The Pennsylvania State University

The Graduate School

College of Engineering

ON STATE OF CHARGE ESTIMATION ACCURACY FOR BATTERY CELLS CONNECTED IN SERIES

A Thesis in

Mechanical Engineering

by

Jariullah Safi

© 2015 Jariullah Safi

Submitted in Partial Fulfillment

of the Requirements

for the Degree of

Master of Science

May 2015

The thesis of Jariullah Safi was reviewed and approved* by the following:

Hosam Fathy
Assistant Professor of Mechanical Engineering
Thesis Advisor

Sean Brennan
Professor of Mechanical Engineering

Christopher Rahn
Professor of Mechanical Engineering

Karen A. Thole
Professor of Mechanical Engineering
Department Head of Mechanical and Nuclear Engineering

*Signatures are on file in the Graduate School.

Abstract

This thesis studies the joint estimation of state of charge (SOC) and current sensor bias in a series string of battery cells. Specifically we present - for the first time - a rigorous analysis of the benefits of estimating the SOC of multiple cells in a series string together along with supporting simulation evidence and discuss the role and benefits of SOC heterogeneity in a string of identical cells. While our approach applies generally to most battery chemistries, this thesis focuses on Lithium-Ion cells. We show through derivation of a Fisher information (FI) matrix using a linear first order integrator model of a cell that estimate variance decreases as a larger number of series connected cells is modeled together. The same derivation concludes that the presence of cells with a higher open-circuit voltage vs SOC slope can improve estimate variance even in small strings. We then test the results of the derivation in simulation for both an initial condition estimation problem and a state estimation problem representative of online estimation in battery management systems. This process introduces a modified dual polarization type equivalent circuit model of Lithium Ion batteries. This model maps a combination of the diffusion states and the state of charge to the output voltage: a nuance inspired by the output equation of a physics-based battery model. A discussion of model identification and the design of an extended Kalman filter (EKF) for a string composed of an arbitrary number of cells follows. For these purposes the initial conditions of all cells are unknown and the string current measurement is assumed to be biased. We then study the performance of the filter as a function of the number of modeled cells, the voltage vs SOC slope of individual cells (achieved by using different initial conditions for some cells), and the statistics of the current sensor. Monte Carlo simulations against a high fidelity physics based model of a Lithium-Iron-Phosphate string of cells show that increasing the number of modeled cells and heterogeneity in cell initial conditions can improve estimator performance in the presence of current noise: a result that extends earlier analyses of the impact of current bias. The thesis concludes with a discussion of the results as well as some pack design recommendations.

Table of Contents

List of Figures

List of Tables

Acknowledgments

First and foremost, I will begin by thanking Allah for giving me the opportunity to get to this point, and for the strength to get past it. I would also like to thank my mother, Amatur Rauf, for raising me well and my father, Muhammad Zafrullah, for showing me the meaning of persistence. My brother Anser for helping to raise me and for supporting our family when we needed it most. My brother Ahsan for showing me that Math and Programming are powerful allies. My brother Danyal for always listening when I needed to scream out in frustration, and for giving me invaluable advice about graduate life and life in general and my sister Bano for always being there for me. My sisters in law Maria, Huma, and Hibba have all helped shape me into the person I am now. Thank you for that and also for also giving me the gift of the most wonderful nephews and niece ever.

Life is never complete without friends; they are the family we choose. The Bowens are there for me when I need them. Thank you Claire and thank you Mack for being there for me and for the memories (and for showing me that a kilt is actually quite awesome to wear). Thank you Michelle for showing me that Dr. Who becomes even better paired with homemade food and wonderful company, now if only we could decide on a location for the bakery. Ted, Nishit, and Hauser, thank you for making my stay at Penn State a delightful one. There are also old friends, those close to my heart but geodetically far. Khurram, Daud, and Humayun, thank you for everything. I hope you are all well, wherever you are.

Colleagues at Penn State deserve my thanks as well. First and foremost are my advisors: Drs. Fathy, Brennan, and Anstrom. Thank you for supporting me through my degree and for providing me the guidance needed to finish this work. I would also like to thank Dr. Rahn for agreeing to be the reader for my thesis and for providing valuable feedback on the work and Dr. Langelaan for teaching me the basics of estimation theory.

I want to thank the CoOL people at Penn State. Mike generated current trajectories that I use in this thesis and he was there to listen to me complain about graduate life. Donald is always

good to bounce ideas off of and listens to me complain about graduate life. Sergio, Mohammad, Abdullah, Ji, and Michelle are always good sources of stimulating conversation and listen to me complain about graduate life. Beeney says I should stop complaining about graduate life. If only I had the will, I could find my Wei again...

I would also like to acknowledge some of the ControlFreaks in my life. Jesse and Mangus, thanks for the 5pm gaming hour. Kshitij, you were my office mate for two years and you survived, bravo! (also thank you for always listening and dispensing wonderful advice). Bobby and Kelilah, lunch with you guys is never dull, especially when it involves pouring hot syrup. Patrick, I hope you get to fight a real fire one day.

Finally I would like to acknowledge the financial support from the government agency ARPA-E.

Dedication

To my mother.

And my mother.

And my mother.

And my father.

And to my brothers, their wives, and their children.

Chapter 1

Introduction

This thesis studies the joint estimation of state of charge (SOC) and current sensor bias in a series string of battery cells using both: 1) Fisher information analysis validated through the use of a modified equivalent circuit model; and 2) extended Kalman filtering against a high fidelity physics-based model.

Estimating cell SOC is one of the main tasks that a battery management system (BMS) must perform [2]. Defined as the ratio of charge contained to maximum usable capacity, knowledge of SOC gives the user an insight into both the energy contained in and potential power draw from a battery pack [19, 36]. This is particularly important in electric vehicle applications. It would be unfortunate for a battery pack to get depleted or be unable to provide the necessary amount of power to climb a steep hill without prior warning. As such, researchers have and continue to contribute knowledge to this field [20].

1.1 Literature Review

It is possible to classify the literature on SOC determination by either the model used or the estimation strategy. For estimation and control purposes it is possible to mathematically represent Li-Ion cells either as "black boxes" using machine learning techniques [4] or using two model structures: 1) modeling of the electrochemical and physics processes inside a cell [9] and 2) approximating the cell as an electrical circuit [14]. Physics-based models are typically simplified for control purposes in two ways: 1) reduction of the number of equations by making assumptions about the operating conditions [26] and 2) by reducing the model in the frequency domain using a method called Pade Approximations [31]. Most battery estimation work has focused on equivalent circuit models during the past decade and a half [38]. This is the result of the relative simplicity and adequate predictive power of such models. The present work also focuses on these models.

The lowest-order dynamic mathematical representation of a Li-Ion battery comes in the form of a nonlinear capacitor with a series resistance. The capacitor voltage - also the rest voltage - is a function of the SOC. Such a model includes both the integrator effect of charge accumulation

and the low frequency impedance of a cell. It also leads to the most elementary method of SOC estimation: integrating the input current (a.k.a Ah or coulomb counting) [20]. Any errors introduced because of current sensor bias or initial condition uncertainty can be partially resolved by inverting the battery's voltage during rest periods [5, 17]. Model based estimation methods that use voltage feedback are more commonly studied because they can make use of the voltage measurements for each cell available in typical vehicle battery packs [2],

An early example of model based estimation for Li-Ion cells comes from Plett who used an equivalent circuit model and the extended Kalman filter (EKF) [21]. The EKF is a technique that allows the use of the linear Kalman filter with non-linear models. The literature presents many variants of the EKF method. Some examples are:

- Santhanagopalan and White use it with a simplified electrochemical model [25]
- He et al. use it with an adaptive law for better filter performance [12]
- Xiong et al. estimate the state of charge for a pack using an average model [37]
- Di Domenico et al. consider a physics inspired model with thermal effects and use an adaptive law for better performance [7].
- Fang et al. use an iterated version of the EKF that minimizes the error between predicted and measured voltage at each time-step [10].

Other methods including the unscented Kalman filter [27], particle filtering [28], and support vector regression [11] have also been explored. This work, however, focuses on EKFs. These techniques tend to be designed for individual cells. Vehicle battery packs can contain hundreds of cells separated into multiple parallel strings [2]. This complicates the estimation process because of increased computational needs and the fact that each cell may behave differently [3].

There has been some work in the last few years on estimation for an entire battery pack. Most approaches use an average pack model with local corrections for individual cells [6, 22, 23, 33]. Plett [22] also focuses on simultaneous estimation of current sensor bias for a battery pack. Estimating current bias is important owing to the inherent difficulties in current measurement. He et al. [13] and Liu et al. [16] recognize this fact and the present authors explored it in a conference paper [24].

1.2 Contributions

All of the methods discussed so far suffer from the same fundamental problem, namely the flatness of the cell's voltage response curve. The detrimental effect of this lack of sensitivity is an increase in SOC estimation error in a "middle" SOC range, typically between about 30% and 80% SOC, as shown mathematically by Sharma et al. and Lin et al. [15, 29]. To alleviate this problem, Wang et al. proposes changing the negative electrode of the cell in a way that would cause small jumps in voltage at pre-defined SOC points [34, 35]. This technique would improve estimation accuracy near those SOC values and can be very helpful for hybrid electric vehicle battery packs. In contrast, the author's past paper [24] explored an algorithmic approach to this problem by using the information from multiple cells connected in a series string together. Simulations show that estimation accuracy could improve for series connected strings even if all of them were in the flat region. We further showed that having one cell with a different initial condition than others - specifically in the non flat region - also improves estimation accuracy.

This thesis is an extension of the above paper with a deeper look at two hypotheses:

1. Considering more than one cell in an SOC estimator will cause the estimate variance to decrease, even if all cells are in the flat region.

2. Adding one cell from the non flat region in a string will also improve estimate variance.

The major contributions of this thesis are:

1. Derivation of the Cramer-Rao lower bound (CRLB) on estimate variance for SOC estimation in a series string of cells.

2. A simulation based study of the performance of joint estimation of SOC and current bias in a series string as a function of the number of cells, variety of initial conditions, and the statistics of the current sensor. This study validates the results of the CRLB derivation.

To accomplish the above we also introduce a modified equivalent circuit (MEC) model inspired by Docimo et al. [8] and present a detailed design for an EKF for a multi-cell string with an arbitrary number of cells.

The rest of the thesis is organized as follows:

Chapter 2 contains information on the various models that appear in this thesis and how they relate to one another. The model identification strategy and parameters for the MEC model is also presented.

Chapter 3 presents the detailed derivation of the CRLB using the lowest fidelity model. It also contains comparisons to the results from the author's past paper [24].

Chapter 4 contains information on how to design an EKF for SOC estimation using the MEC model.

Chapter 5 shows the setup and results of Monte Carlo simulations using a high fidelity physics-based model to represent the real battery and the MEC model for estimation. A discussion shows that these simulations echo the analytical results of Chapter 3.

Chapter 6 concludes the thesis.

Chapter 2

Models used for Analytical Study, Estimation, and Simulation of the Battery

The contributions of this thesis rely on three different but closely related models of a battery cell:

1. A commercial high fidelity physics-based model for simulating a real battery and generating voltage data for given current trajectories. The model is used for Monte Carlo simulations of SOC estimation accuracy, thereby providing a simulation platform for validating the analytical results defined in this thesis.

2. A third order nonlinear modified equivalent circuit model similar in complexity to models typically used for online control and estimation purposes. The model is identified from the above physics based model. It is then used for design of the state estimator (EKF), which is ultimately used for the Monte Carlo simulations.

3. A first order linearized integrator model used in the CRLB derivation because of its simplicity. This model is similar to the one used in [24] and [15].

The rest of this chapter presents more detail on these models including mathematical representations and model identification information.

2.1 High Fidelity Physics-Based Model

To expedite testing of the hypotheses presented in this thesis, we use a high fidelity physics-based model to simulate battery behavior in lieu of a real cell. This model is contained in EC Power's commercial tool Autolion 1D. More information is available at EC Power's Website [1]. Specifically, we simulate a Lithium-Iron-Phosphate 32113 battery cell with a nominal capacity of 4.3 Ah and rest voltage limits between 2.0 V and 3.4 V.

2.2 Third Order Nonlinear Modified Equivalent Circuit Model

From a state space perspective there are three state variables in the modified 2RC (M2RC) model. The first state variable represents the average SOC of a cell. Equation 2.1 shows the time evolution of this state variable with the assumption that $x_1(0) = SOC_0$.

$$\dot{x}_1(t) = \frac{u(t)}{C} \tag{2.1}$$

Where the input $u(t)$ is the applied external current and C is the usable capacity of the cell. Note that this state variable is a pure integrator and describes the capacitive nature of a battery cell.

The remaining two state variables are represented by the first order model in equation 2.2 for $i \in \{1, 2\}$.

$$\dot{x}_i(t) = -\frac{1}{\tau_i} x_i(t) + \frac{1}{C_i} u(t) \tag{2.2}$$

Note that C_i is the capacitance of the corresponding RC pair and τ_i represents the time constant. These state variables provide an abstraction to the various diffusion like processes that take place on a microscopic level inside a battery cell and models the associated lag using two first order systems. Given a long enough rest period (e.g. parking during work hours) these state variables will diminish to zero. In this work we assume that each test starts on a rested battery and use zero as the initial value for these state variables.

Equation 2.3 shows an expression for the terminal voltage $y(t)$ of this model.

$$y(t) = g\left(x_1 + x_2 + x_3\right) + u(t)r \tag{2.3}$$

Where r is the ohmic resistance and $g(x)$ is the voltage measured from the cell with zero input after a long rest (i.e. open circuit voltage).

2.2.1 Identification Tests

Two simulation tests help identify the capacitances, time constants, resistance, and $g(x)$ for the 3^{rd} order model from the higher fidelity simulation.

1. Discharge a cell from 100% SOC to 0% SOC and charge it back to 100% SOC using a very small constant current.

2. Subject the cell to a series of pulses at various SOCs.

The **first** test is useful for determining the cell capacity C by multiplying the current value with the time it takes to completely charge or discharge the cell. Equation 2.4 defines the cell capacity as the average of the charge and discharge capacities.

$$C = \frac{|I_{chg} \times T_{chg}| + |I_{dis} \times T_{dis}|}{2} \tag{2.4}$$

Note that the 0% and 100% SOC values are defined by the low and high voltage limits set by cell manufacturers. Normalizing the test time so that it goes from 0 to 1 for charging and from 1 to 0 for discharging gives $g_{chg}(x)$ and $g_{dis}(x)$. Averaging these functions gives $g(x)$ as equation 2.5 shows.

$$g(x) = \frac{g_{chg}(x) + g_{dis}(x)}{2} \tag{2.5}$$

The **second** test depends on the results of the first because knowledge of C and $g(x)$ helps determine the remaining components of the model. Fig. 2.1 shows the current and SOC trajectories for this test.

Using the voltage response from the pulse test it is possible to find the remaining parameters by solving a least squares problem. Equation 2.6 sets up this minimization problem for $\vec{\theta} = \{r, \tau_1, \tau_2, C_1, C_2\}$, and table 2.1 shows the resulting parameter estimates.

$$\vec{\theta} = \arg\min_{\vec{\theta}} \left\| \vec{ym} - \vec{y}\left(\vec{\theta}\right) \right\|_2^2 \tag{2.6}$$

2.2.2 Results of Identification Tests

EC Power's Autolion 1D generates the voltage trajectories for identification of the M2RC model. Fig. 2.2 shows the voltage curves as a function of SOC derived from the first test.

The test also gives $Q = (4.288 + 4.279)/2 = 4.284$ Ah. Using these results, the M2RC model, and a nonlinear least squares minimization routine leads to the results of Table 2.1. In addition,

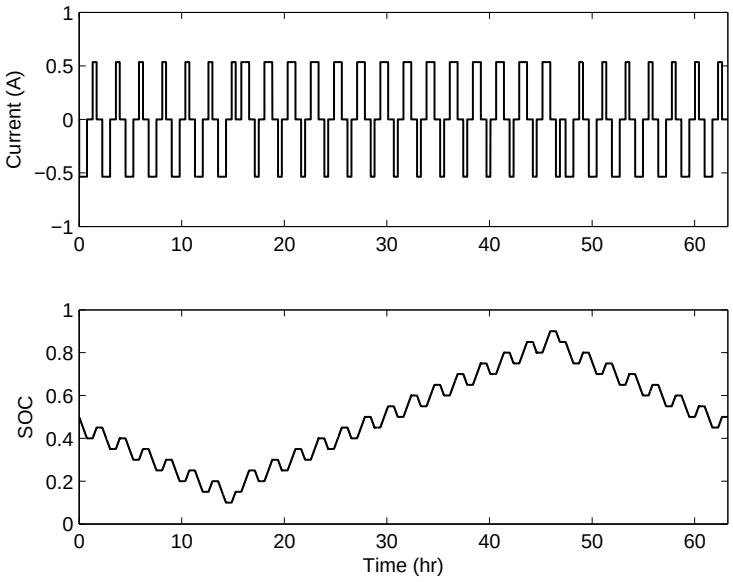

Figure 2.1. Current and SOC Trajectory for Pulse Test

Table 2.1. Model Parameters

Parameter	Estimated Value
r	0.0074 Ω
τ_1	113 s
τ_2	1.7×10^5 s
C_1	2.6 Ah
C_2	260 Ah

Fig. 2.3 shows the true voltage superimposed with the best model output based on identified parameters for the pulse test. There is good agreement between the data and the fit.

It is necessary to validate the model's predictive powers by subjecting it to an input different from the pulse test. We apply a repeated current trajectory derived from the Federal Test Procedure (FTP) 75 that discharges the cell from 95% to 5% SOC. Fig. 2.4 shows a normalized histogram of the error between truth and the M2RC model. The errors are almost entirely contained in a ±5mV bound and are centered about zero.

2.3 The First Order Linearized Integrator Model

The first order integrator model of a battery assumes that, from an estimation perspective, the most important characteristics of a battery are the facts that 1) it integrates the current and 2)

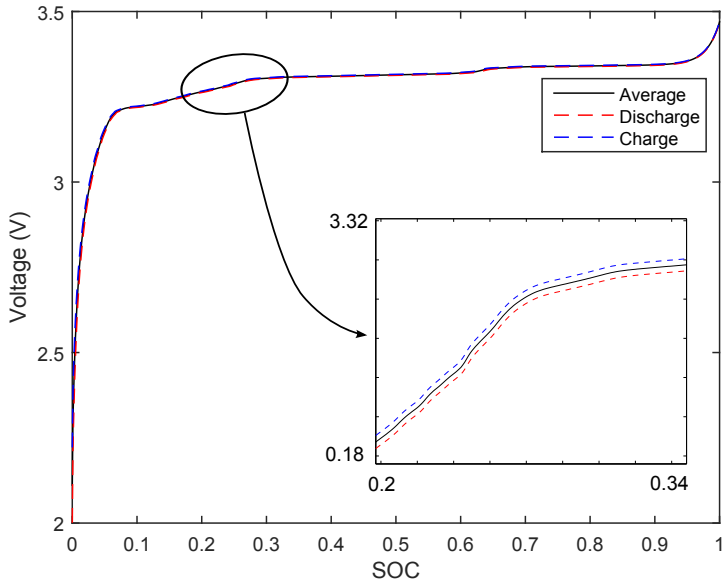

Figure 2.2. Open Circuit Voltage Trajectories

it has a non-linear voltage response as a function of the SOC. Equations 2.7 and 2.8 show the continuous time representation of the model equations.

$$\dot{x}(t) = \frac{u(t)}{C} \tag{2.7}$$

$$y(t) = g\left(x(t)\right) + u(t)r \tag{2.8}$$

Where x is the SOC for this model and the rest of the symbols have the same interpretation as that in the M2RC model. On close examination it is apparent that this model is the same as the M2RC model but without the lag states. The state dynamics of this model are linear but the output equation is not. Linearizing the model about a reference state of charge changes the output to equation 2.9.

$$y(t) = \alpha x(t) + u(t)r \tag{2.9}$$

Where α is the slope of $g(\cdot)$ at the reference point. This representation allows for the use of linear analysis tools to predict estimator performance as a function of the output sensitivity (which is

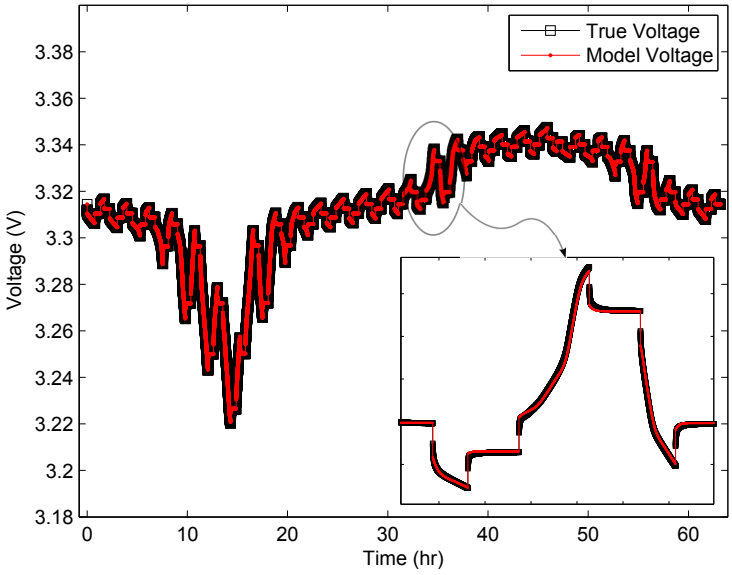

Figure 2.3. Voltage Trajectories from the Model Fit

directly related to α). The next chapter presents such an analysis. Specifically, we derive a lower bound for SOC estimation variance using a linearized integrator model.

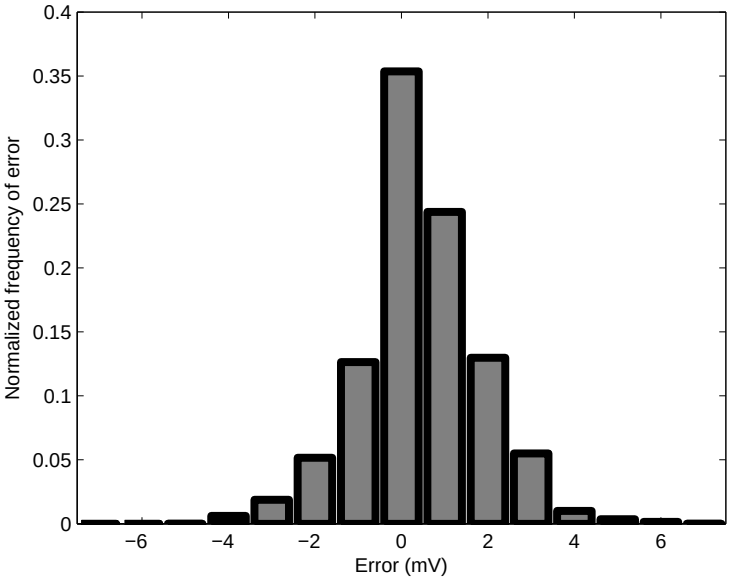

Figure 2.4. Histogram of Errors Induced by the M2RC Model

Chapter 3

Analytical Derivation of SOC Estimation Accuracy using the Fisher Information Matrix

This chapter's focus is on the computation of the Fisher information (FI) matrix for initial SOC estimation in a series string of battery cells using the linearized integrator model. Inversion of the Fisher information matrix gives a best case lower bound for estimator covariance; this fact is given by the Cramer-Rao lower bound (CRLB) theorem [18]. The resulting variance expression for individual cell SOC estimation allows us to analyze the results of [24] and lays the ground for further, more realistic, testing of our hypotheses using Autolion and the M2RC model.

First we present the FI expression for a general linear system. We then compute it for a string of battery cells and invert it. Finally we analyze the resulting variance expression and compare it to the results from [24].

3.1 Fisher Information Matrix for a Free Linear System with Gaussian Measurement Noise

Consider a discrete-time free linear system (or one in which the input is known perfectly), with the system state at time k being $\vec{x}_k$ and the measured output being $\vec{y}_k$.

$$\vec{x}_{k+1} = A\vec{x}_k \tag{3.1}$$

$$\vec{y}_k = C\vec{x}_k + v_k \tag{3.2}$$

Suppose $v_k \sim \mathcal{N}\left(0, \sigma_v^2\right)$ is an independent identically distributed normal noise process with zero mean and variance σ_v^2. If the initial condition $\vec{x}_0$ is unknown and the matrices A and C are perfectly known then it is possible to set up a static estimation problem by writing the output as follows:

$$\begin{bmatrix} \vec{x}_1 \\ \vec{x}_2 \\ \vdots \\ \vec{x}_N \end{bmatrix} = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^N \end{bmatrix} \vec{x}_0 \tag{3.3}$$

If we define $\mathbb{C} = \text{blockdiag}\begin{bmatrix} C & C & \cdots & C \end{bmatrix}_{N \times N}$ where C is repeated N times and define $\mathbb{A} = \begin{bmatrix} A^T & \left(A^2\right)^T & \cdots & \left(A^N\right)^T \end{bmatrix}^T$, then the measurement over a given time period takes the form:

$$\vec{Y} = \mathbb{C}\mathbb{A}\vec{x}_0 + \vec{v} \tag{3.4}$$

Note that $\vec{Y}$ is a noise process with mean $\mathbb{C}\mathbb{A}\vec{x}_0$ and covariance $\Sigma_v = \sigma_v^2 I_{N \times N}$. The Fisher information matrix for the problem in equation 3.4 can be computed by using the sensitivity of the output $\vec{Y}$ to the initial conditions $\vec{x}_0$. Equation 3.5 shows the resulting expression.

$$\mathbb{I} = \mathbb{A}^T \mathbb{C}^T \Sigma_v^{-1} \mathbb{C}\mathbb{A} \tag{3.5}$$

The CRLB theorem states that a lower bound on the covariance of the "best case" estimate of $\vec{x}_0$ is given by the inverse of the FI matrix. Note that $\Sigma_v^{-1} = \frac{1}{\sigma_v^2} I_N$.

3.2 Fisher Information Matrix for Initial Value Estimation in a String of Battery Cells

The Fisher information expression of equation 3.5 is a general expression that will work for any A and C matrix combination. Assuming a string of M cells connected in series and subjected to a current input u which is measured as u_m with a constant bias b, the linearized discrete state space model for the i^{th} cell is given in Equations 3.6 and 3.7,

$$x_{k+1}^{(i)} = x_k^{(i)} + \frac{u_k}{q} \ , \tag{3.6}$$

$$y_k^{(i)} = \alpha^{(i)} x_k^{(i)} + u_k r \ , \tag{3.7}$$

where we have assumed a time step of 1 second and have discretized a linearized continuous integrator model of a battery cell. The capacity of the cell is q and the internal resistance of the cell is r. The slope of the cell's rest voltage vs. SOC is $\alpha^{(i)}$. The measured current input is $u_m = u + b$. We make the bias b a state variable in the system in order to estimate it. Given the assumption that b is constant, the A matrix for the entire string of M cells takes the form in equation 3.8,

$$A = \begin{bmatrix} I_M & -\vec{q} \\ \vec{0} & 1 \end{bmatrix}, \tag{3.8}$$

where $\vec{q} = \frac{1}{q} \begin{bmatrix} 1 & \cdots & 1 \end{bmatrix}^T_{M \times 1}$. Similarly, equation 3.9 shows the C matrix.

$$C = \begin{bmatrix} \alpha_1 & 0 & \cdots & 0 & -r \\ 0 & \alpha_2 & \cdots & 0 & -r \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & \alpha_M & -r \end{bmatrix} = \begin{bmatrix} \Omega_M & -\vec{r} \end{bmatrix} \tag{3.9}$$

Note that the block notation for the matrices is for ease of the incoming derivation.

Using the A and C matrices we can construct $\mathbb{A}$ and $\mathbb{C}$. Equations 3.10 through 3.12 show the relevant matrices.

$$A^i = \begin{bmatrix} I_M & -i\vec{q} \\ \vec{0} & 1 \end{bmatrix} \tag{3.10}$$

$$\mathbb{A} = \begin{bmatrix} I_M & -\vec{q} \\ \vec{0} & 1 \\ I_M & -2\vec{q} \\ \vec{0} & 1 \\ \vdots & \vdots \\ I_M & -N\vec{q} \\ \vec{0} & 1 \end{bmatrix} \tag{3.11}$$

$$\mathbb{C} = \begin{bmatrix} \Omega_M & -\vec{r} & & & \cdots & & \\ & & \Omega_M & -\vec{r} & \cdots & & \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ & & & & \cdots & \Omega_M & -\vec{r} \end{bmatrix} \tag{3.12}$$

Equation 3.13 shows the product of the matrices $\mathbb{C}\mathbb{A}$.

$$\mathbb{C}\mathbb{A} = \begin{bmatrix} \Omega_M & -\Omega_M\vec{q} - \vec{r} \\ \Omega_M & -2\Omega_M\vec{q} - \vec{r} \\ \Omega_M & -3\Omega_M\vec{q} - \vec{r} \\ \vdots & \vdots \\ \Omega_M & -N\Omega_M\vec{q} - \vec{r} \end{bmatrix} \tag{3.13}$$

The resulting FI matrix then has the form in equation 3.14

$$\mathbb{I} = \frac{1}{\sigma_v^2} \begin{bmatrix} N\Omega_M^2 & -\Omega_M \sum_{i=1}^{N} \left(i\Omega_M\vec{q} + \vec{r}\right) \\ -\left(\sum_{i=1}^{N} \left(i\Omega_M\vec{q} + \vec{r}\right)^T\right)\Omega_M & \sum_{i=1}^{N} \left(i\Omega_M\vec{q} + \vec{r}\right)^T \left(i\Omega_M\vec{q} + \vec{r}\right) \end{bmatrix} \tag{3.14}$$

3.3 Inversion of Fisher Information Matrix

The information matrix can be partitioned as a 2×2 block diagonal matrix. Equation 3.15 shows the inverse for such a matrix.

$$\mathbb{I} = \frac{1}{\sigma_v^2} \begin{bmatrix} a & b \\ c & d \end{bmatrix} \Rightarrow \mathbb{I}^{-1} = \sigma_v^2 \begin{bmatrix} a^{-1} + a^{-1}b\left(d - ca^{-1}b\right)^{-1} ca^{-1} & -\left(a - bd^{-1}c\right)^{-1} bd^{-1} \\ -d^{-1}c\left(a - bd^{-1}c\right)^{-1} & d^{-1} + d^{-1}c\left(a - bd^{-1}c\right)^{-1} bd^{-1} \end{bmatrix} \tag{3.15}$$

The first element of this matrix contains the individual variance terms for each cell's SOC estimates. Systematic computation of this element follows. First an expansion of d.

$$d = \sum_{i=1}^{N} \left(i\vec{q}^T\Omega_M + \vec{r}^T\right)\left(i\Omega_M\vec{q} + \vec{r}\right) \tag{3.16}$$

$$
\begin{aligned}
&= \sum_{i=1}^{N} \vec{q}^T \Omega_M^2 \vec{q} i^2 + i\vec{q}^T \Omega_M \vec{r} + i\vec{r}^T \Omega_M \vec{q} + \vec{r}^T \vec{r} \\
&= \vec{q}^T \Omega_M^2 \vec{q} \sum_{i=1}^{N} i^2 + \vec{q}^T \Omega_M \vec{r} \sum_{i=1}^{N} i + \vec{r}^T \Omega_M \vec{q} \sum_{i=1}^{N} i + \vec{r}^T \vec{r} \sum_{i=1}^{N} 1 \\
&= \vec{q}^T \Omega_M^2 \vec{q} \left(\frac{N}{6} (N+1)(2N+1) \right) + \vec{q}^T \Omega_M \vec{r} \left(\frac{N}{2} (N+1) \right) + \vec{r}^T \Omega_M \vec{q} \left(\frac{N}{2} (N+1) \right) + N \vec{r}^T \vec{r}
\end{aligned}
$$

The form of the matrices and vectors involved in equation 3.16 leads to the equality $\vec{q}^T \Omega_M \vec{r} = \vec{r}^T \Omega_M \vec{q}$. This reduces d to equation 3.17.

$$
d = \vec{q}^T \Omega_M^2 \vec{q} \left(\frac{N}{6} (N+1)(2N+1) \right) + N(N+1) \vec{q}^T \Omega_M \vec{r} + N \vec{r}^T \vec{r} \tag{3.17}
$$

Now, an expansion of $ca^{-1}b$:

$$
ca^{-1}b = \left(\sum_{i=1}^{N} \left(i\Omega_M \vec{q} + \vec{r} \right)^T \right) \Omega_M \frac{1}{N} \left(\Omega_M^2 \right)^{-1} \Omega_M \sum_{i=1}^{N} \left(i\Omega_M \vec{q} + \vec{r} \right) \tag{3.18}
$$

Again, because of the structure of the Ω_M matrix, $\Omega_M \frac{1}{N} \left(\Omega_M^2 \right)^{-1} \Omega_M = \frac{1}{N} I_M$. Equation 3.19 shows the rest of the computation for this term.

$$
\begin{aligned}
ca^{-1}b &= \frac{1}{N} \left(\sum_{i=1}^{N} \left(i\Omega_M \vec{q} + \vec{r} \right)^T \right) \left(\sum_{i=1}^{N} \left(i\Omega_M \vec{q} + \vec{r} \right) \right) \\
&= \frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{N} \left(i\vec{q}^T \Omega_M + \vec{r}^T \right) \left(j\Omega_M \vec{q} + \vec{r} \right) \\
&= \frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{N} ij\vec{q}^T \Omega_M^2 \vec{q} + j\vec{r}^T \Omega_M \vec{q} + i\vec{q}^T \Omega_M^2 \vec{r} + \vec{r}^T \vec{r} \\
&= \frac{1}{N} \left(\frac{1}{4} N^2 (N+1)^2 \vec{q}^T \Omega_M^2 \vec{q} + N^2 (N+1) \vec{q}^T \Omega_M^2 \vec{r} + N^2 \vec{r}^T \vec{r} \right) \\
&= \left(\frac{1}{4} N (N+1)^2 \vec{q}^T \Omega_M^2 \vec{q} + N(N+1) \vec{q}^T \Omega_M^2 \vec{r} + N \vec{r}^T \vec{r} \right)
\end{aligned} \tag{3.19}
$$

These expressions lead to equation 3.20.

$$\begin{aligned} d - ca^{-1}b &= \vec{q}^T \Omega_M^2 \vec{q} \left[\frac{1}{6} N (N+1)(2N+1) - \frac{1}{4} N (N+1)^2 \right] \\ &= \frac{1}{12} \vec{q}^T \Omega_M^2 \vec{q} \left[2N (N+1)(2N+1) - 3N (N+1)^2 \right] \\ &= \frac{1}{12} \vec{q}^T \Omega_M^2 \vec{q} \left[N \left(N^2 - 1 \right) \right] \end{aligned} \tag{3.20}$$

Equation 3.21 shows the computation for the bc term.

$$\begin{aligned} bc &= \Omega_M \sum_{i=1}^{N} \left(i\Omega_M \vec{q} + \vec{r} \right) \left(\sum_{i=1}^{N} \left(i\Omega_M \vec{q} + \vec{r} \right)^T \right) \Omega_M \\ &= \Omega_M \left(\sum_{i=1}^{N} \sum_{j=1}^{N} \left(i\Omega_M \vec{q} + \vec{r} \right) \left(j\Omega_M \vec{q} + \vec{r} \right)^T \right) \Omega_M \\ &= \Omega_M \left(\sum_{i=1}^{N} \sum_{j=1}^{N} \left(i\Omega_M \vec{q} + \vec{r} \right) \left(j\vec{q}^T \Omega_M + \vec{r}^T \right) \right) \Omega_M \\ &= \Omega_M \left(\sum_{i=1}^{N} \sum_{j=1}^{N} ij\Omega_M \vec{q}\vec{q}^T \Omega_M + i\Omega_M \vec{q}\vec{r}^T + j\vec{r}\vec{q}^T \Omega_M + \vec{r}\vec{r}^T \right) \Omega_M \\ &= \Omega_M \left(\frac{1}{4} N^2 (N+1)^2 \Omega_M \vec{q}\vec{q}^T \Omega_M + \frac{1}{2} N^2 (N+1) \Omega_M \vec{q}\vec{r}^T + \frac{1}{2} N^2 (N+1) \vec{r}\vec{q}^T \Omega_M + N^2 \vec{r}\vec{r}^T \right) \Omega_M \end{aligned} \tag{3.21}$$

Now, the bca^{-1} term is in equation 3.22.

$$bca^{-1} = \frac{N}{4} (N+1)^2 \Omega_M^2 \vec{q}\vec{q}^T + \frac{N}{2} (N+1) \Omega_M^2 \vec{q}\vec{r}^T \Omega_M^{-1} + \frac{N}{2} (N+1) \Omega_M \vec{r}\vec{q}^T + N\Omega_M \vec{r}\vec{r}^T \Omega_M^{-1} \tag{3.22}$$

Equations 3.23 to 3.26 compute individual portions of equation 3.22.

$$\Omega_M^2 \vec{q}\vec{q}^T = \begin{bmatrix} \alpha_1^2 & & & 0 \\ & \alpha_2^2 & & \\ & & \ddots & \\ 0 & & & \alpha_M^2 \end{bmatrix} \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & \cdots & 1 \end{bmatrix}_{M \times M}$$

$$= \begin{bmatrix} \alpha_1^2 & \alpha_1^2 & \cdots & \alpha_1^2 \\ \alpha_2^2 & \alpha_2^2 & \cdots & \alpha_2^2 \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_M^2 & \alpha_M^2 & \cdots & \alpha_M^2 \end{bmatrix}_{M\times M} \tag{3.23}$$

$$\begin{aligned} \Omega_M^2 \vec{q}\vec{r}^T \Omega_M^{-1} &= \begin{bmatrix} \alpha_1^2 & \alpha_1^2 & \cdots & \alpha_1^2 \\ \alpha_2^2 & \alpha_2^2 & \cdots & \alpha_2^2 \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_M^2 & \alpha_M^2 & \cdots & \alpha_M^2 \end{bmatrix}_{M\times M} \begin{bmatrix} \frac{1}{\alpha_1} & & & 0 \\ & \frac{1}{\alpha_2} & & \\ & & \ddots & \\ 0 & & & \frac{1}{\alpha_M} \end{bmatrix} \\ &= \begin{bmatrix} \alpha_1 & \frac{\alpha_1^2}{\alpha_2} & \cdots & \frac{\alpha_1^2}{\alpha_M} \\ \frac{\alpha_2^2}{\alpha_1} & \alpha_2 & \cdots & \frac{\alpha_2^2}{\alpha_M} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\alpha_M^2}{\alpha_1} & \frac{\alpha_M^2}{\alpha_2} & \cdots & \alpha_M \end{bmatrix}_{M\times M} \end{aligned} \tag{3.24}$$

$$\Omega_M \vec{r}\vec{q}^T = \begin{bmatrix} \alpha_1 & \alpha_1 & \cdots & \alpha_1 \\ \alpha_2 & \alpha_2 & \cdots & \alpha_2 \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_M & \alpha_M & \cdots & \alpha_M \end{bmatrix} \tag{3.25}$$

$$\Omega_M \vec{r}\vec{r}^T \Omega_M^{-1} = \begin{bmatrix} 1 & \frac{\alpha_1}{\alpha_2} & \cdots & \frac{\alpha_1}{\alpha_M} \\ \frac{\alpha_2}{\alpha_1} & 1 & \cdots & \frac{\alpha_2}{\alpha_M} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\alpha_M}{\alpha_1} & \frac{\alpha_M}{\alpha_2} & \cdots & 1 \end{bmatrix}_{M\times M} \tag{3.26}$$

Then bca^{-1} has the form in equation 3.27.

$$
\begin{aligned}
bca^{-1} &= \frac{q^2}{4}N(N+1)^2\begin{bmatrix} \alpha_1^2 & \alpha_1^2 & \cdots & \alpha_1^2 \\ \alpha_2^2 & \alpha_2^2 & \cdots & \alpha_2^2 \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_M^2 & \alpha_M^2 & \cdots & \alpha_M^2 \end{bmatrix} + \frac{qr}{2}N(N+1)\begin{bmatrix} \alpha_1 & \frac{\alpha_1^2}{\alpha_2} & \cdots & \frac{\alpha_1^2}{\alpha_M} \\ \frac{\alpha_2^2}{\alpha_1} & \alpha_2 & \cdots & \frac{\alpha_2^2}{\alpha_M} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\alpha_M^2}{\alpha_1} & \frac{\alpha_M^2}{\alpha_2} & \cdots & \alpha_M \end{bmatrix} \\
&\quad + \frac{qr}{2}N(N+1)\begin{bmatrix} \alpha_1 & \alpha_1 & \cdots & \alpha_1 \\ \alpha_2 & \alpha_2 & \cdots & \alpha_2 \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_M & \alpha_M & \cdots & \alpha_M \end{bmatrix} + Nr^2\begin{bmatrix} 1 & \frac{\alpha_1}{\alpha_2} & \cdots & \frac{\alpha_1}{\alpha_M} \\ \frac{\alpha_2}{\alpha_1} & 1 & \cdots & \frac{\alpha_2}{\alpha_M} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\alpha_M}{\alpha_1} & \frac{\alpha_M}{\alpha_2} & \cdots & 1 \end{bmatrix} \qquad (3.27)
\end{aligned}
$$

Equation gives the form of the covariance block for the cell initial SOCs without the substitution of bca^{-1},

$$
\begin{aligned}
\Sigma_S &= \sigma_v^2\left(a^{-1} + a^{-1}b\left(d - ca^{-1}b\right)^{-1}ca^{-1}\right) \qquad (3.28) \\
&= \frac{\sigma_v^2}{N}\left(\Omega_M^2\right)^{-1}\left(I_M + 12 \times \frac{bca^{-1}}{\vec{q}^T\Omega_M^2\vec{q}\left[N\left(N^2-1\right)\right]}\right), \qquad (3.29)
\end{aligned}
$$

where the numerator inside the parentheses is a scalar equal to $d - ca^{-1}b$. Since the variance for individual variables is the corresponding diagonal element of Σ_S, the resulting expression for variance of the m^{th} cell is shown in equation 3.30.

$$
\Sigma_S(m,m) = \frac{\sigma_v^2}{N\alpha_m^2}\left(1 + \frac{12}{N^2-1}\left[\frac{\frac{q^2\alpha_m^2}{4}(N+1)^2 + qr\alpha_m(N+1) + r^2}{q^2\sum_{i=1}^{M}\alpha_i^2}\right]\right) \qquad (3.30)
$$

3.4 Analysis of the Variance Expression

Equation 3.30 confirms a few intuitions about this estimation problem:

1. Variance decreases as more measurements become available.

2. Variance is large if the m^{th} cell's voltage vs SOC slope α_m is small. This makes sense

because it represents smaller output sensitivity.

3. Increasing voltage sensor variance increases estimate variance.

The expression also presents new analytical insights, some of which are previously only seen in the simulation results of the author's conference paper [24]. We begin by noting that for a cell with a given α value, assuming a fixed number of measurements N, variance cannot be smaller than $\sigma_v^2/(N\alpha_m^2)$. Next, we make the assumption that N is a relatively large number. This asymptotic analysis allows us to remove the linear and constant terms and focus on the quadratic terms inside the parenthesis [1]. This reduces the variance expression to equation 3.31.

$$\Sigma_S(m,m) \approx \frac{\sigma_v^2}{N\alpha_m^2}\left(1+\frac{3\alpha_m^2}{\sum_{i=1}^{M}\alpha_i^2}\right) \tag{3.31}$$

Equation 3.31 provides two insights similar to the hypotheses from the introduction:

1. Considering more than one cell in an SOC estimator will cause the estimate variance to decrease because of an increase in the $\sum_{i=1}^{M}\alpha_i^2$ term.

2. Considering multiple cells each with a small α_i will have the same effect on variance as considering one cell with a much larger α_i.

The above insights may be tested by plotting the right hand side of the expression in the parentheses in equation 3.30 for a string of cells with a small value of α and then adding to it a cell with a relatively larger value of α. Figure 3.1 shows such a plot that assumes that α_m is 0.05, there are M such cells, and there may or may not be an additional cell with $\alpha = 0.5$. This further supports the insights and also implies diminishing returns as a function of the number of low α cells.

The next section examines some parallels between the insights and some of the author's previous simulation work.

3.5 Comparison with Results of a Simplified Simulation

Both insights of the last section are consistent with the Monte Carlo results of the author's conference paper [24]. In the paper, an EKF estimates the SOC and unknown current sensor

[1] As an illustration, the quadratic term will dominate by two orders of magnitude for a single minute of measurements, three orders for two minutes, and four orders for one hour.

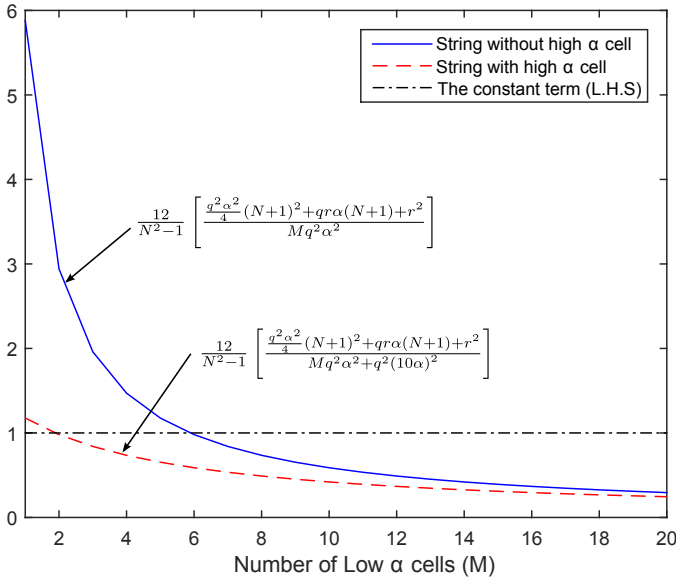

Figure 3.1. Comparison of the L.H.S and R.H.S terms in equation 3.30 as a function of string size and heterogeneity

bias for an entire string of cells. The EKF uses the nonlinear integrator model in equations 2.7 and 2.8 and the estimation takes place against input-output data generated through a one hour long simulation of the integrator model subjected to a zero-mean rectangular wave. The assumed model parameters are $C = 3600$ As and $r = 0.1\ \Omega$. The function $g(\cdot)$ in figure 3.2 was measured in house for a Li-Iron-Phosphate cell.

The paper presented and tested two hypotheses very similar to last section's insights but using Monte Carlo simulations.

Figure 3.3 taken from [24] shows results of a Monte Carlo simulation that tests the effect of increasing number of cells in a balanced string. All cells are assumed to have an initial SOC of 50% (i.e. they have small α).

As a larger number of cells from a string is considered, the SOC estimation error decreases by a factor of approximately four. The pattern is consistent with equation 3.31.

The second test in figure 3.4 shows results of a different Monte Carlo simulation involving only two cells. This time one cell's initial SOC is kept constant at 50% but the quantity is varied for a second cell, running a Monte Carlo simulation for each initial condition and saving the resulting statistics.

Note that with only one added cell, with a high enough value of α, the SOC estimation variance

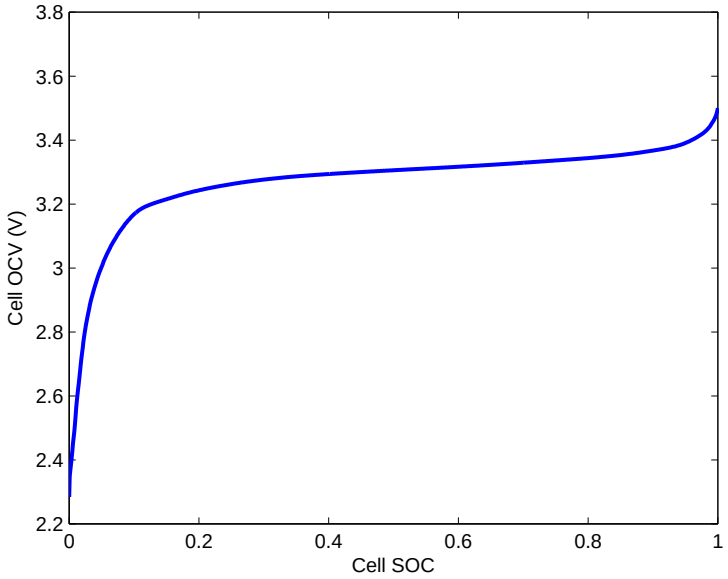

Figure 3.2. Open Circuit Voltage Response of a Lithium-Iron-Phosphate Battery

decreases by more than a factor of four.

Both the derivation in this chapter and the simulation of [24] use the simplest possible model for a battery. It is important to see if these results will be true in a real world scenario. The rest of the thesis tests these hypotheses using Monte Carlo simulation of the Autolion model to represent a real cell along with an EKF for SOC estimation using the M2RC which is similar to the models used in real battery packs.

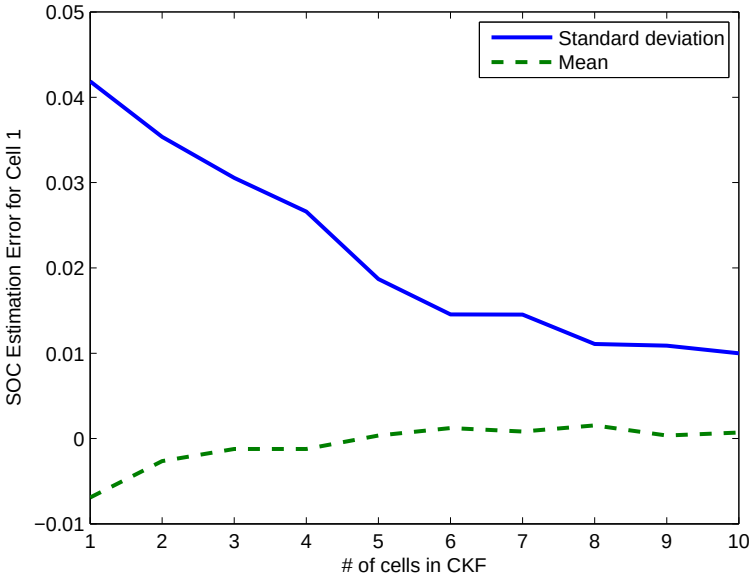

Figure 3.3. Results from [24] consistent with hypothesis 1

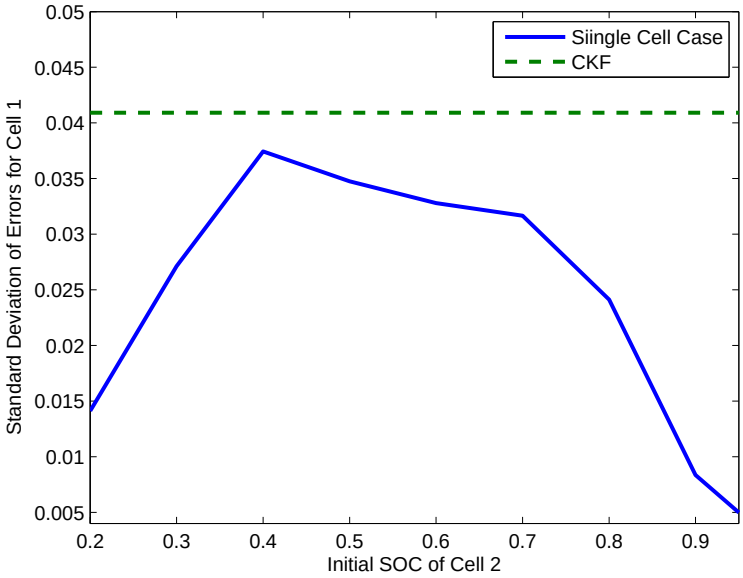

Figure 3.4. Results from [24] consistent with hypothesis 2

Chapter 4

Estimator Design for use in Monte Carlo Simulations

This thesis uses the standard Extended Kalman Filter (EKF) algorithm for SOC estimation. We select the EKF primarily because of its wide spread use in the battery estimation literature. The resulting estimator will be used in the next chapter to estimate SOC for a string of cells against input-output data from Autolion in a Monte Carlo simulation that tests the validity of the previous chapter's hypotheses.

Algorithm 1 provides a summary of the estimator equations adapted from Simon [30]. The rest of the chapter develops a discrete time model for use with the EKF for a single cell as well as a string and then presents the design of various elements associated with the EKF (initial conditions, parameters, etc.).

4.1 Discrete Time Cell Model Development

A discrete time state space representation of the cell model is necessary to apply the EKF. Equations 4.1 and 4.2 give the continuous time representation of the cell model in state space form.

$$\dot{\vec{x}} = A_C \vec{x} + B_C u \tag{4.1}$$

$$A_c = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -\frac{1}{\tau_1} & 0 \\ 0 & 0 & -\frac{1}{\tau_2} \end{bmatrix} \quad B_c = \begin{bmatrix} \frac{1}{C} \\ \frac{1}{C_1} \\ \frac{1}{C_2} \end{bmatrix} \tag{4.2}$$

We assume a time step of one second for all simulations and discretize the system using the zero order hold technique. Equations 4.3 through 4.6 give this quantized form of the model where $k = 0, 1, 2, 3, \cdots, N$ is time in seconds and the initial conditions are the same as the continuous time model.

$$\vec{x}_{k+1} = A\vec{x}_k + Bu_k \tag{4.3}$$

$$y_{k+1} = G\left(\vec{x}_{k+1}\right) + u_{k+1}r \tag{4.4}$$

$$A = \exp\left(A_C\right) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & e^{-\frac{1}{\tau_1}} & 0 \\ 0 & 0 & e^{-\frac{1}{\tau_2}} \end{bmatrix} \tag{4.5}$$

$$B = \left(\int_0^1 \exp\left(A_C\tau\right) d\tau\right) B_C = \begin{bmatrix} \frac{1}{C} \\ \frac{\left(1-e^{-\frac{1}{\tau_1}}\right)\tau_1}{C_1} \\ \frac{\left(1-e^{-\frac{1}{\tau_2}}\right)\tau_2}{C_2} \end{bmatrix} \tag{4.6}$$

Note that $G\left(\vec{x}_{k+1}\right) = g\left(x_{1,k+1} + x_{2,k+1} + x_{3,k+1}\right)$. For future reference we define $D\left(\vec{x}_{k+1}\right) = \frac{dg}{dx}|_{x_{1,k+1}+x_{2,k+1}+x_{3,k+1}}$.

4.2 String Model Equations

Consider a series string of n cells. The measurements available are the terminal voltage for each cell as well as the current applied to the string. The sensors corrupt the true values in the fashion shown in equations 4.7 and 4.8.

$$\vec{Y}_k = \vec{y}_k + \vec{v}_k = \begin{bmatrix} y_{1,k} \\ y_{2,k} \\ \vdots \\ y_{n,k} \end{bmatrix} + \begin{bmatrix} v_{1,k} \\ v_{2,k} \\ \vdots \\ v_{n,k} \end{bmatrix} \tag{4.7}$$

$$U_k = u_k + w_k + b \tag{4.8}$$

Where b is a constant current sensor bias and the noise elements $v_{j,k} \sim \mathcal{N}\left(0, \sigma_v^2\right)$ and $u_k \sim \mathcal{N}\left(0, \sigma_u^2\right)$ are mutually uncorrelated (i.e. if $j \neq l$, or $t \neq k$ then $E\left[v_{j,k}v_{l,t}\right] = 0$ and $E\left[v_{j,k}w_t\right] = 0$). These conditions form the fundamental assumptions for the Kalman filtering algorithm.

In order to jointly estimate the states for each cell in the string as well as the current bias it is necessary to construct a discrete time model for the entire string with an augmented bias state. Equations 4.9, 4.10, and 4.11 show this model.

$$\vec{X}_{k+1} = A_S \vec{X}_k + B_S U_k \tag{4.9}$$

$$\vec{X}_k = \begin{bmatrix} \vec{x}_{1,k} \\ \vec{x}_{2,k} \\ \vdots \\ \vec{x}_{n,k} \\ b \end{bmatrix} \tag{4.10}$$

$$A_S = \begin{bmatrix} A_1 & \cdots & 0 & -B_1 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & A_n & -B_n \\ 0 & \cdots & & 1 \end{bmatrix} \quad B_S = \begin{bmatrix} B_1 \\ \vdots \\ B_n \\ 0 \end{bmatrix} \tag{4.11}$$

4.3 Design of EKF

The EKF is a version of the Kalman filter for estimation of nonlinear dynamic systems. It works by linearizing the model around a best state estimate. The string model has linear dynamics but a nonlinear output. As such the only "extended" part of the EKF used in this paper concerns with the output equation. Algorithm 1 gives a summary of the EKF adapted from Simon [30].

The Q and R matrices are covariances of the current and voltage noise respectively and represent *a priori* knowledge about the sensor characteristics. For the string model these values reduce to $Q = \sigma_u^2$ and $R = \sigma_v^2 I_{n \times n}$. The matrix P represents an estimate of the state estimate covariance and it needs to be properly initialized.

Filter initialization represents the best knowledge about the system at time zero which comes from initial voltage and current measurements. The underlying assumption is that the cell has been at rest for a long time so for each cell x_2 and x_3 start at zero. It is also reasonable to assume that there is no current draw at time zero. As such it is possible to invert the function

$g(x)$ to discern an estimate of the initial SOC. Equation 4.12 shows the initial state vector for the i^{th} cell.

$$\vec{\hat{x}}_{i,0}^{+} = \begin{bmatrix} g^{-1}\left(Y_{i,0}\right) \\ 0 \\ 0 \end{bmatrix} \tag{4.12}$$

Similarly an initial estimate of the bias state comes from the current measurement at time zero (i.e. $\hat{b}_0^+ = U_0$). Owing to noise in the voltage and current sensors these estimates are uncertain. The matrix P_0^+ represents this uncertainty. Since the initial values of zero for x_2 and x_3 is assumed the initial uncertainty about them is practically nonexistent. A conservative estimate of the x_1 variance can be obtained by taking the following Taylor expansion:

$$\hat{x}_{1,0}^{+} = g^{-1}\left(y_0 + v_0\right) \approx g^{-1}\left(y_0\right) + \left.\frac{dg^{-1}}{dy}\right|_{Y_0} v_0 \tag{4.13}$$

The noise element v_0 is a realization of a zero mean Gaussian with variance σ_v^2. Equation 4.14 gives the initial covariance for the i^{th} cell.

$$P_{i,0}^{+} = \begin{bmatrix} \left(\left.\frac{dg^{-1}}{dy}\right|_{Y_0} \sigma_v\right)^2 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{4.14}$$

The initial covariance for $\hat{b}_0$ is the same as the covariance for the current measurement. Equation 4.15 combines this information with the cell state covariance matrices to give the initial covariance.

$$P_0^{+} = \begin{bmatrix} P_{1,0}^{+} & \cdots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & P_{n,0}^{+} & 0 \\ 0 & \cdots & 0 & \sigma_u^2 \end{bmatrix} \tag{4.15}$$

Given the above initial state and covariance estimates, the EKF algorithm proceeds as follows:

Step 1: Initialize filter state estimate and covariance, set $k = 1$

$$\begin{aligned} \hat{\vec{X}}_0^+ &= E\left[\vec{X}_0\right] \\ P_0^+ &= E\left[\left(\vec{X}_0 - \hat{\vec{X}}_0^+\right)\left(\vec{X}_0 - \hat{\vec{X}}_0^+\right)^T\right] \end{aligned}$$

Step 2: Propagate state and covariance matrix

$$\begin{aligned} \hat{\vec{X}}_k^- &= A\hat{\vec{X}}_{k-1}^+ + BU_{k-1} \\ P_k^- &= AP_{k-1}^+A^T + BQB^T \end{aligned}$$

Step 3: Create linearized output matrix

$$H_k = \begin{bmatrix} D\left(\vec{x}_{1,k}\right) & 0 & \cdots & 0 & -r \\ 0 & D\left(\vec{x}_{2,k}\right) & \cdots & 0 & -r \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & D\left(\vec{x}_{n,k}\right) & -r \end{bmatrix}$$

Step 4: Update state and covariance matrix

$$\begin{aligned} K_k &= P_k^- H_k^T\left(H_k P_k^- H_k^T + R\right)^{-1} \\ \hat{\vec{X}}_k^+ &= \hat{\vec{X}}_k^- + K_k\left[\vec{Y}_k - \vec{y}\left(\hat{\vec{X}}_k^-, U_k - \hat{b}\right)\right] \\ P_k^+ &= \left(I - K_k H_k\right)P_k^- \end{aligned}$$

Step 5: Increment k by 1 and go to Step 2.

Algorithm 1: Summary of EKF

This completes the development of the EKF for use in the Monte Carlo simulation. The next chapter presents the various test cases necessary to study the validity of the hypotheses presented in the previous chapter and their results.

Chapter 5

Monte Carlo Studies for Testing of Hypotheses and Variance Expression

This chapter concerns with numerical validation of the variance expression in various scenarios and with testing of the implied hypotheses. The current profile for these tests is derived from one hour of an FTP75 drive cycle on an EV100. All tests assume a voltage sensor with zero mean Guassian noise with a standard deviation of 5 mV.

5.1 Monte Carlo Validation of Variance Expression

The variance expression is for an initial value estimation problem and may be tested via Monte Carlo simulation of a least squares minimization problem of the form in equation 5.1.

$$\hat{x}_0 = \text{argmin}_{x_0} \|y_m - y_p\|_2^2 \tag{5.1}$$

where y_p is voltage predicted by a model and y_m is noisy truth which may be generated by the same model or a higher fidelity model. We do the Monte Carlo simulation for various situations.

1. Linear integrator model for both y_p and y_m.
2. Non-linear integrator model for both y_p and y_m.
3. Non-linear integrator model for y_p and non-linear third-order model for y_m.
4. Non-linear third-order model for both y_p and y_m.
5. Non-linear third-order model y_p and Autolion for y_m.

The first test is a direct validation of the expression and makes the same assumptions. The remaining tests deal with scenarios that break some of the assumptions in order to test the expression's predictive capabilities.

All of the tests assume no current sensing noise and the same current bias of 0.1 A. We test the estimate variance of a Monte Carlo simulation against the derived expression as a function of the number of cells with initial SOC near 50% (a $C50$) in a string that may or may not contain

a cell with initial SOC near 25% (a $C25$). We also track the estimate mean since the variance expression is not strictly guaranteed to hold for a biased estimate.

5.1.1 Linear Integrator Models Only

Assumptions used in constructing this test match the derivation exactly. The models for both data generation and estimation are linear. A $C50$ cell has a corresponding α of 0.0352 whereas a $C25$ cell has an α of 0.5516. Figure 5.1 shows the results of this study.

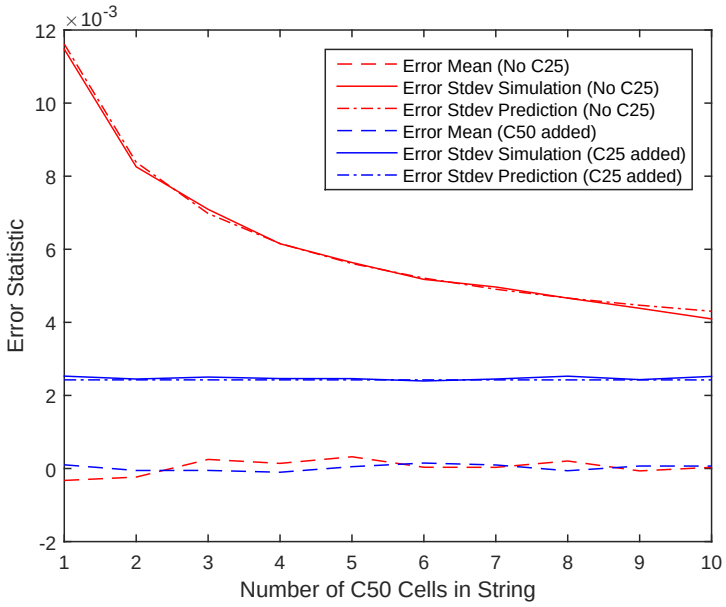

Figure 5.1. Variance Expression Validation: Linear Integrator Models only

The estimates appear to be unbiased and the computed standard deviation lies almost on top of the prediction. This is to be expected for this simulation and serves as a primary validation of the variance expression.

5.1.2 Non-Linear Integrator Models

This test keeps most of the model structure the same as the previous test but changes the output equation of the model to its more realistic non-linear form. The α values used in this simulation are the largest encountered during each test. That is 0.0470 for a $C50$ cell and 0.4653 for a $C25$ cell. Figure 5.2 shows the results of this study.

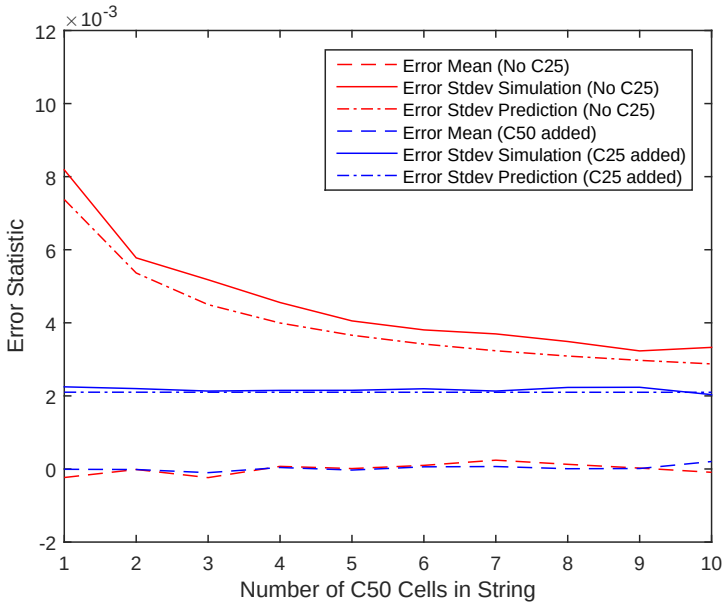

Figure 5.2. Variance Expression Validation: Non-Linear Integrator Models only

The simulation results are still unbiased which means that the CRLB should apply. The predicted variance is either always lower than its computed counterpart, or almost overlapping. This implies that even for non-linear integrator models the variance expression can be used as a predictive tool. The next section loosens some more assumptions and uses a completely different model for both simulation and estimation.

5.1.3 Non-Linear Third Order Models

The third order model is a better abstraction of a real battery than the integrator model. The α values for this simulation are the same as those for the last one. Figure 5.3 shows the results of this test.

These simulation results are unbiased as well. A trend very similar to the last section is seen in the results. The CRLB still holds even though the underlying models are now more complex. This motivates the next subsection which tests what happens when the model used for simulation is more complex than that used for estimation.

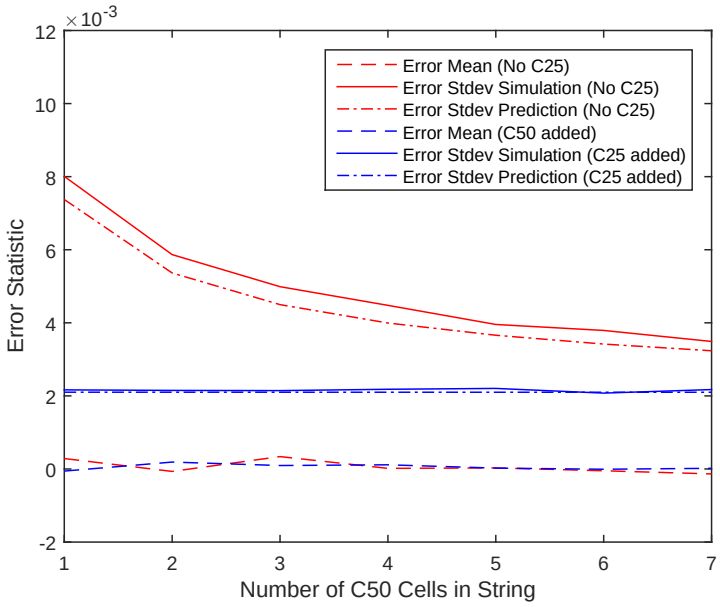

Figure 5.3. Variance Expression Validation: Non-Linear Third Order Models only

5.1.4 Non-Linear Third Order Model for Simulation, Non-Linear Integrator Model for Estimation

Using a higher fidelity model for simulation and a lower fidelity model for estimation is a better representation of reality since all models are ultimately simplifications of the cell's true dynamics. In this case an integrator model is used to predict the voltage of a third order model, both non-linear. Figure 5.4 shows the results of this study.

The estimate means are significantly biased which is likely a result of the model mismatch inherent in this situation. While the presence of a bias removes any guarantees for the variance expression, the CRLB results appear to hold regardless. It is not unheard of for the CRLB to approximate the estimate variance even for biased estimators. This was pointed out by Stoica who also constructed an example where the estimate variance was smaller than the CRLB for a biased estimator, illustrating the loss of the CRLB guarantee [32]. These results imply that the CRLB can be a guideline in the case of model mismatch but any guarantee that it will hold is lost.

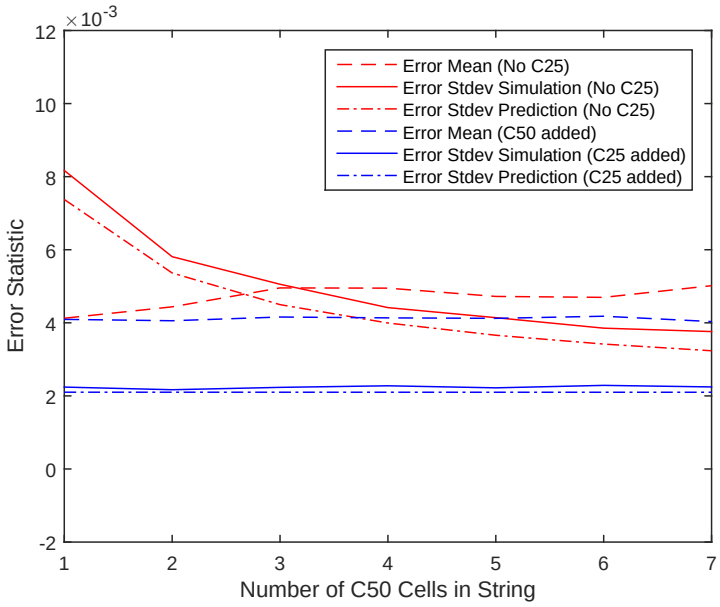

Figure 5.4. Variance Expression Validation: Non-Linear Third Order Model for Simulation, Non-Linear Integrator Model for Estimation

5.1.5 Physics-Based Model for Simulation, Non-Linear Third Order Model for Estimation

Finally we test the initial value estimation scenario most similar to reality. Autolion provides the voltage data from which a third order non-linear model estimates the initial values. We expect the estimates to be biased. Figure 5.5 shows the results of this study. Very large bias in the estimates is seen, especially for the string containing $C50$ cells only. The CRLB still appears to hold similar to the last subsection.

5.2 Testing Instantaneous SOC Estimation Using Extended Kalman Filtering

To evaluate the applicability of the hypotheses in Chapter 3 we run multiple Monte Carlo simulation studies with various configurations of cells in series, this time for instantaneous SOC estimation using an EKF. We ask four major questions:

1. How does constructing an EKF for i cells that are all at the same SOC affect estimator performance as i increases?

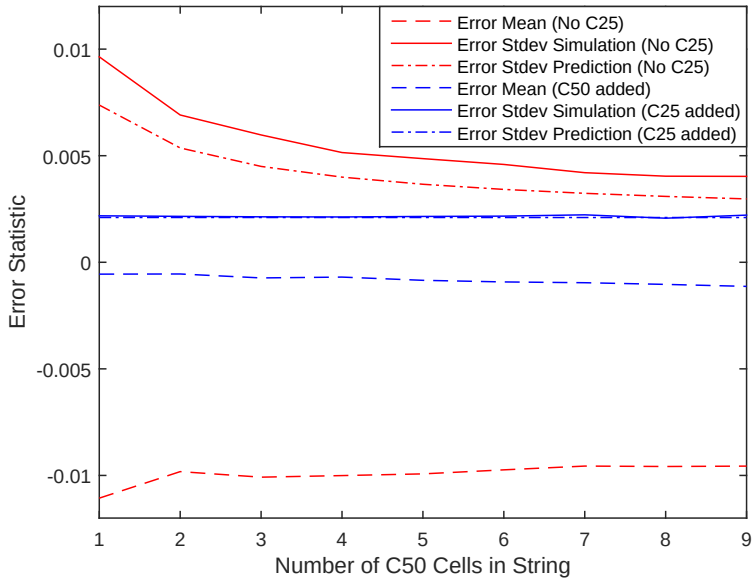

Figure 5.5. Variance Expression Validation: Autolion for Simulation, Non-Linear Third Order Model for Estimation

2. If the SOC of said string is in a flat region of the $g(x)$ curve, how does adding a cell in the non-flat region to this string and to the EKF affect estimator performance?

3. Does the value of current bias affect the estimator performance?

4. How does the value of the current noise affect the estimator performance?

The nominal noise and bias values are consistent with either lab testing of vehicle grade sensors or the literature on pack level SOC estimation [22].

Similar to the variance expression validation, all tests consider estimation accuracy for a cell in the flat region with a starting SOC of 50% (C50). Adding another cell with a starting SOC of 25% (C25) in a non-flat region helps answer question 2. Equation 5.2 shows the set of six cell configurations that we simulate with all combinations of current noise $\sigma_u \in \{0, 0.01, 0.06, 0.1\}$ A and bias $b \in \{0, 0.1, 0.5\}$ A; 72 tests in total.

$$\begin{aligned} string \in \{\{C50\}, \{C50 \times 5\}, \{C50 \times 10\}, \\ \{C50, C25\}, \{C50 \times 5, C25\}, \{C50 \times 10, C25\}\} \end{aligned} \tag{5.2}$$

5.2.1 The Benchmark Single Cell Case

The single cell case ($C50$) serves as a control study and sets reference error statistics for the multi-cell estimators. For the purposes of this paper we focus on the SOC estimation error for only cell 1 which is always a C50. Figure 5.6 shows, as an example, error trajectories from all 1000 runs superimposed on the same plot along with the ensemble mean and a 3σ envelope. The thick black line is the mean and the thick red lines are the envelope.

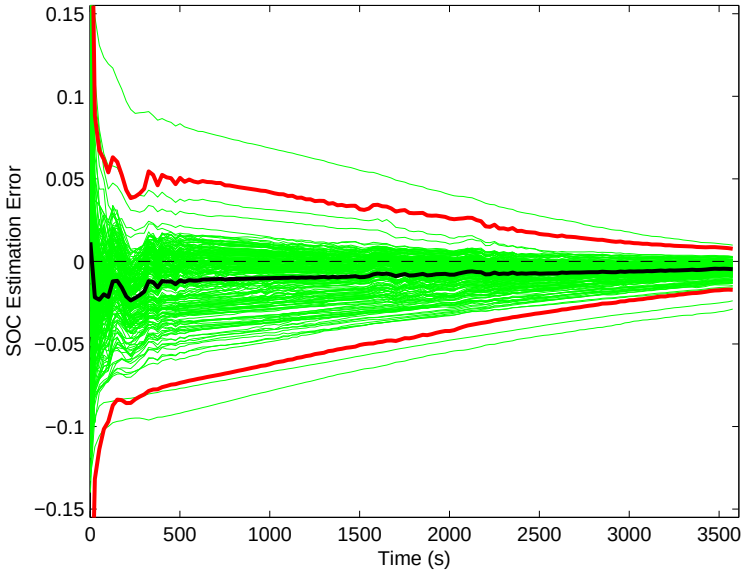

Figure 5.6. Ensemble behavior of error in a $C50$ case

The estimator experiences some transients at the start of the test but settles down by the end. As such it would make sense to base the performance of the estimators on error statistics at the last time step. Figure 5.7 shows the histogram of the ensemble error at the last time step for the test in Figure 5.6.

Since the error distribution appears to be unimodal and Gaussian like, we compare the performance of the estimators using only the first and second moment of this distribution. Specifically an estimator performs better if it produces smaller absolute mean and standard deviation values than the reference case and worse otherwise. The next section explores how the various test cases compare against the benchmark and one another.

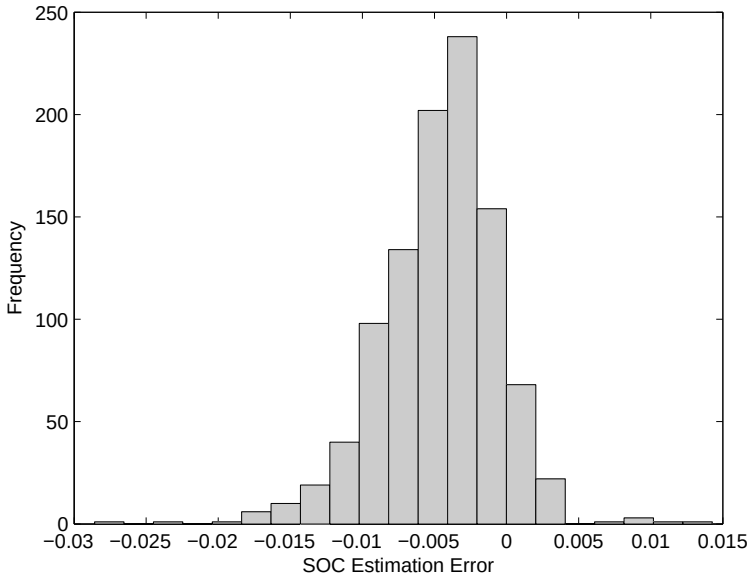

Figure 5.7. Histogram of final error in a $C50$ case

5.2.2 Multi-Cell Cases

For the multi-cell cases we generate the mean and standard deviation mesh plots for all of the cell configurations. Figure 5.8 shows said plot for standard deviation, 5.9 for mean. Black meshes correspond to $C50$, blue to $C50 \times 5$, and red to $C50 \times 10$. Their dashed counterparts represent the same number of $C50$ cells but with the added $C25$.

Note that Figure 5.8 shows that for high values of current noise, a multi-cell estimator configuration always decreases the error standard deviation. This observation echoes, to an extent, the analytical results of Chapter 3 as well as the simulation results in [24]. This lends more credence to the first hypothesis. The remaining sections of this chapter discuss the questions from the last chapter (which in turn are related to the hypotheses of Chapter 3).

5.2.3 Increasing Bias and Current Noise

According to the Monte Carlo results the performance metrics only weakly depend on the current bias value. The noise value, however, has a more obvious impact on the standard deviation of the single cell estimator and on the mean of all of the $C25$ free configurations. The performance of the $C25$ cases appears to be weakly related to the value of current noise. This implies that the

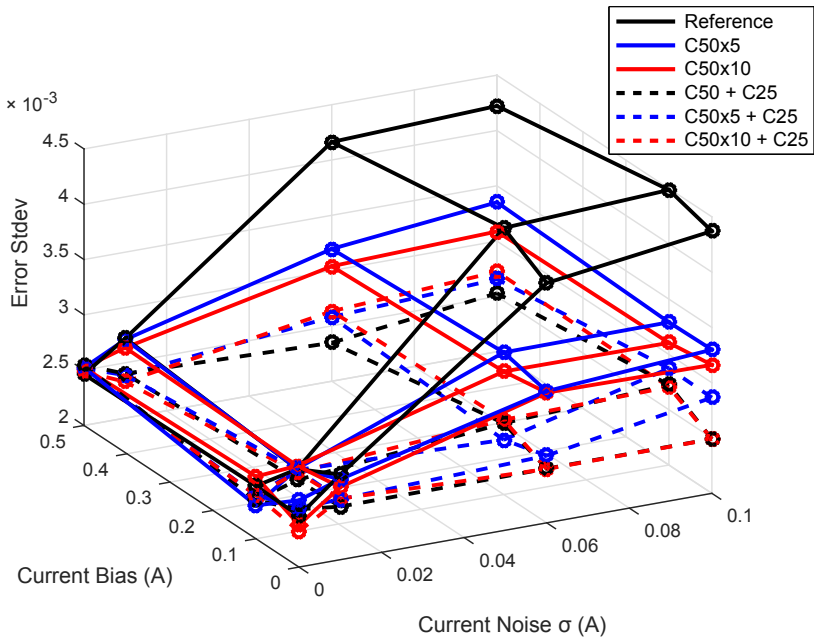

Figure 5.8. Comparison of Standard Deviation

addition of a $C25$ like cell into a current string could not only help improve estimation accuracy but also make the estimators more robust to measurement uncertainties.

Analysis similar to Chapter 2 (but with an added measurement uncertainty term) could be used to determine why addition of the $C25$ cell causes the estimator to be less sensitive to current noise. We leave this as potential future work.

5.2.4 Number of Homogeneous Cells

Increasing the number of $C50$ cells in an estimator improves performance from the standard deviation perspective (as predicted) but negatively impacts the mean. This could be caused by over-reliance on an uncertain model made worse by the weak output sensitivity. The pattern of improving standard deviation but worsening mean continues with the $C25$ cases. As such, in a string with an added $C25$ like cell, an estimator containing more than two cells would not be beneficial.

These results partially align with the predictions from the analysis in Chapter 2. The CRLB assumes a linear unbiased estimator which is not the case for the Monte Carlo study (the model is simple and uncertain, the estimation strategy based on a linear version of the model). As stated before, the CRLB provides a *lower* bound on the variance assuming a best case scenario of a

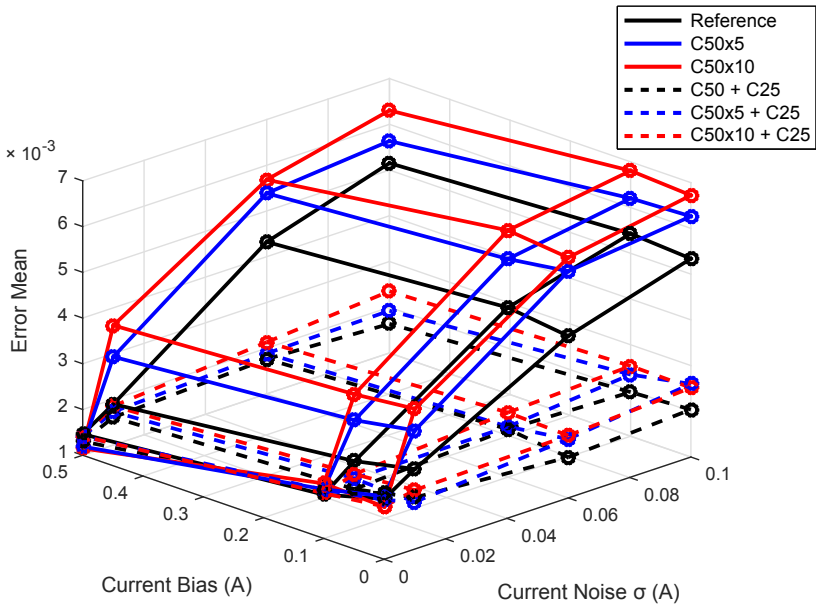

Figure 5.9. Comparison of Mean

linear unbiased estimator. The fact that a nonlinear estimator with biased results still follows the general trend predicted by the CRLB is simply a testament to the predictive power of the method. Further analysis of the EKF specifically will be necessary to determine what leads to worsening of the estimate mean and if it is possible to feasibly construct an unbiased estimator using this method.

5.2.5 Addition of $C25$

The addition of $C25$ always improves estimator performance; this assertion is evidenced by the distance between the solid and dashed meshes in Figures 5.8 and 5.9 and is consistent with Chapter 2's analysis. In a typical battery pack such a configuration would not be feasible as one of the BMS's goals is keeping the cells balanced. On the other hand, a reconfigurable battery pack with the ability to isolate cells in a series string could create these situations from time to time in order to get better estimation accuracy.

The benefits could also be achieved by mixing other cell chemistries with steeper rest voltage curves in a Lithium Iron Phosphate battery pack. This is true because the primary difference between $C50$ and $C25$ is the slope of the OCV curve and equation 3.31 suggests that the addition of a cell with a large slope will improve estimator performance. The introduction of a single such

cell into each string could provide the same benefits as adding a $C25$ while keeping the string relatively balanced.

Chapter 6

Conclusions

The purpose of this thesis is to study the problem of SOC and current sensor bias estimation for Lithium Ion battery strings. Specifically we study the effect of considering multiple cells in a string - with either homogeneous or heterogeneous initial conditions - on estimator performance. Evaluation of estimator accuracy comes from a Fisher Information based analysis of estimator variance using simple models of a string of cells as well as a monte carlo study of error statistics at the end of a one hour long test taken from the FTP-75 driving cycle. Results of the derivation and simulation lead to two major insights:

1. Considering multiple cells together from a balanced string of cells in an estimator may lead to an improvement in the estimator variance but may negatively impact the mean. This result is particularly true for large values of current noise.

2. Introducing an unbalanced cell in a series string such that the new addition is in a non-flat region of the voltage vs SOC curve can lead to to an improvement in both estimator variance and mean that outweighs that seen in the balanced case.

These conclusions imply a benefit to incorporating multiple cells from a series string into the same estimator formulation if permitted by computational ability. To the best of the author's knowledge this is the first time these claims have been made and substantiated using analytical means and simulation together.

For balanced strings of cell - which tends to be the norm in modern vehicle battery packs - the estimator designer would need to decide on the trade-off between error bias and variance. Should the increase in bias outweigh the variance, improvements a multi-cell estimator would no longer be a recommendation.

Assuming the potential for unbalanced strings (e.g. inside a reconfigurable pack architecture) it would be possible to control some cells in a string so that their SOCs reside in the non-flat region of the OCV-SOC curve. This can lead to even more improvement in estimation performance. A special type of battery pack is not necessary to achieve this effect since it is a function of the local sensitivity of a cell's voltage response on SOC. In contemporary pack design using cells from a

different chemistry - one with a less flat voltage vs SOC curve - would lead to similar gains in accuracy.

Bibliography

[1] EC Power AutoLion-1D - Battery Performance and Aging Management | EC Power.

[2] Davide Andrea. *Battery Management Systems for Large Lithium Ion Battery Packs*. Artech House, Boston, 1 edition edition, September 2010.

[3] Thorsten Baumhfer, Manuel Brhl, Susanne Rothgang, and Dirk Uwe Sauer. Production caused variation in capacity aging trend and correlation toinitial cell performance. *Journal of Power Sources*, 247:332–338, February 2014.

[4] Mohammad Charkhgard and Mohammad Farrokhi. State-of-charge estimation for lithium-ion batteries using neural networks and EKF. *Ieee Transactions on Industrial Electronics*, 57(12):4178–4187, December 2010.

[5] Sungwoo Cho, Hyeonseok Jeong, Chonghun Han, Shanshan Jin, Jae Hwan Lim, and Jeonkeun Oh. State-of-charge estimation for lithium-ion batteries under various operating conditions using an equivalent circuit model. *Computers & Chemical Engineering*, 41:1–9, June 2012.

[6] Haifeng Dai, Xuezhe Wei, Zechang Sun, Jiayuan Wang, and Weijun Gu. Online cell SOC estimation of li-ion battery packs using a dual time-scale kalman filtering for EV applications. *Applied Energy*, 95:227–237, July 2012.

[7] Domenico Di Domenico, Eric Prada, and Yann Creff. An adaptive strategy for li-ion battery internal state estimation. *Control Engineering Practice*, 21(12):1851–1859, December 2013.

[8] Donald Docimo, Mohammad Ghanaatpishe, and Hosam K. Fathy. Development and experimental parameterization of a physics-based second-order lithium-ion battery model. page V001T19A003. ASME, October 2014.

[9] Marc Doyle, TF Fuller, and John Newman. Modeling of Galvanostatic Charge and Discharge of the Lithium/Polymer/Insertion Cell. *Journal of the Electrochemical Society*, 140(6):1526–1533, 1993.

[10] Huazhen Fang, Yebin Wang, Zafer Sahinoglu, Toshihiro Wada, and Satoshi Hara. State of charge estimation for lithium-ion batteries: An adaptive approach. *Control Engineering Practice*, 25:45–54, April 2014.

[11] Terry Hansen and Chia-Jiu Wang. Support vector based battery state of charge estimator. *Journal of Power Sources*, 141(2):351–358, March 2005.

[12] Hongwen He, Rui Xiong, Xiaowei Zhang, Fengchun Sun, and JinXin 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*, 60(4):1461–1469, May 2011.

[13] Yao He, XingTao Liu, ChenBin Zhang, and ZongHai Chen. A new model for state-of-charge (SOC) estimation for high-power li-ion batteries. *Applied Energy*, 101:808–814, January 2013.

[14] Xiaosong Hu, Shengbo Li, and Huei Peng. A comparative study of equivalent circuit models for Li-ion batteries. *Journal of Power Sources*, 198:359–367, January 2012.

[15] Xinfan Lin, Patricia Laskowsky, Yonghua Li, and R Dyche Anderson. State of Charge Estimation Error due to Parameter Mismatchin a Generalized Explicit Lithium Ion Battery Model. In *ASME 2011 Dynamics Systems and Control Conference*, pages 1–8, Arlington, 2011.

[16] Xingtao Liu, Zonghai Chen, Chenbin Zhang, and Ji Wu. A novel temperature-compensated model for power li-ion batteries with dual-particle-filter state of charge estimation. *Applied Energy*, 123:263–272, June 2014.

[17] Kong Soon Ng, Chin-Sien Moo, Yi-Ping Chen, and Yao-Ching Hsieh. Enhanced coulomb counting method for estimating state-of-charge and state-of-health of lithium-ion batteries. *Applied Energy*, 86(9):1506–1511, September 2009.

[18] J. P. Norton. *An Introduction to Identification*. Dover Publications, Mineola, N.Y, April 2009.

[19] Lei Pei, Chunbo Zhu, Tiansi Wang, Rengui Lu, and C. C. Chan. Online peak power prediction based on a parameter and state estimator for lithium-ion batteries in electric vehicles. *Energy*, 66:766–778, March 2014.

[20] Sabine Piller, Marion Perrin, and Andreas Jossen. Methods for state-of-charge determination and their applications. *Journal of Power Sources*, 96(1):113–120, June 2001.

[21] G. L. Plett. Extended kalman filtering for battery management systems of LiPB-based HEV battery packs - part 2. modeling and identification. *Journal of Power Sources*, 134(2):262–276, August 2004.

[22] Gregory Plett. Efficient battery pack state estimation using bar-delta filtering. *EVS24*, 2009.

[23] Michael A. Roscher, Oliver S. Bohlen, and Dirk Uwe Sauer. Reliable state estimation of multicell lithium-ion battery systems. *Ieee Transactions on Energy Conversion*, 26(3):737–743, September 2011.

[24] Jariullah Safi, Michael Beeney, Michelle Kehs, Joel Anstrom, Sean Brennan, and Hosam Fathy. Improving SOC accuracy using collective estimation for lithium ion battery cells in series. In *American Control Conference (ACC), 2014*, pages 254–259. IEEE, 2014.

[25] Shriram Santhanagopalan and Ralph E. White. Online estimation of the state of charge of a lithium ion cell. *Journal of Power Sources*, 161(2):1346–1355, October 2006.

[26] Shriram Santhanagopalan and Ralph E. White. Online estimation of the state of charge of a lithium ion cell. *ECS Transactions*, 3(27):191–208, September 2007.

[27] Shriram Santhanagopalan and Ralph E. White. State of charge estimation using an unscented filter for high power lithium ion cells. *International Journal of Energy Research*, 34(2):152–163, February 2010.

[28] Simon Schwunk, Nils Armbruster, Sebastian Straub, Johannes Kehl, and Matthias Vetter. Particle filter for state of charge and state of health estimation for lithium-iron phosphate batteries. *Journal of Power Sources*, 239:705–710, October 2013.

[29] A. Sharma and H.K. Fathy. Fisher identifiability analysis for a periodically-excited equivalent-circuit lithium-ion battery model. In *American Control Conference (ACC), 2014*, pages 274–280, June 2014.

[30] Dan Simon. *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*. Wiley-Interscience, Hoboken, N.J, 1 edition edition, June 2006.

[31] Kandler A. Smith, Christopher D. Rahn, and Chao-Yang Wang. Model-based electrochemical estimation and constraint management for pulse operation of lithium ion batteries. *Ieee Transactions on Control Systems Technology*, 18(3):654–663, May 2010.

[32] Petre Stoica and Randolph L Moses. On biased estimators and the unbiased cramr-rao lower bound. *Signal Processing*, 21(4):349 – 350, 1990.

[33] Fengchun Sun and Rui Xiong. A novel dual-scale cell state-of-charge estimation approach for series-connected battery pack used in electric vehicles. *Journal of Power Sources*, 274:582–594, January 2015.

[34] J. Wang, M. W. Verbrugge, and P. Liu. Composite titanate-graphite negative electrode for improved state-of-charge estimation of lithium-ion batteries. *Journal of The Electrochemical Society*, 157(2):A185–A189, February 2010.

[35] John S. Wang, Elena Sherman, Mark Verbrugge, and Ping Liu. Composite electrodes of disordered carbon and graphite for improved battery state estimation with minimal performance penalty. *Journal of Power Sources*, 196(22):9648–9653, November 2011.

[36] Shuoqin Wang, Mark Verbrugge, John S. Wang, and Ping Liu. Multi-parameter battery state estimator based on the adaptive and direct solution of the governing differential equations. *Journal of Power Sources*, 196(20):8735–8741, October 2011.

[37] Rui Xiong, Fengchun Sun, Xianzhi Gong, and Hongwen He. Adaptive state of charge estimator for lithium-ion cells series battery pack in electric vehicles. *Journal of Power Sources*, 242:699–713, November 2013.

[38] Zhongyue Zou, Jun Xu, Chris Mi, Binggang Cao, and Zheng Chen. Evaluation of model based state of charge estimation methods for lithium-ion batteries. *Energies*, 7(8):5065–5082, August 2014.

Appendix A

Tabulated Monte Carlo Simulation Results

This appendix presents the results of Figures 5.8 and 5.9 in table form for the various test cases as a function of current noise standard deviation σ_{current} and current noise bias b in amperes.

A.1 Data for Error Mean

$b\backslash\sigma_{\text{current}}$	0	0.01	0.06	0.1
0	0.0023	0.0027	0.0045	0.0053
0.1	0.0020	0.0025	0.0047	0.0054
0.5	0.0015	0.0019	0.0043	0.0052

Table A.1. Reference (C50x1) Case, Error Mean

$b\backslash\sigma_{\text{current}}$	0	0.01	0.06	0.1
0	0.0024	0.0036	0.006	0.0063
0.1	0.0021	0.0034	0.0058	0.0062
0.5	0.0012	0.0029	0.0054	0.0056

Table A.2. C50x5 Case, Error Mean

$b\backslash\sigma_{\text{current}}$	0	0.01	0.06	0.1
0	0.0021	0.0041	0.0063	0.0067
0.1	0.0022	0.0039	0.0064	0.0068
0.5	0.0011	0.0036	0.0057	0.0063

Table A.3. C50x10 Case, Error Mean

$b\backslash\sigma_{\text{current}}$	0	0.01	0.06	0.1
0	0.0024	0.0021	0.0019	0.0020
0.1	0.0021	0.0019	0.0020	0.0020
0.5	0.0013	0.0016	0.0018	0.0016

Table A.4. C50 + C25 Case, Error Mean

$b\backslash\sigma_{\text{current}}$	0	0.01	0.06	0.1
0	0.0023	0.002	0.0023	0.0026
0.1	0.0021	0.0021	0.0021	0.0023
0.5	0.0014	0.0017	0.0019	0.0019

Table A.5. C50x5 + C25 Case, Error Mean

$b\backslash\sigma_{\text{current}}$	0	0.01	0.06	0.1
0	0.0022	0.0023	0.0024	0.0025
0.1	0.0020	0.0022	0.0024	0.0025
0.5	0.0014	0.0019	0.0021	0.0023

Table A.6. C50x10, Error Mean

A.2 Data for Error Standard Deviation

$b\backslash\sigma_{\text{current}}$	0	0.01	0.06	0.1
0	0.0025	0.0028	0.0042	0.0044
0.1	0.0025	0.0026	0.0044	0.0045
0.5	0.0025	0.0027	0.0042	0.0042

Table A.7. Reference (C50x1) Case, Standard Deviation

$b\backslash\sigma_{\text{current}}$	0	0.01	0.06	0.1
0	0.0026	0.0027	0.0032	0.0033
0.1	0.0023	0.0026	0.0033	0.0033
0.5	0.0025	0.0027	0.0032	0.0034

Table A.8. C50x5 Case, Standard Deviation

$b\backslash\sigma_{\text{current}}$	0	0.01	0.06	0.1
0	0.0024	0.0027	0.0032	0.0032
0.1	0.0026	0.0026	0.0031	0.0031
0.5	0.0025	0.0026	0.0030	0.0031

Table A.9. C50x10 Case, Standard Deviation

$b \backslash \sigma_{\text{current}}$	0	0.01	0.06	0.1
0	0.0025	0.0025	0.0025	0.0025
0.1	0.0023	0.0025	0.0026	0.0027
0.5	0.0025	0.0024	0.0023	0.0025

Table A.10. C50 + C25 Case, Standard Deviation

$b \backslash \sigma_{\text{current}}$	0	0.01	0.06	0.1
0	0.0025	0.0025	0.0026	0.0029
0.1	0.0024	0.0026	0.0025	0.0029
0.5	0.0025	0.0024	0.0026	0.0027

Table A.11. C50x5 + C25 Case, Standard Deviation

$b \backslash \sigma_{\text{current}}$	0	0.01	0.06	0.1
0	0.0023	0.0026	0.0025	0.0025
0.1	0.0024	0.0025	0.0027	0.0027
0.5	0.0025	0.0023	0.0026	0.0027

Table A.12. C50x10, Standard Deviation