2019

SYSTEM MODELING OF A LITHIUM-ION BATTERY

Tyler Creighton
University of Rhode Island, nerdler33@uri.edu

SYSTEM MODELING OF A LITHIUM-ION BATTERY

BY

TYLER CREIGHTON

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE

REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

IN

ELECTRICAL ENGINEERING

UNIVERSITY OF RHODE ISLAND

2019

MASTER OF SCIENCE THESIS

OF

TYLER CREIGHTON

APPROVED:

Thesis Committee:

Major Professor Richard Vaccaro

Tao Wei

Chengzhi Yuan

Nasser H. Zawia

DEAN OF THE GRADUATE SCHOOL

UNIVERSITY OF RHODE ISLAND

2019

ABSTRACT

Lithium Ion Batteries are widely used in today's portable world. A working model is required in order to safely maximize the output of these batteries, and to efficiently choose appropriate sized systems for any given application. A model of this battery would require parameters that are not easy to measure, and need to be estimated. This work further explored on one existing model for the lithium ion battery, the linear dynamics single particle model, and improved upon it. Data was collected then modeled through the use of several matlab functions, and the final model was then used to predict the behavior of different applications of the battery. A variety of high and low currents in a variety of temperatures was used for both data collection and testing. This design began to explore the hypothesis that a single model can be used for the entire range of the batteries capabilities, instead of the multiple that is currently used. The parameters of this model were then extracted and related to real electro-chemical properties of the battery. Initial testing showed promising results for this hypothesis. The resulting model can be used in the development of future battery based electronics reducing cost and maximizing effectiveness.

ACKNOWLEDGMENTS

Throughout the writing of this work I received a large amount of support and assistance. I would first like to thank my major professor, Dr. R. Vaccaro, whose expertise at both engineering and teaching were immensely valuable and appreciated throughout the course of this thesis.

I would also like to thank the engineers at Electro Standards Laboratory: K. Waterman and J. Turdino. Your help was monumental in the data collection and processing for this study, as well as extracting the original electro-chemical parameters. I would like to thank you for your excellent cooperation and for the opportunity I was given to conduct my research with your guidance.

TABLE OF CONTENTS

LIST OF FIGURES

Figure **Page**

CHAPTER 1

Introduction

The Lithium Ion (Li-Ion) Battery is a widely used source of portable power in many electronics. These batteries, like most other, come in discrete cells that can be stacked together to create a larger power block. The basic dynamics of the Li-Ion battery can be described as the reversible transport of the lithium ions from a negatively charged electrode (anode) to the positively charged electrode (cathode) [3]. These ions can then be moved back to the anode through a recharging process, which allows reuse of the same cell. Over time this leads to the degradation of the battery, changing many of its physical properties. These properties also change with the state of charge (SoC) of the cell, as well as the temperature the cell is being operated at. The combination of these variables makes the Li-Ion cell difficult to model, and a definitive method hasn't yet been created. The specific goal is to create a model based on physical data that accurately matches output voltage based on a specific input current profile. This model will be created by estimating the values of a state space equivalent of the battery, which when combined with state of charge and temperature data, will provide output voltage given an input current. These values will be calculated using the subspace identification method [4], back tracing the measured voltage through the model to calculate both the input and output of the state space block, which in turn will give the parameters of said block.

In order to properly use the cell in complex, rugged designs, a reliable model is required. Many different types of battery models have been attempted, ranging from simple linear equivalent circuit models to sophisticated electro-chemical models involving nonlinearities and partial differential equations. The currently used

models work in idealized scenarios and often deal in finite cases. For this work an improvement upon the electro-chemical single-particle model developed in [2] will be used to create a model that can adapt to state of charge and temperature. This model uses input state of charge and current data to predict the output voltage of a system, going through a dynamics block with adjustable parameters based on SoC, then finally through an Open Circuit Potential (OCP) gain curve. The OCP curve is a non-linear curve as a function of SoC. Temperature will change the values of the OCP curve, and will be the added functionality of the design. The current data is also sent through an additional parameter, a constant over potential gain block, and summed with the output of the OCP block. In practice, having an accurate I to V model of the Li-Ion battery will allow for both a better understanding of the individual parameters of the cells themselves, as well as a baseline to design additional systems that work in conjunction with the cell. This would allow for a realistic approximation of expected results rather than exhaustive testing of the added system.

CHAPTER 2

Review of literature

There are many methods currently used to model a Lithium Ion Cell. It is important to understand what is happening with a Li-ion battery before trying to represent it. The cell can be broken up into two half cells, each with an electrolyte which submerges an electrode, either the cathode or the anode. These two half cells are separated by a medium referred to as a separator. This separator is attributed to the battery's internal resistance, and is often ignored in modeling. The cell behaves similarly in both charging and discharging modes, where lithium ions are transferred from one electrode, depending on the direction of the charge, through the electrolyte and separator into the other electrode [5].

Two major models currently existing for lithium ion cells are the equivalent circuit model and the electro-chemical model. The equivalent circuit approximation uses exclusively electronic components in the form of capacitors and resistors to represent the battery. The resulting model can be seen in figure 1 and is a voltage source going through a resistor, followed by a series of RC circuits [1, 6]. This model has the advantage of being very simple to calculate and use, as it deals with exclusively linear modeling. The voltage source of this model represents the open circuit potential (OCP) of the system, and the first resistor represents the internal resistance and over potential gain of the system. The remaining n RC circuits represent the nth order dynamics of the system. These will be explained in further detail later.

The electro-chemical model is derived directly for the chemistry of the battery rather than its behavior. Two variations of it are the single particle model described in [2, 5, 6, 7, 8, 9] and the Doyle-Fuller-Newman model described in [2, 5]. The

single particle model (SP model) will be the main focus of this work. This models the diffusion process of the ions moving from one electrode to the other through the electrolyte. The SP model can vary in complexity at the cost of accuracy. The particular version of the SP model we are interested in is the Linear Dynamic SP model described in [2]. This model has a linear dynamics block whose output is sent through the nonlinear resting Open Circuit potential block before being added to the over potential gain output of the system. This can be seen in Figure 2. Jenkins describes the Linear Dynamics Model as two linear differential equations, one of which is simply an integrator of the current of the system. As such the design used in this work splits the dynamics block into two portions, a current integrator, and a dynamics block, the design of which will be shown in the next chapter. "The output voltage is a nonlinear function of the input current and both electrode surface concentrations and is given by"

$$V = V_{OCP} + V_{OVER} \tag{1}$$

where OCP is a function of state of charge and OVER is a function of current and state of charge [2]. Estimating state of charge can be achieved through the combination of the two linear equations discussed previously.

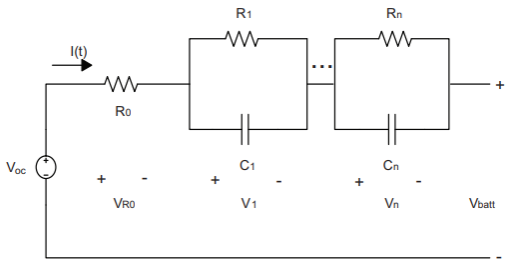

Figure 1. Equivalent Circuit model of the Li-Ion battery [1]

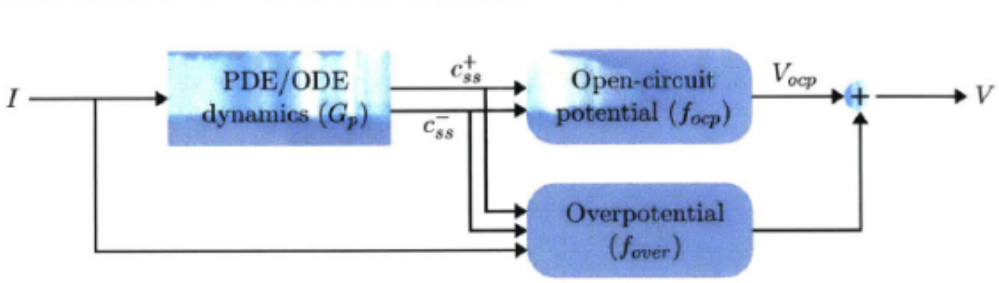

Figure 2. Jenkins model of a linear dynamics single particle model [2]

CHAPTER 3

Model Overview

The ending model of this work is based off of the model described in [2], which can be seen in Figure 3

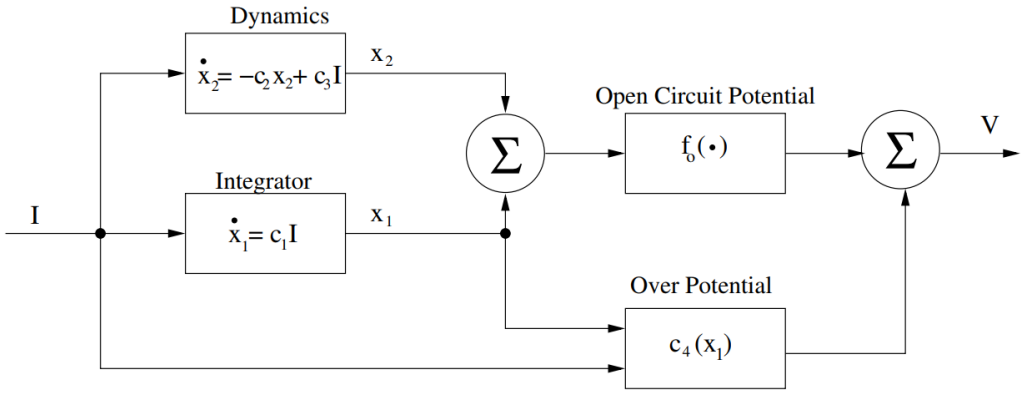

Figure 3. The basic block diagram of the model

The current goes through the dynamics block as well as an integrator, is added together, then goes through the nonlinear resting Open Circuit Potential(OCP) gain before being added with the output of the Over Potential gain. Both OCP and Over Potential are functions of state of charge (SoC). In order to create the dynamics needed for this model, the state space identification toolbox in matlab was used. This function returns the appropriate state space model of a block if both the input and output of the block are known. The input of the block is the known current value, however the output needs to be calculated by tracing the model in reverse. Starting with voltage, the output of the over potential gain is subtracted to reach the OCP gain.

In order to back-trace through the non linear block, an auxiliary function would search the curve for the input based off of the given output. This was realized through a search function based on a binary search algorithm. Starting

with the first and last point of the curve, the midpoint is calculated then compared with the desired output. The midpoint replaces either the begging point or the ending point based on where it was in comparison to the desired value. This process continues until the two points are within an acceptable range, where the midpoint is used as the return value.

Once the input of the OCP block is calculated, the output of the integrator block is a trivial calculation, and is subtracted from the OCP input. This gives the output of the dynamics block which when combined with the input gives the parameters of the block.

3.1 Calculating OCP Curve

In order to calculate the OCP curve, a rest phase of 10 minutes was added to the current profile after every ten minutes of activity shown in Figures 4 and 5. This is used to allow the contribution of the dynamics to settle to zero to isolate the OCP at any given state of charge. The actual settling time of the system varies based on SoC, and in some instances can be closer to an hour, however the actual OCP value can be calculated by fitting an exponential decay to match the modeled ten minute portion to find the actual OCP at the given SoC. The matlab optimization toolbox was used to fit the curve, optimizing over the function: OCP1

$$f = norm(V - c_1(1 - e^{-c_2 * t}) - c_3)^2 \tag{2}$$

Where c_3 is the start point of a segment, $4.62/c_2$ is the settling time, and c_1 is the difference between the start and end point of a segment. $c_1 + c_3$ gives the settled value of the segment, which is the OCP value of the segment.

The current profile was chosen such that each segment is around a five percent gain/loss of SoC. This allowed for enough points to use spline interpolation to create a curve for the full range of SoC. The final OCP curve for the initial model can be seen in figure 6 OCP values for a zero percent SoC and 100 percent SoC

were taken from the data sheet of the battery, since safety cutoffs for the battery prevent accurate readings under these conditions.

3.2 Optimizing parameters and over potential calculations

At the start of the modeling process, it was assumed the Over Potential gain could be simplified to a constant value based on the results of Jenkins [2]. A preliminary value was obtained based on the battery chemistry, and was then optimized with several other parameters including initial SoC, and the first several points of the OCP curve. Initial modeling results showed all segments with greater than a thirty percent SoC fit extremely well with the data, but those prior to this point showed large error. By optimizing over the OCP curve in this range, as well as the over potential gain, these segments matched closer to the measured data, but was still showing considerable error. This can be seen in Figures 7 and 8. However, when modeling just the first two segments of the charge data, a much closer fit was obtained, which can be seen in Figures 9 and 10. Upon closer examination, the over potential constant was much higher in this model, with a value of 0.06 compared to the value of the whole model being closer to 0.045. This process was done multiple times to see the optimal over potential constant for segments 2 and 3 (Figures 11 and 12) and segments 3 and 4 (Figures 13 and 14). The over potential constant continued to decay towards the value found when modeling the entire charge portion of the data. The over potential constant was changed to be a piece-wise linear function, with the first thirty percent SoC being a negative sloped line running through the point 0.06 at 0 SoC and 0.045 at 30 SoC. The entire data set was then modeled using this new over potential calculation. The results of this can be seen in Figures 15 and 16.

At this point, the constant Over Potential gain was changed to have the same constant value over thirty percent SoC, but now includes an exponential decaying

segment from zero to thirty percent SoC. The exponential was modeled with the following functions:

$$\boldsymbol{\alpha} = \frac{4.62}{S} \tag{3}$$

$$\boldsymbol{\beta} = \boldsymbol{\alpha} * (Y - c) \tag{4}$$

fover

$$f = Y - (\frac{\boldsymbol{\beta}}{\boldsymbol{\alpha}}) * (1 - e^{-\boldsymbol{\alpha} * SoC}) \tag{5}$$

where S is the settling time of the exponential, which in this case is the point where the function returns to its constant value, c, or thirty percent SoC, and Y is the Over Potential value at zero SoC. This Y value was obtained by optimizing the over potential value of a single segment, which accurately modeled the segment. The S and Y and c values were added to the optimization of the model as a whole, leaving nine total parameters being optimized at any one time. The over potential curve can be seen in Figure 17

The cost function being optimized over recalculates the spline of the changing OCP curve. It takes the current profile of the given segment and the initial state of charge as input arguments, and generates the SoC curve for the given segment. The previously mentioned inverted OCP spline function is used to get the input value of the OCP block. This is subtracted from the SoC value to retrieve the output of the dynamics block. This output, as well as the input are sent to the system identification toolbox to generate the dynamics parameter. The output to the OCP block is then calculated using the new dynamics block, and its value is added together with the over potential output. This final value is compared to the actual output voltage by minimizing:

$$\frac{||OCP_{out} + Over_{out} - V_{out}||}{||V_{out}||} \tag{6}$$

3.3 Current Profile

The current profile used for the modeling was selected based on a few requirements. Each segment of the profile increases/decreases the SoC by five percent. The system identification toolbox requires a persistently exciting input, requiring a profile that has multiple changes in input current. Both large and small currents were used in the creation of the profile, to accurately reflect possible use cases of the batteries. Each current segment begins and ends with a sharp increase of 1.25 percent SoC over 3.5 minutes, with a 5 minute increase of 1.5 percent SoC in the middle. The profile as a whole allows for an accurate model over the full range of SoC. The basis of the current profile was discussed in [2], however it was modified to have larger and smaller currents to satisfy the interesting input requirement of the system identification, as well as to contain the full range of expected currents on the battery in the field.

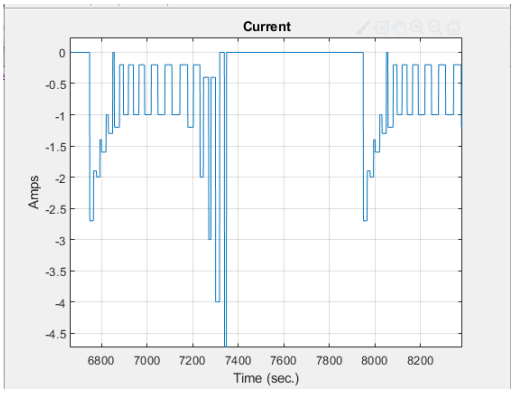

Figure 4. a segment of the current profile used in the modeling

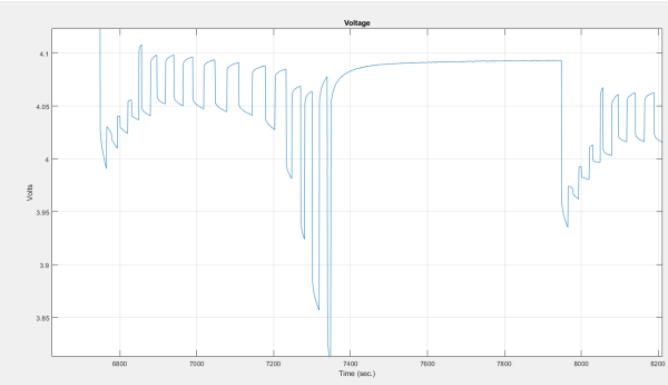

Figure 5. a segment of the voltage response seen from the modeling

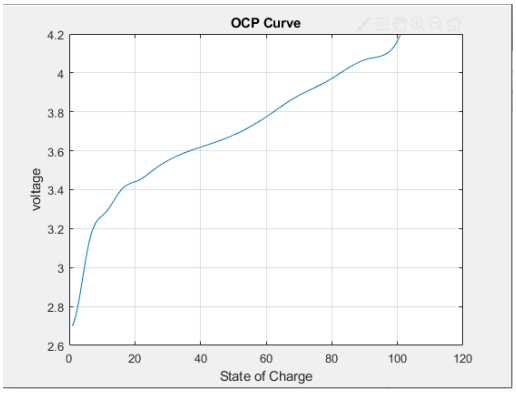

Figure 6. The OCP curve interpolated with matlab spline function

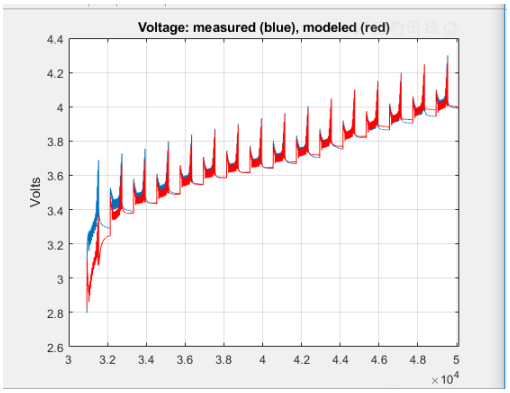

Figure 7. Modeled charge segment with constant over potential gain.

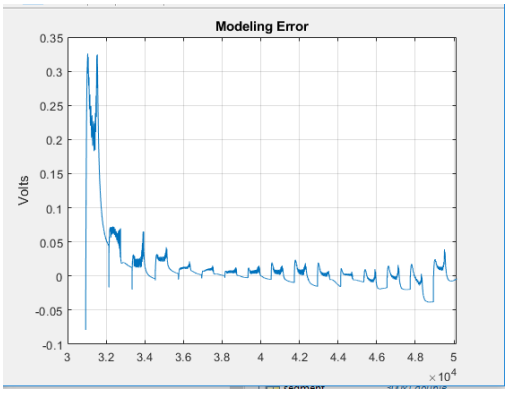

Figure 8. Error of Figure 7, obtained by subtracting modeled from measured voltage

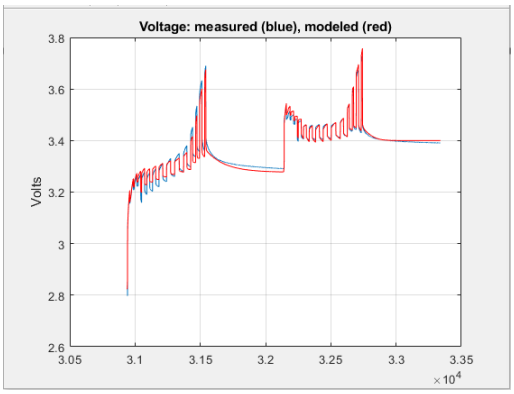

Figure 9. Model of first two segments of charge data with constant over potential gain

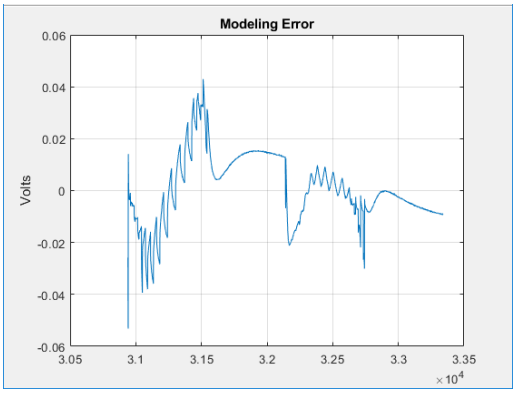

Figure 10. Error of Figure 9, obtained by subtracting modeled from measured voltage

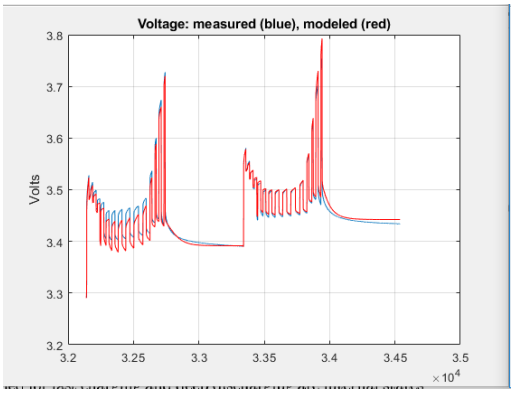

Figure 11. Model of segments 2 and 3 of charge data with constant over potential gain

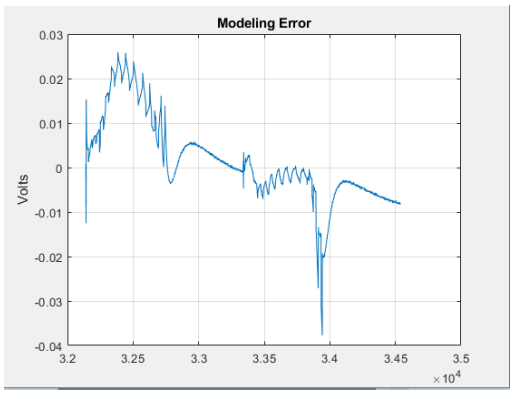

Figure 12. Error of Figure 11, obtained by subtracting modeled from measured voltage

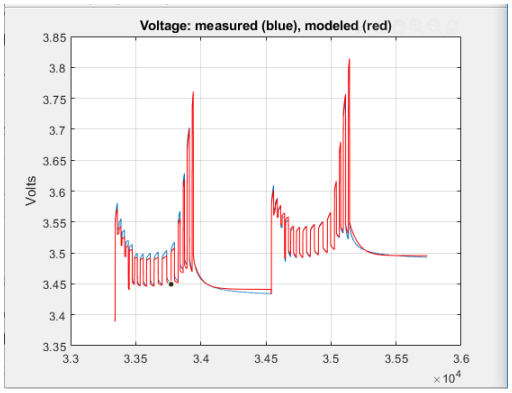

Figure 13. Model of segments 3 and 4 of charge data with constant over potential gain

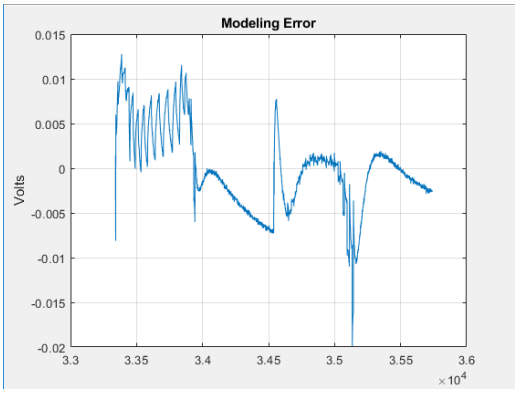

Figure 14. Error of Figure 13, obtained by subtracting modeled from measured voltage

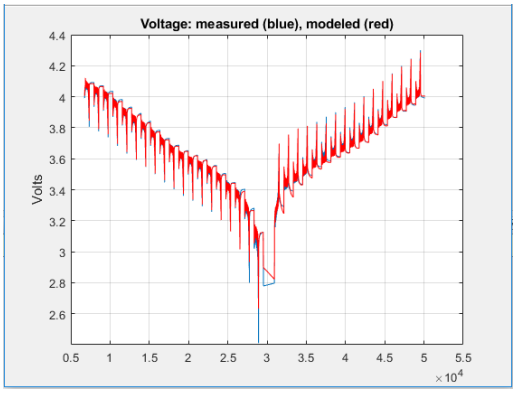

Figure 15. Model of both discharge and charge data using a piecewise linear function for over potential gain

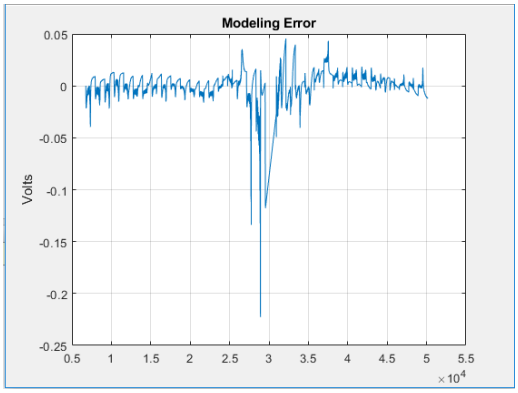

Figure 16. Error of Figure 15, obtained by subtracting modeled from measured voltage

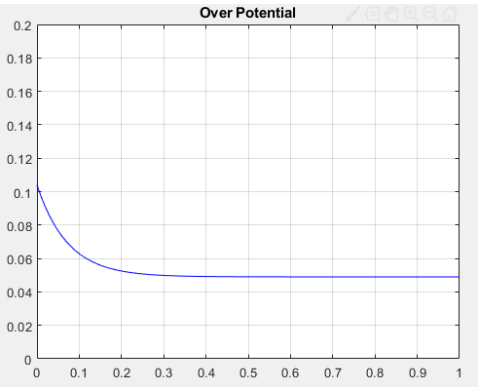

Figure 17. The over potential curve with an exponential decay representation of low SoC values

CHAPTER 4

Results of modeling

The original design and execution of the linear dynamics SP model had multiple instances of the dynamics section for varying states of charge. Each five percent segment of SoC was modeled independently and merged together for a final model. However, once the over potential block was updated to a nonlinear function of state of charge rather than a constant value, this was no longer required. The end result of the modeling showed similar results for both individual models per SoC interval, and a single model for the entire data set. Figures 18 and 19 show the results of using a single model for the entire data set, while Figures 20 and 21 show the results of using multiple models. Looking at the difference it is apparent that the increased complexity and cost of using the multiple models actually yields a worse performance. The ending results were very accurate for higher SoCs, however it starts to vary at lower SoCs. Ignoring the small portion between the charge and discharge portions, there is a sizable error point of 0.13 volts. Outside of this point, the modeled battery behaves within 0.06 volts of the measured battery voltage, with the majority of time being within 0.02 volts.

After the original model was created and verified, several more runs were conducted and modeled using the same initial parameters at both room temperature and a higher temperature of 50C. The cells used in this test were different than the ones in the original model, however all ocp points were within a few mV. The ocp values also did not vary with temperature change. The over potential values did however. Figure 22 shows the initial .35% SoC over potential gain. However, the constant value used for the SoC's greater than .35 was optimized to the same value over both the original model as well as the new runs over both temperatures.

	Fast Pole	Slow Pole	C*B
Room Temp Discharge	-0.2895	-0.0012	0.00077139
Room Temp Charge	-0.3048	0.0001	0.00082668
High Temp Discharge	-0.2897	-0.0001	0.0012
High Temp Charge	-0.3727	-0.0050	0.0014

From this the conclusion follows that these parameters remain relatively static over temperature and cells outside of the initial portion of the over potential gain.

Looking at the other parameters, the eigen values of the a matrix were extracted to obtain the fast and slow poles of the system as well as the unique C*B matrix value to compare the models.

The discharge and charge models for the high temperature data are close enough to be considered the same model, however the dynamics parameters differ between the room temperature discharge and room temperature charge models. The Discharge model from the room temperature data also follows the model for both sets of high temperature data. Additional insight is required to find the difference between discharge and charge sets for this data.

Looking specifically at the results of the new runs, the error is slightly larger than the initial run, however is still within an acceptable margin of around 0.05 V. This error is smaller near the center of the segment and grows as SoC reaches 0 and 1. The results of the additional room temperature run can be seen in Figures 23 and 24 while the results of the high temperature run can be seen in Figures 25 and 26.

4.1 Advantages over Equivalent Circuit Model

This model is an improvement over the equivalent circuit model based on its design. As discussed in chapter 2, the equivalent circuit model has similar components as the linear dynamics SP model, however they are arranged differently. The OCP and over potential values are before the dynamics in that model, which

has the advantage of linear modeling, however it requires multiple dynamics for varying states of charge. While the actual calculations to obtain the model appear to be simpler, the resulting design is inherently more complex. By building the model based on the actual chemistry of the cell, the dynamics block appears before the two nonlinear blocks. While this requires a more complex modeling process to isolate the dynamics, it results in a model that works for all states of charge, and potentially temperatures. Using the methods previously discussed, obtaining the input and output of the linear dynamics is relatively straight forward, and results in an overall simple design for an efficient and accurate model of the lithium ion cell.

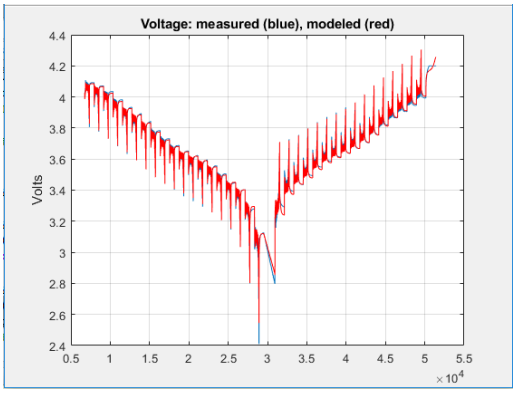

Figure 18. Model of both discharge and charge data using an exponential function for over potential gain, entire charge and discharge data segments were modeled using a single dynamics block

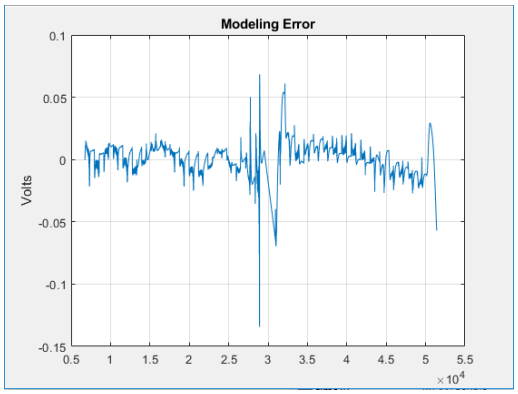

Figure 19. Error of Figure 18, obtained by subtracting modeled from measured voltage

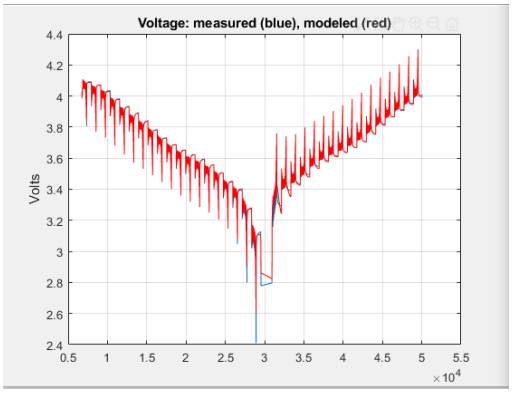

Figure 20. Model of both discharge and charge data using an exponential function for over potential gain, entire charge and discharge data segments were modeled using a separate dynamics block for every 5 percent gain of SoC

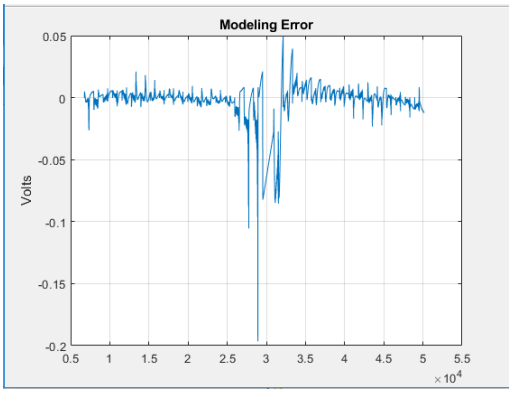

Figure 21. Error of Figure 20, obtained by subtracting modeled from measured voltage

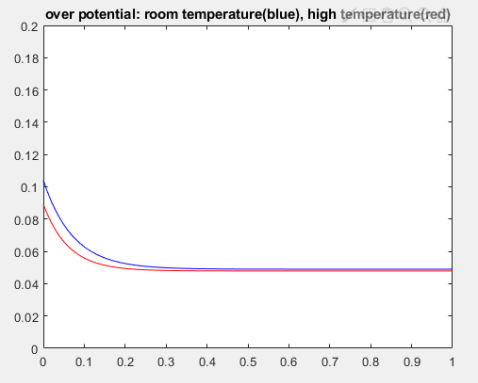

Figure 22. Over Potential gain of high temperature and room temperature data from 0 to 1 SoC

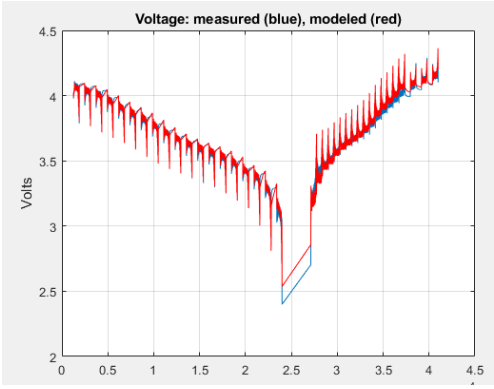

Figure 23. Model of both discharge and charge data at room temperature

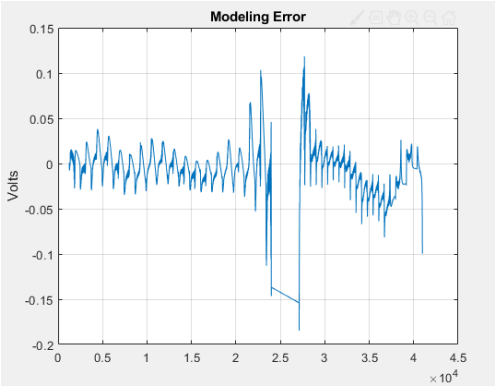

Figure 24. Error of Figure 23, obtained by subtracting modeled from measured voltage

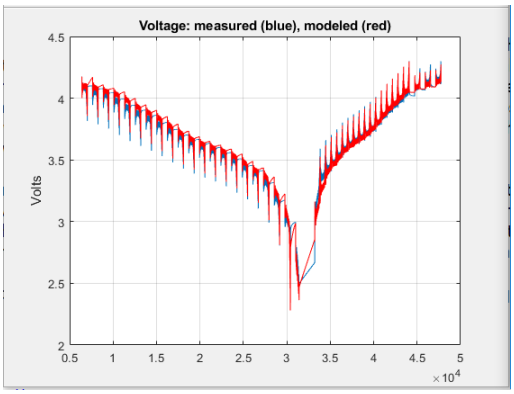

Figure 25. Model of both discharge and charge data at high temperature (50C)

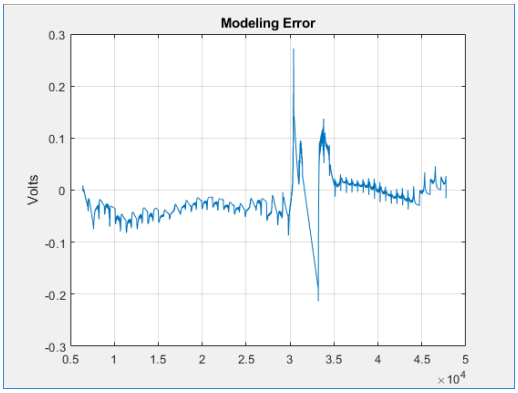

Figure 26. Error of Figure 25, obtained by subtracting modeled from measured voltage

CHAPTER 5

Conclusions and Future Work

Expanding upon this work can be done through higher order dynamics modeling. Through observation, there appears to be two poles in the system. One of which decays within a few minutes, while the other takes significantly longer. The model discussed in this work uses a first order dynamics model, which appears to be sufficient, however a second order design could be pursued to obtain a more accurate representation of the battery.

Other potential avenues for continuation can be seen through creating a fully temperature dependent model. Obtaining more samples of models at different temperature points and interpolating between them to create a full model that can take temperature and current as an input and return an accurate representation of the battery with wildly fluctuating temperature would be the next step in this direction.

Further improving the model to more accurately model high and low SoC values to reduce the larger error near the extremes will be another potential continuation. The model handles extremely well with middling SoC values, but suffers near the extremities of SoC. A potential hybrid design with a few models at the high and low end with a single model encompassing the center portion would take advantage of both designs if the additional precision is required.

Some models had drastically better results than others, using the same methods to both gather and model the data. Further analysis into these difference would help broaden the understanding of the underlying mechanics behind the batteries. This would also help definitively confirm a single model is sufficient to represent all conditions the battery will be in.

Since the parameters of the charge and discharge data sets were so close to each other, it is highly feasible that a single model can be used for both sets. This would not work for the current profiles used in this work due to the discontinuity between the sets, however with a continuous current profile such a model could be constructed. This would be most accurate in a model that ranged from 1-0.02 SoC, as below 0.02 this model starts to degrade.

Currently there is a variable set for the total capacity of a battery. While it remained constant for all tests performed for this work, as a battery is constantly used, its capacity degrades. Due to this, appropriate capacity estimation algorithms will need to be devised to accurately model the full course of a battery's life.

LIST OF REFERENCES

[1] y. Hu, S. Yurkovich, Y. Guezennec, and B. Yurkovich, "A technique for dynamic battery model identification in automotive applications using linear parameter varying structures." *Control Engineering Practice*, vol. 17, pp. 1190–1201, 2009.

[2] B. Jenkins, "Fast adaptive observers for battery management systems," Ph.D. dissertation, Massachusetts Institue of Technology, 2017.

[3] D. W. Limoge, P. Bi, A. Annaswamy, and A. Krupadanam, "A reduced-order model of a lithium-ion cell using the absolute nodal coordinate formulation approach." *IEEE Transactions on Control Systems Technology*, vol. 26, pp. 1001–1014, 2018.

[4] S. Swarup, S. Tan, Z. Liu, Z. Hao, and G. Shi, "Battery state of charge estimation using adaptive subspace identification method." in *2011 9th IEEE International Conference on ASIC*, 2011.

[5] A. Hake and H. Fathy, "Single-particle modeling and experimental parameter identification for a lithium-cobalt-oxide battery cell." Master's thesis, Pennsylvania State University, 2015.

[6] A. S. T. Dao and J. McPhee, "A survey of mathematics-based equivalent-circuit and electrochemical battery models for hybrid and electric vehicle simulation," *Journal of Power Sources*, vol. 256, pp. 410–423, 2014.

[7] S. Moura, "Battery state estimation for a single particle model with electrolyte dynamics." *IEEE Transactions on Control Systems Technology*, vol. 25, pp. 453–468, 2017.

[8] A. Baba and S. Adachi., "Simultaneous state of charge and parameter estimation of lithium-ion battery using log-normalized unscented kalman filter." in *2015 American Control Conference (ACC)*, 2015.

[9] R. Xiong and H. Mu, "Accurate state of charge estimation for lithium-ion battery using dual uncsented kalman filters." in *2017 Chinese Automation Congress (CAC)*, 2017.

AppendixA

make_model.m

```
data=csvread('newdata.csv',1,2);
numSegments = 35;

time1=6749:7348+600; %29510;
%time2=30940:51400;
time2=max(time1)+1:max(time1)+1200;
time3=max(time2)+1:max(time2)+1200;
time4=max(time3)+1:max(time3)+1200;
time5=max(time4)+1:max(time4)+1200;
time6=max(time5)+1:max(time5)+1200;
time7=max(time6)+1:max(time6)+1200;
time8=max(time7)+1:max(time7)+1200;
time9=max(time8)+1:max(time8)+1200;
time10=max(time9)+1:max(time9)+1200;
time11=max(time10)+1:max(time10)+1200;
time12=max(time11)+1:max(time11)+1200;
time13=max(time12)+1:max(time12)+1200;
time14=max(time13)+1:max(time13)+1200;
time15=max(time14)+1:max(time14)+1200;
time16=max(time15)+1:max(time15)+1200;
time17=max(time16)+1:max(time16)+1200;
time18=max(time17)+1:max(time17)+1200;
time19=max(time18)+1:max(time18)+1200;
time20=30940:30939+1200;
time21=max(time20)+1:max(time20)+1200;
time22=max(time21)+1:max(time21)+1200;
time23=max(time22)+1:max(time22)+1200;
time24=max(time23)+1:max(time23)+1200;
time25=max(time24)+1:max(time24)+1200;
time26=max(time25)+1:max(time25)+1200;
time27=max(time26)+1:max(time26)+1200;
time28=max(time27)+1:max(time27)+1200;
time29=max(time28)+1:max(time28)+1200;
time30=max(time29)+1:max(time29)+1200;
time31=max(time30)+1:max(time30)+1200;
time32=max(time31)+1:max(time31)+1200;
time33=max(time32)+1:max(time32)+1200;
time34=max(time33)+1:max(time33)+1200;
time35=max(time34)+1:max(time34)+1200;
```

```
%

 time{1}=time1;
 time{2}=time2;
% time{3}=time3;
% time{4}=time4;
% time{5}=time5;
% time{6}=time6;
% time{7}=time7;
% time{8}=time8;
% time{9}=time9;
% time{10}=time10;
% time{11}=time11;
% time{12}=time12;
% time{13}=time13;
% time{14}=time14;
% time{15}=time15;
% time{16}=time16;
% time{17}=time17;
% time{18}=time18;
% time{19}=time19;
% time(20)=time20;
% time{21}=time21;
% time{22}=time22;
% time{23}=time23;
% time{24}=time24;
% time{25}=time25;
% time{26}=time26;
% time{27}=time27;
% time{28}=time28;
% time{29}=time29;
% time{30}=time30;
% time{31}=time31;
% time{32}=time32;
% time{33}=time33;
% time{34}=time34;
% time{35}=time35;

U{1}=data(time1,2);
YY{1}=data(time1,1);
U{2}=data(time2,2);
YY{2}=data(time2,1);
U{3}=data(time3,2);
YY{3}=data(time3,1);
```

```
U{4}=data(time4,2);
YY{4}=data(time4,1);
U{5}=data(time5,2);
YY{5}=data(time5,1);
U{6}=data(time6,2);
YY{6}=data(time6,1);
U{7}=data(time7,2);
YY{7}=data(time7,1);
U{8}=data(time8,2);
YY{8}=data(time8,1);
U{9}=data(time9,2);
YY{9}=data(time9,1);
U{10}=data(time10,2);
YY{10}=data(time10,1);
U{11}=data(time11,2);
YY{11}=data(time11,1);
U{12}=data(time12,2);
YY{12}=data(time12,1);
U{13}=data(time13,2);
YY{13}=data(time13,1);
U{14}=data(time14,2);
YY{14}=data(time14,1);
U{15}=data(time15,2);
YY{15}=data(time15,1);
U{16}=data(time16,2);
YY{16}=data(time16,1);
U{17}=data(time17,2);
YY{17}=data(time17,1);
U{18}=data(time18,2);
YY{18}=data(time18,1);
U{19}=data(time19,2);
YY{19}=data(time19,1);
U{20}=data(time20,2);
YY{20}=data(time20,1);
U{21}=data(time21,2);
YY{21}=data(time21,1);
U{22}=data(time22,2);
YY{22}=data(time22,1);
U{23}=data(time23,2);
YY{23}=data(time23,1);
U{24}=data(time24,2);
YY{24}=data(time24,1);
U{25}=data(time25,2);
YY{25}=data(time25,1);
```

```
U{26}=data(time26,2);
YY{26}=data(time26,1);
U{27}=data(time27,2);
YY{27}=data(time27,1);
U{28}=data(time28,2);
YY{28}=data(time28,1);
U{29}=data(time29,2);
YY{29}=data(time29,1);
U{30}=data(time30,2);
YY{30}=data(time30,1);
U{31}=data(time31,2);
YY{31}=data(time31,1);
U{32}=data(time32,2);
YY{32}=data(time32,1);
U{33}=data(time33,2);
YY{33}=data(time33,1);
U{34}=data(time34,2);
YY{34}=data(time34,1);
U{35}=data(time35,2);
YY{35}=data(time35,1);


Aint=1;
Cint=1;
Dint=0;
Vmax=1.0;


% parameters


rcap=8.2e-5;
soc_init=.9881;%0.0244;%
fover=0.0479;
fover_init=.0916;
fover_time=.2819;

load OCP
%ocpin_d=[0.05:0.05:0.95];
ocpin_d=[0:0.05:1];
fd=[2.7;fd;4.2];
PP=spline(ocpin_d,fd);
alpha=[soc_init;fover;fover_init;fover_time;
2.7669;3.1335;3.3170;3.4134;3.4567];
```

```
alpha0=alpha;

Ahat_all=[];
Bhat_all=[];
Chat_all=[];
x0_all=[];


OPT=optimset('display','iter','MaxFunEvals',1000);
alpha=fminsearch(@cost_fun,alpha0,OPT,U,YY,...
    PP,rcap,time,numSegments,fd(6:21));
%



ymodel=[];
yyall=[];
uall=[];
yint2=[];
yint0=alpha(1)
f=0;
c2=alpha(3);
c3=alpha(4);
fd=[alpha(5:9);fd(6:21)];
PP=spline(ocpin_d,fd);
for k=1:numSegments
    u=U{k};
    u=u(:);
    yy=YY{k};
    y=zeros(length(yy),1);
    yint=data_gen(1,rcap,1,0,yint0,u);
    yint0=yint(end);
    if(k==19)
       yint0 = 0.0244;
    end
    z=f_over(alpha(2),u,yint,c2,c3);
    yint2=[yint2;yint];

    for k=1:length(y)
        y(k)=inv_ocp_spline(PP,yy(k)-z(k));
    end
    y=y-yint;
    [A,B,C,D,x0hat]=ssm(y,u);
```

```
    Ahat=A
    Bhat=B
    Chat=C
    Ahat_all=[Ahat_all Ahat];
    Bhat_all=[Bhat_all Bhat];
    Chat_all=[Chat_all Chat];
    x0_all=[x0_all x0hat];
    ocp_in=yint+data_gen(A,B,C,D,x0hat,u);
    yyy=f_ocp_spline(PP,ocp_in)+f_over(alpha(2),u,yint,c2,c3);
    yyy=yyy(:);
    ymodel=[ymodel;yyy];
    yyall=[yyall;yy];
    uall=[uall;u];
    f=f+((norm((yyy-yy))/norm(yy))).^(0.5);
end
f
time_all=[time1';time2'];

%%%%%%%%%%%%%%%%%%

figure
plot(time_all,yyall);
hold on
plot(time_all,ymodel,'r');grid
ylabel('Volts')
title('Voltage: measured (blue), modeled (red)')
figure
plot(time_all,yyall-ymodel);grid
ylabel('Volts')
title('Modeling Error')
%%%%%%%%%%%%%%%%%%
```

cost_fun.m

```
function f=esfun_spline_opt2(alpha,U,YY,PP,rcap,time,numSegments,fdi)
f=0;
yint0=alpha(1);
alpha(2)=alpha(2);
c2=alpha(3);
c3=alpha(4);
f1 = alpha(5:9);
ocpin_d=0:0.05:1;
for k=1:numSegments
    u=U{k};
    u=u(:);
    yy=YY{k};
    y1=zeros(length(yy),1);
    yint=data_gen(1,rcap,1,0,yint0,u);
    yint0=yint(end);
    if(k==19)
       yint0 = 0.0244;
    end
    z=f_over(alpha(2),u,yint,c2,c3);
    fd=[f1;fdi];
    PP=spline(ocpin_d,fd);
    for k=1:length(y1)
        y1(k)=inv_ocp_spline(PP,yy(k)-z(k));
    end
    y=y1-yint;
    [A,B,C,D,x0hat]=ssm(y,u);
    yhat=data_gen(A,B,C,D,x0hat,u);
    yyy=f_ocp_spline(PP,yint+yhat)+f_over(alpha(2),u,yint,c2,c3);
    f=f+((norm((yyy-yy))/norm(yy))).^(0.5);
    %keyboard
end
```

BIBLIOGRAPHY

Baba, A. and Adachi., S., "Simultaneous state of charge and parameter estimation of lithium-ion battery using log-normalized unscented kalman filter." in *2015 American Control Conference (ACC)*, 2015.

Chaoui, Hicham, , and Gualous., H., "Adaptive state of charge estimation of lithium-ion batteries with parameter and thermal uncertainties." *IEEE Transactions on Control Systems Technology*, vol. 25, pp. 752–759, 2017.

Dao, A. S. T. and McPhee, J., "A survey of mathematics-based equivalent-circuit and electrochemical battery models for hybrid and electric vehicle simulation," *Journal of Power Sources*, vol. 256, pp. 410–423, 2014.

Hake, A. and Fathy, H., "Single-particle modeling and experimental parameter identification for a lithium-cobalt-oxide battery cell." Master's thesis, Pennsylvania State University, 2015.

Jenkins, B., "Fast adaptive observers for battery management systems," Ph.D. dissertation, Massachusetts Institue of Technology, 2017.

Limoge, D. W., Bi, P., Annaswamy, A., and Krupadanam, A., "A reduced-order model of a lithium-ion cell using the absolute nodal coordinate formulation approach." *IEEE Transactions on Control Systems Technology*, vol. 26, pp. 1001–1014, 2018.

Lin, Xinfan, and Stefanopoulou., A. G., "Analytic bound on accuracy of battery state and parameter estimation." *Journal of The Electrochemical Society*, vol. 162, 2015.

Moura, S., "Battery state estimation for a single particle model with electrolyte dynamics." *IEEE Transactions on Control Systems Technology*, vol. 25, pp. 453–468, 2017.

Plett, G. L., "Extended kalman filtering for battery management systems of lipb-based hev battery packs." *Journal of Power Sources*, vol. 134, pp. 262–276, 2004.

Satadru, D. and Ayalew., B., "Nonlinear observer designs for state-of-charge estimation of lithium-ion batteries." in *2014 American Control Conference*, 2014.

Swarup, S., Tan, S., Liu, Z., Hao, Z., and Shi, G., "Battery state of charge estimation using adaptive subspace identification method." in *2011 9th IEEE International Conference on ASIC*, 2011.

Xiong, R. and Mu, H., "Accurate state of charge estimation for lithium-ion battery using dual uncsented kalman filters." in *2017 Chinese Automation Congress (CAC)*, 2017.

y. Hu, Yurkovich, S., Guezennec, Y., and Yurkovich, B., "A technique for dynamic battery model identification in automotive applications using linear parameter varying structures." *Control Engineering Practice*, vol. 17, pp. 1190–1201, 2009.

Yang and Guodong, "Adaptive state of charge estimation of lithium-ion battery based on battery capacity degradation model." *Energy Procedia*, vol. 152, pp. 514–519, 2018.

Ying, X. and Fahimi, B., "State-space based multi-nodes thermal model for lithium-ion battery." in *2014 IEEE Transportation Electrification Conference and Expo (ITEC)*, 2014.