THE PENNSYLVANIA STATE UNIVERSITY
SCHREYER HONORS COLLEGE

DEPARTMENT OF MECHANICAL ENGINEERING

EXPERIMENTAL VALIDATION OF AN UNKNOWN INPUT ESTIMATION ALGORITHM FOR LITHIUM ION BATTERY APPLICATIONS

PEYMAN NOROUZI
SPRING 2018

A thesis
submitted in partial fulfillment
of the requirements
for a baccalaureate degree
in Mechanical Engineering
with honors in Mechanical Engineering

Reviewed and approved* by the following:

Hosam Fathy
Bryant Early Career Associate Professor of Mechanical and Nuclear Engineering
Thesis Supervisor

Jacqueline O'Connor
Assistant Professor of Mechanical and Nuclear Engineering
Honors Advisor

* Signatures are on file in the Schreyer Honors College.

ABSTRACT

The thesis develops an experimental setup to validate a model-based algorithm for combined state and current estimation in a lithium-ion battery. The algorithm estimates external (input) current based on the measured terminal voltage. This is useful where current measurement is of interest, but it is either not possible to measure directly or it is too costly for smaller budgeted applications. A paper by Mishra et al. [11] has already theoretically analyzed and validated the algorithm. Although important, the theoretical analysis does not offer a comprehensive picture of the algorithm's success because the reduced-order model for estimation cannot fully capture the battery's dynamic. Thus, an experimental investigation becomes necessary. The results of the experiments indicate that the proposed algorithm cannot by itself successfully predict and track the external current and internal state of a battery in the case of Lithium Cobalt Oxide (LCO) and Lithium Iron Phosphate (LFP) lithium-ion batteries. An addition of a 0.5 Ω resistor improves the performance of the algorithm immensely in estimating the input current of the battery. This improvement shows that algorithm can be useful in estimating the input current of a lithium-ion battery without a current sensor if an additional external resistor is used.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to thank Partha P. Mishra, the graduate student at the control optimization lab, for his immense help and contribution throughout my research endeavor. His help not only assisted me vastly in understanding research process better but also taught me many valuable pieces of knowledge related to Lithium-ion Batteries chemistry and control theory applications. Partha was there for me at tough times that I was struggling in and provided me with space to overcome such challenging environments. I truly believe that I am graduating college with a set capability that will allow me to succeed in future projects and research work. In addition, I would like to thank my thesis supervisor Dr. Hosam K Fathy for his useful and valuable feedback during my research journey. His tremendous knowledge in the field of battery systems provided me a secure path to success. Last but not the least, I want to thank my honors advisor, Dr. Jacqueline O'Connor for her mental and emotional support whenever needed. Her positive outlook on my work ethic and potential kept me motivated through the ups and downs of the thesis preparation process.

Chapter 1

Introduction

This work investigates the problem of estimating the input current acting on a lithium-ion battery based on battery terminal voltage measurements. It is motivated by the degree to which an accurate estimation of battery input current can allow a battery management system (BMS) to achieve fault prevention, diagnostics, and power profiling [1]. There are physical sensors that can measure battery input current directly; the two leading types are shunt resistors and Hall Effect sensors [1]. Both sensors have significant limitations. Shunt resistors, for example, inherently increase the power dissipation within a battery pack, whereas Hall Effect sensors are known to be quite expensive for many applications [1]. This leads to the following motivating question: is it possible to estimate the input current acting on a lithium-ion battery inexpensively and accurately? The answer, in short, is yes, by using a validated estimation algorithm.

The focus of this thesis is primarily the estimation of a battery's input current and state of health, given a direct measurement of voltage. The literature on input current estimation of a battery is not extensive [1, 11]. Cambron [1] uses a linear battery model to implicitly estimate the current. Mishra et al. [11] on the other hand, uses a simulation state and input estimation algorithm for estimation of input current in n^{th} order ECM. He also indicates and mentions a theoretical limitation on the input current estimation. The underlying goal of this thesis is to build on his work and experimentally validate the theoretical insight into the input current

estimation behavior. In accordance with the chosen battery structure in Mishra et al., a 3rd order ECM model (n = 3) is used for the experimental validation of the model-based algorithm.

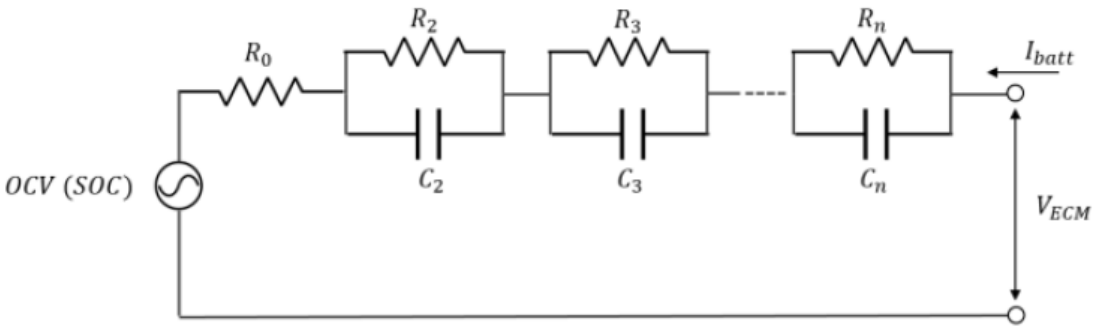

Figure 1. Nth order ECM battery model

The continuous-time state-space equations of the ECM in Figure 1 are:

$$\dot{x}_1 = \frac{I_{batt}}{Q} \tag{1a}$$

$$\dot{x}_i = -\frac{x_i}{\tau_i} + \frac{I_{batt}}{C_i}, \quad i = 2, 3, \dots, n \tag{1b}$$

$$V_{ECM} = OCV(SOC) + \sum_{i=2}^{n} x_i + I_{batt}R_0 \tag{1c}$$

$$\tau_i = R_i C_i, \quad i = 2, 3, \dots, n \tag{1d}$$

The variables in the equations above are defined below in Table 1.

Table 1. Variables in the continuous-time state-space equations of an nth order ECM battery model

I_{batt}	Input Current of the battery
V_{ECM}	Voltage Across the battery
OCV (SOC)	Open Circuit Voltage as a function of State of the Battery
x_i	State of the battery
τ_i	Time Constant
R_i	Resistance
C_i	Capacitance

Chapter 2

Background

There is already a rich literature on lithium-ion battery state (SOC) estimation using techniques such as Proportional-Integral-Derivative (PID) control [2], neural networks [3], and also different variants of the well-known Kalman filter [3, 4 - 6] estimation. However, the focus of this thesis is primarily in the estimation of battery's input current and state of health, given a direct measurement of voltage. To the best of the author's knowledge, the literature on input current estimation of a battery is not extensive [1, 11]. Only two studies have been found on the subject; one experimentally validates its estimation algorithm and the other one's experimental validation is the subject of this thesis. The underlying goal of this thesis is to build on the existing literature in the area of disturbance estimation [7 - 10] to experimentally validate, the ability of a disturbance estimator to monitor battery input current accurately without direct measurement [11].

2.1 State of Charge (SOC) Estimation

As mentioned before, the literature on the estimation of the state of the charge (SOC) of lithium-ion batteries is extensive. All of the referenced literature on estimation of SOC use closed-loop estimation methods, meaning that they use the measured voltages for their estimation. Although vastly different approaches are examined for SOC estimation, the main ideas can be divided into the following categories.

Proportional-Integral-Derivative (PID)

Xu et al. [2] use Proportional-Integral Observer (PI) as an approach for the estimation of the state of charge. The approach was developed by an analysis of the most popular model-based SOC estimation methods. The method is analyzed on a simple RC battery model and then verified using an experimental battery test bench [2]. Figure 2 indicates the block diagram of the PI method in action. The equivalent circuit of the RC battery model, used in the analysis, is demonstrated in Figure 3.

.

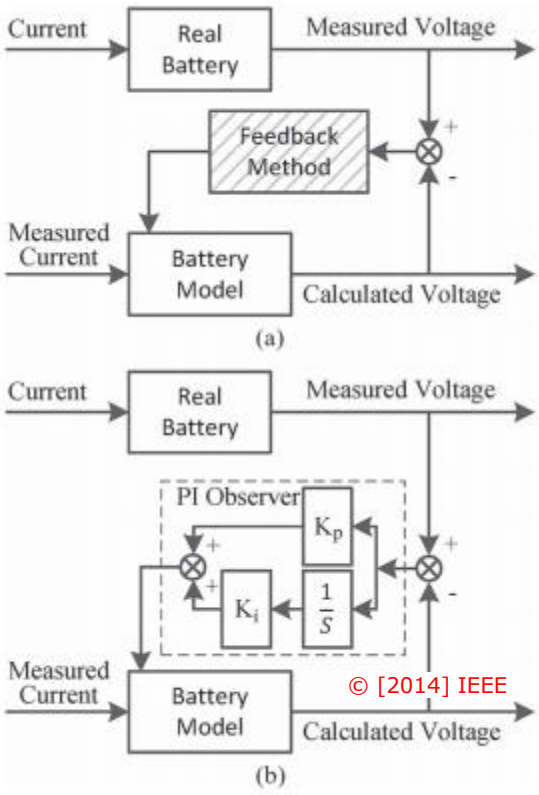

Figure 2. Block diagrams of the proposed method. (a) The general structure of the method. (b) PI Observer as a feedback method [2].

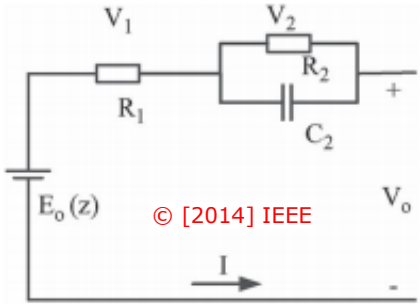

Figure 3. Second order equivalent RC model of Li-ion batteries [2].

Based on the equivalent circuit model, a second-order model is used to count for the modeling error and capacity variation [2]. The equivalent RC model and the system's use both contribute to estimating the state of the charge of the battery. An experimental set-up below in Figure 4 is used to validate the estimated result:

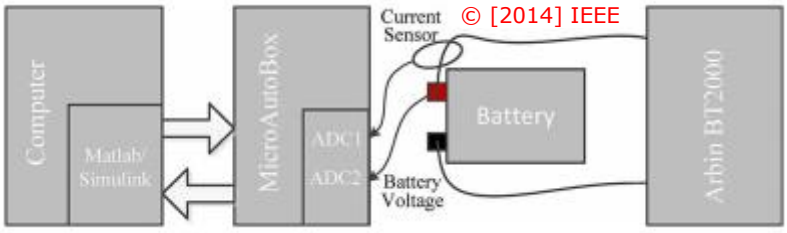

Figure 4. The experimental setup [2].

By comparing the experimental and estimated data, it can be seen that the SOC is predicted within 2% of the error bound. This shows that the approach used to predict the state of the battery is sufficiently accurate. Figure 5 shows and compares the estimated and reference results.

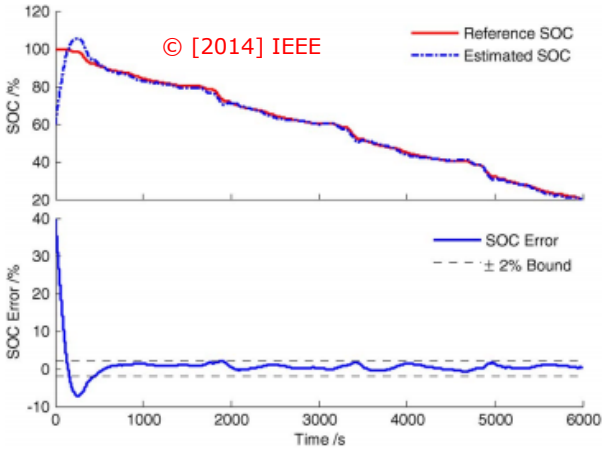

Figure 5. SOC estimation shows to be very close to the reference data [2].

Kalman Filter

Kalman filters are often used for battery SOC estimation. While the literature on this subject is much more extensive, the four references below suffice for providing a summary of different ways that Kalman filters are used in the SOC estimation process. A Kalman filter is a set of equations and an algorithm that estimates desired unknown variables by using a series of measurements taken over time. There are at least two general forms of Kalman filters, the Kalman filter (KF) and the Extended Kalman filter (EKF). The KF is used for linear estimation applications while EKF is a nonlinear version of KF that linearizes the estimation problem at each time step [12]. Because all of the referenced papers' battery model are nonlinear, different forms of EKF are used [3, 4 - 6].

A work by He et al. [5] uses a slightly modified version of EKF, called adaptive extended Kalman filter, on an improved Thevenin battery model (AEKF). A Thevenin

battery model consists of a resistor and an RC pair while the above-improved Thevenin model adds one more RC pair to the battery model [5]. This means that the battery model used is a 3rd order battery model. He et al. shows that AEKF, as described in the paper, estimates the SOC of the battery as accurately as EKF with the benefit of having smaller estimation error [5]. Added to the smaller error, the estimation error, in the case of AEKF, converges to zero faster than the case of EKF [5]. Figures 6 and 7 indicate the estimation accuracy and error accuracy plots, respectively.

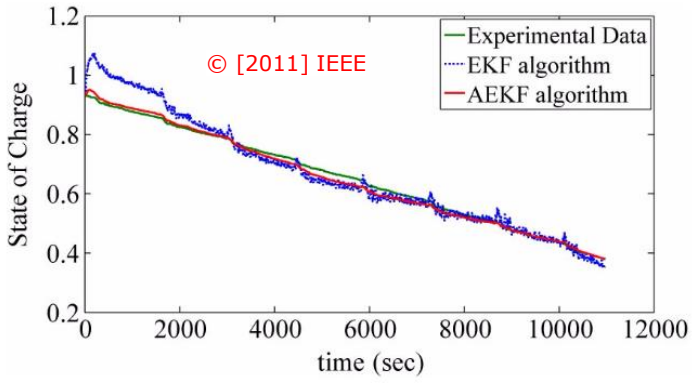

Figure 6. SOC estimation curves showing the difference between EKF, AEKF and experimental data [5].

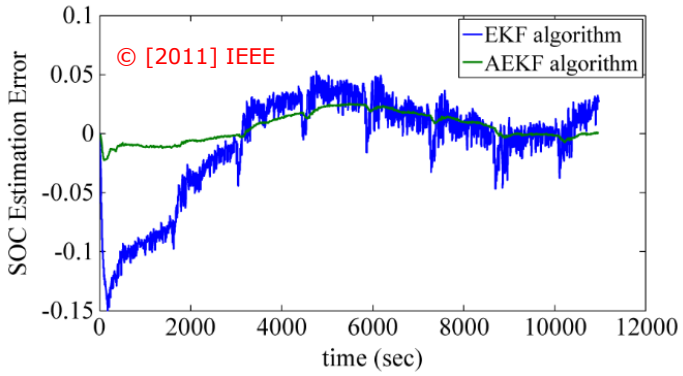

Figure 7. SOC estimation error curves [5].

Li [4] takes a different approach in his paper by approaching the SOC estimation with a 2nd order battery model and an EKF algorithm. In his paper, he uses electrochemical impedance spectroscopy (EIS) to find the three unknowns of his 2nd order battery model. Using fixed resistances in the battery model he recognizes that EKF is able to estimate SOC with a relative error of <3% [4]. He also indicates that constant parameters to describe a battery model constraints accuracy of EKF [4]. If more accuracy is desired, relaxing the constants R1 or R2 in the equivalent circuit model enables EKF algorithm to estimate SOC during charge/discharge more accurately [4].

In addition, Chen et al. [6] show that by having a 3rd order battery model and using a proposed algorithm based on EKF algorithm, one is able to estimate SOC with high accuracy. The proposed algorithm does not rely on initial SOC and is resistant to errors of parameters. To validate their proposed statement, a series of hardware-in-the-loop tests were performed. By comparing the hardware in a loop and simulation data, it is concluded that the proposed model is accurate for use in electric drive vehicles [6].

An interesting paper by Charkhgard et al. [3] approaches the SOC estimation problem using EFK but from a different angle. Instead of having an electrical battery model with one or multiple RC pairs, they use a neural network (NN), trained offline with the collected data from a battery, to model battery's behavior. The NN used in the paper is a radial basis function (RBF) and the model can be seen in Figure 8.

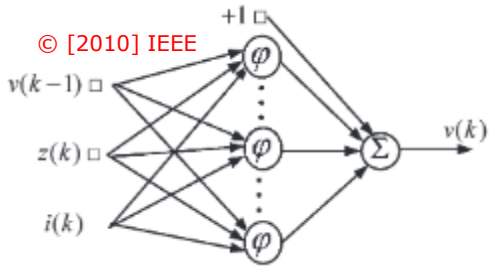

Figure 8. RBF NN model of a battery [3].

The result of the simulation and the experimental data indicate a good accuracy in estimating of SOC with a fast convergence to the actual values [3]. Although successful, the conclusion suggests three issues that can be addressed in future works. They are as following [3]:

a) Applying the method to a battery pack with multiple cells of the same type.

b) Investigation of thermal effects on the proposed method.

c) Investigation of aging on the proposed method.

2.2 Input Current Estimation

As mentioned earlier, the literature on current estimation is narrow. The two referenced papers use two different approaches of disturbance algorithm in order to estimate input current of a battery [1, 11].

Cambron et al. [1] use a derived unknown input observer and a 3rd order battery model for their current estimation. Next, they find the different parameters of the 3rd order battery model through conducting a series of tests on a sample cell. The unknown input observer derivation is a method that uses linear discrete-time system and estimates for the unknowns input values [13]. Cambron et al. [1] use the derivation method presented in [13] closely with one

exception. Instead of a discrete-time system, they consider a continuous time system [1]. To compare the accuracy of the estimated data, a Hall-effect current sensor was used to measure the true current values [1]. The resulting accuracy of the current estimations is shown in Figure 9.

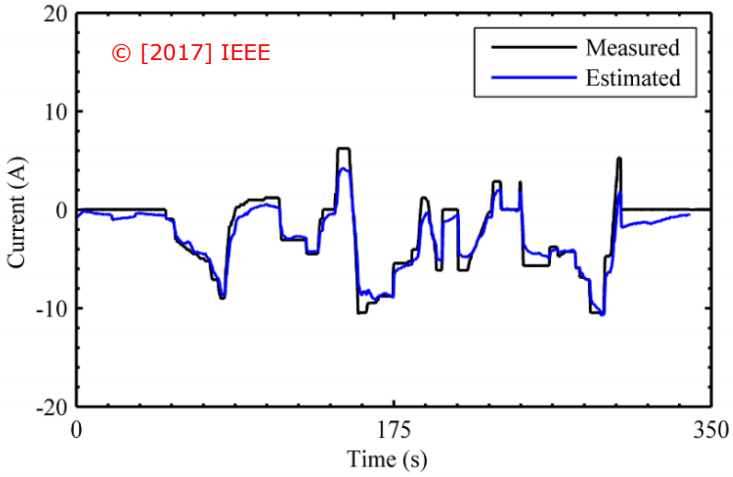

Figure 9. Estimated vs. Measured current accuracy for varying current at 95% state of charge [1].

Mishra et al. [11] on the other hand, adopts the model-based combined state and input estimation technique introduced in Gillijns and Moor. Gillijns et al. [10] extend on their previous work regarding state estimation, proposing an algorithm for a state and input estimation for linear discrete-time systems with direct feedthrough. This is relevant to the battery input current estimation problem since battery state space models have a direct feedthrough term in their output equation in the form of a potential drop across the internal resistance of the cell [10]. The original state algorithm in [10] was developed for a linear state-space model, but Mishra et al. modify it to a nonlinear Li-ion battery ECM [11]. The battery model used by Mishra et al. is of the n^{th} order shown in Figure 10. The algorithm used, which will be experimentally validated in this work, has the following ordered steps [11]:

1) Input estimation: This step uses available battery voltage measurements to estimate the input current or disturbance of the battery model.

2) Measurement update: This step corrects the predicted state of the battery model using the measured voltage.

3) Time update: This step uses the state equation to predict the state of the battery at the next time step.

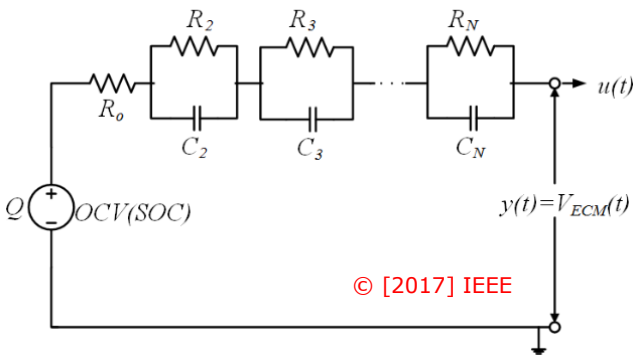

Figure 10. Nth order battery model [11].

In the case of Mishra et al., the algorithm to estimate the battery input current for Lithium-ion batteries is integrated into a PV/battery hybrid unit, and subsequently, the PV generated current. The PV array structure is shown in Figure 11. The simulation results then are compared with the true value for their success evaluation. The results can be seen in Figure 12.

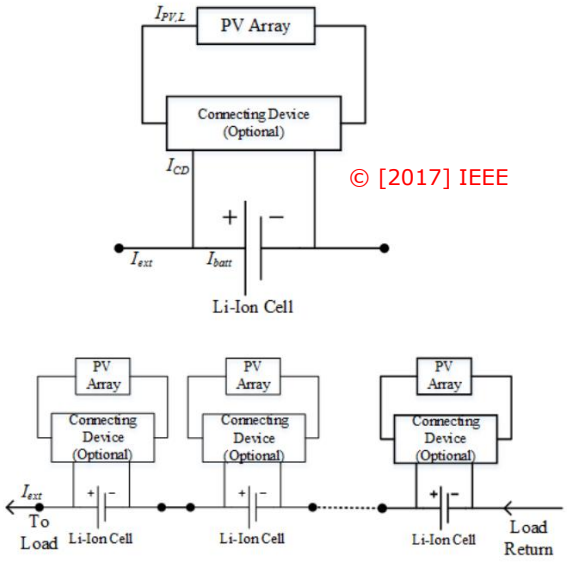

Figure 11. PV/EES hybrid cell (top) and hybrid string (bottom) [11].

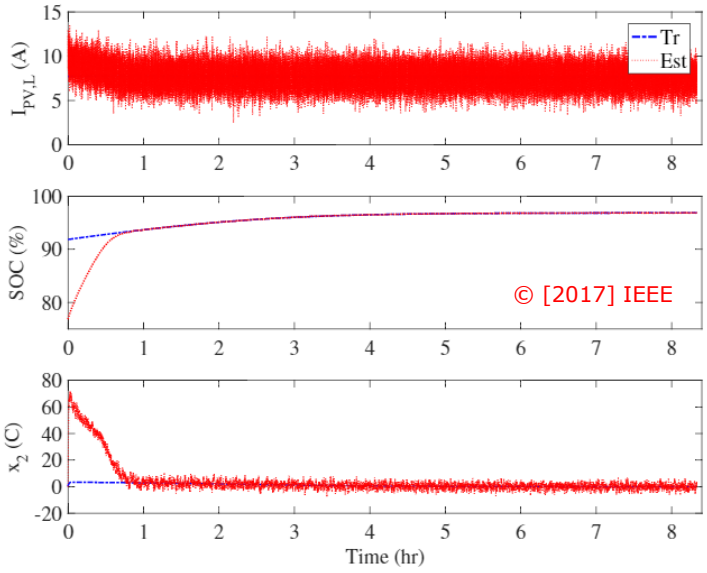

Figure 12. Resulted simulation estimation vs. true values [11].

Chapter 3

Experimental Set-up and Computer Algorithm

The experimental setup is divided into two different sections, Battery Parametrization and Algorithm Validation. Both processes include supplying an input current profile on the batteries in the Arbin Cycler (Laboratory Battery Cycler) and then analyzing the data on the computer using respective algorithms. A picture of the Arbin Cycler is shown in Figure 13. Two Lithium Cobalt Oxide (LCO) and two Lithium Iron Phosphate (LFP) batteries were used in the validation process. Since the two types of batteries have vastly different state-of-charge (SOC) vs. open-circuit-voltage (OCV) curves, choosing them as the experiment's batteries would allow us to have a more comprehensive insight into different territories of lithium-ion battery current estimation. For the sake of both accuracy and calculation speed, a 3rd order model of the lithium-ion battery was modeled then analyzed in MATLAB using the obtained data from the cycler.

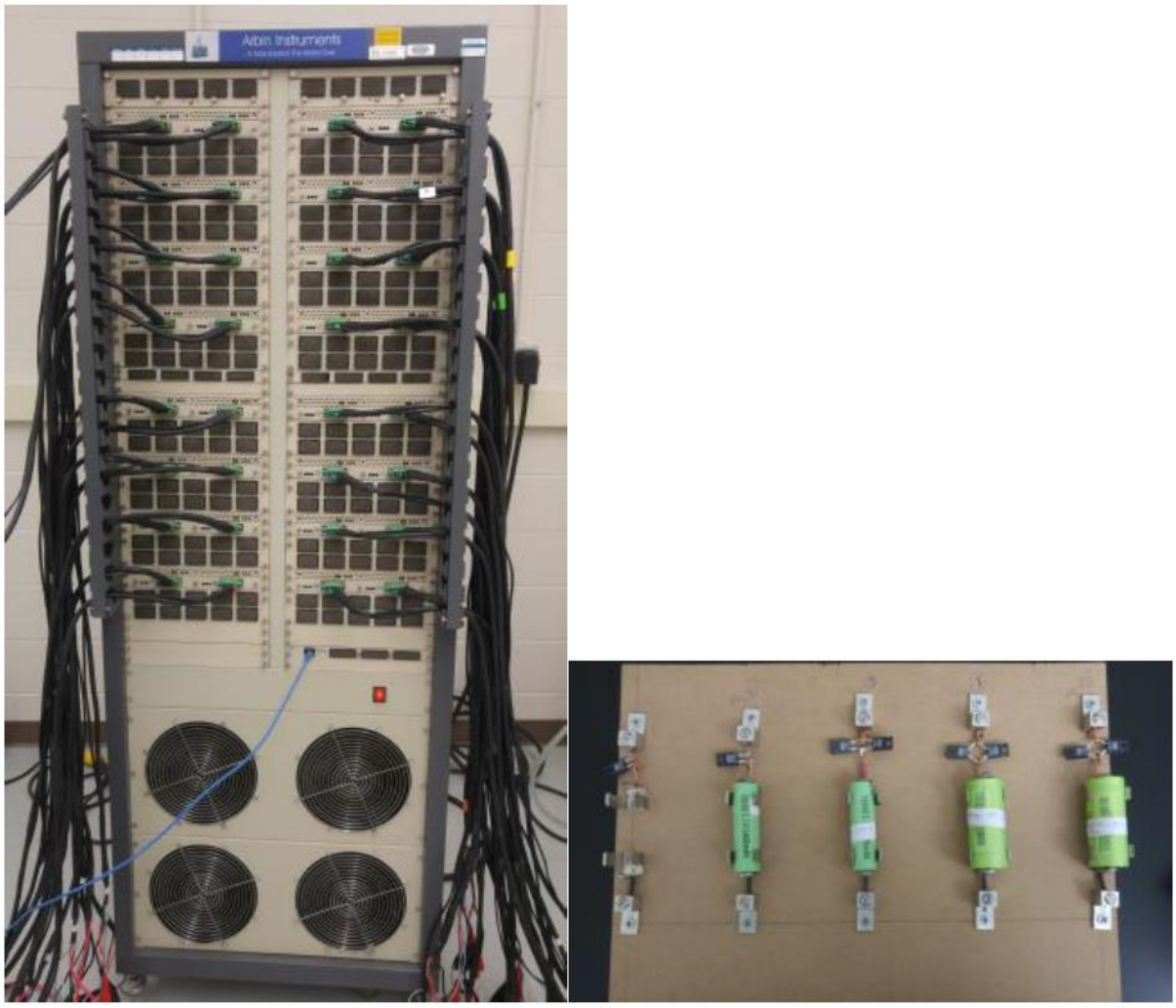

Figure 13. Left picture, the lab cycler used to apply input current profile on the batteries. Right picture, the battery setup with the resistor s in the battery validation process.

3.1 Battery Parameterization

Since a 3rd order ECM model is used, there are 8 different parameters needs to be found. The unknowns are as follows: $R_0, R_i\ (i = 2, 3), C_i\ (i = 2, 3), SOC_{initial}, OCV(SOC), Q$. To find the capacitance (Q) and OCV(SOC) curve of the batteries, a Constant Current Constant Voltage (CCCV) input was used in the Arbin Cycler. As can be seen from Figure 14. The CCCV was

used to slowly charge and discharge the batteries. Q was found by finding the average of battery capacitances while going through the charging and discharging process.

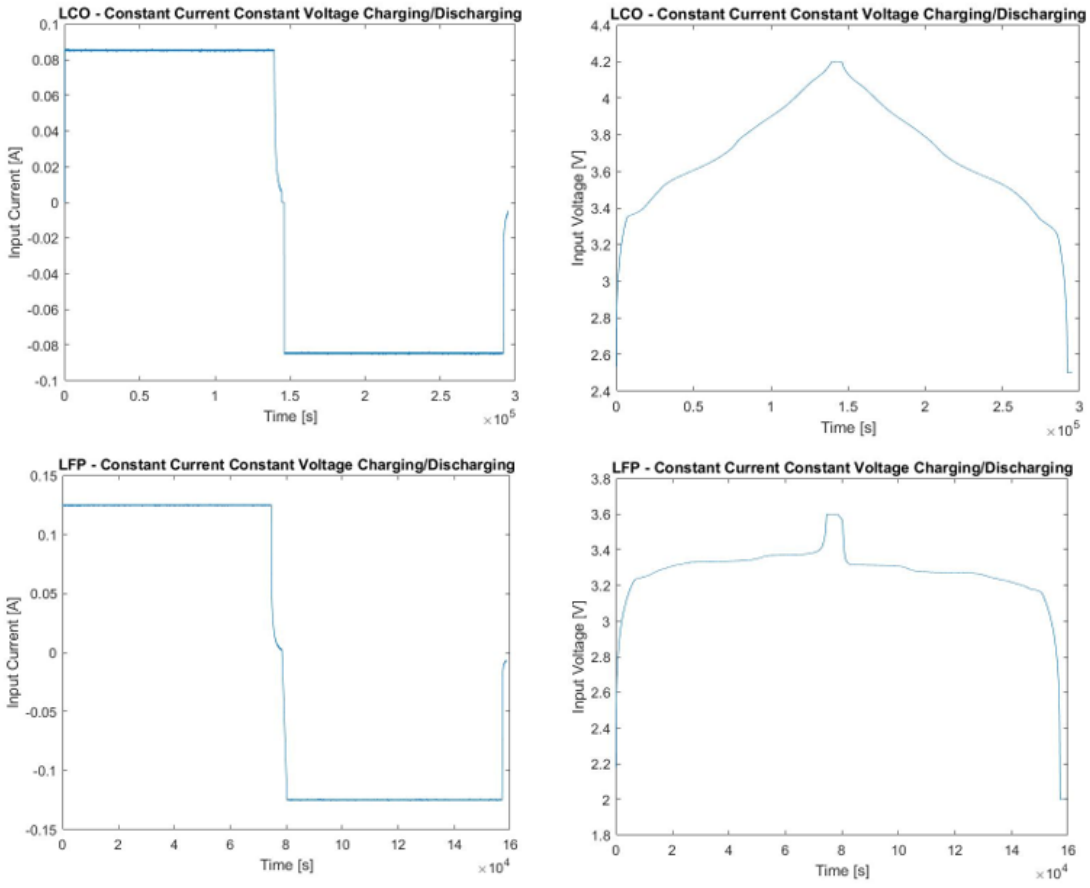

Figure 14. The CCCV input to the batteries. Top two plots, the voltage and current inputs for the LCO batteries. Bottom two plots, the voltage, and current for the LFP batteries.

From equation (1a), the time derivative of charge is equal to input current divided by the already found, Q. Since we need a map of Charge (SOC) in correlation with OCV, we can change the (1a) to a discrete equation shown below:

$$\frac{\Delta x}{dt} = \frac{I_{batt}}{Q} \tag{2a}$$

By implementing the input current and voltage measurements taken from the Arbin into a for loop in MATLAB, we can find our charge and voltage measurements for every time step:

$$\begin{cases} x(1) = 0 \\ x(n) = x(n-1) + (dt \times \frac{I_{batt}(n-1)}{Q}), \\ OCV = V_{ECM} \end{cases} \quad \text{n from 2 to number of input measurements} \qquad (2b)$$

Finally, by plotting OCV vs. SOC, we will have the desired OCV(SOC) mapping curve. To find the remaining unknown of our ECM model, a pulsed current profile was applied to the batteries using the Arbin. Figure 15. indicates the input current profile. This will be explained in detail in Chapter 4.

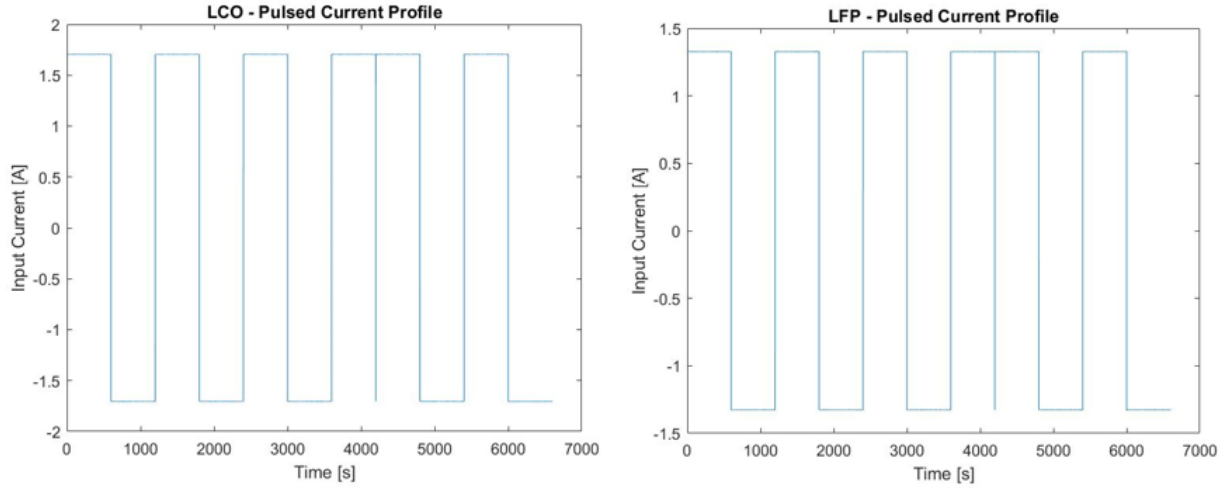

Figure 15. Pulsed input current profile to find the remaining unknown parameters of the batteries. Left, LCO chemistry right, LFP chemistry.

3.2 Algorithm Validation

For the second portion of the experiment, an HEV (Hybrid Electric Vehicle) current cycle was applied to the batteries using the lab cycler. In the HEV cycle, charging and discharging hovers around a decided initial SOC. By the end of the cycling, the SOC should be back to the initial decided SOC. The overall experimental validation procedure can be divided into 5 steps (graphical representation in Figure 16.):

Step 1: Select an input current profile.

Step 2: Precondition the Li-ion cell to an initial SOC.

Step 3: Supply the current profile to charge/discharge the Li-ion cell in the Arbin Cycler and record measured current (I_{true}) and voltage (V_{meas}).

Step 4: Use measured voltage with the input/state estimation algorithm to estimate input current ($\hat{I}$) and SOC of the battery ($\widehat{SOC}$).

Step 5: Compare $\hat{I}$ to I_{true} and $\widehat{SOC}$ to SOC_{true} , which is obtained through the I_{true} integration.

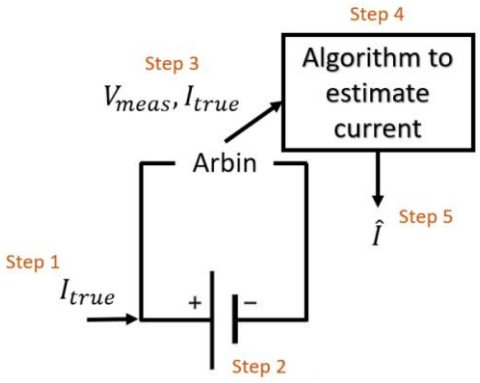

Figure 16. The overall set-up of the validation experiment.

Chapter 4

Battery Parameterization

Chapter 3 explained the general set up of the battery parameterization process. In addition, it talked about how Q and OCV(SOC) curve were found using the CCCV input to the batteries. To be able to find the rest of the ECM model's parameters,$R_0, R_i\ (i = 2, 3), C_i\ (i = 2, 3), SOC_{initial}$, a pulsed input current was applied to the batteries. Since there are more unknowns than the number of equations relating them (equations 1 a-d), a minimum squared error algorithm was used to find the remaining parameters:

$$J = min_{R_0,\tau_1,\tau_2,C_1,C_2,SOC_{initial}} \sum_{j=1}^{m} (V_{meas} - V_{ECM})^2 \quad (4a)$$

The "fminsearch" function in MATLAB solves the equation above by finding the minimum of the unconstrained multivariable ($R_0, \tau_1, \tau_2,\ C_1, C_2, SOC_{initial}$) function by using the derivative-free method. The "fminsearch" function was run 75 times to compute various parameters' values, resulting in different values for J. Afterwards, all the parameters' values were manually reviewed to ensure there were no negative values, since negative numbers would not have a sensical meaning. Next, all the J values were manually revised and the values that were either too large or too far off the rest of the bunch can be identified from the list so that the corresponding parameters values can be removed. Finally, the average of the remaining parameters was calculated to obtain each parameter final value. Table 2. indicates the final values of each parameter.

Table 2. The parameters of the 3rd order ECM battery Model.

Parameters	LCO1	LCO2	LFP1	LFP2
$R_0\ (\Omega)$	0.0591	0.1246	0.0402	0.0234
τ_1	3.3779e+03	2.4183e+03	169.6470	102.8508
τ_2	49.4069	50.2258	23.3531	694.0270
C_1 (F)	3.8174e+04	3.7854e+04	1.3071e+04	5.6685e+03
C_2 (F)	2.2787e+03	2.5034e+03	2.0961e+03	1.6290e+05
$SOC_{initial}$	0.4570	0.4665	0.6648	0.6758
Q (C)	3.3842	3.4361	2.6537	2.6274

Figure 17 shows the obtained OCV(SOC) curve for all of the four batteries (LFPs and LCOs).

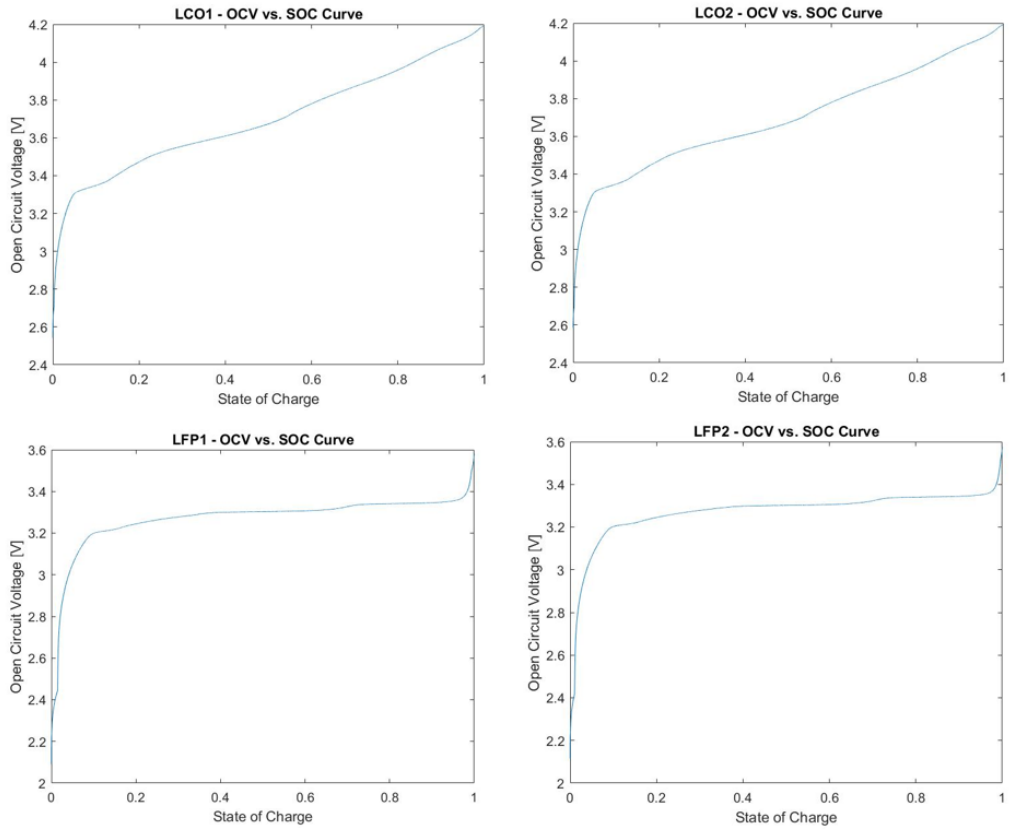

Figure 17. OCV vs. SOC curve. Top two, LCO batteries. Bottom two, LFP batteries.

Chapter 5

Experimental Algorithm Validation

For the validation of the estimation algorithm discussed in Mishra et al. [11], an HEV cycle was applied to all four batteries. The applied current profile can be seen in Figure 18. The proposed algorithm works better in input current estimation as the resistance of the overall system increases [11]. Since the input resistance of a battery is unchangeable, a 0.5 Ω resistor was externally implemented in the Arbin cycler's battery housing; this implementation can be seen in Figure 13. The shown current profile in Figure 21. was applied to the batteries, once with the external resistor attached and once without, so that the proposed statement that "the higher system resistance, the better current estimation" can be experimentally assessed.

The proposed algorithm that is being experimentally validated for its application on the current estimation of lithium-ion batteries was adopted by Mishra et al. [11] from Gillijns and Moor [10]. Since the ECM battery is a nonlinear model and Gillijns and Moor's algorithm works with linear systems, Mishra et al. modified the algorithm, which enables him to apply it to nonlinear systems [11]. The algorithm for estimating current and SOC has four steps that are briefly summarized below. Corresponding equations for each step are taken from Mishra et al. [11] and are shown in Figures 18, 19 and 20. For a more detailed explanation of the algorithm, please refer to Mishra et al. [11]:

Step 1: Initialization

Assuming the initial unbiased estimate of the state of the battery is known, the disturbance and the states of the battery is initialized. It is also assumed that the posterior and a-priori state error covariance matrices are known.

Step 2: Unknown disturbance estimation

Using the available voltage measurement (V_{meas}), the disturbance is estimated using the estimation of a-priori state.

$$C_k = \frac{\partial g}{\partial x_k}\Big|_{\hat{x}_{k|k-1},\hat{d}_{k-1}} = \left[\frac{dOCV}{dx_1}\Big|_{\hat{x}_{1,k|k-1}}, \frac{1}{C_w}, \ldots, \frac{1}{C_w}\right]$$

$$\tilde{R}_k = C_k P^x_{k|k-1} C_k^T + R$$

$$H = \frac{\partial g}{\partial d_k}\Big|_{\hat{x}_{k|k-1},\hat{d}_{k-1}} = R_o$$

$$M_k = (H^T \tilde{R}_k^{-1} H)^{-1} H^T \tilde{R}_k^{-1}$$

$$P_k^d = (H^T \tilde{R}_k^{-1} H)^{-1}$$

where $R = \mathbb{E}(vv^T) > 0$ again is the measurement noise covariance matrix. Using the gain matrix M_k, a new estimate of the disturbance is computed following the relation:

$$\hat{d}_k = M_k(y_k - g(\hat{x}_{k|k-1}, \hat{d}_{k-1}, u_k) + H\hat{d}_{k-1})$$

Figure 18. Corresponding equations for the unknown disturbance estimation step taken from Mishra et al [11].

Step 3: Measurement update

By using the available voltage measurement (V_{meas}), the unbiased minimum variance estimate of the state is found.

$$K_k = P^x_{k|k-1} C_k^T \tilde{R}_k^{-1}$$
$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k(y_k - g(\hat{x}_{k|k-1}, \hat{d}_{k-1}, u_k))$$

The corresponding posterior error covariance and cross-covariance matrices are obtained as:

$$P^x_{k|k} = P^x_{k|k-1} - K_k(\tilde{R}_k - HP^d_k H^T)K_k^T$$
$$P^{xd}_k = (P^{dx}_k)^T = -K_k H P^d_k$$

Figure 19. Corresponding equations for the measurement update step, taken from Mishra et al. [11].

Step 4: Time update

By using our parameterized 3rd order ECM model and the calculated posterior state estimate from the kth step, the a-priori state estimate for the (k+1)th time step can be mathematically estimated.

$$\hat{x}_{k+1|k} = A\hat{x}_{k|k} - Bu_k + B\hat{d}_k$$

Next, we calculate the a-priori state error covariance matrix using error covariance matrices from previous time step following the relations:

$$G = \frac{\partial f}{\partial d_k}\Big|_{\hat{x}_{k-1|k-1}, \hat{d}_{k-1}} = B$$
$$P^x_{k+1|k} = \begin{bmatrix} A & G \end{bmatrix} \begin{bmatrix} P^x_{k|k} & P^{xd}_k \\ P^{dx}_k & P^d_k \end{bmatrix} \begin{bmatrix} A & G \end{bmatrix}^T + Q$$

where $Q = \mathbb{E}(WW^T) \geq 0$ is the process noise covariance.

Figure 20. Corresponding equations for the time update step, taken from Mishra et al. [11].

By the end of the last step of the algorithm, the state and voltage at a time step would allow us to find the next time step's voltage and state. This step essentially enables us to simulate our ECM battery model using equations 1(a-d). After simulating the model, the estimated current ($\hat{I}$) and SOC ($\widehat{SOC}$) is compared with the measured current (I_{true}) and SOC (SOC_{true}) using the Arbin cycler while applying the HEV input current to the batteries.

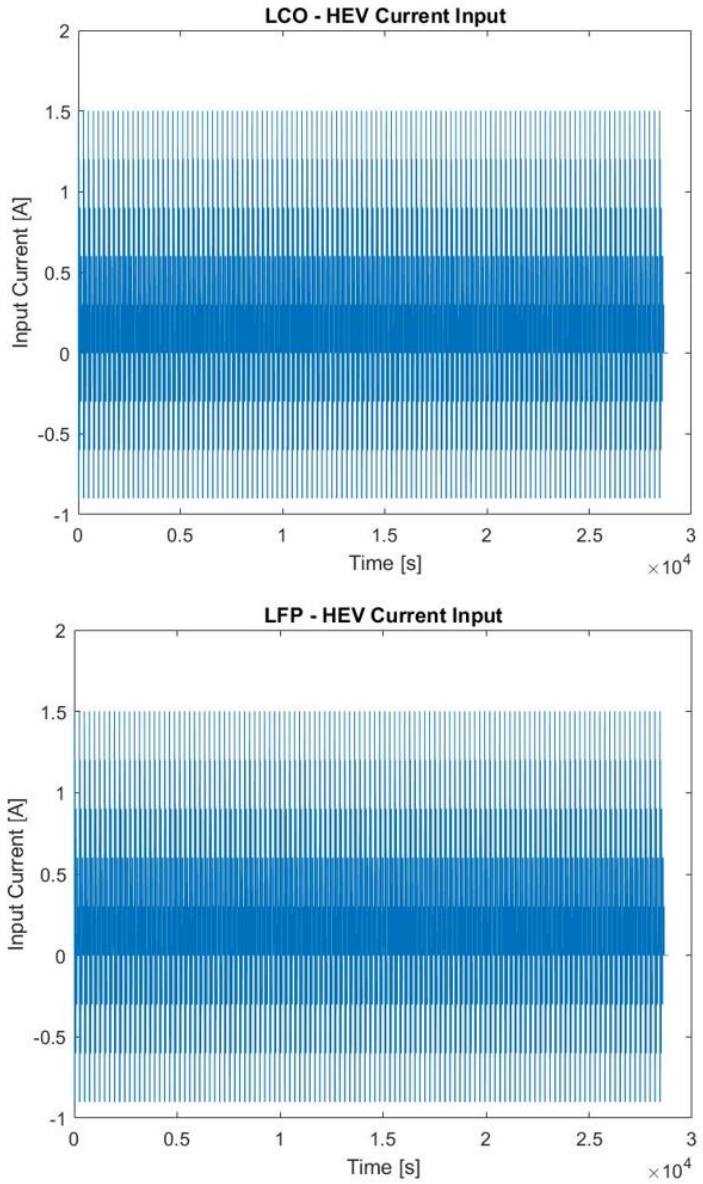

Figure 21. The HEV input current cycle was applied to the batteries with 0 and 0.5 Ω external resistance. Top, LCO cells. Bottom, LFP cells.

Chapter 6

Results

To evaluate the success of the model-based algorithm in estimating the input current of a lithium-ion battery, the results were divided into two sections: one with no external resistor connected across the batteries and the other with 0.5 Ω.

6.1 No External Resistor

After running the algorithm on MATLAB and simulating the ECM battery model using the four steps explained in the last chapter, both true (I_{true}) and estimated ($\hat{I}$) input current values were recorded. Before plotting the results, the current estimation error was calculated by subtracting the true from the estimated values. This is so the effectiveness of the algorithm can be shown in a more efficient way. Figure 22, 23 show the current estimation error vs. time plots for LCO and LFP batteries, respectively.

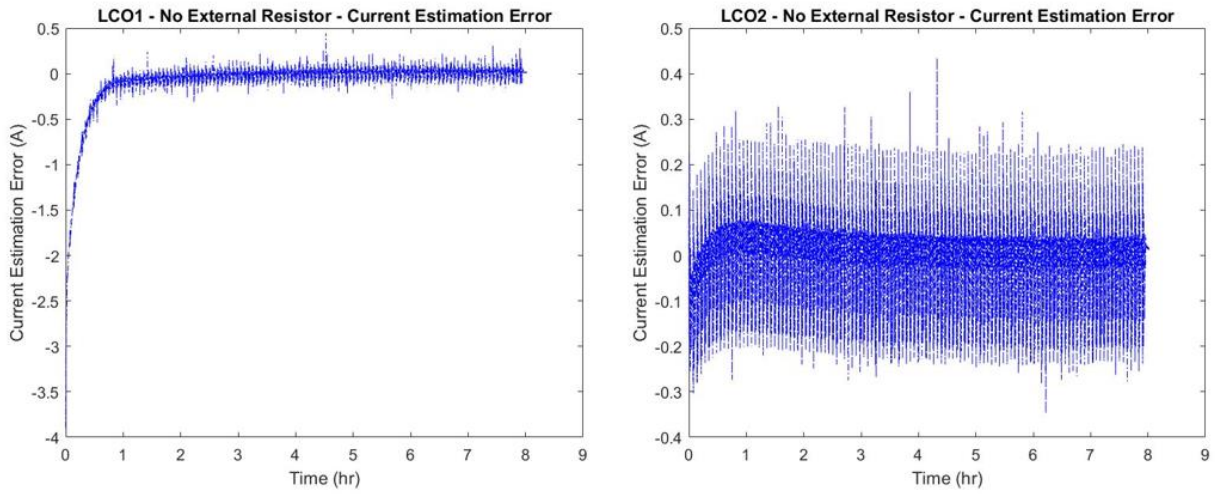

Figure 22. Current estimation error vs. time with no external resistor attached to the cells. Left, LCO1. Right, LCO2.

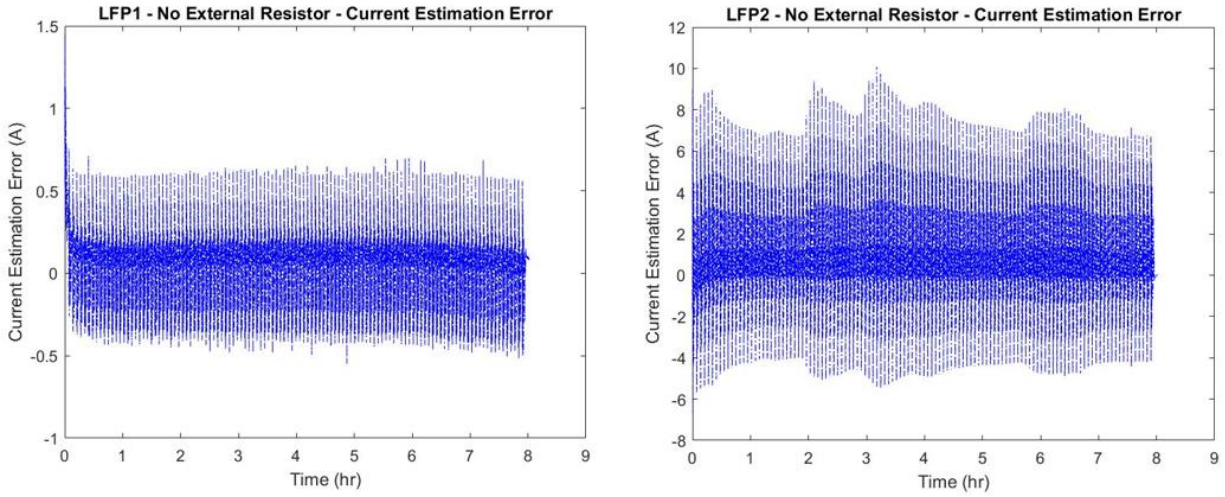

Figure 23. Current estimation error vs. time with no external resistor attached to the cells. Left LFP1. Right, LFP2.

It was expected that the error would start high and then gradually converge to zero in the 8-hour period of the test. In the case of LCO1, the average input current estimation seems to be converging to zero but the diversion from the mean appears to be too high to be acceptable. Even though the current estimation for the LCO1 cells is not perfect, it is more accurate than that of the LCO2 and LFP cells. Although the mean seems to be also converging to a number close to zero, the data is scattered in a wider sense around the arithmetic average. There is also a big difference in the estimation accuracy between the current estimation error of LFP 1 and LFP 2. LFP 2's current estimation is less accurate and precise. While the main reason for this is not clear; the difference may be due to the age difference between the two batteries.

Input current estimation is the main focus of this thesis but using the estimated current to estimate the SOC of the batteries can provide more in-depth insight into the success of the algorithm. As can be seen in Figure 24 and 25, from the four cells LCO1 seems to be converging to the right SOC while the rest of the cells are not. This makes sense because the current

estimation error for these cells did not converge to zero, which shows bias in the system. The estimation of the input current directly affects the SOC estimation.

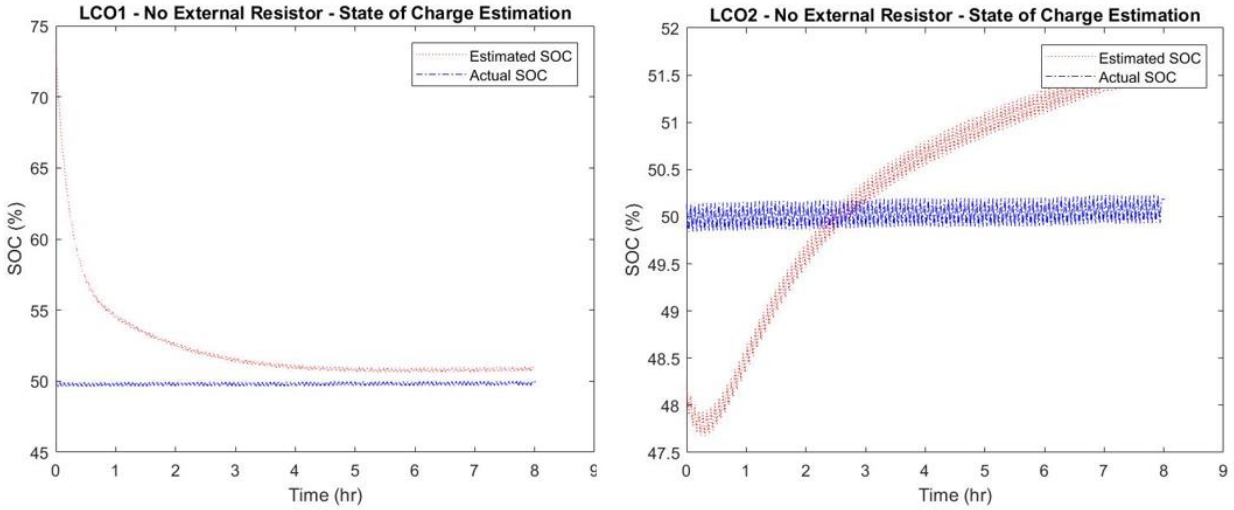

Figure 24. State of charge vs. time with no external resistor attached to the cells. Left, LCO1. Right, LCO2.

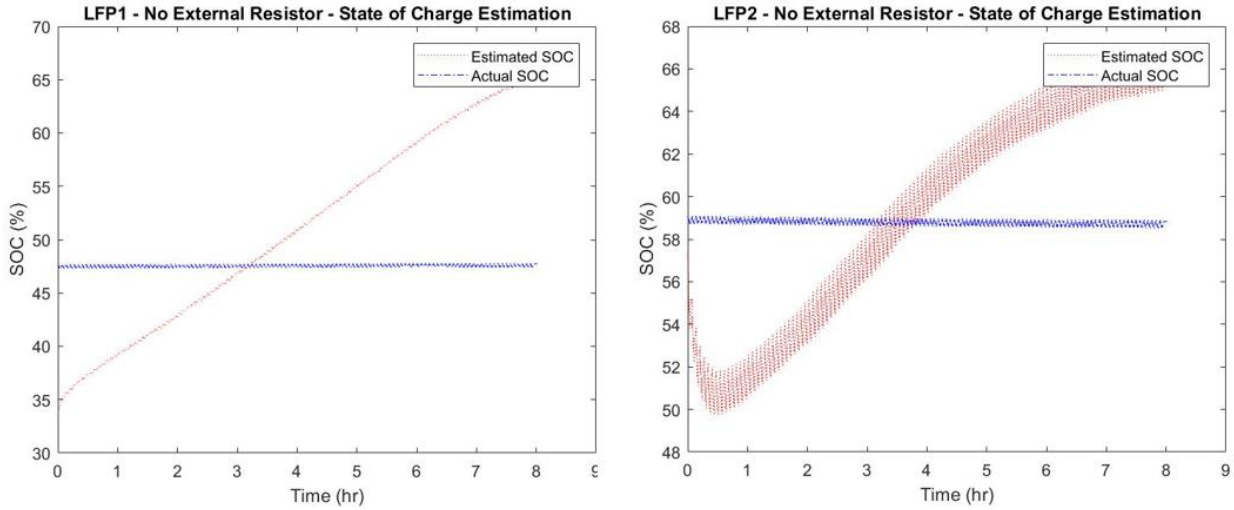

Figure 25. State of charge vs. time with no external resistor attached to the cells. Left, LFP1. Right, LFP2.

6.2 0.5 Ω External Resistor

To see if an addition of 0.5 Ω external resistor to the batteries can improve the performance of the algorithm, the algorithm was run again with the 0.5 Ω external resistance. Figure 26, 27 illustrate the current estimation error vs. time for LCO and LFP batteries while connected to an external 0.5 Ω resistor, respectively.

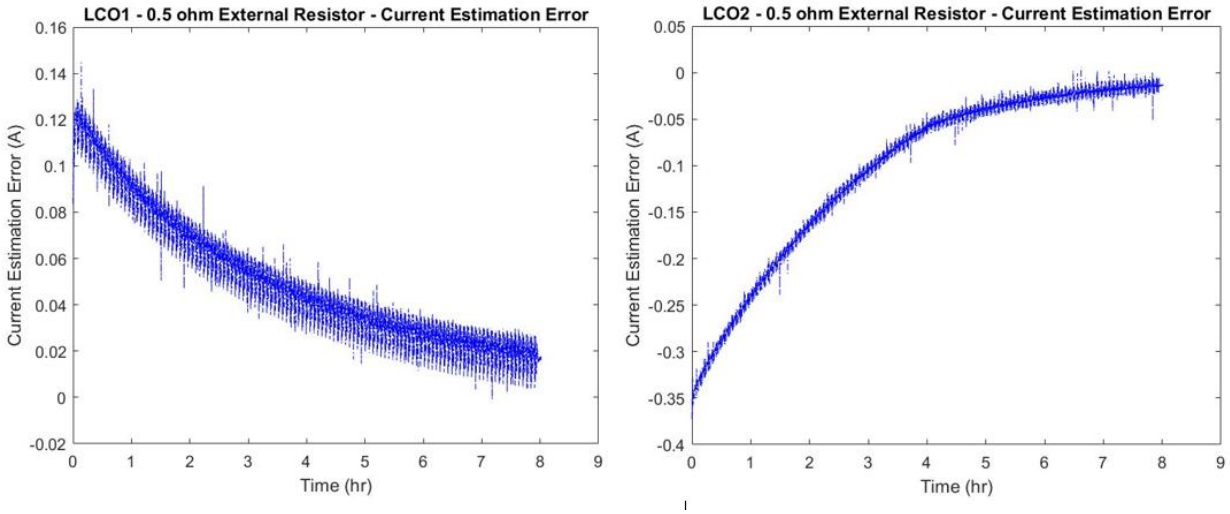

Figure 26. Current estimation error vs. time with 0.5 Ω external resistor attached to the cells. Left, LCO1. Right, LCO2.

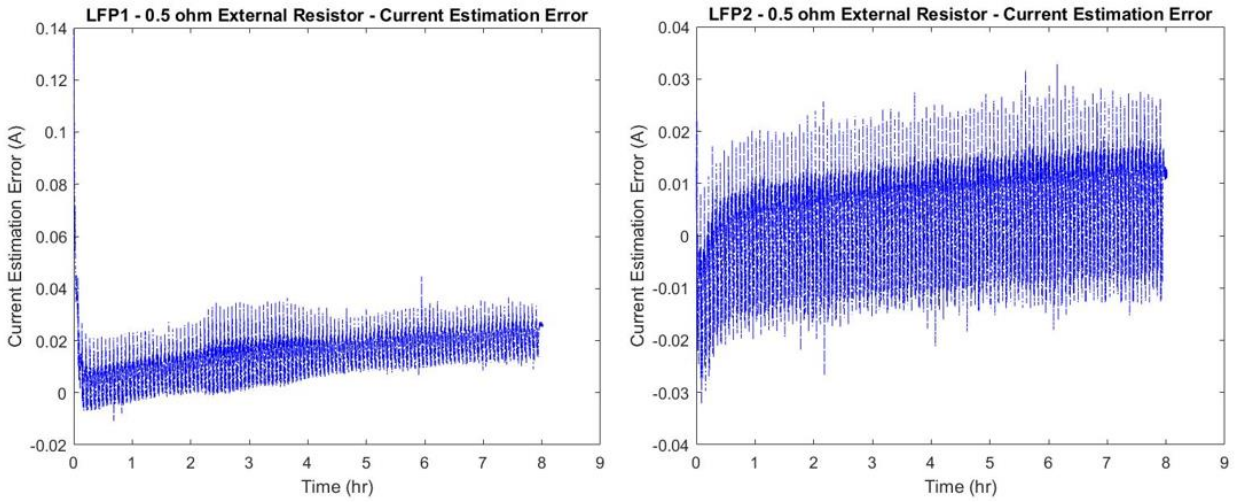

Figure 27. Current estimation error vs. time with 0.5 Ω external resistor attached to the cells. Left, LFP1. Right, LFP2.

As expected by the addition of external resistor, the current estimation error for all four batteries decreased significantly. Both of the cell types' current estimation are more precise and accurate even though, because of the already existing bias, none of the four estimation errors converged to exactly zero. The greatest amount of improvement in the estimation error is in LFP2, which became significantly more precise. All of the four cells estimation errors are small enough to conclude that the algorithm did successfully estimate the input current of the batteries.

Since the current estimation is improved by the addition of the external resistance, it was expected that the SOC estimation would also improve. The SOC estimation of both of the LCO cells improved and in the case of LCO1, it even converged to the right SOC. LFP cells' SOC estimation, on the other hand, did not improve by much and it still seems to be diverging from the actual SOC value. This means that the bias present in the LFP cells' estimation is still overpowering the enhancement in the current estimation by addition of the external resistance. There are suspect reasons for the bias in the estimation that will be discussed in the conclusion section.

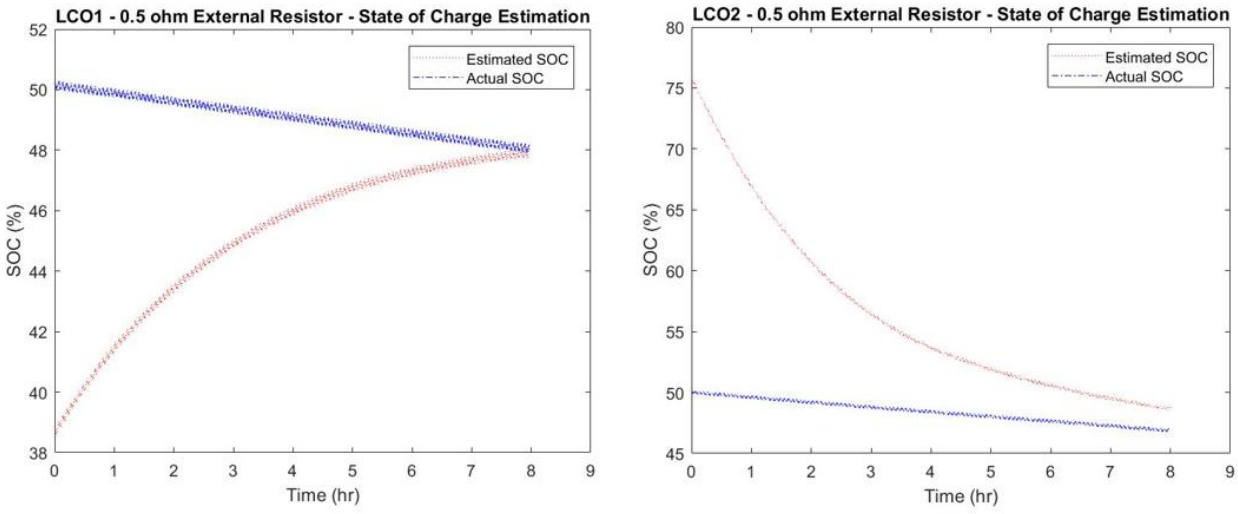

Figure 28. State of charge estimation vs. time with 0.5 Ω external resistor attached to the cells. Left, LCO1. Right, LCO2.

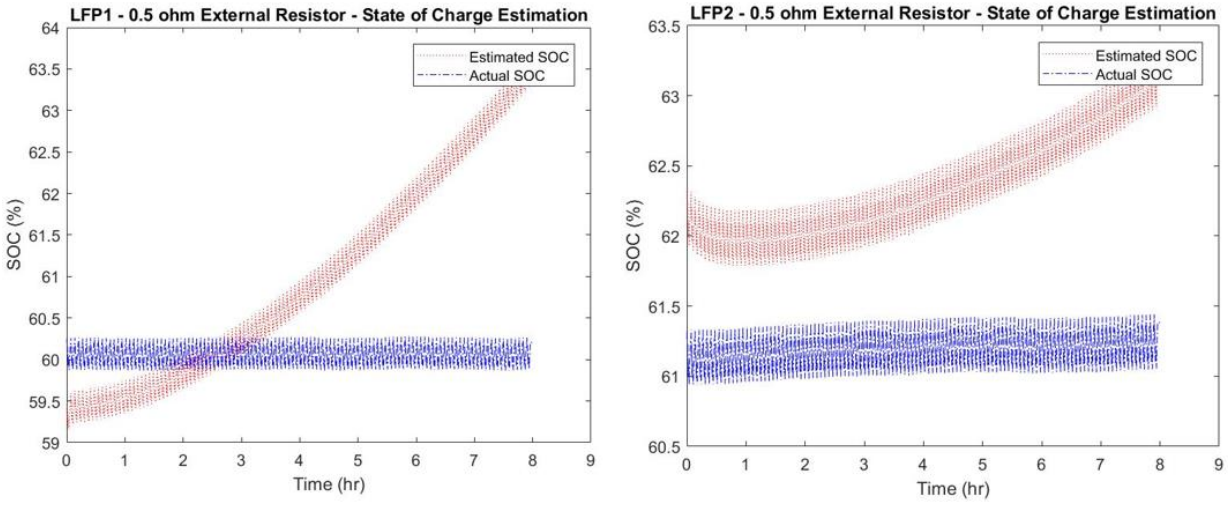

Figure 29. State of charge estimation vs. time with 0.5 Ω external resistor attached to the cells. Left, LFP1. Right, LFP2.

Chapter 7

Conclusion and Ideas for Further Research

This thesis develops an experimental method, aiming to experimentally validate a model-based algorithm discussed in Mishra et al [11]. In Mishra et al., they show the success of the algorithm in input current and SOC estimation using the voltage across the batteries. It is also mentioned in the paper that additional resistance across the batteries would improve the performance of the algorithm in estimating the input current. The findings in this thesis show that without an external resistance across the battery, the algorithm fails to estimate the input current and SOC of the battery effectively. This seems to be because of a bias in the system that the algorithm and the ECM model fail to take into consideration. After an addition of the 0.5 Ω resistor across the batteries, the algorithm can precisely and accurately estimate the input current of the battery to 1 decimal place which is quite desirable. SOC estimation, on the other hand, still suffers from the bias that caused inaccuracy and impreciseness in the case without the external resistance.

There are a couple of factors that may have caused this biased in the estimation. Firstly, the 3rd order ECM battery model that was used to model the lithium-ion batteries may not have been enough to model all the behaviors of the batteries. To accurately model a lithium-ion battery, an infinite order model is needed, which may not be possible to achieve experimentally. The fact that the bias was more apparent in the case of LFP cells compared to the LCO cells can further prove this point since LFP cells are more complex in behavior than LCO cells. In addition, lithium-ion batteries' behavior can be heavily affected by temperature and age. The experiment process took more than a year and was conducted in two different labs with possibly two different temperatures. Temperature and aging were not monitored during the experimental setup and may have caused a discrepancy in the estimation results. Lastly, it is assumed that the parameters during the parameterization process are constant while the battery goes through various cycles of input currents and voltages, which may not be necessarily correct, especially in the LFP

cells. Although with the addition of the 0.5 Ω resistor to the batteries, the algorithm estimated the input current to adequate precision and accuracy. In order to fully validate the algorithm experimentally, a more in-depth research regarding the three limitations above needs to be conducted.

Appendix A

Parameterization Source Code

Main Code:

```
% This is the main program to estimate the unknown parameters of the
third order battery ECM
clearvars;close all; clc

% load all the necessary data files
load('OCV_SOCMap_Q_LFP2.mat')
load('LFP2_Peyman_Characterization_Sergio.mat')

Time_V = Time_V - Time_V(1);
tinp = (linspace(Time_V(1),Time_V(end),length(Time_V)))';
Iinp = interp1(Time_V,I_V,tinp,'pchip');
Vinp = interp1(Time_V,V_V,tinp,'pchip');
samplespace=lhsdesign(50,6);
params_lb = [zeros(1,5) 0.4];
params_ub = [0.3 1e5*ones(1,4) 0.6];
paramsinitguess = ones(length(samplespace),1)*params_lb +
samplespace*diag(params_ub-params_lb);
options = optimset('Display','iter','TolFun',1e-8,'TolX',1e-
8,'MaxFunEvals',10000,'MaxIter',2400);

      for i=1:length(samplespace)

            [params_opt(i,:),Jopt(i)] =
            fminsearch(@(x)costfun(x,Q*3600,Iinp,OCV,SOC,tinp,Vinp),par
            amsinitguess(i,:),options);

      end
```

Costfun Function:

```
function J = costfun(params_unk,Q,u,OCV,SOC,time,Vexp)

      [Vpred,~] = ECM3order(params_unk,time,u,Q,OCV,SOC);
      J = sum((Vpred - Vexp).^2);

end
```

ECM3order Function:

```
% Script for simulating 3rd order ECM for system ID

function [V,States]=ECM3order(params,time,u,Q,OCV,SOC)

      Ro=params(1);tau1=params(2);tau2=params(3);C1 = params(4); C2 =
      params(5);
      A = diag([0;-1/tau1;-1/tau2]);B=[1/Q;1/C1;1/C2];
      C = eye(3); D = 0;
      sysECM = ss(A,B,C,D);
      [ECMstates,~,~] = lsim(sysECM,u,time,[params(6);zeros(2,1)]);
      V =
      interp1(SOC,OCV,ECMstates(:,1),'pchip')+sum(ECMstates(:,2:end),2)
      +u*Ro;
      States = ECMstates;

end
```

Appendix B

Validation Source Code

```
clear all, close all, clc
load('OCV_SOCMap_Q_LFP2.mat')

Cap_Ah = Q;
load('LFP2_Peyman_params.mat')
ext_resist = 0.5;
Rcellp = (params(1)+ext_resist);
tau1 = params(2); tau2 = params(3); C1 = params(4); C2 = params(5);


Sig_u = (0.0000)^2;
Sig_v = 0.0001^2; % Sensor noise variance
load('LFP2_HEV_05_8hours.mat')
time = (linspace(Time_V(1),Time_V(end),length(Time_V)))';
dt = time(2)-time(1);
z_k = interp1(Time_V,V_V,time,'pchip');
Ac=diag([0;-1/tau1;-1/tau2]);% continuous time A matrix
A = expm(Ac*dt); % discrete time A matrix
delt=linspace(0,dt,300);
Bc=[1/(Cap_Ah*3600);1/C1;1/C2]; % continuous time B matrix
B = zeros(3,1);

      for p=2:length(delt)

            B = B+0.5*(delt(p)-delt(p-
            1))*(expm(Ac*delt(p))*Bc+expm(Ac*delt(p-1))*Bc);

      end

[~,ia,ic]=unique(OCV);

I_m = interp1(Time_V,I_V,time,'pchip');
x_true(1) = interp1(OCV(ia),SOC(ia),z_k(1)-I_m(1)*ext_resist,'pchip');

      for i=2:length(time)

            x_true(i) = x_true(i-1)+I_m(i-1)*dt/(Cap_Ah*3600);

      end
```

```
%% NERTSIF

% Initialize observer
Px_post = diag([0.01;0.001*ones(2,1)]);Px_pri=Px_post;
x_post(:,1) = [x_true(1);zeros(2,1)]+sqrtm(Px_post)*randn(3,1);
x_pri(:,1)=x_post(:,1);
Pd = 0.5;
d_est(1) = 0;
Pxd = 0.01*ones(3,1);
Q = diag([B*sqrt(Sig_u)]);
Rk = (Sig_v+Rcellp^2*Sig_u);

      for k = 1:length(time)-1

            k/length(time)*100

            % input estimation
           Ck =
           [(interp1(SOC,OCV,x_pri(1,k)+x_pri(1,k)/1000,'pchip')...
            interp1(SOC,OCV,x_pri(1,k)-
           x_pri(1,k)/1000,'pchip'))/(2*x_pri(1,k)/1000),ones(1,2)];
           R_tilde = Ck*Px_pri*Ck'+Rk;
           Mk = (Rcellp'*R_tilde^-1*Rcellp)^-1*Rcellp'*R_tilde^-1;
           d_est(k)=Mk*(z_k(k)-interp1(SOC,OCV,x_pri(1,k),'pchip')-
           sum(x_pri(2:end,k)));
           Pd=(Rcellp'*R_tilde^-1*Rcellp)^-1;

           % measurement update
           Kk=Px_pri*Ck'*R_tilde^-1; % Kalman Gain
           x_post(:,k)=x_pri(:,k)+Kk*(z_k(k)-
           interp1(SOC,OCV,x_pri(1,k),'pchip')-sum(x_pri(2:end,k))-
           Rcellp*(d_est(k))); % (k-1)*(k>1)+(k==1))
           Px_post = Px_pri-Kk*(R_tilde-Rcellp*Pd*Rcellp')*Kk';
           Pxd = -Kk*Rcellp*Pd;

           % time update or prediction step

           x_pri(:,k+1)=A*x_post(:,k)+B*d_est(k);
           Px_pri=[A,B]*[Px_post,Pxd;Pxd',Pd]*[A,B]'+Q;

           y_est(k)=interp1(SOC,OCV,x_post(1,k),'pchip')+sum(x_post(2:
           end,k))+Rcellp*(d_est(k));

      end

k=k+1;
Ck = [(interp1(SOC,OCV,x_pri(1,k)+x_pri(1,k)/1000,'pchip')-...
            interp1(SOC,OCV,x_pri(1,k)-
x_pri(1,k)/1000,'pchip'))/(2*x_pri(1,k)/1000),ones(1,2)];
```

```
    R_tilde = Ck*Px_pri*Ck'+Rk;
    Mk = (Rcellp'*R_tilde^-1*Rcellp)^-1*Rcellp'*R_tilde^-1;
    d_est(k)=Mk*(z_k(k)-interp1(SOC,OCV,x_pri(1,k),'pchip')-
sum(x_pri(2:end,k)));
Pd=(Rcellp'*R_tilde^-1*Rcellp)^-1;
    Pd_save(k)=Pd;
Kk=Px_pri*Ck'*R_tilde^-1;
    x_post(:,k)=x_pri(:,k)+Kk*(z_k(k)-
interp1(SOC,OCV,x_pri(1,k),'pchip')-sum(x_pri(2:end,k))-
Rcellp*(d_est(k)));

%     x_post_corr(1,k) = x_post(1,k)- 0.038;
    Px_post = Px_pri-Kk*(R_tilde-Rcellp*Pd*Rcellp')*Kk';
    Px_save(:,k)=[Px_post(1,1);Px_post(2,2)];

y_est(k)=interp1(SOC,OCV,x_post(1,k),'pchip')+sum(x_post(2:end,k))+Rce
llp*(d_est(k));
```

BIBLIOGRAPHY

[1] D. Cambron and A. Cramer, "A Lithium-ion Battery Current Estimation Technique Using an Unknown Input Observer", *IEEE Transactions on Vehicular Technology*, vol., no. 99, 2017.

[2] J. Xu, C. Mi, B. Cao, J. Deng, Z. Chen and S. Li, "The State of Charge Estimation of Lithium-Ion Batteries Based on a Proportional-Integral Observer", *IEEE Transactions on Vehicular Technology*, vol. 63, no. 4, pp. 1614-1621, 2014.

[3] M. Charkhgard and M. Farrokhi, "State-of-Charge Estimation for Lithium-Ion Batteries Using Neural Networks and EKF", *IEEE Transactions on Industrial Electronics*, vol. 57, no. 12, pp. 4178-4187, 2010.

[4] M. Li, "Li-ion dynamics and state of charge estimation", *Renewable Energy*, vol. 100, pp. 44-52, 2017.

[5] H. He, R. Xiong, X. Zhang, F. Sun and J. Fan, "State-of-Charge Estimation of the Lithium-Ion Battery Using an Adaptive Extended Kalman Filter Based on an Improved Thevenin Model", *IEEE Transactions on Vehicular Technology*, vol. 60, no. 4, pp. 1461-1469, 2011.

[6] Z. Chen, Y. Fu and C. Mi, "State of Charge Estimation of Lithium-Ion Batteries in Electric Drive Vehicles Using Extended Kalman Filtering", *IEEE Transactions on Vehicular Technology*, vol. 62, no. 3, pp. 1020-1030, 2013.

[7] S. Yong, M. Zhu and E. Frazzoli, "Simultaneous input and state estimation for linear discrete-time stochastic systems with direct feedthrough", *52nd IEEE Conference on Decision and Control*, 2013.

[8] J. Ding, J. Xiao, and Y. Zhang, "Distributed input and state estimation for non-linear discrete-time systems with direct feedthrough", *IET Control Theory & Applications*, vol. 8, no. 15, pp. 1543-1554, 2014.

[9] P. Kitanidis, "Unbiased minimum-variance linear state estimation", *Automatica*, vol. 23, no. 6, pp. 775-778, 1987.

[10] S. Gillijns and B. De Moor, "Unbiased minimum-variance input and state estimation for linear discrete-time systems with direct feedthrough", *Automatica*, vol. 43, no. 5, pp. 934-937, 2007.

[11] P. P. Mishra and H. K. Fathy, “Lithium-ion battery disturbance current estimation, with application to a self-balancing photovoltaic battery storage system,” *2017 American Control Conference (ACC)*, 2017.

[12] S. Julier and J. Uhlmann, "Unscented Filtering and Nonlinear Estimation", *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401-422, 2004.

[13] M. Valcher, "State observers for discrete-time linear systems with unknown inputs", *IEEE Transactions on Automatic Control*, vol. 44, no. 2, pp. 397-401, 1999.

Peyman Norouzi

Academic Vita

316 West Beaver Ave, Apt. 511, State College, PA 16801 | pzn5052@psu.edu | (818)-574-9885

EDUCATION

The Pennsylvania State University, University Park, PA Expected Graduation - May 2018

Bachelor of Science, Mechanical Engineering
Dean's list (Every Semester)
Schreyer Honors College Scholar

RELATED EXPERIENCE

Penn State's Control Optimization Lab — *Researcher* January 2016 – Present
University Park, PA

- Validated the Combined State of Charge Estimation algorithm experimentally to estimate the battery's input current from its input voltage, substantially reducing various electronics' production cost.
- Modeled an 8th order Equivalent Circuit Model (ECM) of a Li-ion battery in MATLAB.
- Parameterized the battery model using a Least Squared Error algorithm in MATLAB.

Robert Bosch LLC — *Process and System Engineering Co-op* May 2017 – August 2017
Charleston, SC

- Performed cycle time studies on the setting process of the plant's tool shop, enhancing employees' efficiency by 14%.
- Conducted several capability studies on the production of the 1.00 mm high pressure bore in multiple diesel injectors.
- Wrote a product quality instruction (PQI) for the honing process of a grinding machine to assist future operators.

PUBLICATIONS

P. Norouzi, P. P. Mishra, and H. K. Fathy, "Experimental Validation of an Unknown Input Estimation Algorithm for Lithium-Ion Battery Applications," *57th American IEEE Decision and Control Conference* (Manuscript under preparation)

LEADERSHIP AND INVOLVEMENT

Penn State Wind Energy Club — *Electrical Team Lead* August 2017 – Present
The Pennsylvania State University, University Park, PA

- Led club's electrical engineering team, teaching new members about the basics of the electrical circuit design by analyzing last year's competition, control, and load circuits.
- Designed the new control and load circuits, following the 2018 Collegiate Wind Energy Competition requirements while improving upon last year's circuit design and performance.

Tau Beta Pi Engineering Honor Society — *Corporate Liaison* August 2017 – Present
The Pennsylvania State University, University Park, PA

- Built a stronger relationship between Penn State's chapter and corporate sponsors via scheduling various company events and fairs.

Operations Committee for THON 2017 — *Member* August 2016 – May 2017
The Pennsylvania State University, University Park, PA

- Contributed to THON 2017 organization, a student-run philanthropy committed to enhancing the lives of children and families impacted by childhood cancer, through coordinating various events throughout the year.

Penn State American Society of Mechanical Engineers — *Member* January 2016 – Present
The Pennsylvania State University, University Park, PA

Phi Kappa Phi Honor Society — *Member* August 2016 – Present
The Pennsylvania State University, University Park, PA

Engineering and Applied Science Interest House — *Webmaster* August 2015 – May 2016
The Pennsylvania State University, University Park, PA

- Improved club's website by implementing a searchable database, making it easier for the members to access their total activity points, required to maintain an active status.

TECHNICAL EXPERTISE

MATLAB, Simulink, C++, OpenCV, ROS, SolidWorks, HTML Coding, qs-STAT

MORE INFORMATION

LinkedIn: linkedin.com/in/peymannorouzi **Website:** personal.psu.edu/pzn5052/