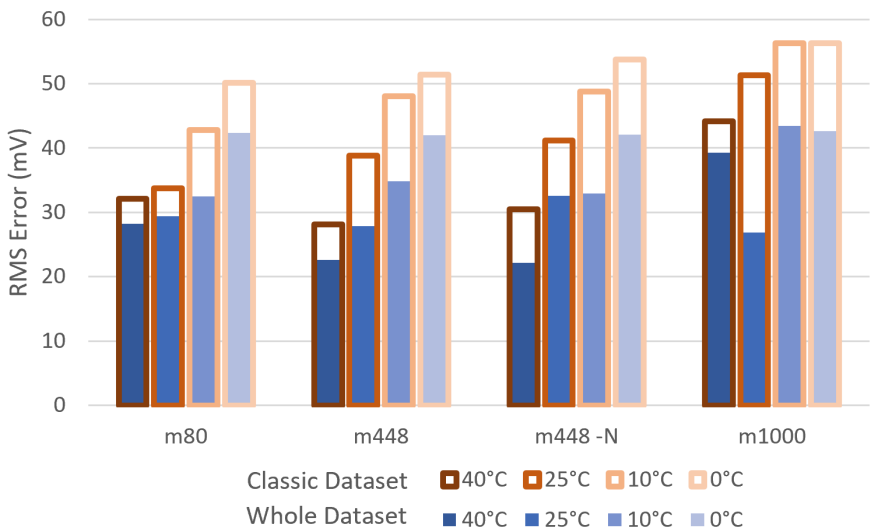


FIGURE 5.7: Voltage estimation, comparison of whole dataset and classic dataset trained networks. The bar graphs represent the average RMSE of test drive cycles UDDS, HWFET, LA92, US06, HWCUST2, and HWGRADE2 which is representative of testing on the whole dataset.

5.6 Range Estimation based on LSTM-RNN Battery model

The following sections will review the performance of the best trained network (training 1) based on the whole dataset.

5.6.1 Model Performance: Best vs. Worst Result for Standard Four Drive Cycles

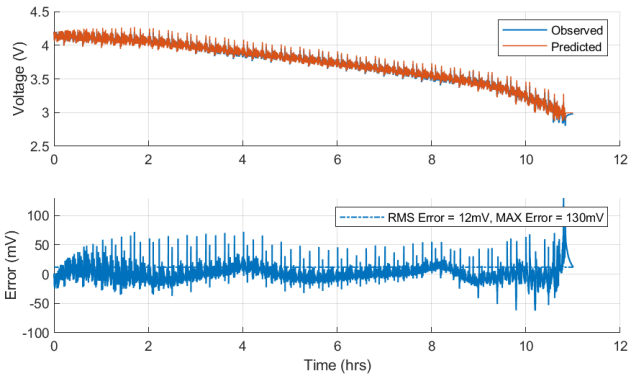
The drive cycle with the best voltage estimation is the LA92 at 40°C, m448-N with an RMSE of 12 mV and MAX error of 130 mV and the worst drive cycle is from the US06 at 0°C m1000 with an RMSE of 57 mV and MAX error of 275 mV. A comparison of the two drive cycles is available in Fig. 5.8. Comparing the LA92 drive cycle in Fig. 5.8a with the US06 in Fig. 5.8c, the US06 has a much higher error at the start and end of the cycle then the LA92. The combination of the lower temperature, higher mass

(1000 kg or towing), as well as the challenging drive cycle of US06 leads to a voltage estimation with higher error. Despite this, the network performs well when viewed on a minute by minute basis. Fig. 5.8b and Fig. 5.8d provide a close-up of the first 5 minutes of the LA92 and US06 drive cycles, respectively.

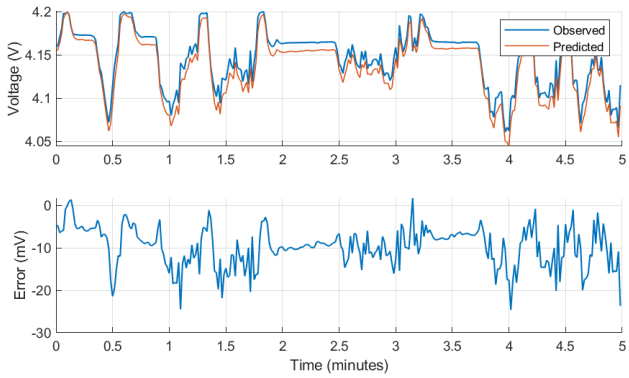
The range estimation for the respective best and worst case scenarios is provided in Fig. 5.9. At a cut-off capacity of 4.514 Ah, the error between the observed distance and estimated distance is -1 km, where negative is an underestimation, for the LA92 drive cycle at 40°C and m448-N. At a cut-off capacity 3.627 Ah, the driving distance error is 1 km for the US06 drive cycle at 0°C and m1000. The error in voltage estimation is not directly related to the error in range and a higher voltage prediction does not always lead to a higher distance error.

5.6.2 Model Performance Over Temperature

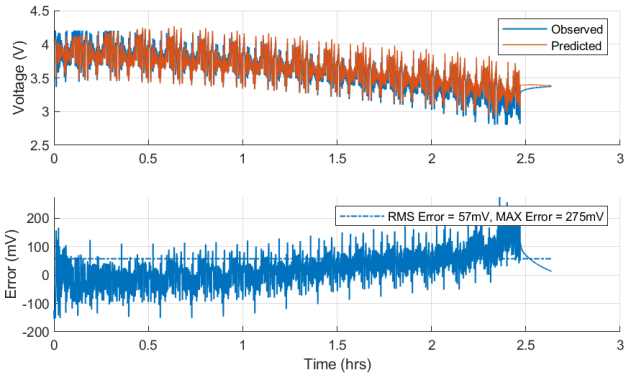
One of the more important factors of battery voltage estimation is temperature. Fig. 5.10 compares the US06 drive cycle, m80, for temperatures 40°C, 25°C, 10°C, and 0°C. From highest to lowest temperature the RMSE is 16 mV, 17 mV, 25 mV, and 34 mV and MAX error of 77 mV, 86 mV, 140 mV, and 187 mV. The higher error at the lower temperatures of 10°C and 0°C can be attributed to larger changes in voltage as well as less training data. The same US06 drive cycle reaches cut-off capacity in less than 3 hours for 0°C while it takes just over 4 hours for 40°C. Overall, the trained LSTM-RNN network performs well at varying temperatures.



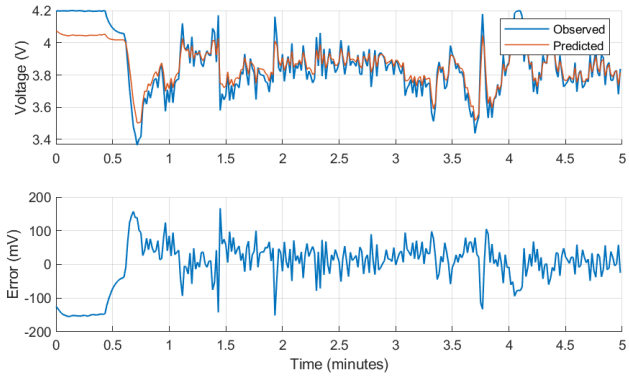
(A) LA92 Drive Cycle at 40°C and m448-N



(B) LA92 Drive Cycle at 40°C from 0 - 5 min

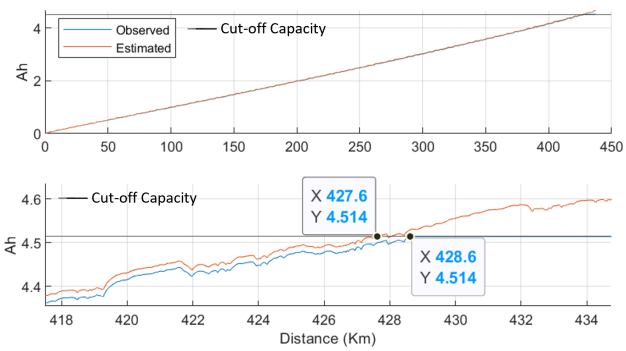


(C) US06 Drive Cycle at 0°C and m1000

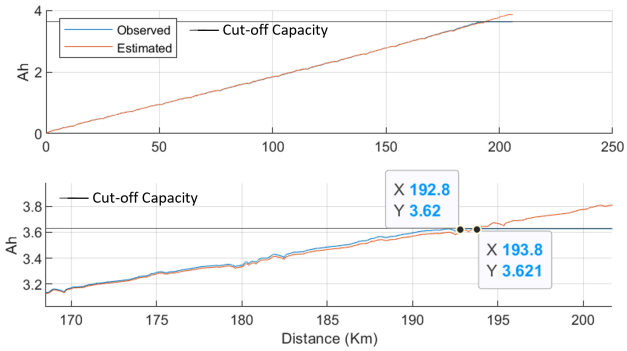


(D) US06 Drive Cycle at 0°C from 0 - 5 min

FIGURE 5.8: LSTM voltage prediction vs. observed (top) and error between predicted and observed (bottom). Close-up of the first 5 minutes of the LA92 and US06 drive cycles.

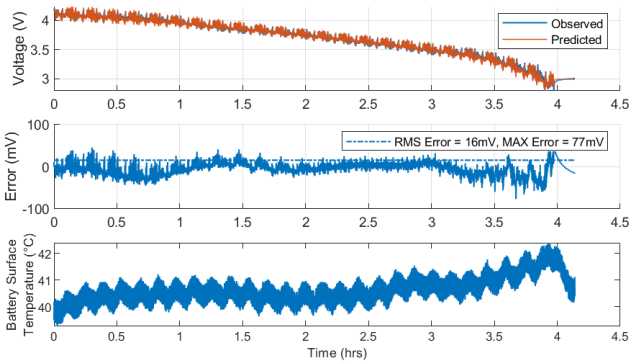


(A) LA92 Drive Cycle at 40°C and m448-N

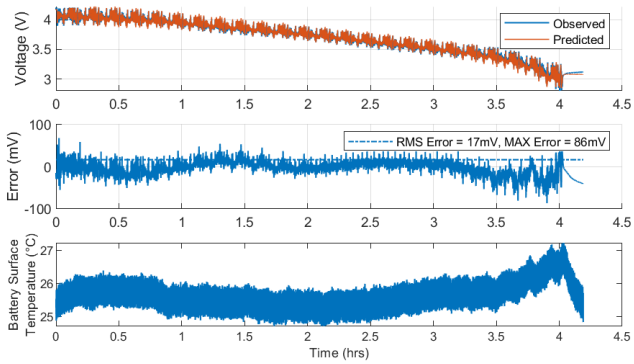


(B) US06 Drive Cycle at 0°C and m1000

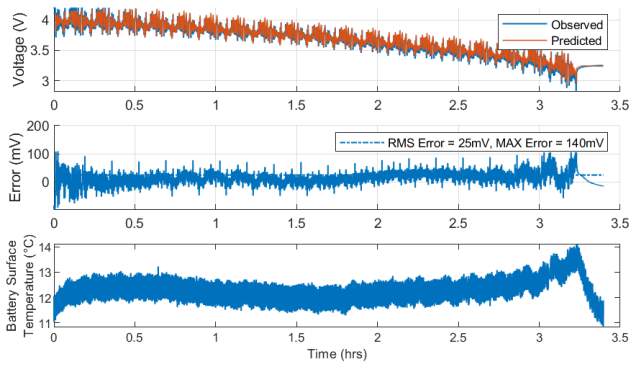
FIGURE 5.9: Range estimation vs. actual (top) and close-up at cut-off capacity of 4.514 Ah for 40°C and 3.627 Ah for 0°C.



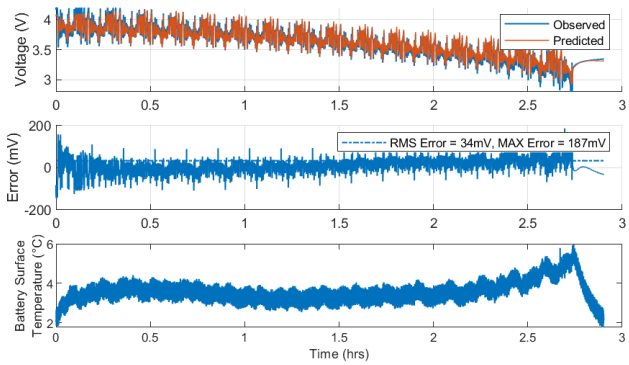
(A) 40°C



(B) 25°C



(C) 10°C



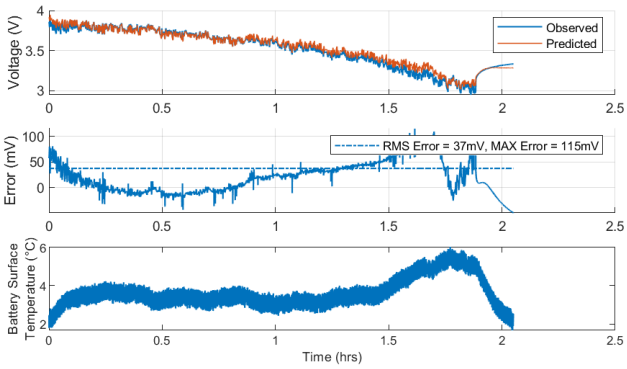
(D) 0°C

FIGURE 5.10: LSTM voltage prediction vs. observed (top), error between predicted and observed (middle), and battery surface temperature (bottom) for m80, US06 drive cycle, and all test temperatures.

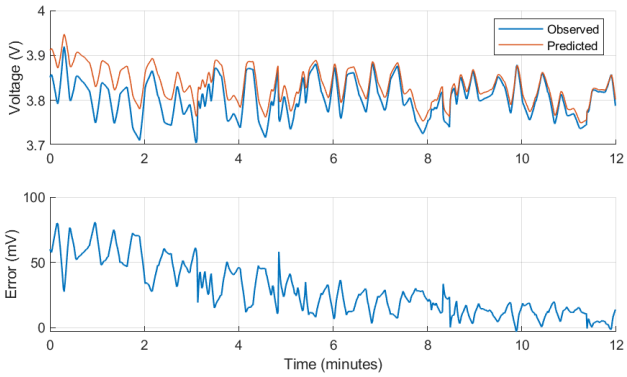
5.6.3 Model Performance in Custom Cycles

The advantage of the dataset collected in this work are the aggressive HWCUST and HWGRADE cycles. Fig. 5.11 illustrates the performance of the network during aggressive highway driving using the HWCUST2 drive cycle at 0°C for m80 and m1000. The RMSE is 37 mV and 38 mV, respectively. Overall, the network performs well, leading to driving range estimation errors, at cut-off capacity, of 1.1 and 1.2 km for the m80 and m1000 cases, respectively. The estimated capacity vs. distance for each case can be seen in Fig. 5.12.

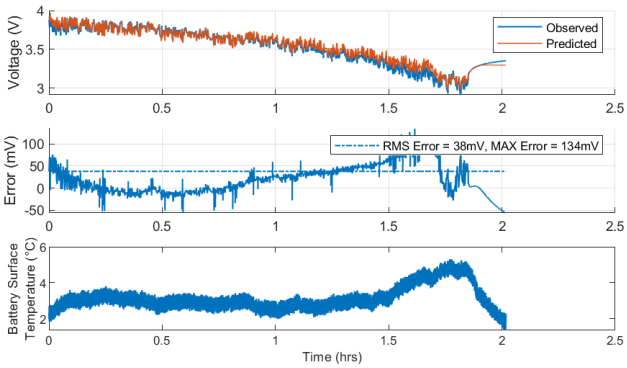
Fig. 5.13 illustrates the performance of the network during an aggressive highway and varying grade drive using the HWGRADE2 cycle at 0°C for m80 and m1000. For the m80 case, the RMSE is 72 mV and a high MAX error of 294 mV, however the range estimation at the cut-off capacity is only 0.7 km. For the m1000 case, the voltage estimation RMSE is 69 mV, MAX error of 343 mV, and a range estimation error of -0.9 km at cut-off capacity.



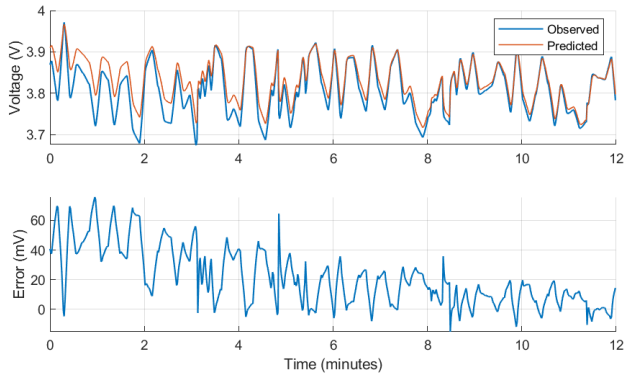
(A) HWCUST2 m80



(B) HWCUST2 m80 from 0 - 12 min

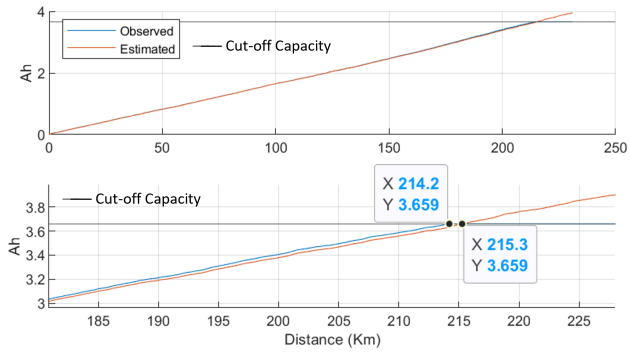


(C) HWCUST2 m1000

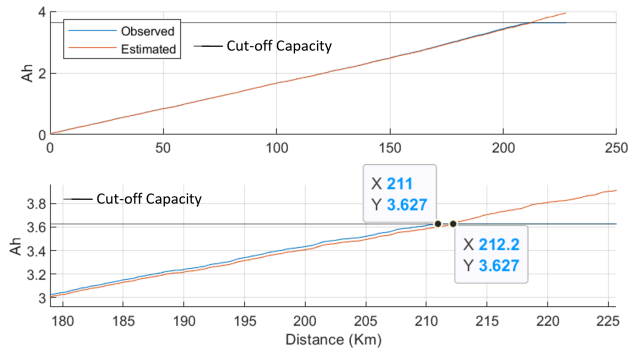


(D) HWCUST2 m1000 from 0 - 12 min

FIGURE 5.11: LSTM voltage prediction vs. observed (top), error between predicted and observed (middle), and battery surface temperature (botton) at 0°C for the full cycles (left). LSTM voltage prediction vs. observed (top), and error between predicted and observed (botton) at 0°C for the first 12 minutes (right).

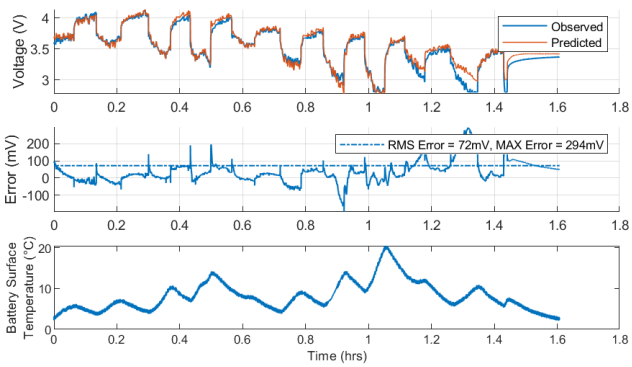


(A) HWCUST2 Drive Cycle at m80 0°C

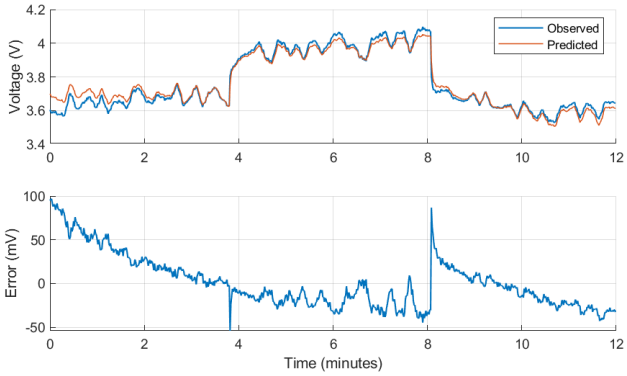


(B) HWCUST2 Drive Cycle at m1000 0°C

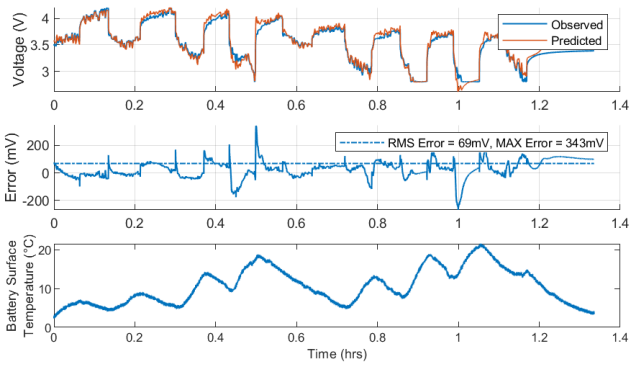
FIGURE 5.12: Range estimation vs. actual (top) and close-up at cut-off capacity of 3.659 Ah for m80 and 3.627 for m1000 (bottom).



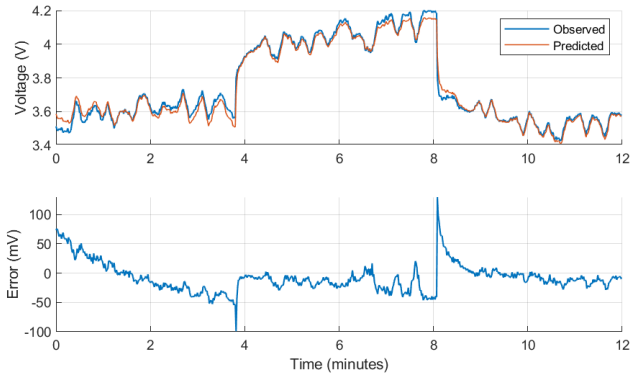
(A) HWGRADE2 m80



(B) HWGRADE2 m80 from 0 - 12 min

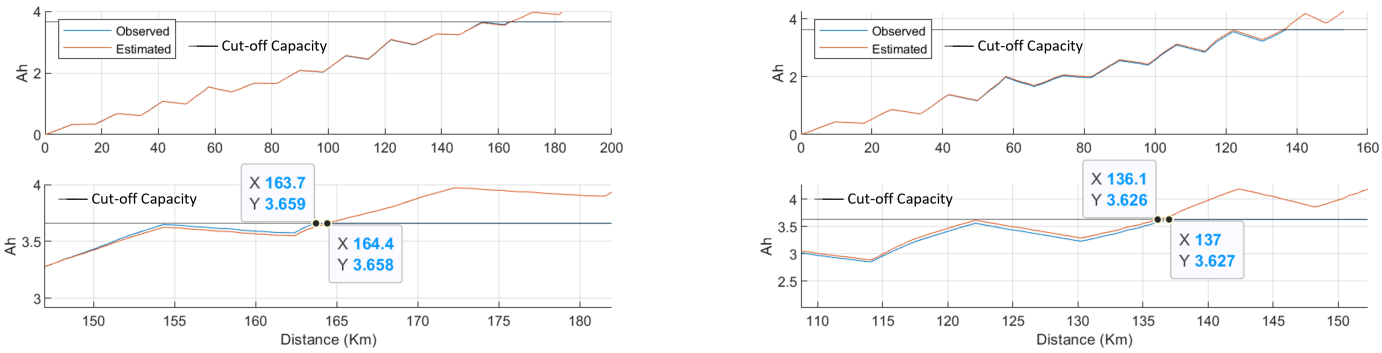


(C) HWGRADE2 m1000



(D) HWGRADE2 m1000 from 0 - 12 min

FIGURE 5.13: LSTM voltage prediction vs. observed (top), error between predicted and observed (middle), and battery surface temperature (botton) at 0°C for the full cycles (left). LSTM voltage prediction vs. observed (top), and error between predicted and observed (botton) at 0°C for the first 12 minutes (right).



(A) HWGRADE2 at m80, 0°C

(B) HWGRADE2 at m1000, 0°C

FIGURE 5.14: Range estimation vs. actual (top) and close-up at cut-off capacity of 3.659 Ah for m80 and 3.627 for m1000 (bottom).

5.6.4 Voltage Estimation Comparison to Literature

The RMSE of the drive cycles, UDDS, HWFET, LA92, US06, HWCUST2, HWGRADE2 for m80 at each temperature is provided in Fig. 5.15. The lowest RMSE is 16 mV for the US06 drive cycle at 40°C while the highest is 72 mV for the HWGRADE2 at 0°C. UDDS exhibits higher RMSE than expected of 47 mV, 45 mV, and 53 mV for 25°C to 0°C, respectively. Even though the dataset takes into account aggressive highway and grade driving, the amount of casual or urban driving is reduced and therefore not estimated as expected. From this, it can be concluded that both urban and highway driving cycles are important for a well rounded battery model.

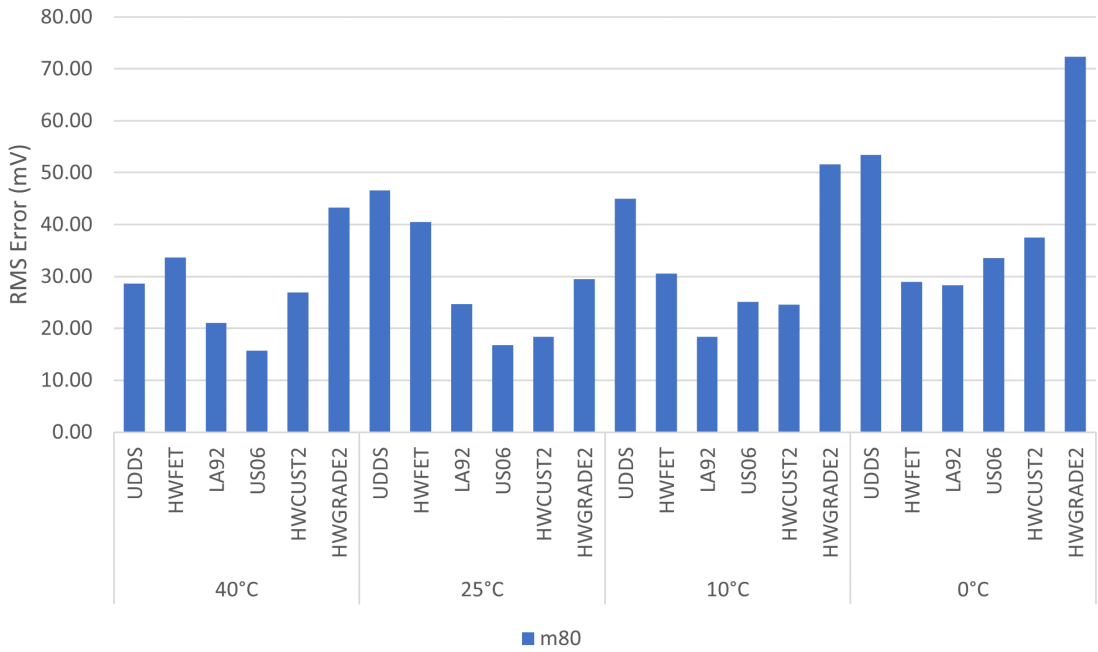


FIGURE 5.15: Voltage estimation RMSE for m80, all temperatures, and all drive cycles.

The work presented here offers a competitive estimation performance when compared to other work mentioned in literature which are shown in Table 5.2. The LSTM-RNN voltage esitmation in this work achieved an average RMSE of 34 mV, MAE of 25 mV, MAX error of 150 mV, and STDDEV of 30 mV for all drive cycles, all temperatures, and all cell parameters. The LA92 drive cycle for the m448-N case had the lowest RMS errors for each temperature, where the highest error was 27 mV. Compared to the LSTM-RNN in [84], the LA92 drive cycle for m448-N for this work achieved and 82% less error. Compared to the GRU-RNN, power input models of the Panasonic NCR18650PF and Sony VTC6 batteries in [21], the LA92 drive cycle for m448-N for this work achieved 53% and 47% less error, respectively.

Method	**Error**	**Temperature**	**Test Case**	**Battery**
LSTM-RNN, thesis results	<27 mV RMSE	40°C, 25°C, 10°C, 0°C	Average of LA92 m448-N	4.5 Ah Tesla/Panasonic 2170
LSTM-RNN [84]	<146 mV RMSE	45°C, 25°C, 0°C	US06 drive cycle (normalized)	2 Ah INR18650-20R
GRU-RNN [21]	<66 mV RMSE	25°C, 10°C, 0°C	NN drive cycle (current input)	2.9 Ah Panasonic NCR18650PF
	<58 mV RMSE		NN drive cycle (power input)	
	<51 mV RMSE		NN drive cycle (power input)	2 Ah Sony VTC6
LSTM-RNN [85]	6 mV MAE	25°C (ambient)	10 Drive Cycles	2.15 Ah Panasonic
SDAE-ELM [86]	3.01 MAPE (%)	5°C - 60°C	Discharge drive cycle from electric buses	Unspecified

TABLE 5.2: Comparison of voltage estimation method results in literature.

5.7 Range Estimation Results

All standard drive cycles at all tested temperatures and cells achieved the goal of range estimation within 5 km. Fig. 5.16 provides the absolute error in kilometers for the m80 cell. The lowest absolute error is 0.05 km for the LA92 drive cycle at 0°C for the m80 kg case. The highest absolute error is 4.78 km for the US06 drive cycle at 25°C for the m1000 kg case. There is no obvious correlation between the range estimation and the criteria used (drive cycles, temperature, mass, HVAC).

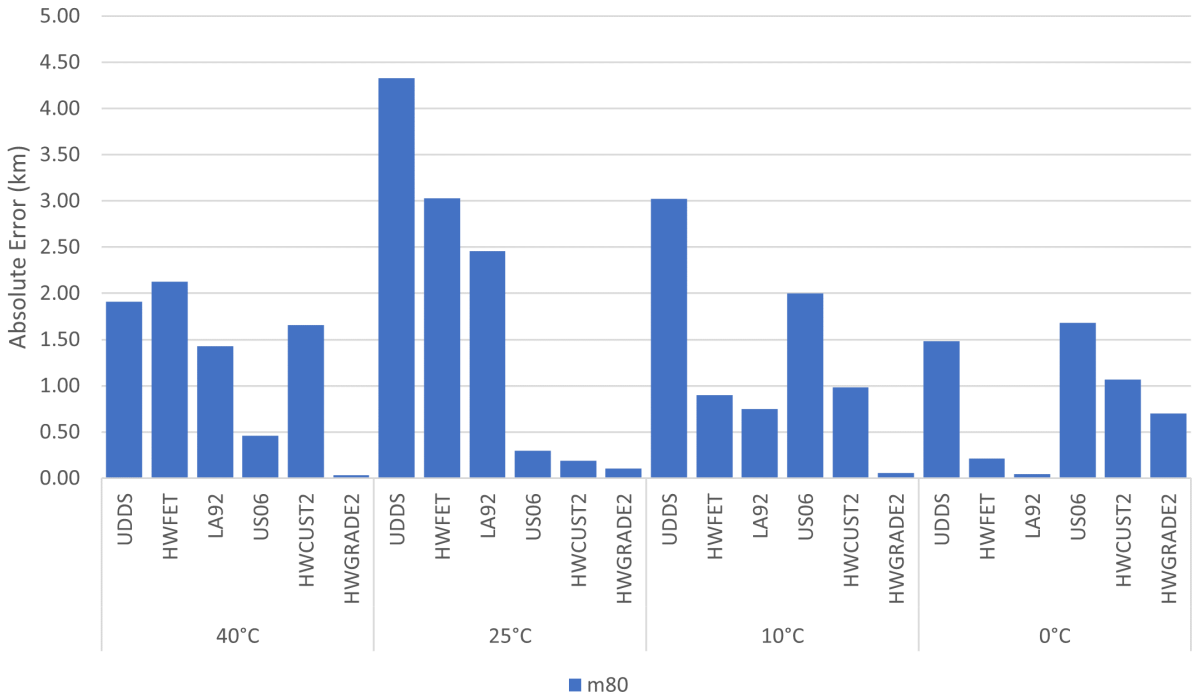


FIGURE 5.16: Absolute error between estimated range and actual range. Estimated range is based on network trained on whole dataset for all drive cycles, all temperatures, and m80 kg.

Using the C/2 discharge test, the energy in Wh is taken at the cut-off capacity and scaled up to the pack level in kWh. The pack energy at cut-off capacity is then divided by the consumption reading (Wh/km) from the vehicle model per temperature and cell to calculate a simple range estimation. Fig. 5.17 compares the simple C/2 range estimation,

the LSTM-RNN battery model range estimation, and the actual range for drive cycles UDDS, LA92, and HWGRADE2 at each temperature for m448-N. The UDDS, LA92, and HWGRADE2 drive cycles illustrate the effects of various driving routes from mild to aggressive. As the drive cycles become more aggressive, the range is reduced and in some cases, the C/2 range error increases. However, the proposed range estimation error stays within 1-3 km. The same can be seen as temperature decreases, where the C/2 range error increases while the proposed model stays consistent.

At 25°C, the more aggressive HWCUST2 cycle range is 195 km shorter than the mild UDDS. However, for the HWGRADE2 drive cycle, the C/2 estimation has an 18 km or 11% range error while the LSTM-based estimation has a -2 km or -1% range error, where the negative means underestimation. This translates to 18 km of range the C/2 or simple range estimation is not capturing. m448-N (cell 3) illustrates the effect of temperature on range estimation.

The graph also includes the battery efficiency, calculated by comparing the energy extracted from each drive cycle to the energy extracted in the C/20 test, both at the cut-off capacity. This graph illustrates the battery model capturing energy loss at the battery. The less intense UDDS and HWFET cycles have a higher efficiency than the more aggressive HWCUST and HWGRADE cycles.

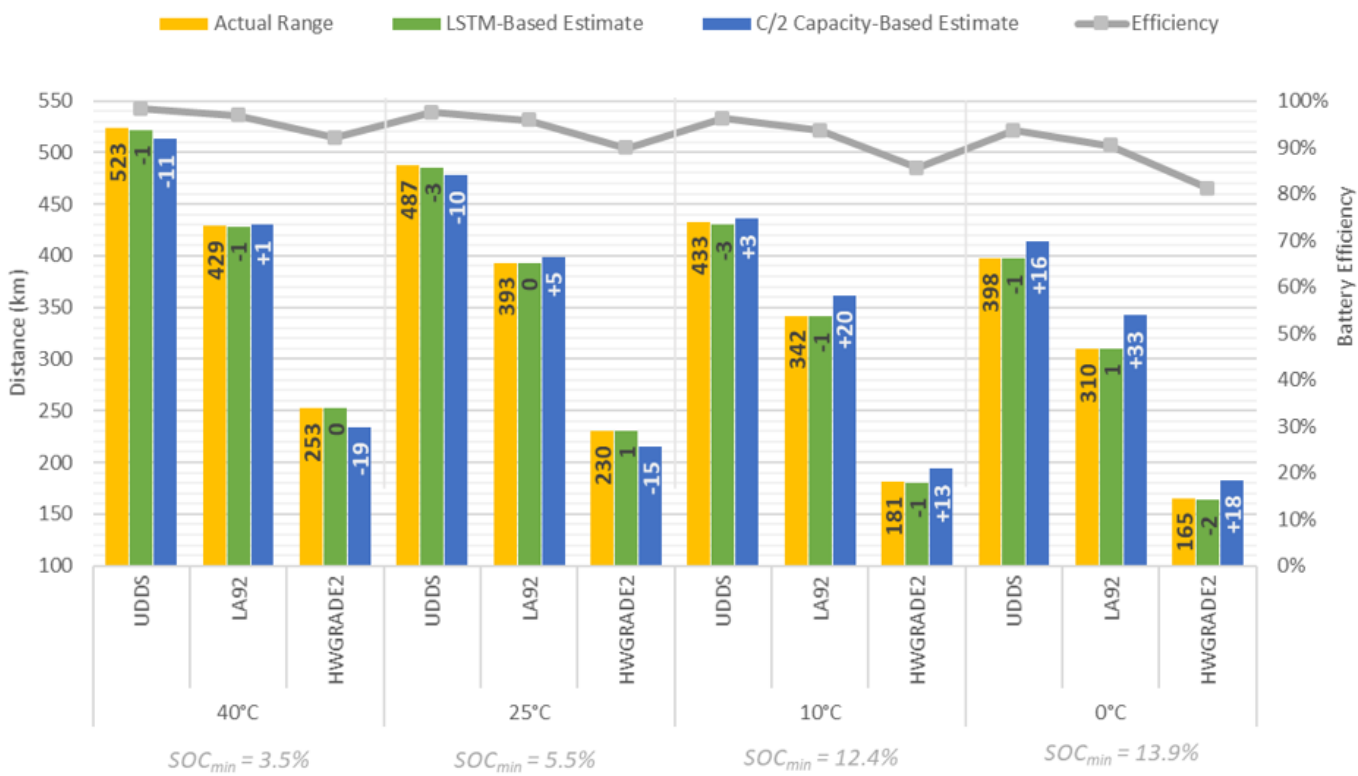


FIGURE 5.17: Actual range vs. simple C/2 range estimation vs. LSTM-based range estimation and efficiency for m448-N battery and select drive cycles. SOC_{min} indicates the cut-off capacity in SOC at the respective temperature.

When comparing the lowest error achieved in this research, the aggressive HWGRADE2 drive cycle for 25°C and m448, seen in Fig. 5.18, achieved a 0 km error at a cut-off capacity of 4.259Ah or $SOC_{min} = 5.5\%$. A range estimation based on an LSTM-RNN battery model leads to a more accurate range estimation in comparison to one based on a simple energy consumption calculation.

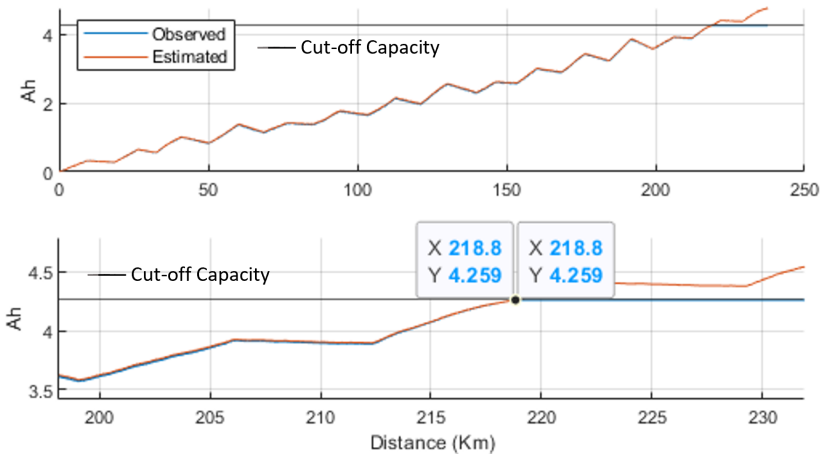


FIGURE 5.18: Range estimation vs. actual (top) and close-up at cut-off capacity of 4.259Ah at 25°C, for the HWGRADE2 drive cycle and m448.

5.7.1 Range Estimation Comparison to Literature

The range estimation results presented here offer competitive performance when compared to other work in literature. The following Table 5.3 is an extension of the table 2.3 in chapter 2. The authors in [42] utilize a similar method to range estimation as the one in this work. Instead of a machine learning based battery model, a 2RC ECM model is used with the battery states estimated using a particle filter and the driving route is estimated using Markov chains. The ECM method has a 91.71% prediction performance at the end of the UDDS cycle. While in this work, the range estimation error compared to the full driving distance at the end of the UDDS cycle ranges from 99.1% to 99.9% even though the voltage estimation results were higher than expected.

The average range error for the whole dataset is 0.29 km and a max error of 4.78 km. The average error is 93% better than the ECM based model in [58] and 90-94% better than the KF model in [54]. While the MAX error is higher by 20% of the results in [58] and in the margin of the results in [54]. These results are in favor of using a machine learning based battery model for range estimation and are evidence of the importance

of proper battery modelling for use in BEV.

Method	**Error**
LSTM-RNN, thesis results	Avg 0.3 km, MAX 4.78 km
Regression Tree, Self-Organizing Maps [53]	1.51 kWh RMSE
XGBoost-LightGBM blend [43]	0.75 km RMSE
Kalman Filter [54]	3-5 km difference at trip end
Particle Filter and Markov Chains [42]	91.71% prediction performance (UDDS last prediction)
Regression Analysis [44]	1.64 km MAE
Kernel Principal Component, fuzzy C clustering, Markov chains, and backpropagation neural networks [57]	5.45 km Max error
Battery equivalent circuit model and SOC estimation [58]	4 km difference at trip end
Artificial Neural Network [48]	94.33% accuracy

Table 5.3: Comparison of Range Estimation Results from Literature

Chapter 6

Conclusions and Future Work

6.1 Concluding Remarks

In summary, the objectives of this research were achieved. A Kalman filter-based SOC estimation function in MATLAB was created and made available on Mathworks. Sample SOC estimation results using publicly available Turnigy battery data was presented. At the moment of writing this thesis, the work has been downloaded over 400 times and has achieved a 5 star rating. This work will allow students and researchers an opportunity to expand battery and state modelling for various application without the need to recreate an SOC estimation algorithm.

An extensive battery dataset comprised of 7 characterization tests, 16 drive cycles, 4 temperatures, and 4 scenarios including a mix of payload mass and HVAC options has been created based on the Tesla/Panasonic 2170 cells. Once available, this dataset will allow other researchers an opportunity to further battery state and modelling estimation.

Accurate range estimation is critical to curb range anxiety and increase the adoption of BEV's. By enhancing the battery model, a more accurate range estimation can be provided to the driver and increase utilizing of the vehicle. A range estimation method was proposed that focused on an LSTM-RNN based battery model. A comparison of

utilizing the extensive dataset versus a classic dataset to train the LSTM-RNN model was provided. SHARCNET was utilized to speed up training of the machine learning battery model from 36 days down to 7 days. 92% of the standard test cases achieved an RMS error of less than 50 mV and all test cases achieved the goal of less than 5 km range estimation error at cut-off capacity.

6.2 Future Work

This thesis includes a tool/function for SOC estimation based on an EKF. With further tuning of the battery parameters and the KF parameters, the RMSE values can be improved further. Furthermore, this function can be expanded upon to add a temperature model for more accurate estimations of the battery temperature. It can also be simplified to remove the temperature dependency as well as adaptive portion by adjusting those respective lines. Currently, battery parameters must be estimated both separately and offline. By adjusting the function to an RLS-EKF algorithm, the parameter estimation can happen in conjunction and on-board a BMS. This will allow the tool to expand from BEVs to other applications such as drones and electric aircraft. Furthermore, the estimation of other battery parameter states such as a dual SOC/SOH estimation option can be added. A dual or multi-state estimation algorithm reduces the computational burden on the BMS by reducing the number of models that need to be run. As the popularity of battery applications grow, so do the need for publicly available battery modelling and battery state estimations tools and functions.

Regarding the range estimation of a BEV, this area of research is fairly new and has many avenues for future projects. The author believe that there are three main areas of future work that together with the research presented in this thesis will results in a complete online range estimation solution. These include driving route estimation, processor-in-the-loop (PIL) testing of algorithm, and integration into a BEV.

A key component of predicting remaining driving range is being able to predict the driving route based on the current location and destination. The driving route can be a simple speed over time profiles generated based on APIs to replace the current driving schedules. By incorporating the route prediction, range estimation can be made available between any two points in the battery SOC.

One of the many challenges in battery algorithm development is being able to achieve the same accuracy on a processor as one can on a normal PC. That is why PIL testing is an important step in the development of the remaining drive estimation. Lastly, the remaining driving range must be incorporated into the overall BMS and vehicle. One such similar method exists in the Tesla Model X. Pictured in Fig. 6.1 is the Tesla Model X infotainment screen with directions from Hamilton, ON to the CN Tower in Toronto, ON. The left hand side provides an estimation of the remaining SOC once arrived at destination and the remaining SOC for a round trip. One piece of information not provided in this picture is the range in terms of miles or kilometers.

Other small avenues of work that can further the research of this thesis include:

- Adding -10°C and -20°C test data to the Tesla/Panasonic battery dataset
- Create and utilize a more sophisticated (forward-looking) vehicle model for future battery testing
- Create a battery model with other machine learning methods such as GRU-RNN, CNN, or even unsupervised learning methods
- Test range estimation from varying SOC levels
- Replicate the research with a different battery cell for a more direct comparison

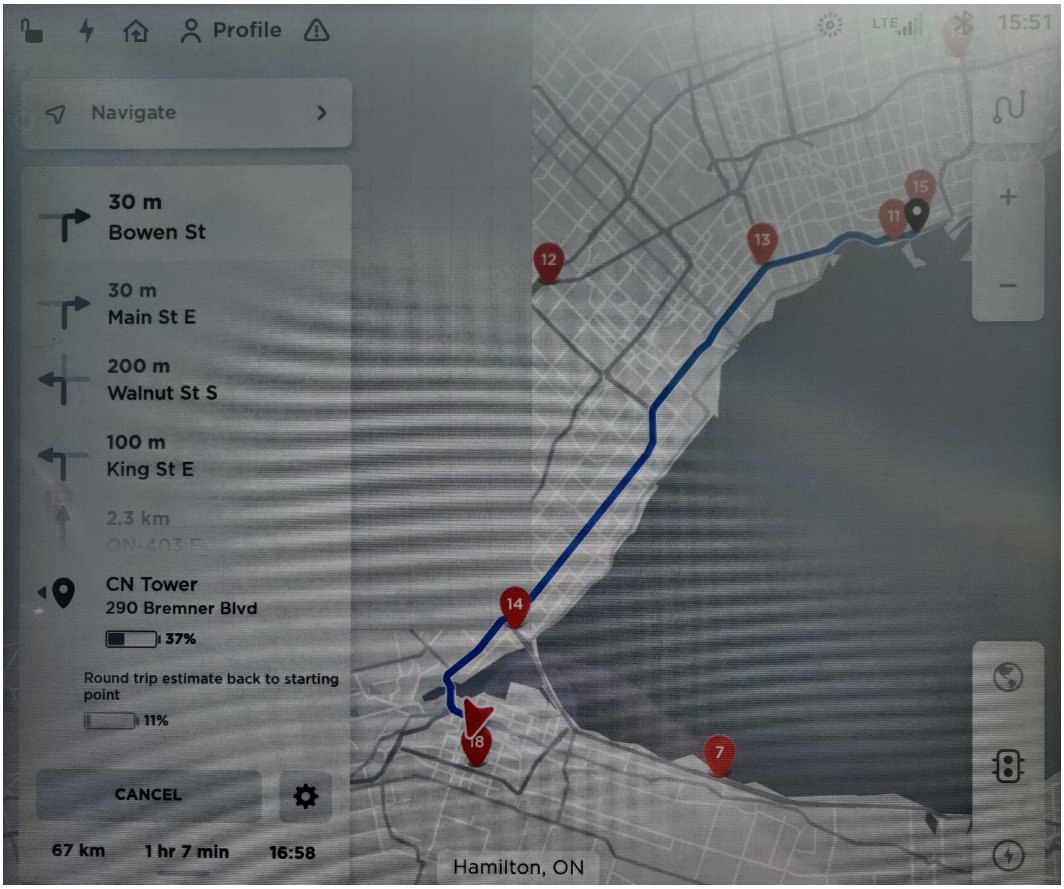


FIGURE 6.1: Tesla Model X infotainment screen: directions and range estimation from Hamilton, ON to the CN Tower in Toronto, ON based on the current SOC.

6.3 Publications

The following conference paper has resulted from this research:

- F. Khanum, E. Louback, F. Duperly, C. Jenkins, P. J. Kollmeyer and A. Emadi, "A Kalman Filter Based Battery State of Charge Estimation MATLAB Function," 2021 IEEE Transportation Electrification Conference & Expo (ITEC), 2021, pp. 484-489, doi: 10.1109/ITEC51675.2021.9490163.

Appendix

State of Charge Estimation Function based on Extended Kalman Filter

main.mlx

```
clc; clear; close all;

load('06-03-19_09.46 825_LA92_0degC_Turnigy_Graphene.mat');
LiPoly.RecordingTime            = meas.Time;
LiPoly.Measured_Voltage         = meas.Voltage;
LiPoly.Measured_Current         = meas.Current;
LiPoly.Measured_Temperature     = meas.Battery_Temp_degC;
nominalCap                      = 4.81; % Battery capacity in Ah taken ...
    from data.
LiPoly.Measured_SOC             = (nominalCap + ...
    meas.Ah).*100./nominalCap;  % Calculate the SOC using Coloumb ...
    Counting for comparison

% Resample input data
LiPoly.RecordingTime            = LiPoly.RecordingTime(1:10:end);
LiPoly.Measured_Voltage         = LiPoly.Measured_Voltage(1:10:end);
LiPoly.Measured_Current         = LiPoly.Measured_Current(1:10:end);
LiPoly.Measured_Temperature     = LiPoly.Measured_Temperature(1:10:end);
LiPoly.Measured_SOC             = LiPoly.Measured_SOC(1:10:end);

% Current Definition: (+) Discharging, (-) Charging
LiPoly.Measured_Current_R       = - LiPoly.Measured_Current;
% Converting seconds to hours
LiPoly.RecordingTime_Hours      = LiPoly.RecordingTime/3600;

[SOC_Estimated, Vt_Estimated, Vt_Error] = ...
    EKF_SOC_Estimation(LiPoly.Measured_Current_R, ...
    LiPoly.Measured_Voltage, LiPoly.Measured_Temperature);

% Terminal Voltage Measured vs. Estimated
figure
plot(LiPoly.RecordingTime_Hours,LiPoly.Measured_Voltage);
hold on
plot(LiPoly.RecordingTime_Hours,Vt_Estimated);
```

```
hold off;
legend('Measured','Estimated EKF');
ylabel('Terminal Voltage[V]');xlabel('Time[hr]');
title('Measured vs. Estimated Terminal Voltage (V) at 0 Deg C')
grid minor

% Terminal Voltage Error
figure
plot(LiPoly.RecordingTime_Hours,Vt_Error);
legend('Terminal Voltage Error');
ylabel('Terminal Voltage Error');
xlabel('Time[hr]');

% SOC Coulomb Counting vs. Estimated
figure
plot (LiPoly.RecordingTime_Hours,LiPoly.Measured_SOC);
hold on
plot (LiPoly.RecordingTime_Hours,SOC_Estimated*100);
hold off;
legend('Coulomb Counting','Estimated EKF');
ylabel('SOC[%]');xlabel('Time[hr]');
title('Coulomb Counting vs. SOC Estimated at 0 Deg C')
grid minor

% SOC Error
figure
plot(LiPoly.RecordingTime_Hours,(LiPoly.Measured_SOC - ...
    SOC_Estimated*100));
legend('SOC Error');
ylabel('SOC Error [%]');
xlabel('Time[hr]');
grid minor

% Calculate RMSE and MAX of Vt and SOC
RMSE_Vt      = sqrt((sum((LiPoly.Measured_Voltage - Vt_Estimated).^2)) ...
    /(length(LiPoly.Measured_Voltage)))*1000 % mV
RMSE_SOC     = sqrt((sum((LiPoly.Measured_SOC - SOC_Estimated*100).^2)) ...
    /(length(LiPoly.Measured_SOC))) % (%)
Max_Vt       = max(abs(LiPoly.Measured_Voltage - Vt_Estimated))*1000 % mV
Max_SOC      = max(abs(LiPoly.Measured_SOC - SOC_Estimated*100)) % (%)
```

EKF_SOC_Estimation.mlx

```
function [SOC_Estimated, Vt_Estimated, Vt_Error] = ...
    EKF_SOC_Estimation(Current, Vt_Actual, Temperature)

load 'BatteryModel.mat'; % Load the battery parameters
load 'SOC-OCV.mat'; % Load the SOC-OCV curve

SOC_Init     = 1; % intial SOC
X            = [SOC_Init; 0; 0]; % state space x parameter intializations
DeltaT       = 1; % sample time in seconds
```

```
Qn_rated    = 4.81 * 3600; % Ah to Amp-seconds
% initialize scatteredInterpolant functions for battery parameters and ...
    SOC-OCV curve
% this function also allows for extrapolation
F_R0    = scatteredInterpolant(param.T,param.SOC,param.R0);
F_R1    = scatteredInterpolant(param.T,param.SOC,param.R1);
F_R2    = scatteredInterpolant(param.T,param.SOC,param.R2);
F_C1    = scatteredInterpolant(param.T,param.SOC,param.C1);
F_C2    = scatteredInterpolant(param.T,param.SOC,param.C2);

SOCOCV  = polyfit(SOC_OCV.SOC,SOC_OCV.OCV,11); % calculate 11th order ...
    polynomial for the SOC-OCV curve
dSOCOCV = polyder(SOCOCV); % derivative of SOC-OCV curve for matrix C

n_x   = size(X,1);
R_x   = 2.5e-5;
P_x   = [0.025 0 0;
0 0.01 0;
0 0 0.01];
Q_x   = [1.0e-6 0 0;
0 1.0e-5 0;
0 0 1.0e-5];

SOC_Estimated   = [];
Vt_Estimated    = [];
Vt_Error        = [];
ik              = length(Current);
% Current         = Current-0.1;

for k=1:1:ik
    T           = Temperature(k); % C
    U           = Current(k); % A
    SOC         = X(1);
    V1          = X(2);
    V2          = X(3);

    % Evaluate the battery parameter scatteredInterpolant
    % functions for the current temperature & SOC
    R0      = F_R0(T,SOC);
    R1      = F_R1(T,SOC);
    R2      = F_R2(T,SOC);
    C1      = F_C1(T,SOC);
    C2      = F_C2(T,SOC);
    % OCV     = F_OCV(T,SOC);
    % OCV     = pchip(param.SOC,param.OCV,SOC); % pchip sample for ...
        unknown or single temperature

    OCV = polyval(SOCOCV,SOC); % calculate the values of OCV at the ...
        given SOC, using the polynomial SOCOCV

    Tau_1       = C1 * R1;
    Tau_2       = C2 * R2;

    a1 = exp(-DeltaT/Tau_1);
```

```
    a2 = exp(-DeltaT/Tau_2);

    b1 = R1 * (1 - exp(-DeltaT/Tau_1));
    b2 = R2 * (1 - exp(-DeltaT/Tau_2));

    TerminalVoltage = OCV - R0*U - V1 - V2;

    if U > 0
        eta = 1; % eta for discharging
    elseif U ≤ 0
        eta = 1; % eta for charging
     end

     dOCV = polyval(dSOCOCV, SOC);
     C_x    = [dOCV -1 -1];

    Error_x   = Vt_Actual(k) - TerminalVoltage;

    Vt_Estimated    = [Vt_Estimated;TerminalVoltage];
    SOC_Estimated   = [SOC_Estimated;X(1)];
    Vt_Error        = [Vt_Error;Error_x];

    A   = [1 0  0;
    0 a1 0;
    0 0  a2];
    B   = [-(eta * DeltaT/Qn_rated); b1; b2];
    X   = (A * X) + (B * U);
    P_x = (A * P_x * A') + Q_x;

    KalmanGain_x = (P_x) * (C_x') * (inv((C_x * P_x * C_x') + (R_x)));
    X            = X + (KalmanGain_x * Error_x);
    P_x          = (eye(n_x,n_x) - (KalmanGain_x * C_x)) * P_x;

    %    Q_x = KalmanGain_x * Error_x * KalmanGain_x';

end
```

References

[1] IEA, “World Energy Balances: Overview”, IEA, Paris, 2021, [Online]. Available: https://www.iea.org/reports/world-energy-balances-overview. [Accessed: 23-Aug-2021].

[2] S. O’Kane, “EU proposes phasing out new internal combustion cars by 2035,” The Verge, 14-Jul-2021. [Online]. Available: https://www.theverge.com/2021/7/14/22576994/european-union-gas-vehicle-ice-ban-2035-electric-cars. [Accessed: 23-Aug-2021].

[3] J. Woo, H. Choi, J. Ahn, "Well-to-wheel analysis of greenhouse gas emissions for electric vehicles based on electricity generation mix: A global perspective", Transportation Research Part D Transport and Environment, 2017, vol. 51, pp. 340-350, doi: 10.1016/j.trd.2017.01.005.

[4] “2021 U.S. electric vehicle Experience (EVX) Ownership Study,” J.D. Power, 21-Jan-2021. [Online]. Available: https://www.jdpower.com/business/press-releases/2021-us-electric-vehicle-experience-evx-ownership-study. [Accessed: 23-Aug-2021].

[5] D. Pevec, J. Babic, A. Carvalho, Y. Ghiassi-Farrokhfal, W. Ketter and V. Podobnik, "Electric Vehicle Range Anxiety: An Obstacle for the Personal Transportation (R)evolution?," 2019 4th International Conference on Smart and Sustainable Technologies (SpliTech), 2019, pp. 1-8, doi: 10.23919/SpliTech.2019.8783178.

[6] M. Nilsson, “Electric Vehicles The Phenomenon of Range Anxiety”, Accessed: 2021, [Online]. Available: http://e-mobility-nsr.eu/fileadmin/user_upload/downloads/info-pool/the_phenomenon_of_range_anxiety_elvire.pdf.

[7] M. Naguib, P. Kollmeyer and A. Emadi, "Lithium-Ion Battery Pack Robust State of Charge Estimation, Cell Inconsistency, and Balancing: Review," in IEEE Access, vol. 9, pp. 50570-50582, 2021, doi: 10.1109/ACCESS.2021.3068776.

[8] G. Plett, “Battery Management Systems, Volume I: Battery Modeling”, Artech, 2015.

[9] MIT Electric Vehicle Team. “A Guide to Understanding Battery Specifications”, December 2008, Accessed: 2021, [Online]. Available: http://web.mit.edu/evt/summary_battery_specifications.pdf.

[10] C. Vidal, O. Gross, R. Gu, P. Kollmeyer and A. Emadi, "xEV Li-Ion Battery Low-Temperature Effects—Review," in IEEE Transactions on Vehicular Technology, vol. 68, no. 5, pp. 4560-4572, May 2019, doi: 10.1109/TVT.2019.2906487.

[11] W. Liu, “Hybrid Electric Vehicle System Modelling and Control”, Wiley, 2017.

[12] A. Emadi, Advanced Electric Drive Vehicles, CRC Press, 2014.

[13] P. Lin, P. Jin, J. Hong, and Z. Wang, “Battery voltage and state of power prediction based on an improved novel polarization voltage model,” Energy Reports, vol. 6, pp. 2299–2308, 2020, doi: http://dx.doi.org/10.1016/j.egyr.2020.08.014.

[14] S. Xiang, G. Hu, R. Huang, F. Guo, and P. Zhou, “Lithium-Ion battery online Rapid STATE-OF-POWER Estimation under multiple constraints,” Energies, vol. 11, no. 2, p. 283, 2018, doi: http://dx.doi.org/10.3390/en11020283.

[15] M. Berecibar, I. Gandiaga, I. Villarreal, N. Omar, J. Van Mierlo, and P. Van den Bossche, “Critical review of state of Health estimation methods of Li-ion batteries for real applications,” Renewable and Sustainable Energy Reviews, vol. 56, pp. 572–587, 2016, doi: https://doi.org/10.1016/j.rser.2015.11.042.

[16] B. Pattipati, C. Sankavaram and K. Pattipati, "System Identification and Estimation Framework for Pivotal Automotive Battery Management System Characteristics," in IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 41, no. 6, pp. 869-884, Nov. 2011, doi: 10.1109/TSMCC.2010.2089979.

[17] M. Naguib, C. Vidal, P. Kollmeyer, P. Malysz, O. Gross, and A. Emadi, “Comparative study between equivalent circuit and recurrent neural network battery voltage models,” SAE Technical Paper Series, 2021, doi: https://doi.org/10.4271/2021-01-0759.

[18] W. He, M. Pecht, D. Flynn, and F. Dinmohammadi, “A physics-based electrochemical model for lithium-ion battery state-of-charge estimation solved by an optimised projection-based method and moving-window filtering,” Energies, vol. 11, no. 8, p. 2120, 2018, doi: https://doi.org/10.3390/en11082120.

[19] C. Zhang, K. Li, S. Mcloone, and Z. Yang, “Battery modelling methods for electric vehicles - a review,” 2014 European Control Conference (ECC), 2014, doi: http://dx.doi.org/10.1115/1.4002475.

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

[21] R. Zhao, P. J. Kollmeyer, R. D. Lorenz and T. M. Jahns, "A Compact Methodology

Via a Recurrent Neural Network for Accurate Equivalent Circuit Type Modeling of Lithium-Ion Batteries," in IEEE Transactions on Industry Applications, vol. 55, no. 2, pp. 1922-1931, March-April 2019, doi: 10.1109/TIA.2018.2874588.

[22] C. Vidal, P. Malysz, P. Kollmeyer and A. Emadi, "Machine Learning Applied to Electrified Vehicle Battery State of Charge and State of Health Estimation: State-of-the-Art," in IEEE Access, vol. 8, pp. 52796-52814, 2020, doi: 10.1109/ACCESS.2020.2980961.

[23] Q.-K. Wang, Y.-J. He, J.-N. Shen, Z.-F. Ma, and G.-B. Zhong, “A unified modeling framework for lithium-ion batteries: An artificial neural network based thermal coupled equivalent circuit model approach,” Energy, vol. 138, pp. 118–132, 2017, doi:https://doi.org/10.1016/j.energy.2017.07.035.

[24] C. Sbarufatti, M. Corbetta, M. Giglio, and F. Cadini, “Adaptive prognosis of lithium-ion batteries based on the combination of particle filters and radial basis function neural networks,” Journal of Power Sources, vol. 344, pp. 128–140, 2017, doi: https://doi.org/10.1016/j.jpowsour.2017.01.105.

[25] Q. Yu, R Xiong, L. Wang, C. Lin, “A Comparative Study on Open Circuit Voltage Models for Lithium-ion Batteries”, Chinese Journal of Mechanical Engineering,vol 31, (2018), doi: https://doi.org/10.1186/s10033-018-0268-8

[26] D. Sun, X. Yu, C. Wang, C. Zhang, R. Huang, Q. Zhou, T. Amietszajew, and R. Bhagat, “State of charge estimation for lithium-ion battery based on an intelligent adaptive extended kalman filter with improved noise estimator,” Energy, vol. 214, p. 119025, 2021 doi: https://doi.org/10.1016/j.energy.2020.119025.

[27] I. Jokić, Ž. Zečević and B. Krstajić, "State-of-charge estimation of lithium-ion batteries using extended Kalman filter and unscented Kalman filter," 2018 23rd International Scientific-Professional Conference on Information Technology (IT), Zabljak, Montenegro, 2018, pp. 1-4, doi: 10.1109/SPIT.2018.8350462.

[28] V. Sangwan, R. Kumar and A. K. Rathore, "State-of-charge estimation for li-ion battery using extended Kalman filter (EKF) and central difference Kalman filter (CDKF)," 2017 IEEE Industry Applications Society Annual Meeting, Cincinnati, OH, USA, 2017, pp. 1-6, doi: 10.1109/IAS.2017.8101722.

[29] L. Haoran, L. Liangdong, Z. Xiaoyin and S. Mingxuan, "Lithium Battery SOC Estimation Based on Extended Kalman Filtering Algorithm," 2018 IEEE 4th International Conference on Control Science and Systems Engineering (ICCSSE), Wuhan, China, 2018, pp. 231-235, doi: 10.1109/CCSSE.2018.8724766.

[30] D. Yang, G. Qi and X. Li, "State-of-charge estimation of LiFePO4/C battery based on extended Kalman filter," 2013 IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC), Hong Kong, China, 2013, pp. 1-5, doi: 10.1109/APPEEC.2013.6837188.

[31] Chirag (2021). Design and Test Lithium Ion Battery Management Algorithms. [Online]. Available: https://www.mathworks.com/matlabcentral/fileexchange/72865-design-and-test-lithium-ion-battery-management-algorithms. [Accessed: 2021].

[32] E. Chemali, P. J. Kollmeyer, M. Preindl, and A. Emadi, “State-of-charge estimation of Li-ion batteries using deep neural networks: A machine learning approach,” Journal of Power Sources, vol. 400, pp. 242–255, 2018, doi: https://doi.org/10.1016/j.jpowsour.2018.06.104

[33] E. Chemali, P. J. Kollmeyer, M. Preindl, R. Ahmed and A. Emadi, "Long Short-Term Memory Networks for Accurate State-of-Charge Estimation of Li-ion Batteries," in IEEE Transactions on Industrial Electronics, vol. 65, no. 8, pp. 6730-6739, Aug. 2018, doi: 10.1109/TIE.2017.2787586.

[34] M. S. Sidhu, D. Ronanki and S. Williamson, "State of Charge Estimation of Lithium-Ion Batteries Using Hybrid Machine Learning Technique," IECON 2019 - 45th Annual Conference of the IEEE Industrial Electronics Society, 2019, pp. 2732-2737, doi: 10.1109/IECON.2019.8927066.

[35] C. Vidal, P. Kollmeyer, M. Naguib, P. Malysz, O. Gross, and A. Emadi, “Robust xEV Battery State-of-Charge Estimator Design using Deep Neural Networks,” in Proc WCX SAE World Congress Experience, Detroit, MI, Apr 2020.

[36] Y. Liu, X. Shu, H. Yu, J. Shen, Y. Zhang, Y. Liu, and Z. Chen, “State of charge prediction framework for lithium-ion batteries incorporating long short-term memory network and transfer learning,” Journal of Energy Storage, vol. 37, p. 102494, 2021, doi: 10.1016/j.est.2021.102494.

[37] C. Vidal, P. Kollmeyer, E. Chemali and A. Emadi, "Li-ion Battery State of Charge Estimation Using Long Short-Term Memory Recurrent Neural Network with Transfer Learning," 2019 IEEE Transportation Electrification Conference and Expo (ITEC), Detroit, MI, USA, 2019, pp. 1-6.

[38] J. C. Álvarez Antón, P. J. García Nieto, C. Blanco Viejo and J. A. Vilán Vilán, "Support Vector Machines Used to Estimate the Battery State of Charge," in IEEE Transactions on Power Electronics, vol. 28, no. 12, pp. 5919-5926, Dec. 2013, doi: 10.1109/TPEL.2013.2243918.

[39] I. A. Majid, R. F. Rahman, N. A. Setiawan and A. I. Cahyadi, "Electric vehicle battery dynamics modelling using support vector machine," 2013 Joint International Conference on Rural Information & Communication Technology and Electric-Vehicle Technology (rICT & ICeV-T), 2013, pp. 1-3, doi: 10.1109/rICT-ICeVT.2013.6741500.

[40] U.S Department of Energy, “2019 Telsa Model 3 Standard Range”, https://www.fueleconomy.gov/feg/Find.do?action=sbs&id=41415, (Accessed 2021).

[41] B. Varga, A. Sagoian, and F. Mariasiu, “Prediction of electric vehicle range: A comprehensive review of current issues and challenges,” Energies, vol. 12, no. 5, p. 946, 2019, https://doi.org/10.3390/en12050946.

[42] J. A. Oliva, C. Weihrauch and T. Bertram, "Model-based remaining driving range prediction in electric vehicles by using particle filtering and Markov chains," 2013 World Electric Vehicle Symposium and Exhibition (EVS27), 2013, pp. 1-10, doi: 10.1109/EVS.2013.6914989.

[43] L. Zhao, W. Yao, Y. Wang and J. Hu, "Machine Learning-Based Method for Remaining Range Prediction of Electric Vehicles," in IEEE Access, vol. 8, pp. 212423-212441, 2020, doi: 10.1109/ACCESS.2020.3039815.

[44] A. Bolovinou, I. Bakas, A. Amditis, F. Mastrandrea and W. Vinciotti, "Online prediction of an electric vehicle remaining range based on regression analysis," 2014 IEEE International Electric Vehicle Conference (IEVC), 2014, pp. 1-8, doi: 10.1109/IEVC.2014.7056167.

[45] I. Miri, A. Fotouhi, and N. Ewin, “Electric vehicle energy consumption modelling and estimation—a case study,” International Journal of Energy Research, vol. 45,

no. 1, pp. 501–520, 2020, https://doi.org/10.1002/er.5700

[46] M. Mruzek, I. Gajdáč, Ľ. Kučera, and D. Barta, “Analysis of parameters influencing electric vehicle range,” Procedia Engineering, vol. 134, pp. 165–174, 2016, doi: https://doi.org/10.1016/j.proeng.2016.01.056.

[47] M. Smuts, B. Scholtz, and J. Wesson, “A critical review of factors influencing the remaining driving range of electric vehicles,” 2017 1st International Conference on Next Generation Computing Applications (NextComp), 2017, doi: 10.1109/NEXTCOMP.2017.8016198.

[48] H. A. Yavasoglu, Y. E. Tetik, and K. Gokce, “Implementation of machine learning based real time range estimation method without destination knowledge for bevs,” Energy, vol. 172, pp. 1179–1186, 2019, https://doi.org/10.1016/j.energy.2019.02.032.

[49] A. Enthaler, T. Weustenfeld, F. Gauterin and J. Koehler, "Thermal management consumption and its effect on remaining range estimation of electric vehicles," 2014 International Conference on Connected Vehicles and Expo (ICCVE), 2014, pp. 170-177, doi: 10.1109/ICCVE.2014.7297537.

[50] C. De Cauwer, W. Verbeke, T. Coosemans, S. Faid, and J. Van Mierlo, “A data-driven method for energy consumption prediction and energy-efficient routing of electric vehicles in real-world conditions,” Energies, vol. 10, no. 5, p. 608, 2017, doi: https://doi.org/10.3390/en10050608

[51] H. Rahimi-Eichi and M. Chow, "Big-data framework for electric vehicle range estimation," IECON 2014 - 40th Annual Conference of the IEEE Industrial Electronics Society, 2014, pp. 5628-5634, doi: 10.1109/IECON.2014.7049362.

[52] Y. Zhang, W. Wang, Y. Kobayashi and K. Shirai, "Remaining driving range estimation of electric vehicle," 2012 IEEE International Electric Vehicle Conference, 2012, pp. 1-7, doi: 10.1109/IEVC.2012.6183172.

[53] B. Zheng, P. He, L. Zhao and H. Li, "A Hybrid Machine Learning Model for Range Estimation of Electric Vehicles," 2016 IEEE Global Communications Conference (GLOBECOM), 2016, pp. 1-6, doi: 10.1109/GLOCOM.2016.7841506.

[54] S. S. Sonalikar and S. D. Shelke, "Estimation of Remaining Range of Electric Vehicle Using Kalman Filter," 2018 International Conference on Inventive Research in Computing Applications (ICIRCA), 2018, pp. 632-636, doi: 10.1109/ICIRCA.2018.8596767.

[55] X. Liu, J. Wu, C. Zhang, and Z. Chen, “A method for state of energy estimation of lithium-ion batteries at dynamic currents and temperatures,” Journal of Power Sources, vol. 270, pp. 151–157, 2014, doi: https://doi.org/10.1016/j.jpowsour.2014.07.107.

[56] K. Li, F. Wei, K. J. Tseng and B. Soong, "A Practical Lithium-Ion Battery Model for State of Energy and Voltage Responses Prediction Incorporating Temperature and Ageing Effects," in IEEE Transactions on Industrial Electronics, vol. 65, no. 8, pp. 6696-6708, Aug. 2018, doi: 10.1109/TIE.2017.2779411.

[57] Pan, Chaofeng, et al. “Driving Range Estimation for Electric Vehicles Based on Driving Condition Identification and Forecast.” AIP Advances, vol. 7, no. 10, Oct. 2017, p. 105206, 10.1063/1.4993945. Accessed 15 Nov. 2021.

[58] M. Ceraolo and G. Pede, "Techniques for estimating the residual range of an electric vehicle," in IEEE Transactions on Vehicular Technology, vol. 50, no. 1, pp. 109-115, Jan. 2001, doi: 10.1109/25.917893.

[59] M. Ismail, R. Dlyma, A. Elrakaybi, R. Ahmed and S. Habibi, "Battery state of charge estimation using an Artificial Neural Network," 2017 IEEE Transportation Electrification Conference and Expo (ITEC), Chicago, IL, USA, 2017, pp. 342-349.

[60] Jun Xu, Chunting Chris Mi, Binggang Cao and Junyi Cao, "A new method to estimate the state of charge of lithium-ion batteries based on the battery impedance model," *Journal of Power Sources*, vol. 233, 2013, pp. 277-284.

[61] A. Nugroho, E. Rijanto, F. D. Wijaya and P. Nugroho, "Battery state of charge estimation by using a combination of Coulomb Counting and dynamic model with adjusted gain," 2015 International Conference on Sustainable Energy Engineering and Application (ICSEEA), Bandung, Indonesia, 2015, pp. 54-58.

[62] I. Baccouche, S. Jemmali, B. Manai, R. Chaibi and N. E. Ben Amara, "Hardware implementation of an algorithm based on kalman filtrer for monitoring low capacity Li-ion batteries," 2016 7th International Renewable Energy Congress (IREC), Hammamet, Tunisia, 2016.

[63] C. Chang, Y. Zheng, and Y. Yu, “Estimation for Battery State of Charge Based on Temperature Effect and Fractional Extended Kalman Filter,” Energies, vol. 13, no. 22, p. 5947, Nov. 2020.

[64] Q.-Z. Zhang, Z.-Y. Wang, and H.-M. Yuan, Estimation for SOC of Li-ion battery based on two-order RC temperature model, 28-Jun-2018. [Online]. Available: https://ieeexplore-ieee-org.libaccess.lib.mcmaster.ca/document/8398150. [Accessed: 2020].

[65] Kollmeyer, Phillip; Skells, Michael (2020), “Turnigy Graphene 5000mAh 65C Li-ion Battery Data”, Mendeley Data, V1, doi: 10.17632/4fx8cjprxm.1

[66] R. Ahmed, "OCV-RRC Models in State Space, Kalman Filtering and State of

Charge Estimation," MECH ENG 754 Management and Control of Electric Vehicle Batteries Lecture 7, McMaster University, Hamilton, Ontario, Fall 2020.

[67] Daoming Sun, Xiaoli Yu, Chongming Wang, Cheng Zhang, Rui Huang, Quan Zhou, Taz Amietszajew, Rohit Bhagat "State of charge estimation for lithium-ion battery based on an Intelligent Adaptive Extended Kalman Filter with improved noise estimator," Energy, vol. 214, 2021.

[68] Kollmeyer, Phillip (2018), “Panasonic 18650PF Li-ion Battery Data”, Mendeley Data, V1, doi: 10.17632/wykht8y7tg.1

[69] Naguib, Mina; Kollmeyer, Phillip; Skells, Michael (2020), “LG 18650HG2 Li-ion Battery Data and Example Deep Neural Network xEV SOC Estimator Script”, Mendeley Data, V1, doi: 10.17632/b5mj79w5w9.1

[70] F. Khanum, E. Louback, F. Duperly, C. Jenkins, P. J. Kollmeyer and A. Emadi, "A Kalman Filter Based Battery State of Charge Estimation MATLAB Function," 2021 IEEE Transportation Electrification Conference & Expo (ITEC), 2021, pp. 484-489, doi: 10.1109/ITEC51675.2021.9490163.

[71] J. Gazzarri, "Modeling Batteries Using Simulink and Simscape," Mathworks, [Online]. Available: https://www.mathworks.com/videos/modeling-batteries-using-simulink-and-simscape-1562930245321.html. [Accessed 19 December 2020].

[72] The Idaho National Laboratory, "Battery Test Manual for Plug-In Hybrid Electric Vehicles," U.S. Department of Energy, Idaho Falls, Idaho, 2010.

[73] Florian Knorn (2021). M-code LaTeX Package. [Online]. Available: https://www.mathworks.com/matlabcentral/fileexchange/8015-m-code-latex-package.

[74] "2019 Tesla Model 3 Standard Range plus RWD - Specifications and Price." EVSpecifications, https://www.evspecifications.com/en/model/bc5896. Accessed 22 Nov. 2021.

[75] P. Kollmeyer, "Electromechanical Modeling and Analysis of a Corbin Sparrow Electric Vehicle using On the Road Data," M.S. Thesis, University of Wisconsin - Madison, 2011.

[76] O. US EPA, "Annual Certification Data for Vehicles, Engines, and Equipment," www.epa.gov, Nov. 08, 2017. https://www.epa.gov/compliance-and-fuel-economy-data/annual-certification-test-data-vehicles-and-engines (accessed Oct. 04, 2021).

[77] P. G. Anselma, P. Kollmeyer, J. Lempert, Z. Zhao, G. Belingardi, and A. Emadi, "Battery state-of-health sensitive energy management of hybrid electric vehicles: Lifetime prediction and ageing experimental validation," Applied Energy, vol. 285, p. 116440, Mar. 2021, doi: 10.1016/j.apenergy.2021.116440.

[78] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," in IEEE Transactions on Signal Processing, vol. 45, no. 11, pp. 2673-2681, Nov. 1997, doi: 10.1109/78.650093.

[79] K. Yao, G. Zweig, M. Hwang, Y. Shi, D. Yu, "Recurrent Neural Networks for Language Understanding" in Interspeech, Aug. 2013, doi: 10.13140/2.1.2755.3285.

[80] H. Sak, A. Senior, F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling", Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH, pp. 338-342, 2014.

[81] R. Pascanu, T. Mikolov, Y. Bengio, “On the difficulty of training Recurrent Neural Networks,” in Proceedings of the 30th International Conference on Machine Learning, 2012, doi: 10.5555/3042817.3043083

[82] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.

[83] D.P. Kingma, J. Ba, Adam: a method for stochastic optimization, CoRR abs/1412.6980, http://arxiv.org/abs/1412.6980.

[84] D. Zhu, J. J. Campbell and G. Cho, "Battery Voltage Prediction Using Neural Networks," 2021 IEEE Transportation Electrification Conference & Expo (ITEC), 2021, pp. 807-812, doi: 10.1109/ITEC51675.2021.9490081.

[85] O. J. Ojo, X. Lin, H. Lang and X. Hu, "A Voltage Fault Detection Method Enabled by A Recurrent Neural Network and Residual Threshold Monitor for Lithium-ion Batteries," 2021 IEEE Transportation Electrification Conference & Expo (ITEC), 2021, pp. 813-820, doi: 10.1109/ITEC51675.2021.9490102.

[86] Li, Shuangqi, et al. “Big Data Driven Lithium-Ion Battery Modeling Method Based on SDAE-ELM Algorithm and Data Pre-Processing Technology.” Applied Energy, vol. 242, May 2019, pp. 1259–1273, 10.1016/j.apenergy.2019.03.154. Accessed 10 Nov. 2021.