|  |
| --- |
| Extended Kalman Filtering of State and Parametric 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. A comparison between*  *a deterministic estimation technique using Youla paramertization and the well-established stochastic estimation technique, Extended Kalman filtering, is performed and analyzed for robust performance?* |

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

4.2.3 State EKF vs State Youla Estimation5

**5. Future Work4**

# 1. Introduction & Literature Review

## Motivation

1. 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].

## Related Work

## Battery Modeling

## Estimation Algorithms

## Objectives

# 2. System Modeling & Analysis

## Overview of Li-Ion Battery

## State of Charge

## Open Circuit Voltage

## Electrical Equivalent Circuit Model

## Continuous Time Model

## Discrete Time Model

## Sensor Bias Modeling

## Current Sensor Bias

## Voltage Sensor Bias

## Observability Analysis

# 3. Algorithms & Implementation

## 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 noise standard-deviation is allowed to increase up to 2% to simulate effects like aging, stress and electromagnetic interference. The bias level is set as 2:5% of the corresponding signal’s maximum value. Bias level is stepped up from 0% to 2.5% and then 2.5% to 5%.

## 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 = 600 and N is the length of the drive cycle.

Since the sampling time is 1s, this corresponding to ignoring the first 10min of data.

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

time (first 600 samples are excluded). 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 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

A linear Kalman Filter and Extended Kalman Filter was implemented on a nonlinear equivalent circuit battery model to estimate the battery’s 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 due to Open Current Voltage depending on SOC. On the other hand, EKF considers the nonlinearities of the battery model by linearization using first order Taylor series about an operating point, this in turn improves the SOC estimation as shown below.

## State EKF vs Dual EKF

## EKF Parameter Variation

## Sensor Bias Estimation

## State EKF vs Youla Estimation

NOTES:

**Model Validation**

**RMS Error**

**Parameter Estimation using Dual EKF**

**Covariance Agreement (Model vs Truth)**

**Biased Vs Unbiased Simulations**

**Robustness**

* Sensor noise
* Parameter Variation

**EKF vs KF Comparison**

## Figures

Figures should be centered on the page. Every figure should be numbered, have a caption, and be cited in the text. For example, see Figure 1. If you have many figures, you may find it useful to use Word’s Cross-Reference feature to keep track of figure, table, and equation numbering.

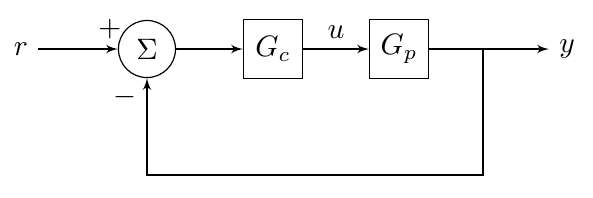


Figure 1 - A simple block diagram as an example of how to structure a figure.

## Tables

Tables of data should be treated like figures: centered, captioned, and cited in the text. For example, see Table 1.

Table 1 - This is a caption.

|  |  |  |
| --- | --- | --- |
| Column 1 Title | Column 2 Title | Column 3 Title |
| 1 | 5 | 9 |
| 2 | 6 | 10 |
| 3 | 7 | 11 |
| 4 | 8 | 12 |

## Equations

Equations should be on their own line and centered. Be sure to define all terms used in the equation. For example,

where is force, is mass, and is acceleration.

# 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