|  |
| --- |
| Extended Kalman Filtering of State and Sensor Bias Estimation of a Li-Ion Battery Model |
| MAE 298 – Estimation Theory Final Project |
| Felipe Valdez  Jonathan Dorsey |

|  |
| --- |
| Abstract |
| *The increasing demand for electric vehicles (EVs) has led to technological advancements in the field of battery technology. State of charge (SOC) estimation is a vital function of the battery management system - the heart of electric vehicles, and Kalman filtering is a common method for SOC estimation. Due to the non-uniformities in tuning and testing scenarios, quantifying performance of SOC estimation algorithms is difficult. In this work, an SOC estimation algorithm is developed, Extended Kalman Filter (EKF), and tested for a variety of scenarios like adding sensor noise and bias to terminal voltage and current, and varying state and parameter initializations. In addition, a Dual EKF is implemented to estimate the voltage and sensor bias and compared against the State EKF for robustness to sensor bias/noise.* |

Table of Contents

**1. Introduction & Literature Review** **2**

1.1 Motivation 2

1.2 Related Work3

1.2.1 Battery Modeling3

1.2.2 Estimation of Algorithms3

1.3 Objectives3

**2. System Modeling & Analysis 4**

2.1 Overview of Li-Ion Battery5

2.1.1 State of Charge5

2.1.2 Open Circuit Voltage5

2.2 Electrical Equivalent Circuit Model5

2.1.1 Continuous Time Model5

2.1.2 Discrete Time Model5

2.3 Sensor Bias Modeling5

2.3.1 Current Sensor Bias5

2.3.2 Voltage Sensor Bias5

2.3.3 Observability Analysis5

**3. Algorithms & Implementation 4**

3.1 Linear Kalman Filter5

3.2 Extended Kalman Filter5

3.3 State & Parametric Estimation5

3.3.1 Dual EKF Estimation5

**4. Results & Discussion 4**

4.1 The Setup5

4.1.1 Simulation Setup5

4.1.2 Performance Indices5

4.2 Simulation Results5

4.2.1 State KF vs EKF5

4.2.2 State EKF vs Dual EKF 5

4.2.3 Sensor Bias Estimation5

**5. Future Work4**

# 1. Introduction & Literature Review

Introduce your project and briefly review the sources you used for this paper. This is expected to be 1-2 papers at most. Cite references in the text using IEEE style [1].

## Motivation

## Related Work

## Battery Modeling

## Estimation Algorithms

## Objectives

# 2. System Modeling & Analysis

## Overview of Li-Ion Battery

In any modeling methodology, the first step is to understand the actually physics, mechanisms and governing equations (if available). The focus of this paper being the estimation of Lithium Ion battery, it makes sense to first understand the basic fundamental quantities of interest associated with batteries in general and Li-Ion batteries specifically.

When it comes to Li-Ion batteries, there are predominately two quantities which are of interest to researches. These are the “State of Charge” and the “State of Health” of the battery. While both are being heavily researched, the State of Charge of a system or SOC, is the predominate quantity

## State of Charge

As mentioned above, one of the most important parameters of a battery is the State of Charge or SOC. The SOC of a battery effectively provides a measure of the batteries actually capacity available to the device or end user. This is an important parameter to know since the safety of many batteries, such as the Li-Ion batteries used in this paper, have the potential to be extremely dangerous and even explode or cause fires.

Unfortunately, the SOC of a battery is not a directly measurable quantity and therefore must be estimated in order to make available for application in control of battery management systems. In order to overcome the drawback, this papers presents the Extended Kalman Filter as the estimation technique of choice to reliably and accurately predict the SOC of the battery of interest.

## Open Circuit Voltage

One of the key modeling tools which is employed in this paper is the relationship between the SOC and the Open Circuit Voltage (OCV) of a battery. It has been experimentally shown that is for Li-Ion batteries, the OCV, can be computed as a function of the batteries SOC. While determining the relationship between these two quantities requires very precise and well executed experimental measurements.

For the purposes of this paper, the experimental relationship between these two quantities are assumed to be given. However, even given this data, the OCV/SOC relationship is typically nonlinear and normally requires either linearization-based estimation schemes (such as Kalman Filter) or nonlinear approximation such as (Extended Kalman Filter).

## Electrical Equivalent Circuit Model

Regardless of the techniques used to estimate the SOC, the OCV is a critical quantity in that it allows researchers to model batteries in terms of electrical circuits, and appl

One of the predominate issues with controlling or estimating battery parameters from first principle models is the required complexity of the fundamental dynamics and mechanisms of a battery. For example, the first principle model of a Li-Ion battery is modeled using partial differential equations (PDEs). Needless to say, the complexity of PDE models are far from practically applicable straight from derivation and often require extensive computational resources to solve numerically.

To bypass this problem, it is desired to use simplified low order dynamic models that are numerically tractable for the intended application. This leads to the use of “Equivalent Circuit Models,” or EMCs. The benefit of EMCs is their inherent ease of derivation and application which becomes apparent in commercial uses where computing overhead is extremely limited, for cost considerations.



(soc)

This paper will use the “Dual Polarity” equivalent circuit model as it is not only one of the most popular LI-Ion battery models in commercial use today but also the relative ease of reframing the model into an Extended Kalman Filter (EKF).

The circuit schematic for the DP model is shown above. Notice that the terminal voltage of the battery (U\_L) is easily shown to be related to the dynamics of the open circuit voltage (U\_oc), the series resistance (R0) and the two resistor-capacitor circuits. By application of basic circuit rules (KCL and KVL), the dynamics of the system can easily be derived with only

## Continuous Time Model

As mentioned previously, by using standard circuit analysis, we can extract the dynamic behavior of the system.

By applying KVL around the complete loop of the circuit, we get the following expression for the terminal voltage of the circuit as a function of the internal elements of the circuits.

By applying KCL to both RC branches we derive the following equations…

By including the expression for SOC with the equations defined above, the continuous time state space model can be written as…

A noticeable feature of this state space is the linear behavior of the state equations and the nonlinear behavior of the output equations. Therefore, the system is inherently nonlinear, indicating that estimating the SOC for the nonlinear model would most likely require at the very least an Extended Kalman Filter (EKF) or even more advanced methods use as the Unscented Kalman Filter (UKF).

## Discrete Time Model

While the continuous time model is an important start in the process of estimating the system, in todays computer age, it is significantly easier to implement discrete time models on modern computers which inherently are limited to finite numerical representations of numbers.

To transform the continuous time model into a discrete time state space, the closed form discretization formulas (shown below) were applied the appropriate matrices and vectors of the continuous time model to produce the following discrete state and output equations.

## Sensor Bias Modeling

When setting up an estimation problem, it is important to incorporate physical phenomena that exist under real world conditions whenever possible. One such condition that is common is suboptimal sensor measurements being used as inputs into estimators. The predominate means through which this is exemplified is the presence of a measurement bias or offset. Typically, the bias of a sensor is calibrated for or at least guaranteed accurate within some specified tolerance. Since this bias will always be present and may change with time, it is useful to estimate the value of these biases.

In this paper, a Dual Extended Kalman Filter is implemented to estimate the terminal voltage and current sensor bias along with the state of charge using an augmented state-space model involving the bias. Bias is modeled as random-walk where, it is assumed to be essentially constant but is capable of varying slowly over time, driven by some process modeled using a small fictitious noise. Both voltage and current sensor bias are considered independently in this paper.

In order to generate data that exhibits a bias, the current and voltage biases were included in either the input current data or the terminal voltage measurement respectively, as constant offsets. This provides known biases which facilitate initial validation of the bias estimation scheme as well as provide some small means of tuning the estimator for future testing.

## Current Sensor Bias

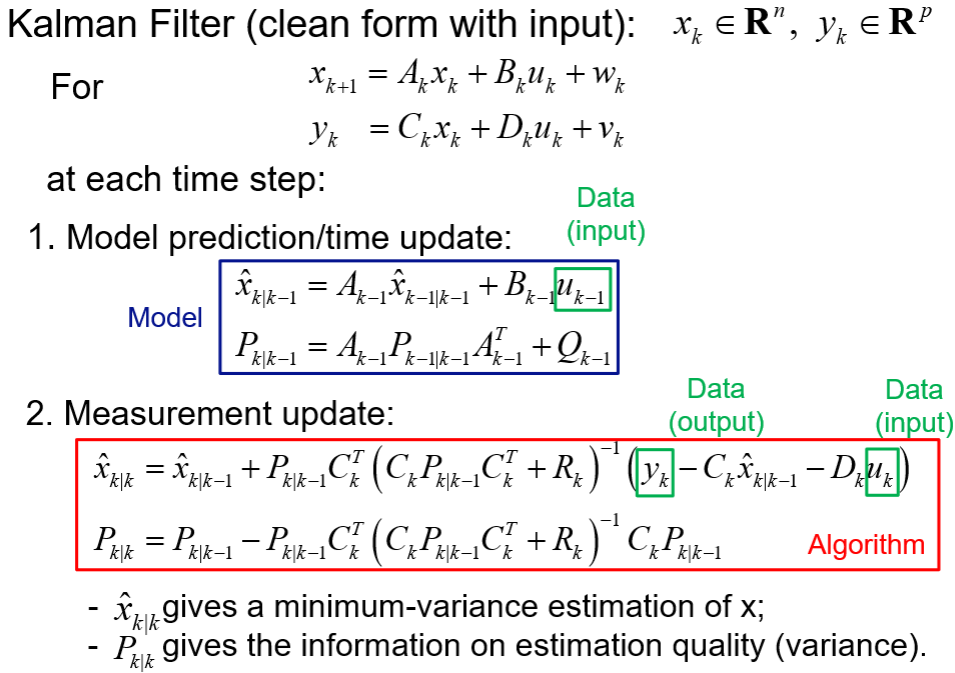
## Voltage Sensor Bias

## Observability Analysis

# 3. Algorithms & Implementation

 do we need these plots for Lin’s lookup table data that we are using?

## Linear Kalman Filter



## Extended Kalman Filter

## State & Parametric Estimation

## Dual EKF



# 4. Results & Discussion

## The Setup

## Simulation Setup

The overall simulation setup consists of the estimation algorithms KF, EKF and DEKF used with the battery model explained previously. For this study, MATLAB is used to simulate the measurements required for the Kalman algorithms….

A typical BMS is equipped with current, voltage and temperature sensors which have limited accuracy due to intrinsic measurement noise and bias. In this work, we only considered current and voltage measurement bias. To test estimation algorithms under different sensor properties, noise and bias are added to both current as well as terminal voltage signals. The noise added is Gaussian with zero-mean and a standard deviation of 1% of the corresponding signal’s maximum value. The bias level is set as 20mV for Voltage and 12.5mA for current.

## Performance Indices

Root mean square error: RMSE is the square root of mean of square of all errors. It

is calculated using the actual and estimated values, and is computed for SOC as well

as terminal voltage. It denotes the estimation accuracy.

Infinity Norm of SOC Error: It gives the worse-case measure of the SOC error and is

given by where n is the number of samples over the whole simulation time and N is the length of the drive cycle (the sampling time is 0.1s).

Variance of SOC Error: It refers to the average variance of SOC error over whole simulation

time. Variance measures the estimate’s uncertainty and is denoted by . With every new measurement, the Kalman filter aims to reduce uncertainty and hence, the variance ideally decreases and remains constant at steady-state.

## Simulation Results

## Model Validation

A model validation was conducted to verify the derived 2nd Order Equivalent Circuit Model (ECM). This verification was performed by generating data of actual SOC and terminal voltage (with process noise) from a “true” model of a 3rd Order ECM given time and battery current data as inputs. The noise added is Gaussian with zero-mean and a standard deviation of .1% of the corresponding signal’s maximum value. The simulated “true” data was then used on a derived 3rd order circuit battery model with an EKF implementation for validation. The results are shown below.

1. b.

c. d.

**Figure X**. blah blah blah

After concluding that the derived 3rd order ECM’s SOC closely matched the “true” SOC data, a 2nd order ECM was derived and used as a framework to design an Extended Kalman Filter and Dual EKF to estimate the SOC and Voltage/Current Bias as it will be explained in the next sections.

## State KF vs EKF – SOC Estimation

A linear Kalman Filter and Extended Kalman Filter was implemented on a nonlinear equivalent circuit battery model to estimate its state of charge. Using the nonlinear data on a Linear KF gives inaccurate results and it can be seen that EKF performs a lot better. This is due to the fact that Linear KF does not consider nonlinearities of the system coming from the open circuit voltage dependence on SOC. On the other hand, EKF considers the nonlinearities of the battery model by linearization of the nonlinear model using first order Taylor series about an operating point, this in turn significantly improves the SOC estimation as shown below.

## State EKF vs Dual EKF -SOC Estimation in the Presence of Sensor Bias

The performance of State and Dual EKF in estimating SOC was compared. A terminal voltage bias was added to the “true” model and the generated “true” data was used by the EKF and DEKF algorithms for SOC estimation. It can be noticed that DEKF performs better in estimating SOC when compared to EKF with the presence of voltage bias. Dual EKF has approximately 2% improvement in SOC estimation when comparing their root mean square values as seen in table 1.



Table 1 – State EKF vs DEKF Indices with Sensor Voltage Bias

|  |  |  |
| --- | --- | --- |
| **Index** | **State EKF** | **Dual EKF** |
|  | 0.0280 (2.80%) | 0.0116 (1.16%) |
|  | 1.0005 | 1.0003 |
|  | 0.0102 (1.02%) | 0.0102 (1.02%) |
|  | 8.53x10-6 | 8.74x10-6 |



Same as previously, a current sensor bias of 25mA was added to the “true” model and its generated data was used by both algorithms to estimate the state of charge. The figures below show that both EKF and DEKF perform well under a constant current bias. Table 2 show the performance indices which indicate almost identical performance for both.



Table 1 – State EKF vs DEKF Indices with Sensor Current Bias

|  |  |  |
| --- | --- | --- |
| **Index** | **State EKF** | **Dual EKF** |
|  | 0.0051 (0.51%) | 0.0051 (0.51%) |
|  | 0.9842 | 0.9842 |
|  | 0.0012 (0.12%) | 0.0012 (0.12%) |
|  | 1.084x10-5 | 1.0813x10-5 |

## EKF Parameter Variation

A common challenge encountered when implementing some estimation scheme like EKF is the effect of parameter mismatch, or physical variability of a specific parameter. Often, the solution to this is to apply principles of system identification through physical or simulated testing. However, this method is often time consuming and must be performed under precise testing conditions to output the most accurate parameter value possible. Additionally, operational use and wear can lead to parameters to evolve and change with continued service. To address this concern, the robustness of the estimation methodology being implemented should be reviewed to ensure that estimation failure will not occur due to minor parametric variability or system/model parameter mismatch.



Notice in the figures (???????) shown above, how the plots differ from the actual and the tuned EKF estimators. In plot(???), a series of SOC estimation trials were performed that varied each parameter by +-5% and would record the parameters that produced the maximum RMS SOC Error.

After performing a few further trials, it became clear that the predominate parameter responsible for the loss of tracking between the tuned EKF and EKF with mismatched parameters was the series resistance R0.

The reason for this behavior is hypothesized to be the large sensitivity of ohmic voltage drop in the DP Model as a function of R0 and the large input current. To validate the effect the series resistance has on the EKF performance, a series of sequential tests were performed that vary R0 by ± 20% to observe the estimated response.

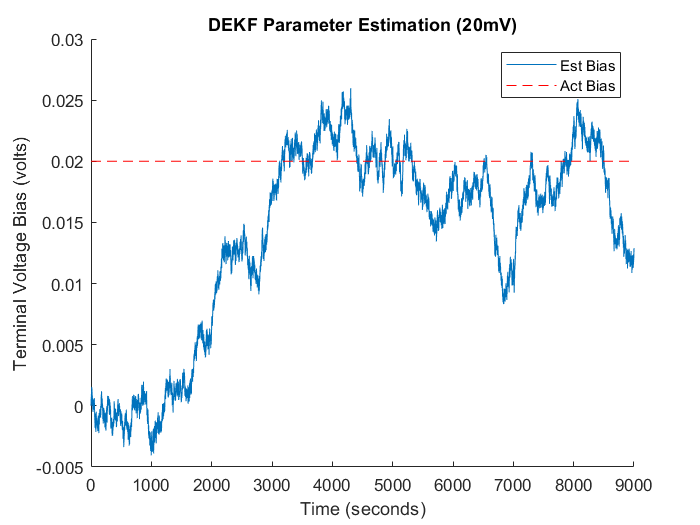
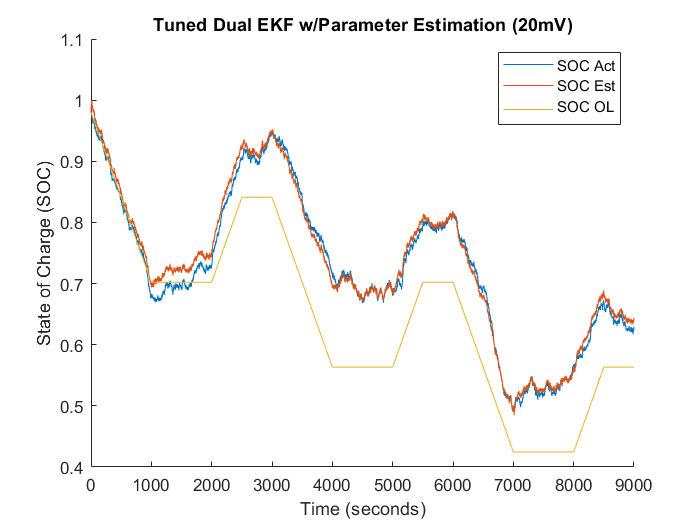
As shown in plot (????) the variability of R0 has a demonstrable effect on the non-zero current input behavior of the SOC estimator. The physical intuition behind this behavior would suggest the high input current through R0, creates an artificially voltage drop which acts like an output bias skewing the measurement update routine of the EKF algorithm.

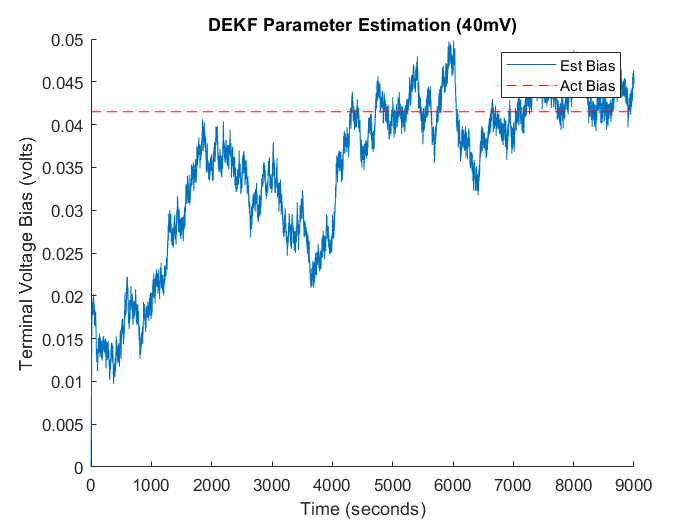
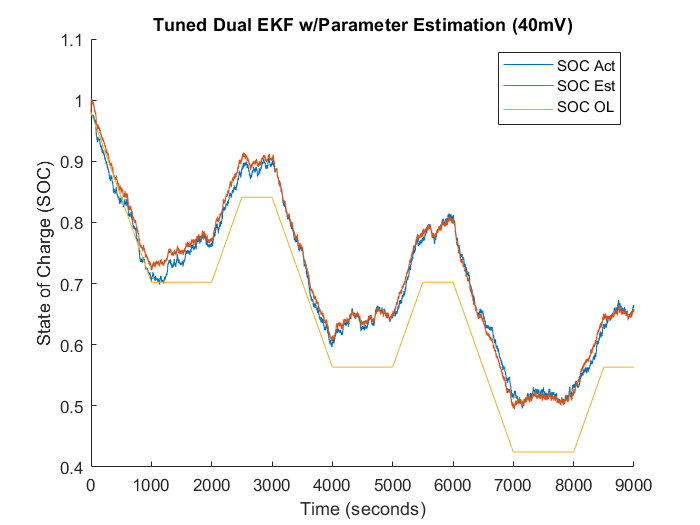
It should be further noted, that all other parameter variations aside, when R0 is tuned to match the approximate value of the true battery, the estimation results are , relatively speaking, fairly close the actual SOC.

## Sensor Bias Estimation

Voltage Bias

As the System Model and Analysis section briefly covered, every ‘real’ system exhibits some sort of noise or bias that presents the likelihood of the distorting and degrading acceptable estimation performance. To mitigate this behavior, the following plots demonstrate the benefits of implementing a tuned Dual EKF as means of estimating the SOC and the sensor bias (voltage bias). It should be noted that the ‘actual’ data used as ground truth in this simulation was reperformed with a constant 20mV. This prescribed value allows for the following comparison in plot (???) to be made revealing a general trend of the system to approximately oscillate about the actual voltage bias.





Current Bias

The results of the DEKF with current sensor bias is demonstrated below. A constant current bias of 0.25% of full scale was used from an online sensor specification data sheet. DEKF performs well in the case of constant current sensor bias with SOC average error at 0.24%. The indices

And also indicate good performance. The algorithm is able to track the bias which is modeled as random-walk. The bias is initialized at zero (no bias) as the level is generally unknown. The bias estimation error settles within 8500 seconds when tuning its fictitious noise to be very small (10-8.4).

# Future Work

Briefly summarize your project and its findings. Discuss any open questions or potential avenues for further research.

UKF, PF, Adaptive EKF, Gain Scheduled EKF, MHE.

# References

Use IEEE format for your references. It is useful but not necessary to use Word’s built in features for references and bibliographies.

|  |  |
| --- | --- |
| [1] | IEEE Periodicals, "IEEE Reference Guide," IEEE, Piscataway, NJ, 2018. |

# Supplemental Material

Include all Matlab code (Matlab has a “publish” feature that will help format your code nicely for Word). If you have Simulink models, include pictures of the models and code for any user-defined functions. If applicable, include additional figures and any other important work that you did not include in the body.

## Matlab Code

### File 1

(code here)

### File 2

(code here)

## Simulink Models

### Model 1

(image here)

(code for user-defined functions here)

## Additional Figures

## Anything Else