

University of Windsor

Scholarship at UWindor

Electronic Theses and Dissertations

Theses, Dissertations, and Major Papers

1-1-2019

Novel Approaches for State of Charge Modeling in Battery Management Systems

Mostafa Shaban Mohamed Ahmed
University of Windsor

Follow this and additional works at: <https://scholar.uwindsor.ca/etd>

Recommended Citation

Ahmed, Mostafa Shaban Mohamed, "Novel Approaches for State of Charge Modeling in Battery Management Systems" (2019). *Electronic Theses and Dissertations*. 8157.
<https://scholar.uwindsor.ca/etd/8157>

This online database contains the full-text of PhD dissertations and Masters' theses of University of Windsor students from 1954 forward. These documents are made available for personal study and research purposes only, in accordance with the Canadian Copyright Act and the Creative Commons license—CC BY-NC-ND (Attribution, Non-Commercial, No Derivative Works). Under this license, works must always be attributed to the copyright holder (original author), cannot be used for any commercial purposes, and may not be altered. Any other use would require the permission of the copyright holder. Students may inquire about withdrawing their dissertation and/or thesis from this database. For additional inquiries, please contact the repository administrator via email (scholarship@uwindsor.ca) or by telephone at 519-253-3000ext. 3208.

**NOVEL APPROACHES FOR STATE OF CHARGE MODELING IN
BATTERY MANAGEMENT SYSTEMS**

by

Mostafa Shaban Mohamed Ahmed

A Thesis

Submitted to the Faculty of Graduate Studies
through the Department of Electrical and Computer Engineering
in Partial Fulfillment of the Requirements for
the Degree of Master of Applied Science at the
University of Windsor

Windsor, Ontario, Canada

© 2019 Mostafa Shaban Mohamed Ahmed

Novel Approaches for State of Charge Modeling in Battery Management Systems

by
Mostafa Shaban Mohamed Ahmed

APPROVED BY:

J. Ahamed
Department of Mechanical, Automotive and Materials Engineering

M. Azzouz
Department of Electrical and Computer Engineering

B. Balasingam, Advisor
Department of Electrical and Computer Engineering

December 2, 2019

Declaration of Co-Authorship / Previous Publication

Co-Authorship

I hereby declare that this thesis incorporates material that is result of joint research, as follows: Chapters 2 of this thesis was co-authored with professor Balasingam and Sheikh Arif Raihan, and chapter 3 was co-authored with professor Balasingam who provided supervision and guidance during the research and writing process. In all cases, the key ideas, primary contributions, data analysis, interpretation, and writing were performed by the author.

I am aware of the University of Windsor Senate Policy on Authorship and I certify that I have properly acknowledged the contribution of other researchers to my thesis, and have obtained written permission from each of the co-author(s) to include the above material(s) in my thesis.

Previous Publication

Thesis chapter	Publication title/full citation	Publication status
2	Ahmed, Mostafa Shaban & Raihan, Sheikh Arif & Balasingam, Balakumar, “A Scaling Approach for Improved State-of-Charge Representation in Li-ion Batteries”, 2019	To be submitted
3	Ahmed, Mostafa Shaban & Balasingam, Balakumar, “A Tutorial On Linear State-Space Model Parameter Estimation Using the EM Algorithm”, 2019	To be submitted
2	Ahmed, Mostafa Shaban & Balasingam, Balakumar, “A Scaling Approach for Improved Open Circuit Voltage Modeling in Li-ion Batteries”, IEEE Electrical Power and Energy Conference, Montreal, 2019	Published

I certify that I have obtained a written permission from the copyright owner(s) to include the above published material(s) in my thesis. I certify that the above material describes work completed during my registration as a graduate student at the University of Windsor.

General

I declare that, to the best of my knowledge, my thesis does not infringe upon anyone’s copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my thesis. I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and

the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

Abstract

One of the key steps of any battery management system design is the representation of the open circuit voltage (OCV) as a function of the state of charge (SOC). The OCV-SOC relationship is very non-linear that is often represented using a polynomial that has log and inverse terms that are not defined around SOC equal to zero or one. The traditional response to this problem was only at the software level. In this thesis, I present a formal scaling approach to the OCV-SOC characterization in Li-ion batteries. I show that, through formal modeling and optimization, the traditional approach to OCV-SOC modeling can be significantly improved by selecting the proper value of ϵ . When the proposed technique is used a decrease in the maximum SOC error of 9% is reported. The proposed approach is tested on data collected from multiple cells over various temperatures for OCV-SOC characterization and the results are presented.

State-space model (SSM) and the Kalman filter have several applications in the emerging areas of automation and data science including in battery SOC estimation. In many such applications, the application of Kalman filtering requires model identification with the help of the observed data. I present the formulas with derivations for linear state-space model parameter estimation using the expectation maximization (EM) algorithm. Particularly, I derive the formulas for different special SSM cases of practical interest, such as the continuous white noise acceleration (CWNA) model. Through simulation, I show the benefits of these derivations for the special models in comparison with the generalized approach.

Contents

Declaration of Co-Authorship / Previous Publication	iii
Abstract	vi
List of Tables	x
List of Figures	xi
1 Introduction	1
1.1 Background about battery managment systems	1
1.2 Battery Fuel Gauge	4
1.3 Organization of the Thesis	5
1.4 Bibliography	6
2 A Scaling Approach for Improved State-of-Charge Representation in Li-ion Batteries	9
2.1 Introduction	9
2.2 Open Circuit Voltage Characterization Method	14
2.3 Proposed Scaling Approach	20
2.3.1 Proposed Scaling Approach	20
2.3.2 Optimization of the Scaling Parameter	21
2.4 Scaling Approach Results	22
2.4.1 Single Battery Results	23

2.4.2	Multiple Temperature Results	24
2.4.3	Multiple Batteries at Multiple Temperatures Results	26
2.5	State of Charge Tracking with Scaling	27
2.5.1	Scaled Extended Kalman Filter	28
2.5.2	Scaled EKF Results	30
2.6	Conclusions and Discussions	32
2.7	Bibliography	35

3 Linear State-Space Model Parameter Estimation Using the EM Algorithm 39

3.1	Introduction	39
3.2	Problem Definition	41
3.3	Review of the EM Algorithm	42
3.4	Scalar State-Space Model	44
3.4.1	Complete Data Likelihood	45
3.4.2	Expectation	45
3.4.3	Maximization	46
3.5	General State-Space Model	47
3.5.1	Complete Data Likelihood	47
3.5.2	Expectation	48
3.5.3	Maximization	49
3.6	CWNA State-Space Model	51
3.6.1	Complete Data Likelihood	52
3.6.2	Expectation	53
3.6.3	Maximiation	53
3.7	Results	53
3.7.1	Scalar SSM	54
3.7.2	CWNA SSM	57

3.8	Conclusion	58
3.8.1	Matrix Identities	59
3.8.2	Prediction, Filtering & Smoothing	59
3.9	Bibliography	60
4	Future Work	64
4.1	Bibliography	66
5	Conclusion	67
	Vita Auctoris	69

List of Tables

2.1	Comparison of RMSE	25
2.2	Number of cells tested for each temperature	27
3.3	Different SSM scenarios	54

List of Figures

1.1	Battery management system (BMS). Battery fuel gauge (BFG), cell balancing circuitry (CBC) and optimal charging algorithm (OPA) are the three important blocks of a BMS.	2
1.2	Battery fuel gauge. Individual blocks of the BFG [13].	4
2.3	OCV vs. SOC curve of a Li-ion battery. This particular curve is obtained from a Samsung EB575152 battery. The state of charge (SOC) is indicated as a ratio; sometimes SOC is referred to as a percentage.	15
2.4	Equivalent circuit model of a battery during slow charge/discharge. It must be noted that the above equivalent circuit model is suitable when the battery experiences constant current of very low amplitude. This model allows us to estimate the OCV-SOC curve.	16
2.5	Measured voltage and current. Voltage and current during charge and discharge at 30°C.	19
2.6	OCV-SOC modeling error. <i>Top row:</i> OCV vs. SOC, Voltage error vs. SOC and SOC error vs. OCV <i>Middle row:</i> OCV vs. SOC, Voltage error vs. SOC and SOC error vs. OCV <i>Bottom row:</i> RMSE as a function of ϵ for different temperatures.	23

2.7	Modeling error vs ϵ. Top Left: ϵ vs Temperature. The ϵ that had the minimum RMSE for each temperature. Top Right: ϵ frequency as a function of ϵ . Bottom Left: RMSE_{av} vs ϵ . Bottom Right: RMSE vs ϵ over all temperatures shown as a box plot (the central (red) mark of the box is median; the edges of the box are 25th and 75th percentile values; the whiskers extend to the most extreme data points not considered outliers, and outliers are plotted individually with a '+').	26
2.8	SOC tracking using EKF with and without scaling. Current and voltage profile used along the the SOC tracking.	31
2.9	Average RMSE for different Batteries. Average RMSE vs ϵ for different Li-ion batteries over multiple temperatures and multiple cells as indicated in Table 2.2.	33
2.10	Boxplot for different batteries at multiple temperatures. RMSE vs ϵ for multiple temperatures and multiple cells as indicated in Table 2.2. Shown as a box plot (the central (red) mark of the box is median; the edges of the box are 25th and 75th percentile values; the whiskers extend to the most extreme data points not considered outliers, and outliers are plotted individually with a '+'). The best ϵ values are those with small RMSE in terms of mean and variance.	34
3.11	Scalar SSM parameter estimation. The estimated LLh is shown for different scenarios; the true LLh is shown for comparison; the results are averaged over 1000 Monte-Carlo run with good initialization. . . .	55
3.12	RMSE of our estimate. Root mean square error (RMSE) in estimates averaged over 1000 Monte-Carlo runs good initialization experiment for all four cases.	56

3.13 Scalar SSM parameter estimation.	The estimated likelihood is shown for different scenarios; the true likelihood is shown for comparison; the results are averaged over 1000 Monte-Carlo run with bad initialization.	57
3.14 Matrix SSM case 1.	The estimated likelihood is shown for the matrix form case 1; the true likelihood is shown for comparison; the results are for 1000 Monte-Carlo run.	58

Chapter 1

Introduction

1.1 Background about battery management systems

The use of rechargeable (secondary) batteries is steadily on the rise with wide applications from small electronic devices to electric vehicles (EV) and aerospace applications. Rechargeable batteries need to be managed by a battery management system (BMS) to ensure the safety, efficiency and reliability of the devices powered by them. Depending on the application the BMS is required to estimate different states of the battery. These states can be state of charge (SOC), remaining useful life (RUL) and time to shutdown (TTS) among other things. In some application the BMS has to balance the SOC or temperature of individual cells. Also, the BMS is responsible for choosing the optimal charging algorithm for some applications such as EVs.

Figure 1.1 shows the components and responsibilities of the BMS. The first component is the battery fuel gauge (BFG). The BFG uses offline data to characterize the battery model. It then estimates the battery states using the voltage, current and temperature measurements and the offline data. BFG estimates battery states such as SOC and state of health (SOH). The second component of a BMS is the cell balancing circuitry (CBC). Which is responsible for balancing the SOC of all the individual cells (which is very important because when cells are unbalanced the

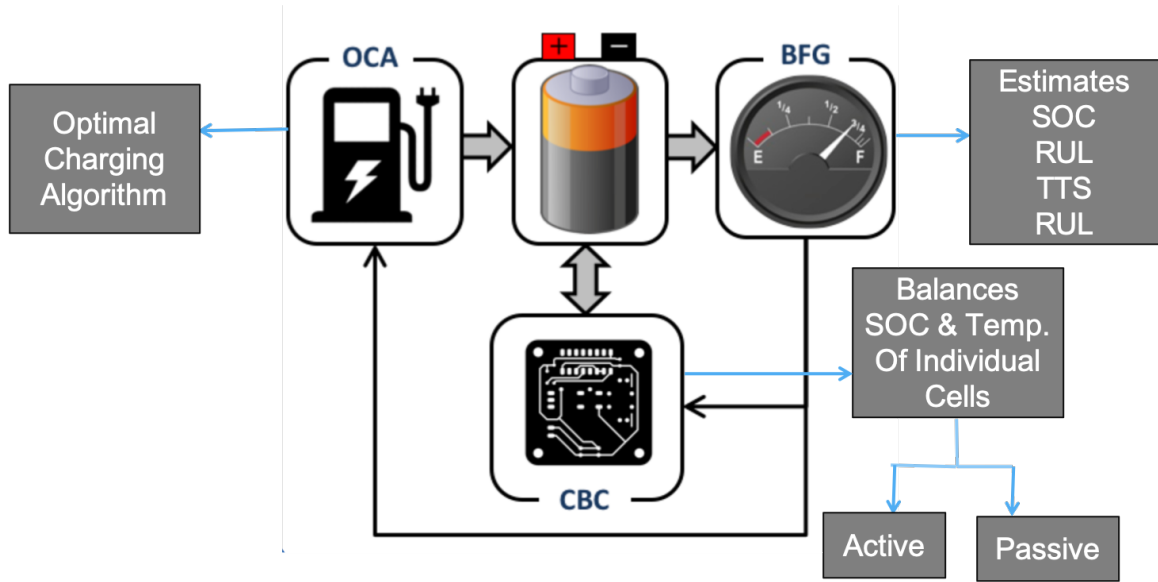


Figure 1.1: **Battery management system (BMS)**. Battery fuel gauge (BFG), cell balancing circuitry (CBC) and optimal charging algorithm (OCA) are the three important blocks of a BMS.

battery is not performing at its most optimal point). The third component is the optimal charging algorithm (OCA). This component is responsible for choosing the most optimal charging algorithm to the battery the fastest without compromising its health over time.

Although current BMS accuracy are good enough for applications such as mobile phones where an error in the estimated states can be non-critical for the operation of the device, in sensitive applications, such as EVs, the accuracy of the estimated states has to be very high, otherwise there can be serious downfalls. This is why more research is being done now to make the BMS more accurate and adaptable to changes.

With the current battery pack prices \$200-\$300/kWh in 2016 and 2017 [1], and with EVs having an average battery capacity of around 80 kWh, the battery pack alone costs around \$24,000. By managing the battery pack correctly a good BMS can extend its life and save money. Furthermore, the charge time of EVs is one of the open research challenges, where there is a trade off between the charge time and

battery life [2]. Less charging time means using higher current to charge the battery which lead to faster deterioration in the battery capacity. The goal is to find the optimal charging profile/pattern where the charging time is minimized with minimal compromise to the battery capacity over time.

Accurate SOC estimation is crucial for all aspects of a BMS. In [3] the authors proposed an approach to estimate the SOC based on dynamic impedance technique. While [4] proposed an approach based on multi-model switching strategy. A fractional order extended Kalman filter approach was proposed in [5]. With all these different approaches in hand the next important question is how to evaluate the accuracy of the these approaches since the true SOC is almost never known. This highlights the importance of BFG evaluation. The work [6,7] proposed different methods to evaluate the accuracy of the BFG. Another important issue is that all the current BMS rely heavily on offline characterization of the OCV-SOC model [8] which can change over time and by different usage profiles. For the BMS to be efficient it needs to track these changes effectively.

There is much to be known about the battery life and how a battery ages so that SOH and battery capacity can be accurately estimated. Even though many techniques are proposed for SOH estimation [9–12], when applied to real world scenarios and tested on real data the performance seems inadequate. Same can be said about the battery capacity estimation techniques proposed in the literature such as [13,14].

Finally, EV batteries are discarded once they reach 70-80 % of their original capacity [15]. However, this doesn't mean that they can't be reused in other applications. Such batteries can be used as a stationary storage systems in residential buildings and can improve the overall environmental sustainability [16]. For such applications fast and accurate battery modeling is required which is one of the focuses of current research.

1.2 Battery Fuel Gauge

BFG is the first and most important part of the BMS. This is because neither of the other two components of the BMS (CBC and OCA) can work properly without the estimates that the BFG provides. BFG accuracy is very important because it can affect the performance of the other BMS components.

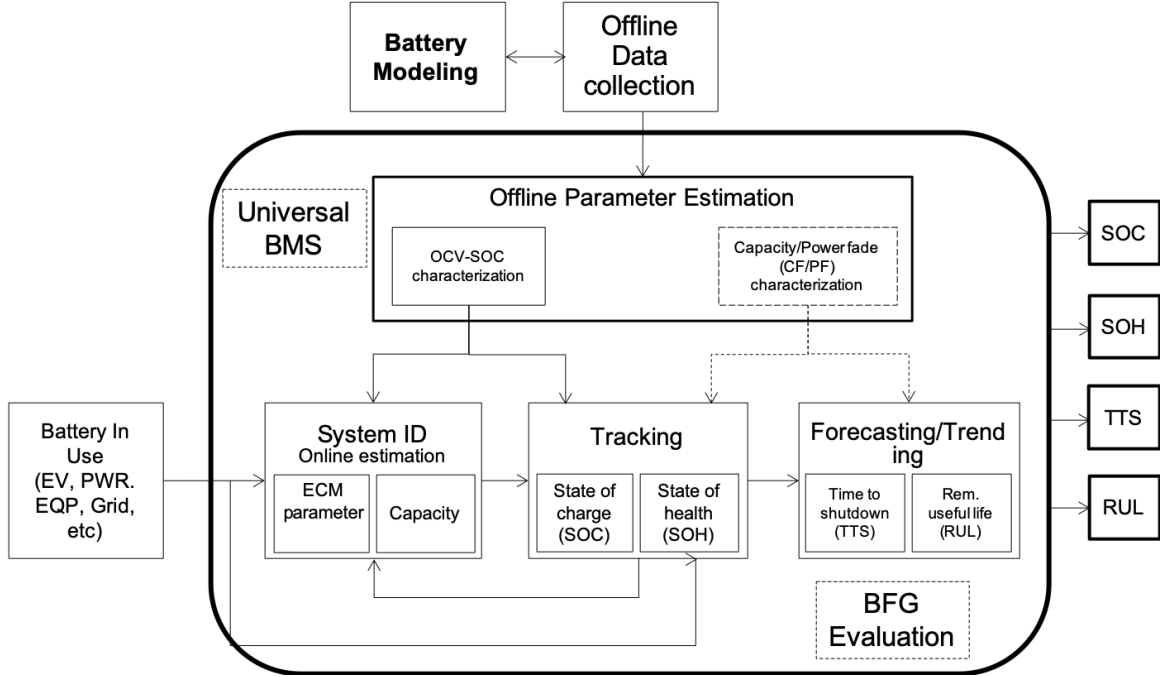


Figure 1.2: **Battery fuel gauge.** Individual blocks of the BFG [13].

Figure 1.2 shows the individual blocks of a BFG. First some offline data is collected, it is then used to get the OCV-SOC characterization and capacity/power fade characterization. Using these offline characterization the BMS can now estimate the ECM parameters and the battery capacity. Having all these estimates the BMS can track the SOC and SOH of the battery. It can also forecast the TTS and RUL of the battery.

Having the estimates provided by the BFG we need to know how accurate these estimates are. We can then use BFG evaluation to evaluate how good our estimates

are. There are many metrics that can be used to evaluate the accuracy of the BFG such as: coulomb counting metric, OCV metric, TTV metric and TTS metric. These metrics individually or combined can be used to evaluate the accuracy of the BFG.

1.3 Organization of the Thesis

I choose to present this thesis structured according to the manuscript format rather than the traditional format. The chapters consist of manuscripts previously written and submitted/published by the author, with first authorship. Chapters 2, and 3 are included in this thesis as written at the times of their submissions, with minor alterations to format and slight modifications to content in order to maintain a cohesive thesis structure. As prescribed by the manuscript format, abstracts have also been omitted.

While a traditional thesis commonly contains a general literature review and problem statement, and since each chapter has its own literature review I have chosen to omit these sections from the introduction. Each of the chapters will provide a literature review and problem statement in the introduction which serve the purpose of familiarizing the reader with both the context of the research and relevant literature. To include a general literature review and problem statement in this thesis would be to introduce unnecessary redundancy.

The remainder of this thesis is organized as follows: Chapter 2, presents a novel linear scaling approach for OCV-SOC modeling. This approach is shown to minimize the offline OCV-SOC characterization error. It also provides a formal approach that can be used to find the optimal scaling factor. Furthermore, the scaling approach can be incorporated in the online SOC estimation technique to minimize the error in the estimation. Chapter 3 is a tutorial on how to use the expectation maximization (EM) algorithm on the state space model (SSM) to estimate unknown model parameters. It shows the derivation of the EM algorithm for scalar, matrix form

and continuous white noise acceleration (CWNA). Additionally, in the results section, it shows and compares the results of different cases that are tested. Moreover, chapter 4 details some possible future work. Particularly it is shown how the EM algorithm can be applied to the BMS state equation for SOC estimation. The battery capacity estimation equation is derived in this chapter along with an idea on how the OCV-SOC model can be estimated online. Finally, chapter 5 concludes this thesis.

1.4 Bibliography

- [1] I.-Y. L. Hsieh, M. S. Pan, Y.-M. Chiang, and W. H. Green, “Learning only buys you so much: Practical limits on battery price reduction,” *Applied Energy*, vol. 239, pp. 218 – 224, 2019. pages 2
- [2] J. Schoch, J. Gaerttner, A. Schuller, and T. Setzer, “Enhancing electric vehicle sustainability through battery life optimal charging,” *Transportation Research Part B: Methodological*, vol. 112, pp. 1 – 18, 2018. pages 3
- [3] M.-H. Hung, C.-H. Lin, L.-C. Lee, and C.-M. Wang, “State-of-charge and state-of-health estimation for lithium-ion batteries based on dynamic impedance technique,” *Journal of Power Sources*, vol. 268, pp. 861 – 873, 2014. pages 3
- [4] Y. Wang, C. Zhang, and Z. Chen, “A method for state-of-charge estimation of li-ion batteries based on multi-model switching strategy,” *Applied Energy*, vol. 137, pp. 427 – 434, 2015. pages 3
- [5] K. S. Mawonou, A. Eddahech, D. Dumur, D. Beauvois, and E. Godoy, “Improved state of charge estimation for li-ion batteries using fractional order extended kalman filter,” *Journal of Power Sources*, vol. 435, p. 226710, 2019. pages 3
- [6] B. Balasingam, G. Avvari, B. Pattipati, K. Pattipati, and Y. Bar-Shalom, “Robust battery fuel gauge algorithm development, part 3: State of charge tracking,”

- in *2014 International Conference on Renewable Energy Research and Application (ICRERA)*, pp. 110–115, IEEE, 2014. pages 3, 10
- [7] B. Balasingam, G. Avvari, K. Pattipati, and Y. Bar-Shalom, “Performance analysis results of a battery fuel gauge algorithm at multiple temperatures,” *Journal of Power Sources*, vol. 273, pp. 742–753, 2015. pages 3
 - [8] B. Pattipati, B. Balasingam, G. Avvari, K. R. Pattipati, and Y. Bar-Shalom, “Open circuit voltage characterization of lithium-ion batteries,” *Journal of Power Sources*, vol. 269, pp. 317–333, 2014. pages 3, 10, 14, 25
 - [9] L. Fang, J. Li, and B. Peng, “Online estimation and error analysis of both soc and soh of lithium-ion battery based on dekf method,” *Energy Procedia*, vol. 158, pp. 3008 – 3013, 2019. Innovative Solutions for Energy Transitions. pages 3
 - [10] D. Andre, C. Appel, T. Soczka-Guth, and D. U. Sauer, “Advanced mathematical methods of soc and soh estimation for lithium-ion batteries,” *Journal of Power Sources*, vol. 224, pp. 20 – 27, 2013. pages 3
 - [11] Y. Li, M. Abdel-Monem, R. Gopalakrishnan, M. Berecibar, E. Nanini-Maury, N. Omar, P. van den Bossche, and J. V. Mierlo, “A quick on-line state of health estimation method for li-ion battery with incremental capacity curves processed by gaussian filter,” *Journal of Power Sources*, vol. 373, pp. 40 – 53, 2018. pages 3
 - [12] X. Tang, C. Zou, K. Yao, G. Chen, B. Liu, Z. He, and F. Gao, “A fast estimation algorithm for lithium-ion battery state of health,” *Journal of Power Sources*, vol. 396, pp. 453 – 458, 2018. pages 3
 - [13] B. Balasingam, G. V. Avvari, B. Pattipati, K. Pattipati, and Y. Bar-Shalom, “Robust battery fuel gauge algorithm development, part 2: Online battery-capacity estimation,” in *2014 International Conference on Renewable Energy Research and Application (ICRERA)*, pp. 104–109, Oct 2014. pages xi, 3, 4, 10

- [14] G. L. Plett, “Extended kalman filtering for battery management systems of lipb-based HEV battery packs: Part 3. State and parameter estimation,” *Journal of Power sources*, vol. 134, no. 2, pp. 277–292, 2004. pages 3, 10
- [15] S. Saxena, C. L. Floch, J. MacDonald, and S. Moura, “Quantifying ev battery end-of-life through analysis of travel needs with vehicle powertrain models,” *Journal of Power Sources*, vol. 282, pp. 265 – 276, 2015. pages 3
- [16] M. A. Cusenza, F. Guarino, S. Longo, M. Ferraro, and M. Cellura, “Energy and environmental benefits of circular economy strategies: The case study of reusing used batteries from electric vehicles,” *Journal of Energy Storage*, vol. 25, p. 100845, 2019. pages 3

Chapter 2

A Scaling Approach for Improved State-of-Charge Representation in Li-ion Batteries

2.1 Introduction

Rechargeable batteries remain the primary source for storing electrical energy. Rechargeable batteries have wide range of applications from electric vehicles to mobile phones and many wireless devices used in day to day life [1].

Li-ion batteries have proven to be far superior than any other existing type of batteries in terms of their energy density. Li-ion batteries have high energy and power density and don't suffer from memory effect [2]. They also have reasonable life cycle [3]. Li-ion batteries have low self discharge rate, good charging and discharging efficiency and they work efficiently over a broad range of temperature [4]. Due to that, there is high demand on Li-ion batteries in the automobile and aerospace applications.

A battery management system (BMS) ensures the safety, efficiency and reliability of a battery system. The BMS needs to estimate some important states of the battery such as state of charge (SOC), state of health (SOH) and remaining useful life (RUL)

need to be estimated. Battery fuel gauge (BFG) algorithm is used to estimate these states [5]. A BFG consists of several offline and online modeling and parameter estimation modules, such as, OCV-SOC modeling, where OCV stands for open circuit voltage, equivalent circuit model (ECM) parameter estimation, battery capacity estimation [6], SOC tracking [6], remaining power prediction [7] and RUL estimation [8], etc. Implementing the BFG algorithm is challenging because it is affected by many battery parameters that vary according to the temperature, age and battery cycles [9]. The focus of this chapter is at the OCV-SOC modeling step where the objective is to develop an accurate offline model that shows the SOC as a function of the open circuit voltage (OCV). We also investigate the effect of using the proposed approach on online estimations of the SOC while the battery is in use.

In order to estimate the OCV-SOC model parameters [10], a fully charged battery is slowly discharged until it is empty; it is then slowly charged until it becomes full again. The {Voltage , Current} pair is collected during this entire time to estimate the OCV-SOC model parameters using a least-square estimation approach [10]. Several models were tried in the past, from simple linear models to complex polynomial models involving log and inverse terms. A comparative review of several OCV-SOC models can be found in [10]. Of the many OCV-SOC models studied in [10], the combined+3 model is considered for further analysis in this chapter. Compared to other models, the combined +3 model is simplistic at the same time it allows to capture salient features of the OCV-SOC curve.

The combined+3 model relates the OCV and SOC as follows

$$V_0(s) = k_0 + \frac{k_1}{s} + \frac{k_2}{s^2} + \frac{k_3}{s^3} + \frac{k_4}{s^4} + k_5s + k_6\ln(s) + k_7\ln(1-s) \quad (2.1)$$

where s denoted the SOC $\in [0, 1]$ and $V_0(s)$ denotes the OCV. The range of OCV depends on the type of battery; For a single li-ion cell this range is OCV $\in [3, 4.2]$. The “modeling objective” is to estimate the model parameters k_0, k_1, \dots, k_7 . By

collecting (OCV,SOC) data points spanning the entire range of OCV and SOC, the model parameter vector $k = [k_0, k_1, \dots, k_7]$ can be linearly estimated using a least square approach. However, as it can be seen from (2.1) the model is not defined closer to SOC = 0 or SOC = 1. Traditional response to this problem is to “discard” data that is closer to SOC = 0 or SOC = 1. In this chapter, we show that such a discarding approach results in a significant worst case error and propose a better approach to remedy this problem.

In this chapter we present the details of a novel scaling technique and detail its performance improvement over existing methods. We also propose an optimization approach that can be used to find the scaling factor ϵ (see (2.21) for more details) that minimizes the root mean square error (RMSE) over a wide range of temperatures. We then show how the proposed scaling technique can be used for online SOC estimation using an extended Kalman filter (EKF). Furthermore, we present the results for online SOC estimation with and without scaling for simulated data and show how the online SOC estimation can be improved using the scaling approach.

The contribution of this chapter are listed below:

- *A novel scaling approach.* We introduce a novel scaling approach that can be used for OCV-SOC model parameter estimation.
- *An approach to optimize the scaling factor.* We formulate a mathematical approach to calculate the optimal value for the scaling factor that minimizes the RMSE the modeling error.
- *Results for the scaling and optimization approaches.* We test both approaches on nine different batteries over a wide range of temperature and show the results.
- *A novel approach for online SOC estimation.* We present a novel approach for online SOC estimation by combining EKF and the proposed scaling approach.
- *Objective performance analysis.* We present the effect of using the online SOC

estimation approach compared to a typical EKF (without scaling).

This chapter is organized as follows: section 2.2 explains the battery model that was used and the OCV-SOC function. It also explains in detail how to derive the OCV-SOC model. The novel scaling approach is introduced in section 2.3. We also formulate a mathematical approach to find the scaling factor that leads to the minimum modeling error. Furthermore, section 2.4 shows the results of the proposed approach when tested and validated on real data. Additionally, section 2.5 shows how to use the proposed scaling approach for online estimation of the SOC. We also present in this section the results for tracking the SOC with and without the use of the scaling approach. Finally, section 2.6 concludes the chapter.

List of Notations

C_{batt}	Battery capacity in Ampere hour (see (2.4))
Δ_k	Time difference between two measurements (2.4)
ϵ	Scaling factor (2.21)
ϵ_{opt}	Optimal value for ϵ that has corresponding minimum RMSE (2.27)
\mathbf{e}	Voltage error vector in the OCV-SOC model (2.22)
$G[k]$	Control variable of the Kalman filter at time k (2.34)
$h[k]$	Hysteresis at time k (2.6)
$h'[k+1]$	Linearization of measurement model (4)
$i[k]$	Measured current at time k (2.4)
k_0, k_1, \dots, k_7 ..	OCV-SOC model parameter for combined+3 model (2.1) (also see \mathbf{k})
\mathbf{k}	OCV parameter vector (2.17)

$\hat{\mathbf{k}}$	Estimate of OCV parameter vector \mathbf{k} (2.18)
\mathbf{k}_o	OCV parameter vector without R_0 (2.11)
$\hat{\mathbf{k}}_o$	Estimate of \mathbf{k}_o (2.19)
$m[k+1]$	Inovation variance of Kalman filter (6)
$n_v[k]$	Voltage measurement noise at time k (2.5)
\mathbf{n}	$N \times 1$ vector each row is $n_v[k]$ (2.16)
$P_s[k]$	State variance of the Kalman filter (2)
$\mathbf{p}[k]^T$	Measurement model for the least square algorithm (2.10)
$\mathbf{p}_o(s[k])^T$	First eight elements of $\mathbf{p}[k]^T$ (2.12)
\mathbf{P}	$N \times 8$ matrix each row is $\mathbf{p}[k]^T$ (2.15)
R_0	Series resistance of the battery (2.6)
R_h	Hysteresis equivalent resistance (2.9)
$R_{0,h}$	Effective resistance (2.9)
RMSE	Root mean square error in voltage of the OCV-SOC model (2.23)
$\text{RMSE}(\epsilon_i, t_j)$..	RMSE for scaling factor ϵ_i and temperature t_j (2.24)
$\text{RMSE}_{\text{av}}(\epsilon_i)$..	RMSE for scaling factor ϵ_i averaged over all temperatures (2.26)
s	State of charge (2.1)
s'	Scaled state of charge (2.20)
$s[k]$	SOC at time k (2.2)
σ_i	Standard deviation of current measurement (2.36)

σ_s	Standard deviation of SOC estimate (2.36)
σ_v	Standard deviation of voltage measurement (5)
$S[k+1 k]$	Innovation variance of Kalman filter (5)
$u[k]$	Control input of the Kalman filter at time k (2.35)
$v[k]$	True terminal voltage at time k (2.5)
$V_o(s)$	OCV at SOC s (2.1)
$V_o(s[k])$	OCV at SOC $s(k)$ (2.2)
$\hat{V}_o(s)$	Estimate of $V_o(s)$ (2.19)
$V_o(x_s[k])$	observation model of the Kalman filter at time k (2.38)
\mathbf{v}	$N \times 1$ vector of voltage measurements $z_v[k]$ (2.14)
$W[k+1]$	Gain of Kalman filter (7)
$x_s[k]$	State of the Kalman filter at time k (2.32)
$\hat{z}[k+1 k]$	Measurement prediction of Kalman filter (3)
$z_v[k]$	Measured terminal voltage at time k (2.5)

2.2 Open Circuit Voltage Characterization Method

In this section, we summarize the approach presented in [10] to OCV characterization. The OCV of a Lithium-ion battery varies with its state of charge (SOC) in a non-linear fashion as shown by a sample curve in Figure 2.3.

The set of OCV-SOC parameters $\{k_0, k_1, k_2, k_3, k_4, k_5, k_6, k_7\}$ in combined+3 model, can be estimated offline, i.e., through a custom experimental data collection process followed by a parameter estimation step. Next, such an approach presented in [10]

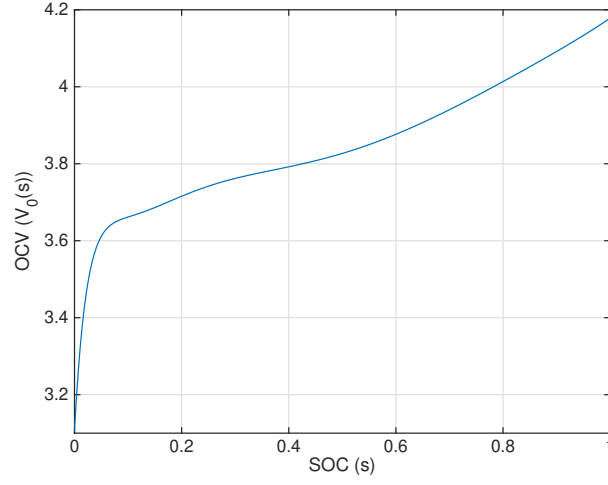


Figure 2.3: **OCV vs. SOC curve of a Li-ion battery.** This particular curve is obtained from a Samsung EB575152 battery. The state of charge (SOC) is indicated as a ratio; sometimes SOC is referred to as a percentage.

for experimental data collection and the estimation of the OCV-SOC parameters is briefed.

For accurate enough estimation of the OCV-SOC parameters, we need the $\{V_o(s), s\}$ pairs spanning $s \in [0, 1]$. The data was collected by discharging the battery from full-to-empty and then charging it back from empty-to-full with a very low current ($C/30$ to $C/40$). Assuming that the data is recorded every time index k , the OCV-SOC relationship in (2.1) can be re-written as

$$V_o(s[k]) = k_0 + \frac{k_1}{s[k]} + \frac{k_2}{s^2[k]} + \frac{k_3}{s^3[k]} + \frac{k_4}{s^4[k]} + k_5 s[k] + K_6 \ln(s[k]) + k_7 \ln(1 - s[k]) \quad (2.2)$$

where $k_0, k_1, k_2, k_3, k_4, k_5, k_6$ and k_7 are the parameters corresponding to one of the models of OCV-SOC characterization (see (2.1)).

Figure 2.4 shows the equivalent circuit of a battery when it is slowly charged or

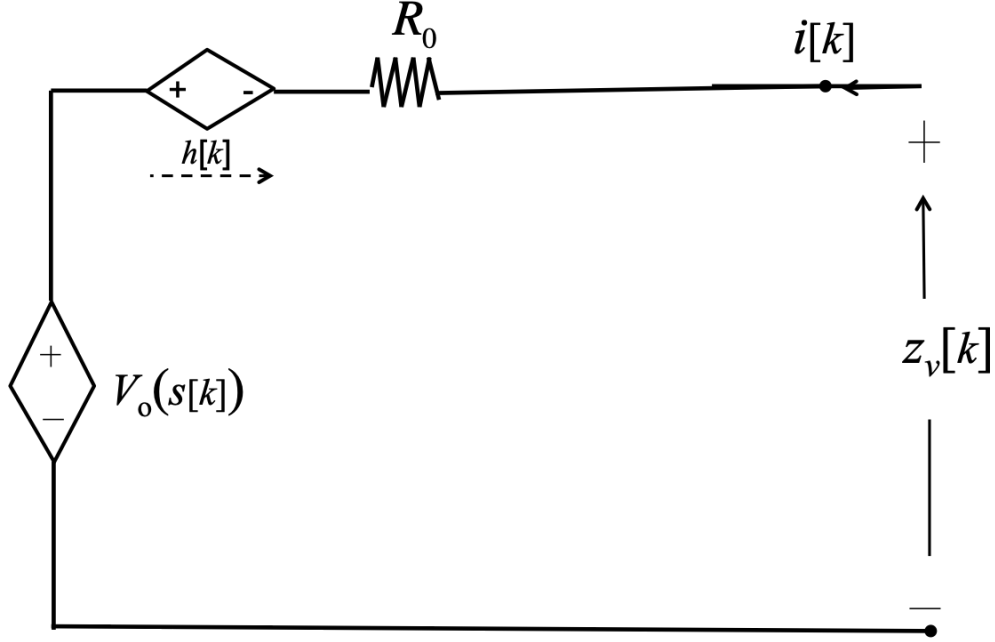


Figure 2.4: **Equivalent circuit model of a battery during slow charge/discharge.** It must be noted that the above equivalent circuit model is suitable when the battery experiences constant current of very low amplitude. This model allows us to estimate the OCV-SOC curve.

discharged with a constant rate. First, we define the SOC at a given time as

$$s[k] \triangleq s \quad \text{at time } k \quad (2.3)$$

The true SOC can be calculated using the Coulomb counting equation. The authors in [11] show different sources of error in the Coulomb counting equation, that can be a result of the integration error, the battery capacity error or current measurement error. Since the current during the OCV experiment is usually kept constant through highly accurate programmable charge/load devices such as Arbin tester [12], we assume that there is no noise in the measured current. Also, the integration error can be neglected because constant current is used [11]. Therefore, we can use the Coulomb counting equation with confidence.

The Coulomb Counting equation is given below,

$$s[k+1] = s[k] + \frac{\Delta_k i[k]}{3600 C_{\text{batt}}} \quad (2.4)$$

where Δ_k is the time difference between two measurements, $i[k]$ is the current through the battery and C_{batt} is the battery capacity in Ampere hour (Ah), considering the voltage measurement errors, the measured voltage is written as

$$z_v[k] = v[k] + n_v[k] \quad (2.5)$$

where $n_v[k]$ is the voltage measurement noise which is modeled as white Gaussian with standard deviation (s.d.) σ_v . During the OCV experiment i.e., when the battery is being slowly charged/discharged, the terminal voltage can be written as

$$z_v[k] = V_o(s[k]) + h[k] + i[k]R_0 + n_v[k] \quad (2.6)$$

where $h[k]$ is the hysteresis or voltage “pull” which is a function of current and SOC of the battery [13]. Since the OCV test is performed at a very low current, we assume that the hysteresis is proportional to the current only [12], i.e.

$$h[k] \propto i[k] \quad (2.7)$$

Hence, (2.6) can be rewritten as

$$z_v[k] = V_o(s[k]) + i[k]R_{0,h} + n_v[k] \quad (2.8)$$

where the *effective resistance*

$$R_{0,h} = R_0 + R_h \quad (2.9)$$

is the summation of the battery series resistance R_0 and the *constant-current hysteresis equivalent resistance*, R_h .

The parameters of the linear OCV-SOC model in (2.2) can be written as,

$$z_v[k] = \underbrace{[\mathbf{p}_o(s[k])^T \quad i[k]]}_{\mathbf{p}[k]^T} \underbrace{\begin{bmatrix} \mathbf{k}_o \\ R_{0,h} \end{bmatrix}}_{\mathbf{k}} + n_v[k] \quad (2.10)$$

where

$$\mathbf{k}_o = [k_0 \ k_1 \ k_2 \ k_3 \ k_4 \ k_5 \ k_6 \ k_7]^T \quad (2.11)$$

and

$$\mathbf{p}_o(s[k])^T = [1 \ \frac{1}{s[k]} \ \frac{1}{s^2[k]} \ \frac{1}{s^3[k]} \ \frac{1}{s^4[k]} \ s[k] \ \ln(s[k]) \ \ln(1 - s[k])] \quad (2.12)$$

By considering a batch of N voltage observations, (2.10) can be written as

$$\mathbf{v} = \mathbf{P}\mathbf{k} + \mathbf{n} \quad (2.13)$$

where

$$\mathbf{v} = [z_v[1] \ z_v[2] \ \dots \ z_v[t_N]]^T \quad (2.14)$$

$$\mathbf{P} = [\mathbf{p}[1] \ \mathbf{p}[2] \ \dots \ \mathbf{p}[t_N]]^T \quad (2.15)$$

$$\mathbf{n} = [n[1] \ n[2] \ \dots \ n[t_N]]^T \quad (2.16)$$

$$\mathbf{k} = [k_0 \ k_1 \ k_2 \ k_3 \ k_4 \ k_5 \ k_6 \ k_7 \ R_{0,h}]^T \quad (2.17)$$

The least squares estimate of the parameter vector is given by

$$\hat{\mathbf{k}} = (\mathbf{P}^T \mathbf{P})^{-1} \mathbf{P}^T \mathbf{v} \quad (2.18)$$

Now, for a given SOC, s , the corresponding OCV estimate $\hat{V}_o(s)$ is computed as

$$\hat{V}_o(s) = \mathbf{p}_o(s)^T \hat{\mathbf{k}}_o \quad (2.19)$$

where $\hat{\mathbf{k}}_o$ is formed by the first 8 elements of $\hat{\mathbf{k}}$.

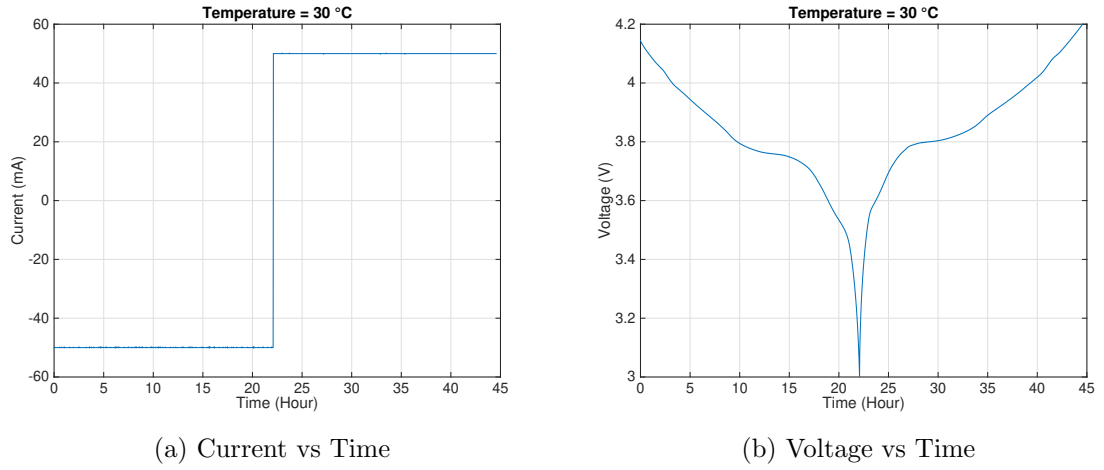


Figure 2.5: **Measured voltage and current.** Voltage and current during charge and discharge at 30°C.

Figure 2.5 shows the discharging/charging process when low current ($C/30$) was used to drain/charge the battery. After computing the SOC using Coulomb counting, as shown in (2.4), it can be used along with the measured voltage and current values to compute the OCV-SOC model using (2.18).

Finally, the OCV-SOC modeling can be summarized as follows:

Summary: OCV-SOC Modeling at Temperature T

1. Fully charge the battery at T_{\max}
2. Bring the battery to temperature T
3. Collect $v[k], i[k]$ during steps 4) and 5)
4. Slow-discharge the battery at C/N rate until empty
5. Slow-charge the battery at C/N rate until full
6. Compute battery capacity at T
7. Compute SOC $s[k]$ using Coulomb counting through (2.4)
8. Estimate the model parameters through (2.18)

2.3 Proposed Scaling Approach

2.3.1 Proposed Scaling Approach

One of the problems of the offline OCV-SOC modeling is that models such as the *combined+3 model* and others are not defined at the SOC values $s = 0$ and $s = 1$. Considering that the OCV-SOC model has ‘log(s)’ and ‘1/s’ terms, value of SOC that is closer to “0” and “1” will cause numerical issues. In this section we present a formal approach to solve this problem.

The proposed scaling approach maps the range of $s \in [0, 1]$ to $s' \in [\epsilon, 1 - \epsilon]$ where

$$s' = s(1 - 2\epsilon) + \epsilon \quad (2.20)$$

and

$$\epsilon \in (0, 0.5) \quad (2.21)$$

where ϵ is the *scaling factor*. From here on, we use s' to indicate *scaled SOC* and s to indicate *unscaled(true) SOC*. Now s' will be used instead of s in (2.2) and the entire OCV-SOC parameter estimation procedure described in Section 2.2. Here, it must be noted that s' does not go to 0 or 1 – it always stays ϵ away from these extreme values.

2.3.2 Optimization of the Scaling Parameter

In this subsection, we formulate the optimization problem. First, let us denote the voltage modeling error as

$$\mathbf{e} = \mathbf{v} - \mathbf{P}\hat{\mathbf{k}} \quad (2.22)$$

where $N \times 1$ vector \mathbf{e} is the voltage error in the OCV-SOC model. Now we will calculate the RMSE for the OCV-SOC model as follows,

$$\text{RMSE} = \frac{\mathbf{e}^T \mathbf{e}}{N} \quad (2.23)$$

where RMSE is the root mean square error in voltage of the OCV-SOC model. The above experiment can be repeated in different temperatures. Now, let us denote the voltage modeling error at $\epsilon = \epsilon_i$, and temperature $t = t_j$ as $\mathbf{e}(\epsilon_i, t_j)$. Then, the RMSE at $\epsilon = \epsilon_i$ and $t = t_j$ is given as

$$\text{RMSE}(\epsilon_i, t_j) = \frac{\mathbf{e}(\epsilon_i, t_j)^T \mathbf{e}(\epsilon_i, t_j)}{N} \quad (2.24)$$

for m different values of ϵ and n different values of t , the computed RMSE becomes the following matrix,

$$\begin{bmatrix} \text{RMSE}(\epsilon_1, t_1) & \text{RMSE}(\epsilon_1, t_2) & \dots & \text{RMSE}(\epsilon_1, t_n) \\ \text{RMSE}(\epsilon_2, t_1) & \text{RMSE}(\epsilon_2, t_2) & \dots & \text{RMSE}(\epsilon_2, t_n) \\ \vdots & \vdots & \vdots & \vdots \\ \text{RMSE}(\epsilon_m, t_1) & \text{RMSE}(\epsilon_m, t_2) & \dots & \text{RMSE}(\epsilon_m, t_n) \end{bmatrix} \quad (2.25)$$

Through our analysis of the the data we were able to conclude that averaging the RMSE for each ϵ over the entire range of temperature is a good estimate of the RMSE (these results are shown in the next section). This can be done by taking each row from $\text{RMSE}(\epsilon_i, t_j)$ and calculating the $\text{RMSE}_{\text{av}}(\epsilon_i)$ by,

$$\text{RMSE}_{\text{av}}(\epsilon_i) = \frac{\sum_{j=1}^n \text{RMSE}(\epsilon_i, t_j)}{n} \quad (2.26)$$

The optimal value for ϵ is the one that has the minimum RMSE_{av} and it can be calculated as follows,

$$\epsilon_{\text{opt}} = \arg \min_{\epsilon} (\text{RMSE}_{\text{av}}(\epsilon)) \quad (2.27)$$

2.4 Scaling Approach Results

This section is divided into three subsection. Subsection 2.4.1 shows the results of using different scaling factors ϵ on real data collected from a single battery Samsung EB575152. Subsection 2.4.2 shows the results of using different scaling factors ϵ on a single battery Samsung EB575152 at multiple temperatures. Finally, subsection 2.4.3 shows the results of using different scaling factors ϵ on multiple batteries at multiple temperatures.

2.4.1 Single Battery Results

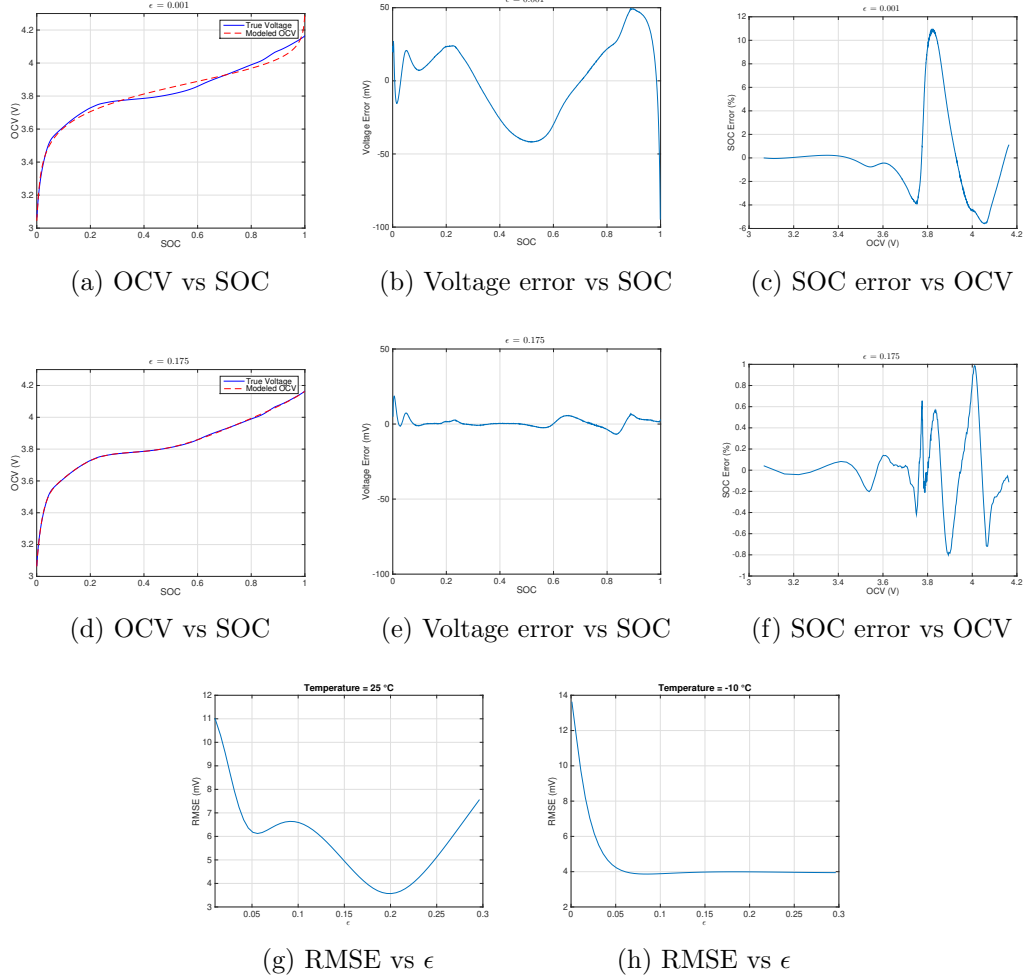


Figure 2.6: **OCV-SOC modeling error. *Top row:*** OCV vs. SOC, Voltage error vs. SOC and SOC error vs. OCV

Middle row: OCV vs. SOC, Voltage error vs. SOC and SOC error vs. OCV

Bottom row: RMSE as a function of ϵ for different temperatures.

Figure 2.6 shows the difference between a proper scaling approach and intuitive tweaks (often done at the programming level) during battery characterization. It is evident from Figure 2.6 that using different scaling factors can have a significant impact on the voltage modeling error. When $\epsilon = 0.001$ was used Figure 2.6(a)¹ it

¹This is to emulate how the numerical instability problem is typically addressed at the programming level.

showed $\text{RMSE} = 25.6\text{mV}$. When $\epsilon = 0.175$ was used Figure 2.6(a) (this ϵ is close to the optimized value) it showed $\text{RMSE} = 3.3\text{mV}$. This shows that using a very small value of ϵ is not always the best way of mapping the SOC (s).

Figure 2.6(b) and Figure 2.6(e) show that the voltage modeling error can be significantly reduced by using the appropriate scaling factor. It also shows that absence of scaling, which is the traditional way for OCV-SOC modeling, can lead to gigantic error in OCV-SOC modeling. Additionally, Figure 2.6(c) shows that not using scaling leads to an SOC error of more than 10% around the nominal voltage of the battery. On the other hand, when scaling is used in Figure 2.6(f) it shows a maximum SOC error of 1%. By using the proposed scaling technique the max SOC error can be reduced by 9%. The effect of this modeling error on the online SOC estimation is shown in Section 2.5.2.

Figure 2.6(g) shows the RMSE as a function of ϵ at 25°C . It can be seen that in this case $\epsilon = 0.2$ result in the minimum RMSE equal to 3.6mV . It should be noted that this is not the only way the RMSE behaves for different ϵ . Figure 2.6(h) shows a different way in which the RMSE changes with ϵ . The lowest RMSE happens at $\epsilon = 0.1$ with $\text{RMSE} = 3.9\text{mV}$, the graph is flat after that. However, for $\epsilon = 0.2$ the $\text{RMSE} = 4\text{mV}$. It is clear that the difference is not that significant in terms of the RMSE. Other relations of RMSE vs ϵ can appear but the goal is to find an ϵ value that is best over all temperatures. We address this issue in the next subsection.

2.4.2 Multiple Temperature Results

Data sets were collected from the Samsung EB575152 over temperature ranging from -25 to 50°C . The data sets had contained voltage, current and time measurements. The sampling rate for this experiment was 1 second. Four cells were tested for each temperature and 4 data sets were collected one for each cell. The OCV-SOC model was computed over all temperatures for ϵ ranging from 0.001 to 0.3. The value for ϵ that showed the smallest RMSE was computed. Figure 2.7(a) shows the range of

Table 2.1: Comparison of RMSE

	Proposed	[10]	[10]	[14]
Model	Combined+3	Combined+3	Combined	Combined
RMSE(mV)	4.1	10	18	5.3

temperature and the corresponding ϵ that showed to have the smallest RMSE.

It is clear from Figure 2.7(a) that the value for ϵ that is most appropriate over most temperatures is around $\epsilon = 0.18$. So we do some further data analysis and try to reach the best ϵ that would result in the least RMSE. Figure 2.7(b) shows the number of times each ϵ showed a corresponding minimum RMSE. From the first look at Figure 2.7(b) it seems that the optimal value for ϵ is between $\epsilon = 0.15$ and $\epsilon = 0.19$ since this range has the most number of corresponding minimum RMSE. However, Figure 2.6(h) shows that different ϵ can have very small RMSE. This is very important because it means that although one ϵ might show a minimum RMSE for certain temperature it might not be suitable for other temperatures.

In order to find ϵ_{opt} , first we compute the $\text{RMSE}(\epsilon, t)$ for every ϵ in our range, this is done for all the temperatures and the box plot is shown in 2.7(c). We then calculate the $\text{RMSE}_{\text{av}}(\epsilon_i)$ using (2.26). In order to find the single optimal value for ϵ_{opt} over all temperatures the average $\text{RMSE}_{\text{av}}(\epsilon_i)$ was computed and the result is presented in Figure 2.7(d).

From the data presented in Figure 2.7(c)&(d) we propose to use $\epsilon = 0.17$ since it shows to have the minimum RMSE_{av} over all temperatures at a value of $\text{RMSE} = 4.1\text{mV}$ and this value also has the smallest variance as it can be seen from the boxplot in Figure 2.7(c).

Table 2.1 compares the RMSE derived in this chapter and the RMSE in [10] which used the combined model and combined+3 model. It also shows the RMSE shown in [14] which only used the combined model. Table 2.1 shows that using the proposed scaling approach yields lower RMSE for the OCV-SOC model compared to [10] & [14].

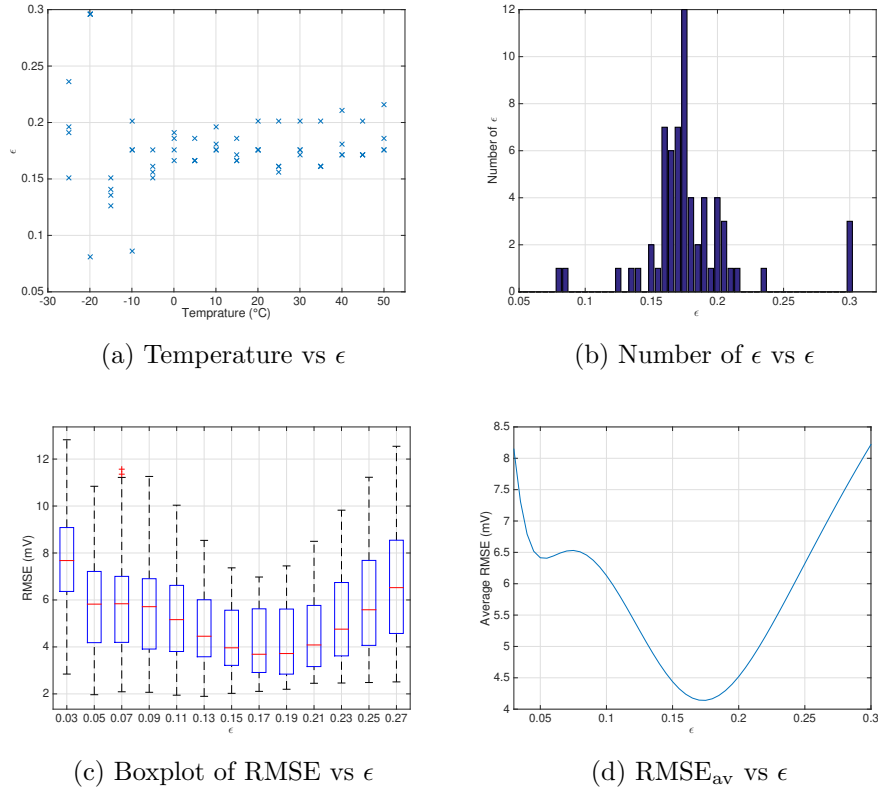


Figure 2.7: **Modeling error vs ϵ .** **Top Left:** ϵ vs Temperature. The ϵ that had the minimum RMSE for each temperature. **Top Right:** ϵ frequency as a function of ϵ . **Bottom Left:** RMSE_{av} vs ϵ . **Bottom Right:** RMSE vs ϵ over all temperatures shown as a box plot (the central (red) mark of the box is median; the edges of the box are 25th and 75th percentile values; the whiskers extend to the most extreme data points not considered outliers, and outliers are plotted individually with a '+').

2.4.3 Multiple Batteries at Multiple Temperatures Results

In this subsection we present the results of using the proposed scaling approach on the following batteries: LG LGIP-530B, Nokia BP-4L, Blackberry RIM FS 1, Blackberry RIM M S1, Samsung AB463651, Samsung EB504465, Samsung EB555157VA and Samsung EBL1A2GBA. The nominal capacities for these batteries vary from 1Ah to 2Ah. Table 2.2 shows the number of cells that were tested at different temperatures.

Figure 2.9 summarizes the $\text{RMSE}_{\text{av}}(\epsilon_i)$ for all the batteries that were used to test the scaling approach. It can be observed that there an ϵ that shows a significantly

Table 2.2: Number of cells tested for each temperature

Temp (°C)	-25	-20	-15	-10	-5	0	5	10	15	20	25	30	35	40	45	50
Samsung EB575152	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
LG LGIP-530B	2	2	1	2	1	2	2	2	2	2	2	2	2	2	3	2
Nokia BP-4L	4	4	4	4	4	4	4	4	4	4	4	4	4	4	3	4
Blackberry RIM FS 1	2	2	1	2	2	2	2	2	2	2	1	2	1	2	1	0
Blackberry RIM M S1	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
Samsung AB463651	1	1	1	1	1	1	4	1	1	1	4	1	4	1	4	0
Samsung EB555157VA	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
Samsung EBL1A2GBA	1	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
Samsung EB504465	4	4	4	3	4	3	4	4	4	4	4	3	4	3	4	2

smaller value of corresponding RMSE_{av} . This shows that choosing the correct scaling factor for your battery is of paramount importance to minimize the modeling error.

Figure 2.10 shows the corresponding box plot for the batteries over all temperatures. It is very clear that the values that ϵ_{opt} not only show the minimum RMSE_{av} but also the smallest variance over all temperatures.

It should be noted that most of the graphs in Figure 2.9 and Figure 2.10 look similar. However, they show slightly different ϵ_{opt} . It can be seen from Figure 2.9 that ϵ_{opt} is within the range of $[0.175, 0.2]$ and it has a corresponding RMSE_{av} within this range $[3, 4.45]$ mV. Additionally, Figure 2.10 shows that the ϵ_{opt} has the smallest RMSE variance over all temperatures.

2.5 State of Charge Tracking with Scaling

The primary application of the OCV-SOC parameter is to estimate the SOC in real time. The authors in [15] presented a literature review on the estimation of the SOC for Li-ion batteries. Many algorithms can be used for SOC estimation and tracking such as: Kalman filter [16], Unscented Kalman filter [17–19], Fading Kalman filter [20], Particle Filter [21] among many others. Out of the many SOC tracking algorithms developed, the Extended Kalman filter (EKF) based approach is well known [16, 22–25]. However, all the EKF approach presented in the past did not

consider scaling. In this section, we re-introduce the EKF based approach to SOC tracking with proper incorporation of scaling and discuss its benefits.

2.5.1 Scaled Extended Kalman Filter

In order to derive the SOC tracking equations, we will start with the Coulomb counting equation (2.4)

$$s[k+1] = s[k] + \frac{\Delta_k i[k]}{3600C_{\text{batt}}} \quad (2.28)$$

where the SOC is assumed to be in $s[k] \in [0, 1]$. First, let us re-write the above by replacing $s[k]$ with the scaled version of SOC $s'[k]$ that is obtained based on (2.20) as

$$\frac{s'[k+1]}{1-2\epsilon} = \frac{s'[k]}{1-2\epsilon} + \frac{\Delta_k i[k]}{3600C_{\text{batt}}} \quad (2.29)$$

where

$$s'[k] = s[k](1-2\epsilon) + \epsilon \quad (2.30)$$

The scaled version of the Coulomb counting equation is then

$$s'[k+1] = s'[k] + (1-2\epsilon) \frac{\Delta_k i[k]}{3600C_{\text{batt}}} \quad (2.31)$$

Now, let us denote the scaled SOC at time k as

$$x_s[k] \triangleq s'[k] \quad (2.32)$$

and write the process model of the EKF as follows

$$x_s[k+1] = x_s[k] + G[k]u[k] + n_i[k] \quad (2.33)$$

where

$$G[k] = \frac{(1 - 2\epsilon)\Delta_k}{3600C_{\text{batt}}} \quad (2.34)$$

$$u[k] = z_i[k] \quad (2.35)$$

and the noise $n_i[k]$ is the current measurement noise which is assumed to be zero-mean Gaussian with standard deviation σ_i . The standard deviation σ_s for the state estimation x_s can be written as

$$\sigma_s = (1 - 2\epsilon) \frac{\Delta_k \sigma_i}{3600C_{\text{batt}}} \quad (2.36)$$

For the battery equivalent circuit model provided in Figure 2.4, the measurement model can be written as

$$z_v[k] = V_o(x_s[k]) + i[k]R_{0,h} + n_v[k] \quad (2.37)$$

where the noise $n_v[k]$ is the voltage measurement noise which is assumed to be zero-mean Gaussian with standard deviation σ_v .

The observation model above is non-linear in terms of the SOC, i.e.,

$$\begin{aligned} V_o(x_s[k]) = & K_0 + \frac{K_1}{x_s[k]} + \frac{K_2}{x_s^2[k]} + \frac{K_3}{x_s^3[k]} + \frac{K_4}{x_s^4[k]} + K_5 x_s[k] \\ & + K_6 \ln(x_s[k]) + K_7 \ln(1 - x_s[k]) \end{aligned} \quad (2.38)$$

Now, the *online SOC tracking problem* can be formally stated as follows: Given $z_v[k]$ and $z_i[k]$, the the voltage and current measurements respectively, at time k , recursively estimate the (scaled) SOC of the battery $\hat{x}_s(k|k)$ and the associated estimation error covariance $P_s(k|k)$. The Algorithm 1 summarizes the extended Kalman filter approach to SOC tracking.

The Algorithm 1 works by taking as an input the previous state $\hat{x}_s[k|k]$, previous

covariance $P_s[k|k]$, current measurement $z_i[k+1]$ and voltage measurement $z_v[k+1]$. It outputs the state $\hat{x}_s[k+1|k+1]$ and covariance estimate $P_s[k+1|k+1]$ using (8) and (9) respectively. In the process, it calculates the state prediction $\hat{x}_s[k+1|k]$ using (1), state prediction variance $P_s[k+1|k]$ using (2) and measurement prediction $\hat{z}[k+1|k]$ using (3). Finally, the innovation variance $S[k+1|k]$, innovation $m[k+1]$ and filter gain $W[k+1]$ are calculated using (5),(6) and (7) respectively.

Algorithm 1

$[\hat{x}_s[k+1|k+1], P_s[k+1|k+1]] =$

EKF-SOC($\hat{x}_s[k|k], P_s[k|k], z_i[k+1], z_v[k+1]$)

- 1: State prediction: $\hat{x}_s[k+1|k] = \hat{x}_s[k|k] + G[k]u[k]$
 - 2: State prediction variance: $P_s[k+1|k] = P_s[k|k] + \sigma_s^2$
 - 3: Measurement prediction: $\hat{z}[k+1|k] = V_o(\hat{x}_s[k+1|k]) + z_i[k+1]R_{0,h}$
 - 4: Linearization of observation model: $h'[k+1] = -\frac{K_1}{\hat{x}_s^2[k+1|k]} - \frac{2K_2}{\hat{x}_s^3[k+1|k]} - \frac{3K_3}{\hat{x}_s^4[k+1|k]} - \frac{4K_4}{\hat{x}_s^5[k+1|k]} + K_5 + \frac{K_6}{\hat{x}_s[k+1|k]} - \frac{K_7}{1-\hat{x}_s[k+1|k]}$
 - 5: Innovation variance: $S[k+1|k] = \sigma_v^2 + h'[k+1]P[k+1|k]h'[k+1]'$
 - 6: Innovation: $m[k+1] = z_v[k+1] - \hat{z}[k+1|k]$
 - 7: Filter gain: $W[k+1] = P[k+1|k]h'[k+1]'S[k+1]^{-1}$
 - 8: State estimate: $\hat{x}_s[k+1|k+1] = \hat{x}_s[k+1|k] + W[k+1]m[k+1]$
 - 9: State estimate variance: $P_s[k+1|k+1] = P_s[k+1|k] - W[k+1]S[k+1]W[k+1]'$
-

2.5.2 Scaled EKF Results

In this subsection, we present a numerical analysis of the proposed EKF algorithm for SOC tracking. The goal of the analysis is to demonstrate the performance of the algorithm with and without scaling. In order to do that, we need data that is free of other uncertainties such as parameter estimation errors. First, we explain how such evaluation data was simulated for two different analyses:

1. *SOC tracking without scaling.* For this analysis we set $\epsilon = 0.001$ allowing the *scaled* SOC to vary between $s' = x_s[k] = 0.001$ and $s' = x_s[k] = 0.999$ — this is almost equivalent to not scaling at all.
2. *SOC tracking with scaling.* For this analysis we set $\epsilon = 0.175$, to its optimal value. This allows the scaled SOC to vary between $s' = x_s[k] = 0.175$ and

$$s' = x_s[k] = 0.825.$$

Under the above two assumptions the data for the analysis is constructed as follows.

1. First, the OCV parameters K_0, \dots, K_7 were estimated based on the assumed ϵ as discussed above. The data for the OCV parameter estimation was collected from Samsung EB575152 battery (this is real OCV characterization data - not simulated one).
2. Then, the estimated OCV parameters, a certain current profile shown in Figure 2.8(a), and the observation model (2.37) is used to simulate the voltage across the battery terminals. The simulated voltage is shown in Figure 2.8(b).

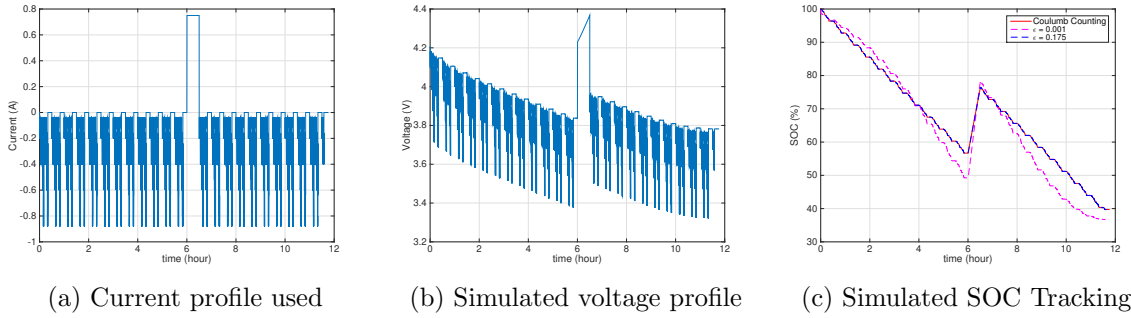


Figure 2.8: **SOC tracking using EKF with and without scaling.** Current and voltage profile used along the the SOC tracking.

The SOC tracking algorithm described in Section 2.5.1, is used to track the SOC of the battery for two different cases: with and without scaling. Figure 2.8(c) shows the SOC tracking results. Since simulated current profile is being used we can apply Coulomb counting to estimate the true SOC.

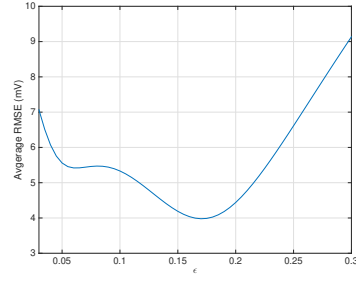
The average load is kept constant implying that the SOC should decrease linearly on an average sense. However, the unscaled version of the SOC estimates are not seen to decrease/increase in a linear fashion. The performance of the scaled version of the SOC tracking satisfies this expectation. The Coulomb counting method provides a

reference SOC because it was computed based on true value of the battery capacity and noiseless current.

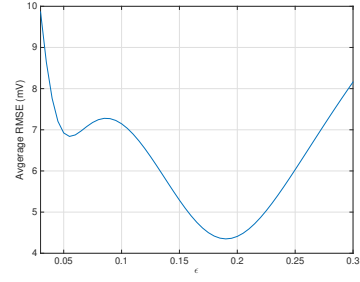
It is clear from this figure that the scaled version of the EKF is superior to the unscaled version and it follows the true SOC compared to the un-scaled version that deviates from the true SOC. The reason for the unstable performance can be traced back to two things. First, the SOC modeling error that is shown in Section 2.4.1 Figure 2.6(c). Second, the numerical instability issue resulting from computing $\log(x_s[k])$ and $1/x_s[k]$ when $x_s[k]$ approaches zero and from computing $\log(1 - x_s[k])$ when $x_s[k]$ approaches one. The proposed scaling scheme effectively avoided this situation in the EKF.

2.6 Conclusions and Discussions

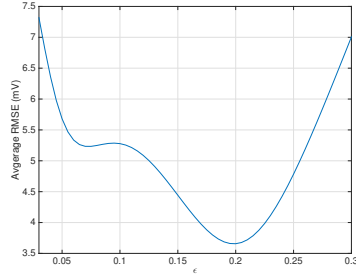
In this chapter we considered the problem of OCV-SOC characterization in Li-ion batteries. OCV-SOC models often employ log and inverse terms that will lead to numerical instability and increased errors during characterization. In this chapter we proposed a solution to this problem using a linear scaling approach. The proposed scaling approach uses a scaling factor ϵ that can be anywhere between 0 and 0.5. We developed an approach to find the optimal value of this parameter that leads to the minimum modeling error and is stable across multiple temperatures. More batteries should be tested to see if the optimal scaling factor found here is consistent with other batteries. We showed that by using the proposed scaling approach the maximum SOC error can be decreased by 9%. The proposed approach in this chapter was tested on multiple batteries at multiple temperatures ranging from -25°C to 50°C . We also used EKF to track the SOC online and showed that estimation error can be significantly decreased by using the appropriate scaling factor ϵ .



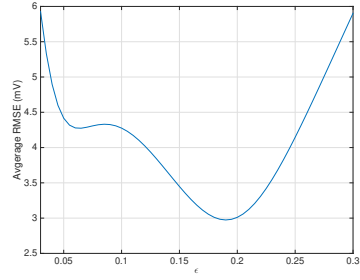
(a) LG LGIP-530B



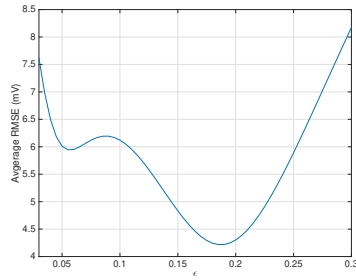
(b) Nokia BP-4L



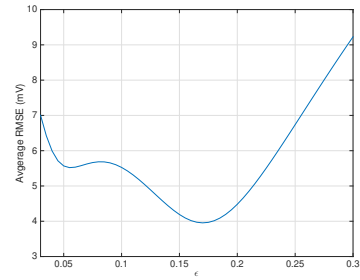
(c) Blackberry RIM FS 1



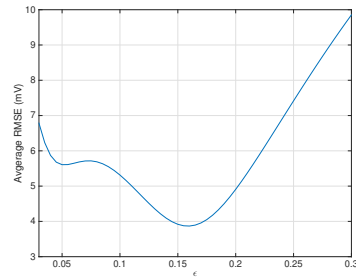
(d) Blackberry RIM M S1



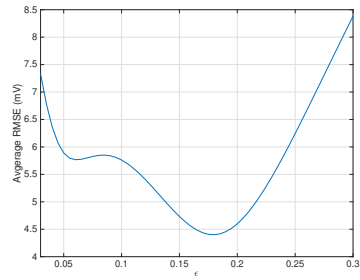
(e) Samsung AB463651



(f) Samsung EB555157VA

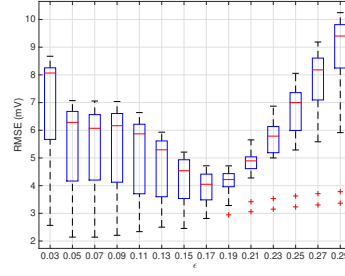


(g) Samsung EBL1A2GBA

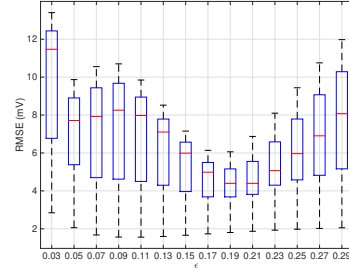


(h) Samsung EB504465

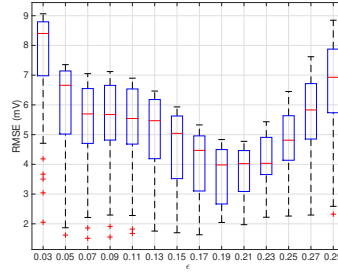
Figure 2.9: **Average RMSE for different Batteries.** Average RMSE vs ϵ for different Li-ion batteries over multiple temperatures and multiple cells as indicated in Table 2.2.



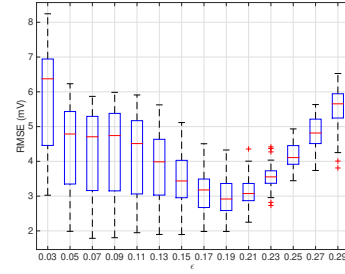
(a) LG LGIP-530B



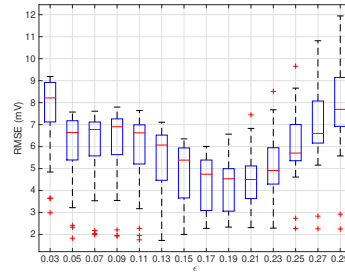
(b) Nokia BP-4L



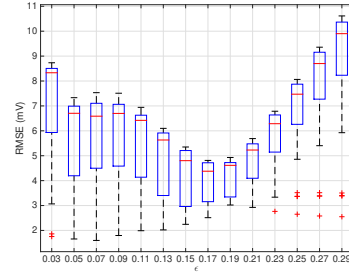
(c) Blackberry RIM FS 1



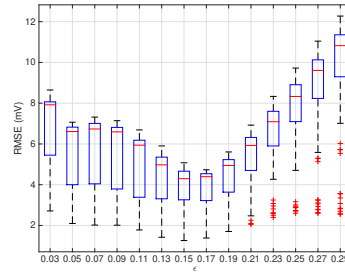
(d) Blackberry RIM M S1



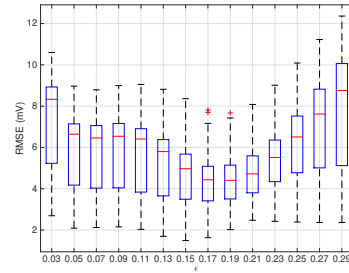
(e) Samsung AB463651



(f) Samsung EB555157VA



(g) Samsung EBL1A2GBA



(h) Samsung EB504465

Figure 2.10: **Boxplot for different batteries at multiple temperatures.** RMSE vs ϵ for multiple temperatures and multiple cells as indicated in Table 2.2. Shown as a box plot (the central (red) mark of the box is median; the edges of the box are 25th and 75th percentile values; the whiskers extend to the most extreme data points not considered outliers, and outliers are plotted individually with a '+'). The best ϵ values are those with small RMSE in terms of mean and variance.

2.7 Bibliography

- [1] N. Nitta, F. Wu, J. T. Lee, and G. Yushin, “Li-ion battery materials: present and future,” *Materials Today*, vol. 18, no. 5, pp. 252 – 264, 2015. pages 9
- [2] A. Al-Haj Hussein and I. Batarseh, “A review of charging algorithms for nickel and lithium battery chargers,” *IEEE Transactions on Vehicular Technology*, vol. 60, pp. 830–838, March 2011. pages 9
- [3] M. A. Roscher, J. Assfalg, and O. S. Bohlen, “Detection of utilizable capacity deterioration in battery systems,” *IEEE Transactions on Vehicular Technology*, vol. 60, pp. 98–103, Jan 2011. pages 9
- [4] M. A. Hannan, M. M. Hoque, A. Hussain, Y. Yusof, and P. J. Ker, “State-of-the-art and energy management system of lithium-ion batteries in electric vehicle applications: Issues and recommendations,” *IEEE Access*, vol. 6, pp. 19362–19378, 2018. pages 9
- [5] G. Avvari, B. Pattipati, B. Balasingam, K. Pattipati, and Y. Bar-Shalom, “Experimental set-up and procedures to test and validate battery fuel gauge algorithms,” *Applied energy*, vol. 160, pp. 404–418, 2015. pages 10
- [6] B. Balasingam, G. Avvari, B. Pattipati, K. Pattipati, and Y. Bar-Shalom, “Robust battery fuel gauge algorithm development, part 3: State of charge tracking,” in *2014 International Conference on Renewable Energy Research and Application (ICRERA)*, pp. 110–115, IEEE, 2014. pages 3, 10
- [7] G. L. Plett, “Extended kalman filtering for battery management systems of lipb-based HEV battery packs: Part 3. State and parameter estimation,” *Journal of Power sources*, vol. 134, no. 2, pp. 277–292, 2004. pages 3, 10

- [8] G. won You, S. Park, and D. Oh, “Real-time state-of-health estimation for electric vehicle batteries: A data-driven approach,” *Applied Energy*, vol. 176, pp. 92 – 103, 2016. pages 10
- [9] B. Balasingam, G. V. Avvari, B. Pattipati, K. Pattipati, and Y. Bar-Shalom, “Robust battery fuel gauge algorithm development, part 2: Online battery-capacity estimation,” in *2014 International Conference on Renewable Energy Research and Application (ICRERA)*, pp. 104–109, Oct 2014. pages xi, 3, 4, 10
- [10] B. Pattipati, B. Balasingam, G. Avvari, K. R. Pattipati, and Y. Bar-Shalom, “Open circuit voltage characterization of lithium-ion batteries,” *Journal of Power Sources*, vol. 269, pp. 317–333, 2014. pages 3, 10, 14, 25
- [11] K. Movassagh, S. A. Raihan, and B. Balasingam, “A novel approach to coulomb counting method for estimating the state-of-charge of batteries,” *IEEE Canada Electrical and Energy Conference, Accepted*, 2019. pages 16
- [12] G. L. Plett, *Battery Management Systems, Volume I: Battery Modeling*. Artech House Publishers, 2015. pages 16, 17
- [13] 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*, vol. 134, no. 2, pp. 262–276, 2004. pages 17
- [14] C. Huang and L. Wang, “Gaussian process regression-based modelling of lithium-ion battery temperature-dependent open-circuit-voltage,” *Electronics Letters*, vol. 53, no. 17, pp. 1214–1216, 2017. pages 25
- [15] R. Zhang, B. Xia, B. Li, L. Cao, Y. Lai, W. Zheng, H. Wang, and W. Wang, “State of the art of lithium-ion battery soc estimation for electrical vehicles,” *Energies*, vol. 11, no. 7, 2018. pages 27

- [16] G. L. Plett, “Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs: Part 1. background,” *Journal of Power sources*, vol. 134, no. 2, pp. 252–261, 2004. pages 27
- [17] Y. Zheng, M. Ouyang, X. Han, L. Lu, and J. Li, “Investigating the error sources of the online state of charge estimation methods for lithium-ion batteries in electric vehicles,” *Journal of Power Sources*, vol. 377, pp. 161 – 188, 2018. pages 27
- [18] C. Zou, C. Manzie, D. Nešić, and A. G. Kallapur, “Multi-time-scale observer design for state-of-charge and state-of-health of a lithium-ion battery,” *Journal of Power Sources*, vol. 335, pp. 121 – 130, 2016. pages 27
- [19] F. Zheng, Y. Xing, J. Jiang, B. Sun, J. Kim, and M. Pecht, “Influence of different open circuit voltage tests on state of charge online estimation for lithium-ion batteries,” *Applied Energy*, vol. 183, pp. 513 – 525, 2016. pages 27
- [20] K. Lim, H. A. Bastawrous, V.-H. Duong, K. W. See, P. Zhang, and S. X. Dou, “Fading kalman filter-based real-time state of charge estimation in lifepo4 battery-powered electric vehicles,” *Applied Energy*, vol. 169, pp. 40 – 48, 2016. pages 27
- [21] B. Xia, Z. Sun, R. Zhang, and Z. Lao, “A cubature particle filter algorithm to estimate the state of the charge of lithium-ion batteries based on a second-order equivalent circuit model,” *Energies*, vol. 10, no. 4, 2017. pages 27
- [22] G. Dong, J. Wei, and Z. Chen, “Kalman filter for onboard state of charge estimation and peak power capability analysis of lithium-ion batteries,” *Journal of Power Sources*, vol. 328, pp. 615 – 626, 2016. pages 27
- [23] R. Xiong, X. Gong, C. C. Mi, and F. Sun, “A robust state-of-charge estimator for multiple types of lithium-ion batteries using adaptive extended kalman filter,” *Journal of Power Sources*, vol. 243, pp. 805 – 816, 2013. pages 27

- [24] J. Wei, G. Dong, and Z. Chen, “On-board adaptive model for state of charge estimation of lithium-ion batteries based on kalman filter with proportional integral-based error adjustment,” *Journal of Power Sources*, vol. 365, pp. 308 – 319, 2017. pages 27
- [25] Z. Wei, K. J. Tseng, N. Wai, T. M. Lim, and M. Skyllas-Kazacos, “Adaptive estimation of state of charge and capacity with online identified battery model for vanadium redox flow battery,” *Journal of Power Sources*, vol. 332, pp. 389 – 398, 2016. pages 27

Chapter 3

Linear State-Space Model Parameter Estimation Using the EM Algorithm

3.1 Introduction

State-space models (SSM) and Kalman filtering have a wide ranging applications: Aerospace systems [1], autonomous vehicles [2], wireless communication (channel estimation) [3], robotics [4], battery state of charge (SoC) estimation [5], chaotic signals [6] [7], computer vision [8], power system state estimation [9], seismology [10], simultaneous localization and mapping [11], and weather forecasting [12], are just few examples. Recently, advances in sensory technology and communication has ushered to the era of internet of things (IoTs) where everything from household items to individual parts of equipment and vehicles are equipped with the technology to continuously collect data. State-space models and Kalman filtering proves to be a useful tool in analyzing these data for effective information fusion and system automation [13]. The Kalman filter provides the best instantaneous estimate to a linear-Gaussian SSM given that the model parameters are known. In many of the emerging applications,

such as IoTs, the underlying physical system is not fully understood, as such deriving an accurate state-space model is not yet feasible; there are also cases where the SSM is partially known.

One approach to solve this problem is to add SSM parameters as additional states to the state vector in a Kalman filter [14]. Another approach is to run different models at the same time and run an adaptive filter that chooses the best model. The authors in [15] proposed a reduced state estimation technique that makes use of a multiple model estimator [14], where an adaptive filtering algorithm chooses the best model from different predefined model dynamics in real time. The generalized pseudo-Bayesian estimator and the interactive multiple model estimators procedure are among the best known examples of this type of method [14]. A survey on this method is given in [16, 17]. However, this approach only performs well in scenarios where the model dynamics are predictable but when applied to models with high uncertainty they don't perform well. Additionally, they often require a long time for computation [18].

The third and final approach, which is the one used in this chapter, works by breaking the problem into an iterative process of estimating the state parameters and estimating the state itself. The first part is to start with an initial guess of what the state parameters are and then estimate the states using a Kalman filter or an extended Kalman filter. The second part is to use these estimates states to estimate the model parameters. With each iteration the estimates should converge to the true value. For this approach it can be formulated as expectation maximization (EM) algorithm proposed in [19, 20]. This chapter serves as a tutorial on using the EM algorithm to estimate unknown state parameters for a SSM. The goal of this chapter is to present the approach and equation in an easy to understand way with all the derivations shown in details so the reader can follow easily. We also test the algorithm on different scenarios and compare the results.

This chapter is structured as follows, section 3.3 is a review of the EM algorithm

and explains in details the steps of the algorithm. In section 3.4 shown how the EM algorithm is applied on a scalar SSM and shows in detail the derivation of the unknown parameters. Furthermore, the EM algorithm is derived for a matrix case of the SSM and the derivation is shown in detail in section 3.5. Then, the discretized continuous white noise acceleration (CWNA) model is presented in section 3.6. Additionally, in section 3.7 we run a statistical test for different scenarios and compare the results, also a comparison between the general form presented in section 3.5 is compared with the CWNA form presented in section 3.6 where a scalar parameter is estimated instead of a matrix. Finally, section 3.8 concludes the chapter.

3.2 Problem Definition

Let us consider a linear state-space model consisting of a process of an $m_x \times 1$ state \mathbf{x}_k and its $m_z \times 1$ observation \mathbf{z}_k summarized by

$$\begin{aligned}\mathbf{x}_{k+1} &= \mathbf{F}\mathbf{x}_k + \mathbf{v}_k \\ \mathbf{z}_k &= \mathbf{H}\mathbf{x}_k + \mathbf{w}_k\end{aligned}\tag{3.1}$$

where \mathbf{F} is an $m_x \times m_x$ state transition matrix, \mathbf{H} is an $m_z \times m_x$ observation matrix, the process and measurement noise $\mathbf{v}(k)$ and $\mathbf{w}(k)$, respectively, are assumed to be zero-mean Gaussian noise vectors with the following covariance matrices

$$\begin{aligned}E\{\mathbf{v}_k \mathbf{v}_k^T\} &= \mathbf{Q} \\ E\{\mathbf{w}_k \mathbf{w}_k^T\} &= \mathbf{R}\end{aligned}\tag{3.2}$$

Given the observation \mathbf{z}_k and the knowledge of the initial state $\mathbf{x}_0 \sim \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$ the Kalman filter [14] produces the best estimate of \mathbf{x}_k if the *model parameters* $\mathbf{F}, \mathbf{H}, \mathbf{Q}$ and \mathbf{R} of the SSM are known.

The objective of this chapter is to estimate the model parameters assuming only

the observation \mathbf{z}_k , The general problem definition of SSM identification can be formally stated as follows: *Given a batch of observations \mathbf{z}_k , $k = 1, 2, \dots, n$, estimate the model parameters $\mathbf{F}, \mathbf{H}, \mathbf{Q}$ and \mathbf{R} as well as the parameters $\boldsymbol{\mu}_0$ and $\boldsymbol{\Sigma}_0$ of the initial state $\mathbf{x}_0 \sim \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$.*

The general SSM identification problem defined above was already solved in [20] using the EM algorithm. In the next two sections we summarize this general SSM identification approach. Then we will present specific state-space models of practical importance and derive their SSM parameters using similar approach.

3.3 Review of the EM Algorithm

In this section, we give a brief review of the Expectation Maximization (EM) algorithm [20–22]. Consider the following scenario where N observations

$$\mathcal{Z} = \{z_i\}_{i=1}^N \tag{3.3}$$

are generated from a set of parameters $\boldsymbol{\Theta}$. The likelihood of the parameter $\boldsymbol{\Theta}$ given the above observations (that is assumed independently and identically distributed – iid) is given by

$$\mathcal{L}(\boldsymbol{\Theta}) = p(\mathcal{Z}|\boldsymbol{\Theta}) = \prod_{i=1}^N p(z_i|\boldsymbol{\Theta}) \tag{3.4}$$

Now, the maximum likelihood estimation of the parameters is given by

$$\hat{\boldsymbol{\Theta}}_{\text{ML}} = \arg \max_{\boldsymbol{\Theta}} \mathcal{L}(\boldsymbol{\Theta}) \tag{3.5}$$

The above optimization in (3.5) is sometimes intractable because the nature of the observation model. The EM algorithm offers an approximate, iterative way to perform this optimization.

The EM algorithm starts by considering that the observed data \mathcal{Z} is *incomplete*; i.e., it assumes that there is a certain information that the observations \mathcal{Z} doesn't have; let us denote this *missing information* as \mathcal{X} . With that, the *complete data* is written as $\mathcal{Z} = (\mathcal{Z}, \mathcal{X})$. Using Bayes' theorem, the joint density function of this complete data can be written as

$$p(z|\Theta) = p(z, x|\Theta) = p(z|x, \Theta)p(x|\Theta) \quad (3.6)$$

Now, instead of writing the likelihood function as a function of Θ , such as $\mathcal{L}(\Theta)$ in (3.4), a new function is defined as follows

$$Q(\Theta, \Theta^{i-1}) = E [\log p(\mathcal{Z}, \mathcal{X}|\Theta)|\mathcal{Z}, \Theta^{i-1}] \quad (3.7)$$

where the expectation is with respect to the probability density function of \mathcal{X} ; as such, the resulting quantity $Q(\Theta, \Theta^{i-1})$ is not a function of \mathcal{X} . In summary, we introduced a variable \mathcal{X} to define the likelihood function and then removed that same variable by taking an expectation. The positive effect of this process is that the resulting $Q(\Theta, \Theta^{i-1})$ is in a form that can be easily maximized.

Similar to the likelihood function $\mathcal{L}(\Theta)$, $Q(\Theta, \Theta^{i-1})$ is dependent on Θ ; in addition, it is also dependent on Θ^{i-1} , the *initial guess* on Θ . Now, the expectation step above is formally written as:

Expectation Step:

$$\begin{aligned} Q(\Theta, \Theta^{i-1}) &= E [\log p(\mathcal{Z}, \mathcal{X}|\Theta)|\mathcal{Z}, \Theta^{i-1}] \\ &= \int_x \log p(\mathcal{Z}, x|\Theta) f(x|\mathcal{Z}, \Theta^{i-1}) dx \end{aligned} \quad (3.8)$$

It must be noted that the probability density of x , $f(x|\mathcal{Z}, \Theta^{i-1})$, is not yet defined; how to find such a density is one of the practical aspects of EM algorithm design. In the next section, we will show how to select (and estimate) such a density using

Gaussian mixture density estimation as an example. It is important to note that such selection will be different for each type of density, such as a Poisson density or Bernoulli density; and it will differ depending on the problem.

Now, the new value of Θ is obtained as:

Maximization Step:

$$\Theta^{(i)} = \arg \max_{\Theta} Q(\Theta, \Theta^{i-1}) \quad (3.9)$$

The important difference between the cost function $\mathcal{L}(\Theta)$ and $Q(\Theta, \Theta^{i-1})$ is that unlike $\mathcal{L}(\Theta)$, $Q(\Theta, \Theta^{i-1})$ can be optimized in a closed form. Starting from an initial guess for Θ^{i-1} , the EM algorithm iterates between the *Expectation Step* and the *Maximization Step* until the estimated parameters converge.

3.4 Scalar State-Space Model

Using usual notations, a scalar SSM is given as follows

$$\begin{aligned} x_{k+1} &= Fx_k + v_k \\ z_k &= Hx_k + w_k \end{aligned} \quad (3.10)$$

where F , H are scalars and the process and measurement noise $v(k)$ and $w(k)$, respectively, are assumed to be zero-mean noise with the following variance

$$\begin{aligned} E\{v_k^2\} &= \sigma_v^2 \\ E\{w_k^2\} &= \sigma_w^2 \end{aligned} \quad (3.11)$$

and the initial state is assumed to be normal, $x_0 \sim \mathcal{N}(\mu_0, \sigma_0)$.

Given a batch of observations z_k , $k = 1, 2, \dots, n$, the objective is to estimate F, H, σ_v^2 and σ_w^2 as well as the initial state mean μ_0 and σ_0 .

3.4.1 Complete Data Likelihood

The joint density of the complete data (i.e., assuming X_n is observed) is written as

$$p(\mathcal{Z}_n, \mathcal{X}_n | \Theta) = p(x_0) \prod_{k=1}^n p(x_k | x_{k-1}) \prod_{k=1}^n p(z_k | x_k) \quad (3.12)$$

where $\Theta = \{\sigma_v, \sigma_w\}$ and

$$\begin{aligned} p(x_0) &= \frac{1}{\sqrt{2\pi}\sigma_0} \exp \left\{ -\frac{(x_0 - \mu_0)^2}{2\sigma_0^2} \right\} \\ p(x_k | x_{k-1}) &= \frac{1}{\sqrt{2\pi}\sigma_v} \exp \left\{ -\frac{(x_k - Fx_{k-1})^2}{2\sigma_v^2} \right\} \\ p(z_k | x_k) &= \frac{1}{\sqrt{2\pi}\sigma_w} \exp \left\{ -\frac{(z_k - Hx_k)^2}{2\sigma_w^2} \right\} \end{aligned} \quad (3.13)$$

Now, the complete-data likelihood of Θ can be written as

$$\begin{aligned} -2 \ln p(\mathcal{X}_n, \mathcal{Z}_n | \Theta) &= c + \ln \sigma_0^2 + \frac{(x_0 - \mu_0)^2}{\sigma_0^2} + n \ln \sigma_v^2 \\ &\quad + \sum_{k=1}^n \frac{(x_k - Fx_{k-1})^2}{\sigma_v^2} + n \ln \sigma_w^2 + \sum_{k=1}^n \frac{(z_k - Hx_k)^2}{\sigma_w^2} \end{aligned} \quad (3.14)$$

where c is a constant.

3.4.2 Expectation

The expectation step is written as

$$Q(\Theta, \Theta^{i-1}) = E \{ \ln p(\mathcal{Z}, \mathcal{X} | \Theta) | \mathcal{Z}, \Theta^{i-1} \} = E \{ -2 \ln p(\mathcal{X}_n, \mathcal{Y}_n | \Theta) \} \quad (3.15)$$

which can be written as

$$\begin{aligned} Q(\Theta, \Theta^{i-1}) &= \ln \sigma_0^2 + \frac{1}{\sigma_0^2} ((x_{0|n} - \mu_0)^2 + P_{0|n}) + n \ln \sigma_v^2 + \frac{1}{\sigma_v^2} (S_{11} - 2FS_{10} + F^2S_{00}) \\ &\quad + n \ln \sigma_w^2 + \frac{1}{\sigma_w^2} \left(\sum_{k=1}^n (z_k - Hx_{k|n})^2 + H^2P_{k|n} \right) \end{aligned} \quad (3.16)$$

where

$$S_{11} = \sum_{k=1}^n (x_{k|n}^2 + P_{k|n}) \quad (3.17)$$

$$S_{10} = \sum_{k=1}^n (x_{k|n}x_{k-1|n} + P_{k,k-1|n}) \quad (3.18)$$

$$S_{00} = \sum_{k=1}^n (x_{k-1|n}^2 + P_{k-1|n}) \quad (3.19)$$

3.4.3 Maximization

To maximize for F we will take the partial derivative of (3.16) with respect to F and equate it to zero. Where we get,

$$\hat{F} = \frac{S_{10}}{S_{00}} \quad (3.20)$$

We then maximize for σ_v^2 , and it will be as follows

$$\hat{\sigma}_v^2 = \frac{1}{n} (S_{11} - 2FS_{10} + F^2S_{00}) \quad (3.21)$$

if we maximize for F , we can substitute (3.20) in (3.21) then we get

$$\hat{\sigma}_v^2 = \frac{1}{n} \left(S_{11} - \frac{S_{10}^2}{S_{00}} \right) \quad (3.22)$$

We then maximize for H and it can be written as

$$\hat{H} = \frac{\sum_{k=1}^n z_k x_{k|n}}{\sum_{k=1}^n x_{k|n}^2 + P_{k|n}} \quad (3.23)$$

We can then maximize for σ_w^2 . It can be shown to be as follows

$$\widehat{\sigma_w^2} = \frac{1}{n} \left(\sum_{k=1}^n (z_k - Hx_{k|n})^2 + H^2 P_{k|n} \right) \quad (3.24)$$

if we maximize for H using (3.23) then

$$\widehat{\sigma_w^2} = \frac{1}{n} \left(\sum_{k=1}^n z_k^2 - \frac{(\sum_{k=1}^n z_k x_{k|n})^2}{\sum_{k=1}^n x_{k|n}^2 + P_{k|n}} \right) \quad (3.25)$$

The mean and covariance of the initial state can be estimated as follows

$$\widehat{\mu}_0 = x_{0|n} \quad (3.26)$$

$$\widehat{\sigma}_0^2 = (x_{0|n} - \mu_0)^2 + P_{0|n} \quad (3.27)$$

(3.27) can be written as follows if we substitute (3.26) in it

$$\widehat{\sigma}_0^2 = P_{0|n} \quad (3.28)$$

3.5 General State-Space Model

3.5.1 Complete Data Likelihood

The joint density of the complete data (i.e., assuming X_n is observed) is written as

$$p(\mathcal{Z}_n, \mathcal{X}_n | \Theta) = p(\mathbf{x}_0) \prod_{k=1}^n p(\mathbf{x}_k | \mathbf{x}_{k-1}) \prod_{k=1}^n p(\mathbf{z}_k | \mathbf{x}_k) \quad (3.29)$$

where $\Theta = \{\mathbf{Q}, \mathbf{R}\}$ and

$$\begin{aligned}
p(\mathbf{x}_0) &= \frac{1}{(2\pi)^{\frac{m}{2}} |\Sigma_0|^{\frac{1}{2}}} \exp \left\{ -\frac{1}{2} (\mathbf{x}_0 - \boldsymbol{\mu}_0)^T \Sigma_0^{-1} (\mathbf{x}_0 - \boldsymbol{\mu}_0) \right\} \\
p(\mathbf{x}_k | \mathbf{x}_{k-1}) &= \frac{1}{(2\pi)^{\frac{m}{2}} |\mathbf{Q}|^{\frac{1}{2}}} \exp \left\{ -\frac{1}{2} (\mathbf{x}_k - \mathbf{F} \mathbf{x}_{k-1})^T \mathbf{Q}^{-1} (\mathbf{x}_k - \mathbf{F} \mathbf{x}_{k-1}) \right\} \\
p(\mathbf{z}_k | \mathbf{x}_k) &= \frac{1}{(2\pi)^{\frac{m}{2}} |\mathbf{R}|^{\frac{1}{2}}} \exp \left\{ -\frac{1}{2} (\mathbf{z}_k - \mathbf{H} \mathbf{x}_k)^T \mathbf{R}^{-1} (\mathbf{z}_k - \mathbf{H} \mathbf{x}_k) \right\}
\end{aligned} \tag{3.30}$$

Now, the complete-data likelihood of Θ can be written as

$$\begin{aligned}
-2 \ln p(\mathcal{X}_n, \mathcal{Z}_n | \Theta) &= c + \ln |\Sigma_0| + (\mathbf{x}_0 - \boldsymbol{\mu}_0)^T \Sigma_0^{-1} (\mathbf{x}_0 - \boldsymbol{\mu}_0) + n \ln |\mathbf{Q}| + n \ln |\mathbf{R}| \\
&+ \sum_{k=1}^n (\mathbf{x}_k - \mathbf{F} \mathbf{x}_{k-1})^T \mathbf{Q}^{-1} (\mathbf{x}_k - \mathbf{F} \mathbf{x}_{k-1}) + \sum_{k=1}^n (\mathbf{z}_k - \mathbf{H} \mathbf{x}_k)^T \mathbf{R}^{-1} (\mathbf{z}_k - \mathbf{H} \mathbf{x}_k)
\end{aligned} \tag{3.31}$$

where c is a constant.

3.5.2 Expectation

The expectation step is written as

$$Q(\Theta, \Theta^{i-1}) = E \{ \ln p(\mathcal{Z}, \mathcal{X} | \Theta) | \mathcal{Z}, \Theta^{i-1} \} = E \{ -2 \ln p(\mathcal{X}_n, \mathcal{Y}_n | \Theta) \} \tag{3.32}$$

which can be written as

$$\begin{aligned}
Q(\Theta, \Theta^{i-1}) &= \ln |\Sigma_0| + \text{tr} \{ \Sigma_0^{-1} [\mathbf{P}_{0|n} + (\mathbf{x}_{0|n} - \boldsymbol{\mu}_0)(\mathbf{x}_{0|n} - \boldsymbol{\mu}_0)^T] \} \\
&+ n \ln |\mathbf{Q}| + \text{tr} \{ \mathbf{Q}^{-1} [\mathbf{S}_{11} - \mathbf{S}_{10} \mathbf{F}^T - \mathbf{F} \mathbf{S}_{10}^T + \mathbf{F} \mathbf{S}_{00} \mathbf{F}^T] \} \\
&+ n \ln |\mathbf{R}| + \text{tr} \{ \mathbf{R}^{-1} [\mathbf{M}_{11} - \mathbf{M}_{10} \mathbf{H}^T - \mathbf{H} \mathbf{M}_{10}^T + \mathbf{H} \mathbf{M}_{00} \mathbf{H}^T] \}
\end{aligned} \tag{3.33}$$

where

$$\mathbf{S}_{11} = \sum_{k=1}^n (\mathbf{x}_{k|n} \mathbf{x}_{k|n}^T + \mathbf{P}_{k|n}) \quad (3.34)$$

$$\mathbf{S}_{10} = \sum_{k=1}^n (\mathbf{x}_{k|n} \mathbf{x}_{k-1|n}^T + \mathbf{P}_{k,k-1|n}) \quad (3.35)$$

$$\mathbf{S}_{00} = \sum_{k=1}^n (\mathbf{x}_{k-1|n} \mathbf{x}_{k-1|n}^T + \mathbf{P}_{k-1|n}) \quad (3.36)$$

$$\mathbf{M}_{11} = \sum_{k=1}^n \mathbf{z}_k \mathbf{z}_k^T \quad (3.37)$$

$$\mathbf{M}_{10} = \sum_{k=1}^n \mathbf{z}_k \mathbf{x}_{k|n}^T \quad (3.38)$$

$$\mathbf{M}_{00} = \sum_{k=1}^n (\mathbf{x}_{k|n} \mathbf{x}_{k|n}^T + \mathbf{P}_{k|n}) \quad (3.39)$$

3.5.3 Maximization

To maximize for \mathbf{F} we will take the partial derivative of (3.16) with respect to \mathbf{F} and equate it to zero. [23] can be used to derive the differentiation of matrices.

By using the fact that \mathbf{Q}^{-1} and \mathbf{S}_{00} are symmetric and using (3.69)(3.70)(3.71) we can show that

$$\frac{\partial Q(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{i-1})}{\partial \mathbf{F}} = -2\mathbf{Q}^{-1}\mathbf{S}_{10} + 2\mathbf{Q}^{-1}\mathbf{F}\mathbf{S}_{00} = 0 \quad (3.40)$$

After premultiplying by \mathbf{Q} we get,

$$\hat{\mathbf{F}} = \mathbf{S}_{10}\mathbf{S}_{00}^{-1} \quad (3.41)$$

We then maximize for \mathbf{Q} . We can use (3.65)(3.66)(3.67)(3.68) to show that

$$\frac{\partial Q(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{i-1})}{\partial \mathbf{Q}} = n\mathbf{Q}^{-1} - \mathbf{Q}^{-1} [\mathbf{S}_{11} - \mathbf{S}_{10}\mathbf{F}^T - \mathbf{F}\mathbf{S}_{10}^T + \mathbf{F}\mathbf{S}_{00}\mathbf{F}^T] \mathbf{Q}^{-1} = 0 \quad (3.42)$$

We will pre and post multiply by \mathbf{Q} ,

$$\hat{\mathbf{Q}} = \frac{1}{n} [\mathbf{S}_{11} - \mathbf{S}_{10}\mathbf{F}^T - \mathbf{F}\mathbf{S}_{10}^T + \mathbf{F}\mathbf{S}_{00}\mathbf{F}^T] \quad (3.43)$$

If we maximize for \mathbf{F} , we can substitute (3.41) in (3.43) then we get

$$\hat{\mathbf{Q}} = \frac{1}{n} [\mathbf{S}_{11} - \mathbf{S}_{10}\mathbf{S}_{00}^{-1}\mathbf{S}_{10}^T] \quad (3.44)$$

We then maximize for \mathbf{H} by noting that \mathbf{R}^{-1} and \mathbf{M}_{00} are symmetric and using (3.69)(3.70)(3.71) and it can be written as

$$\hat{\mathbf{H}} = \mathbf{M}_{10}\mathbf{M}_{00}^{-1} \quad (3.45)$$

We can then maximize for \mathbf{R} using (3.65)(3.66)(3.67)(3.68). It can be shown to be as follows

$$\hat{\mathbf{R}} = \frac{1}{n} [\mathbf{M}_{11} - \mathbf{M}_{10}\mathbf{H}^T - \mathbf{H}\mathbf{M}_{10}^T + \mathbf{H}\mathbf{M}_{00}\mathbf{H}^T] \quad (3.46)$$

if we maximize for \mathbf{H} using (3.45) then (3.46) can be written as

$$\hat{\mathbf{R}} = \frac{1}{n} [\mathbf{M}_{11} - \mathbf{M}_{10}\mathbf{M}_{00}^{-1}\mathbf{M}_{10}^T] \quad (3.47)$$

The initial mean and covariance can be shown to be as follows,

$$\boldsymbol{\mu}_0 = \mathbf{x}_{0|n} \quad (3.48)$$

$$\boldsymbol{\Sigma}_0 = \mathbf{P}_{0|n} + (\mathbf{x}_{0|n} - \boldsymbol{\mu}_0)(\mathbf{x}_{0|n} - \boldsymbol{\mu}_0)^T \quad (3.49)$$

if we estimate the initial mean using (3.48) then the initial covariance can be written

as follows

$$\boldsymbol{\Sigma}_0 = \mathbf{P}_{0|n} \quad (3.50)$$

3.6 CWNA State-Space Model

$$\mathbf{x} = \begin{bmatrix} x & \dot{x} \end{bmatrix}^T \quad (3.51)$$

The state vector above is modeled to undergo the following *process model*

$$\mathbf{x}(k+1) = \mathbf{F}\mathbf{x}(k) + \mathbf{v}(k) \quad (3.52)$$

where the elements of the 2×1 vector $\mathbf{v}(k)$ are assumed to be *zero-mean Gaussian noise*,

$$\mathbf{F} = \begin{bmatrix} 1 & \Delta T \\ 0 & 1 \end{bmatrix} \quad (3.53)$$

where ΔT is the sampling time that is assumed to be a constant. The *process noise covariance*, \mathbf{Q} , can be shown to be [14]

$$\mathbf{Q} = E \{ \mathbf{v}_k \mathbf{v}_k^T \} = \tilde{q} \begin{bmatrix} \frac{1}{3} \Delta T^3 & \frac{1}{2} \Delta T^2 \\ \frac{1}{2} \Delta T^2 & \Delta T \end{bmatrix} = \tilde{q} \bar{\mathbf{Q}} \quad (3.54)$$

where $\bar{\mathbf{Q}}$ is the covariance of the process noise.

$$|\mathbf{Q}| = \tilde{q}^2 \frac{1}{12} \Delta T^4 \quad (3.55)$$

$$\bar{\mathbf{Q}}^{-1} = \tilde{q}^{-1} \begin{bmatrix} 12\Delta T^{-3} & -6\Delta T^{-2} \\ -6\Delta T^{-2} & 4\Delta T^{-1} \end{bmatrix} \quad (3.56)$$

$$z(k) = \mathbf{H}\mathbf{x}(k) + w(k) \quad (3.57)$$

where

$$\mathbf{H} = \begin{bmatrix} 1 & 0 \end{bmatrix} \quad (3.58)$$

where $w(k)$ is assumed to be zero-mean Gaussian noise with standard deviation σ_v .

Consequently, the measurement model covariance is written as

$$\mathbf{R} = E \{ \mathbf{w}_k \mathbf{w}_k^T \} = \sigma_w^2 \quad (3.59)$$

The problem formulation is the same as in Subsection 3.5 except now \mathbf{F} & \mathbf{H} are known. Same steps can be used to reach (3.31) but now it can be written as

3.6.1 Complete Data Likelihood

$$\begin{aligned} -2 \ln p(\mathcal{X}_n, \mathcal{Z}_n | \boldsymbol{\Theta}) = & c + \ln |\boldsymbol{\Sigma}_0| + (\mathbf{x}_0 - \boldsymbol{\mu}_0)^T \boldsymbol{\Sigma}_0^{-1} (\mathbf{x}_0 - \boldsymbol{\mu}_0) + n \ln \left(\tilde{q}^2 \frac{1}{12} \Delta T^4 \right) \\ & + \tilde{q}^{-1} \sum_{k=1}^n (\mathbf{x}_k - \mathbf{F}\mathbf{x}_{k-1})^T \bar{\mathbf{Q}}^{-1} (\mathbf{x}_k - \mathbf{F}\mathbf{x}_{k-1}) + n \ln |\mathbf{R}| \\ & + \sum_{k=1}^n (\mathbf{z}_k - \mathbf{H}\mathbf{x}_k)^T \mathbf{R}^{-1} (\mathbf{z}_k - \mathbf{H}\mathbf{x}_k) \end{aligned} \quad (3.60)$$

3.6.2 Expectation

The expectation step can be written as

$$\begin{aligned}
Q(\Theta, \Theta^{i-1}) = & \ln |\Sigma_0| + \text{tr} \{ \Sigma_0^{-1} [\mathbf{P}_{0|n} + (\mathbf{x}_{0|n} - \boldsymbol{\mu}_0)(\mathbf{x}_{0|n} - \boldsymbol{\mu}_0)^T] \} \\
& + n \ln \left(\tilde{q}^2 \frac{1}{12} \Delta T^4 \right) + \tilde{q}^{-1} \text{tr} \{ \bar{\mathbf{Q}}^{-1} [\mathbf{S}_{11} - \mathbf{S}_{10} \mathbf{F}^T - \mathbf{F} \mathbf{S}_{10}^T + \mathbf{F} \mathbf{S}_{00} \mathbf{F}^T] \} \\
& + n \ln |\mathbf{R}| + \text{tr} \{ \mathbf{R}^{-1} [\mathbf{M}_{11} - \mathbf{M}_{10} \mathbf{H}^T - \mathbf{H} \mathbf{M}_{10}^T + \mathbf{H} \mathbf{M}_{00} \mathbf{H}^T] \}
\end{aligned} \tag{3.61}$$

for the sake of simplification we will use the following equation

$$t = \text{tr} \{ \bar{\mathbf{Q}}^{-1} [\mathbf{S}_{11} - \mathbf{S}_{10} \mathbf{F}^T - \mathbf{F} \mathbf{S}_{10}^T + \mathbf{F} \mathbf{S}_{00} \mathbf{F}^T] \} \tag{3.62}$$

3.6.3 Maximiation

Maximizing for \tilde{q}

$$\frac{\partial Q(\Theta, \Theta^{i-1})}{\partial \tilde{q}} = 2n\tilde{q}^{-1} - \tilde{q}^{-2}t = 0 \tag{3.63}$$

$$\hat{\tilde{q}} = \frac{t}{2n} \tag{3.64}$$

Now \mathbf{Q} can be estimated easily using (3.54), and \mathbf{R} can be estimated using (3.24) since it is scalar.

3.7 Results

In this section, we demonstrate the performance of the proposed algorithms using simulated data. The performance comparison is done in terms of the following measures:

- *True Log-Likelihood:* We calculate the value of the true Log-Likelihood (LLh) $[\ln p(\mathcal{X}_n, \mathcal{Z}_n | \Theta)]$ using (3.14). We substitute in this function the true value of the terms $(x_n, z_n, F, Q, H$ and $\mathbf{R})$ that was used to simulated the data. When

all ground truth are assumed known, the true LLh will be at its highest value.

- *Estimated likelihood*: This is the value of the Log-Likelihood function shown in eq (3.14) $[\ln p(\mathcal{X}_n, \mathcal{Z}_n | \Theta)]$ when we substitute the estimated values of the unknown parameters with the estimated values ($\widehat{x}_n, \widehat{F}, \widehat{Q}, \widehat{H}$ and \widehat{R}). The closer it is to the true LLh value the more accurate our estimation is.
- *Root mean square error (RMSE)*: We calculate the value of the RMSE in the estimated parameters ($\widehat{x}_n, \widehat{F}, \widehat{Q}, \widehat{H}$ and \widehat{R}). This measure was only done for the scalar case.

In this section, we present several numerical studies to evaluate the performance of the SSM identification methods summarized in this chapter. Table 3.3 shows a list of four different scenarios for the evaluation of the proposed algorithm. For this test we assumed that the initial state and covariance are known for simplicity.

Table 3.3: Different SSM scenarios

	F	H	Q	R
Case 1	Known	Known	Est.	Est.
Case 2	Known	Est.	Est.	Est.
Case 3	Est.	Known	Est.	Est.
Case 4	Est.	Est.	Est.	Est.

3.7.1 Scalar SSM

In Figure 3.11 we show the EM algorithm for the four different scenarios listed in Table 3.3 and the corresponding unknowns that were estimated. The value of F is changed from 0.1 to 0.9 by increments of 0.1 i.e., as F increases it reaches the boundary of stability. For each value of F the true LLh value was calculated and compared to the estimated LLh of the four cases listed in Table 3.3. This test was done with a good initialization of the unknown values. The initial values of the unknowns Q, R and H

was chosen randomly from the range $[0.5, 1.5]$ of the true value. While the initial value for F was randomly chosen from the range $[0, 1]$ of the true value. The comparison of the LLh values is shown in the Figure 3.11.

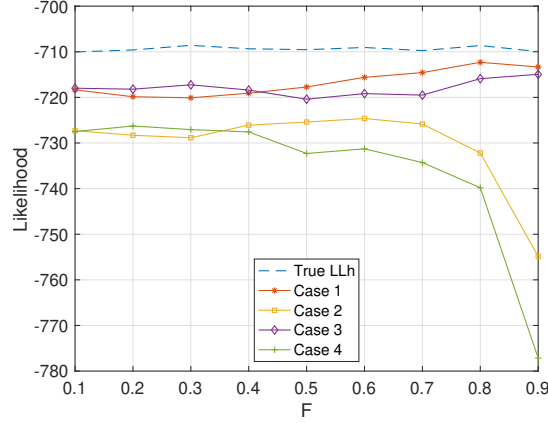


Figure 3.11: **Scalar SSM parameter estimation.** The estimated LLh is shown for different scenarios; the true LLh is shown for comparison; the results are averaged over 1000 Monte-Carlo run with good initialization.

In Figure 3.12 we show the root mean square error (RMSE) in the estimation of the unknown parameters. It is clear from the graph that as the error decreases the estimated LLh gets closer to the true LLh value indicating that our estimate is close to the true value. Figure 3.11 shows that case 2 and case 4 always have lower estimated LLh compared to cases 1 and case 3. The only common thing between case 2 and case 4 that is different from case 1 and case 3 is that H is being estimated, where in case 1 and 3 H is known. As a result, we can conclude that when H is being estimated the performance drops significantly specially when F is closer to the stability boundary. Furthermore, Figure 3.12 shows that the reason for this decline in the performance is due to the significant error in the estimation of Q in case 2 and case 4 as F gets closer to the stability boundary.

The same scenarios were tested again with the only difference of having a worse initialization of the unknown parameters Q , R and H were it was initialized randomly in the range $[0, 5]$ of the true value. However, F had the same initialization as before

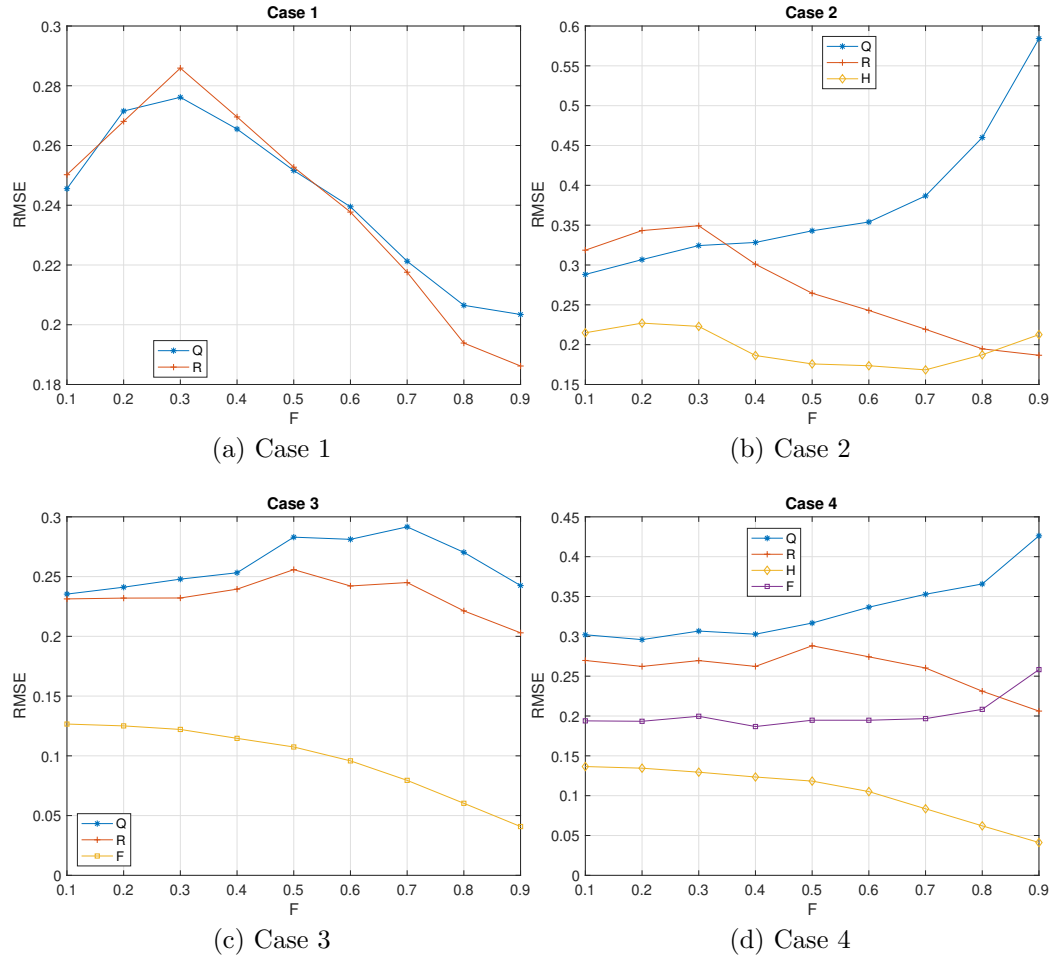


Figure 3.12: **RMSE of our estimate.** Root mean square error (RMSE) in estimates averaged over 1000 Monte-Carlo runs good initialization experiment for all four cases.

to ensure the system stability, this is shown in Figure 3.13. When compare to Figure 3.11 we can see that the two figures have the same pattern where the estimated LLH increases with F for case 1 and case 3, while it drops with F for case 2 and case 4. However, the only difference being that the estimated LLh is less for all cases when compared to its good initialization counterpart. This shows that a better initialization leads to a better estimate and an estimated LLh that is closer to the true LLh. It is also important to mention that around 1% of the data for this run didn't converge and were considered outliers, as a result, they were removed.

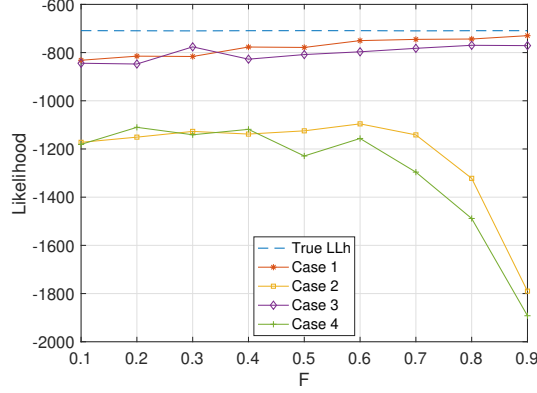
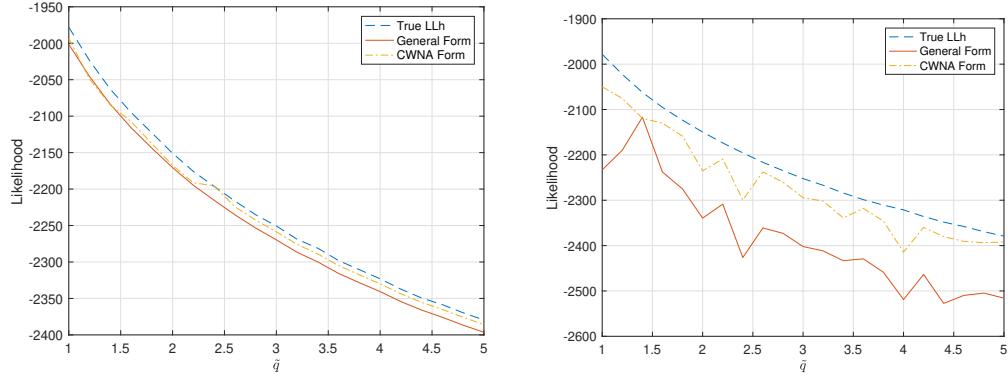


Figure 3.13: **Scalar SSM parameter estimation.** The estimated likelihood is shown for different scenarios; the true likelihood is shown for comparison; the results are averaged over 1000 Monte-Carlo run with bad initialization.

3.7.2 CWNA SSM

In this section we compare between the general matrix form presented in section 3.5 and the CWNA model shown in section 3.6. We apply case 1 for both algorithms where only \mathbf{Q} and \mathbf{R} are being estimated. For the moderate initialization of \mathbf{Q} and \mathbf{R} it was initialized randomly in the range $[2, 3]$ of the true value. While, the range for bad initialization was $[0, 20]$ of the true value.

Figure 3.14 shows the true LLh and estimated LLh for different values of \tilde{q} and compares between the moderate and bad initialization. The figure shows that using the CWNA form where only \tilde{q} is being estimated instead of the whole \mathbf{Q} matrix results in at least equal accuracy if not a better estimate when moderate initialization is used. It can be seen that even when the bad initialization is used, which is a more realistic scenario, the CWNA form is always performing better, as it is always converging faster and more accurately to the true value of \mathbf{Q} and \mathbf{R} . This can be seen in Figure 3.14 (b) where the estimated LLh of the CWNA form is always closer to the True LLh and the estimated LLh of the general form is always further away from the true LLh.



(a) 1000 Monte Carlo runs with moderate initialization (b) 1000 Monte Carlo runs with bad initialization

Figure 3.14: **Matrix SSM case 1.** The estimated likelihood is shown for the matrix form case 1; the true likelihood is shown for comparison; the results are for 1000 Monte-Carlo run.

3.8 Conclusion

In conclusion, we formulated the problem in SSM where a Kalman filter can't be used due to unknown state transition parameters or unknown state and measurement covariance. We proposed the use of the EM algorithm to estimate the unknown system parameters. Additionally, we showed all the steps of the EM algorithm with all the equations that were used to reach the final formulas. Furthermore, we introduced the use of the EM algorithm on the CWNA model and shown how this model can be used to reduce the complexity of the EM algorithm where a scalar quantity is being estimated instead of a matrix. Finally, the EM algorithm was tested on different scenarios for the scalar case. For all four cases of the SSM the RMSE and estimated log-likelihood were reported and their performance compared. Furthermore, the results show that using the CWNA will at least have the same performance accuracy when used if not even higher accuracy compared to using the general form.

3.8.1 Matrix Identities

Here we show some matrix identities that were used through out the chapter; these identities were obtained from [23]. The numbers on the left of the equation is the equation number in the matrix cookbook and the numbers on the right are used to refer to these equations throughout the chapter.

$$(4) \quad (\mathbf{A} + \mathbf{B})^T = \mathbf{A}^T + \mathbf{B}^T \quad (3.65)$$

$$(5) \quad (\mathbf{AB})^T = \mathbf{B}^T \mathbf{A}^T \quad (3.66)$$

$$(57) \quad \frac{\partial \ln |\mathbf{X}|}{\partial \mathbf{X}} = (\mathbf{X}^{-1})^T = (\mathbf{X}^T)^{-1} \quad (3.67)$$

$$(100) \quad \frac{\partial}{\partial \mathbf{X}} \text{tr}(\mathbf{AX}^{-1}\mathbf{B}) = -(\mathbf{X}^{-1}\mathbf{BAX}^{-1})^T \quad (3.68)$$

$$(101) \quad \frac{\partial}{\partial \mathbf{X}} \text{tr}(\mathbf{AXB}) = \mathbf{A}^T \mathbf{B}^T \quad (3.69)$$

$$(104) \quad \frac{\partial}{\partial \mathbf{X}} \text{tr}(\mathbf{AX}^T) = \mathbf{A} \quad (3.70)$$

$$(118) \quad \frac{\partial}{\partial \mathbf{X}} \text{tr}(\mathbf{AXBX}^T\mathbf{C}) = \mathbf{A}^T \mathbf{C}^T \mathbf{XB}^T + \mathbf{CAXB} \quad (3.71)$$

3.8.2 Prediction, Filtering & Smoothing

Here we show below the equations for prediction, filtering then the smoothing respectively.

Prediction:

$$\mathbf{x}_{k|k-1} = \mathbf{F}\mathbf{x}_{k-1|k-1} \quad (3.72)$$

$$\mathbf{P}_{k|k-1} = \mathbf{F}\mathbf{P}_{k-1|k-1}\mathbf{F} + \mathbf{Q} \quad (3.73)$$

Filtering:

$$\mathbf{x}_{k|k} = \mathbf{x}_{k|k-1} + \mathbf{K}_k [\mathbf{z}_k - \mathbf{H}\mathbf{x}_{k|k-1}] \quad (3.74)$$

$$\mathbf{P}_{k|k} = [\mathbf{I} - \mathbf{K}_k \mathbf{H}] \mathbf{P}_{k|k-1} \quad (3.75)$$

Where the kalman gain is \mathbf{K}_k is given as

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{F}^T [\mathbf{F} \mathbf{P}_{k|k-1} \mathbf{F}^T + \mathbf{R}] \quad (3.76)$$

Smoothing:

$$\mathbf{x}_{k-1|n} = \mathbf{x}_{k-1|k-1} + \mathbf{J}_{k-1} [\mathbf{x}_{k|n} - \mathbf{x}_{k|k-1}] \quad (3.77)$$

$$\mathbf{P}_{k-1|n} = \mathbf{P}_{k-1|k-1} + \mathbf{J}_{k-1} [\mathbf{P}_{k|n} - \mathbf{P}_{k|k-1}] \mathbf{J}_{k-1}^T \quad (3.78)$$

$$\mathbf{J}_{k-1} = \mathbf{P}_{k-1|k-1} \mathbf{F}^T [\mathbf{P}_{k|k-1}]^{-1} \quad (3.79)$$

Finally, the Lag-one covariance smoother can be used to derive $\mathbf{P}_{k,k-1|n}$ using the following equations

$$\mathbf{P}_{n,n-1|n} = [\mathbf{I} - \mathbf{K}_n \mathbf{H}] \mathbf{F} \mathbf{P}_{n-1|n-1} \quad (3.80)$$

$$\mathbf{P}_{k-1,k-2|n} = \mathbf{P}_{k-1|k-1} \mathbf{J}_{k-2}^T + \mathbf{J}_{k-1} [\mathbf{P}_{k,k-1|n} - \mathbf{F} \mathbf{P}_{k-1|k-1}] \mathbf{J}_{k-2}^T \quad (3.81)$$

3.9 Bibliography

- [1] Z. Guang, L. Yuyang, and B. Xingzi, “Conservative term constrained kalman filter for autonomous orbit determination,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 54, pp. 783–793, April 2018. pages 39
- [2] W. Zhang, Z. Wang, C. Zou, L. Drugge, and M. Nybacka, “Advanced vehicle state monitoring: Evaluating moving horizon estimators and unscented kalman

- filter,” *IEEE Transactions on Vehicular Technology*, vol. 68, pp. 5430–5442, June 2019. pages 39
- [3] L. Reggiani, L. Dossi, L. Barletta, and A. Spalvieri, “Extended kalman filter for mimo phase noise channels with independent oscillators,” *IEEE Communications Letters*, vol. 22, pp. 1200–1203, June 2018. pages 39
- [4] S. Tully and H. Choset, “A filtering approach for image-guided surgery with a highly articulated surgical snake robot,” *IEEE Transactions on Biomedical Engineering*, vol. 63, pp. 392–402, Feb 2016. pages 39
- [5] M. Partovibakhsh and G. Liu, “An adaptive unscented kalman filtering approach for online estimation of model parameters and state-of-charge of lithium-ion batteries for autonomous mobile robots,” *IEEE Transactions on Control Systems Technology*, vol. 23, pp. 357–363, Jan 2015. pages 39
- [6] C. Pantaleon and A. Souto, “Comments on ”an aperiodic phenomenon of the extended kalman filter in filtering noisy chaotic signals”,” *IEEE Transactions on Signal Processing*, vol. 53, pp. 383–384, Jan 2005. pages 39
- [7] H. Leung, Zhiwen Zhu, and Zhen Ding, “An aperiodic phenomenon of the extended kalman filter in filtering noisy chaotic signals,” *IEEE Transactions on Signal Processing*, vol. 48, pp. 1807–1810, June 2000. pages 39
- [8] S. Y. Chen, “Kalman filter for robot vision: A survey,” *IEEE Transactions on Industrial Electronics*, vol. 59, pp. 4409–4420, Nov 2012. pages 39
- [9] J. Zhao, A. Gómez-Expósito, M. Netto, L. Mili, A. Abur, V. Terzija, I. Kamwa, B. Pal, A. K. Singh, J. Qi, Z. Huang, and A. P. S. Meliopoulos, “Power system dynamic state estimation: Motivations, definitions, methodologies, and future work,” *IEEE Transactions on Power Systems*, vol. 34, pp. 3188–3198, July 2019. pages 39

- [10] X. Deng and Z. Zhang, “Automatic multihorizons recognition for seismic data based on kalman filter tracker,” *IEEE Geoscience and Remote Sensing Letters*, vol. 14, pp. 319–323, March 2017. pages 39
- [11] S. A. Holmes, G. Klein, and D. W. Murray, “An $o(n^2)$ square root unscented kalman filter for visual simultaneous localization and mapping,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, pp. 1251–1263, July 2009. pages 39
- [12] M. Dhanya and A. Chandrasekar, “Improved rainfall simulation by assimilating oceansat-2 surface winds using ensemble kalman filter for a heavy rainfall event over south india,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, pp. 7721–7726, Dec 2014. pages 39
- [13] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, “Internet of things: A survey on enabling technologies, protocols, and applications,” *IEEE Communications Surveys Tutorials*, vol. 17, pp. 2347–2376, Fourthquarter 2015. pages 39
- [14] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons, 2004. pages 40, 41, 51
- [15] P. Mookerjee and F. Reifler, “Reduced state estimator for systems with parametric inputs,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 40, pp. 446–461, April 2004. pages 40
- [16] E. Mazor, A. Averbuch, Y. Bar-Shalom, and J. Dayan, “Interacting multiple model methods in target tracking: a survey,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, pp. 103–123, Jan 1998. pages 40

- [17] X. Rong Li, Youmin Zhang, and Xiaorong Zhi, “Multiple-model estimation with variable structure. iv. design and evaluation of model-group switching algorithm,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 35, pp. 242–254, Jan 1999. pages 40
- [18] A. Zia, T. Kirubarajan, J. P. Reilly, D. Yee, K. Punithakumar, and S. Shirani, “An em algorithm for nonlinear state estimation with model uncertainties,” *IEEE Transactions on Signal Processing*, vol. 56, pp. 921–936, March 2008. pages 40
- [19] A. P. Dempster, N. M. Laird, and D. B. Rubin, “Maximum likelihood from incomplete data via the em algorithm,” *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 1–38, 1977. pages 40
- [20] R. H. Shumway and D. S. Stoffer, “Time series regression and exploratory data analysis,” in *Time Series Analysis and its Applications*, pp. 47–82, Springer, 2011. pages 40, 42
- [21] G. McLachlan and T. Krishnan, *The EM Algorithm and Extensions*, vol. 382. John Wiley & Sons, 2007. pages 42
- [22] J. A. Bilmes *et al.*, “A gentle tutorial of the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models,” *International Computer Science Institute*, vol. 4, no. 510, p. 126, 1998. pages 42
- [23] K. B. Petersen and M. S. Pedersen, *The Matrix Cookbook*. November 15, 2012. pages 49, 59

Chapter 4

Future Work

This chapter shows how a battery management system (BMS) can use the EM algorithm to estimate the OCV-SOC model and battery capacity online. First the EM algorithm is shown for a general state space model (SSM). Then, it is applied to our BMS application by using its equations.

The SSM with a control input can be written as follows,

$$x_k = Fx_{k-1} + G_1U_k + v_k \quad (4.1)$$

$$z_k = h(x_k) + a^Tb + w_k \quad (4.2)$$

we can assume that the initial state is known and assume white gaussian noise v_k and w_k .

$$p(x_k|x_{k-1}) = \frac{1}{\sqrt{2\pi}\sigma_v} \exp\left(\frac{-(x_k - Fx_{k-1} - G_1U_k)^2}{2\sigma_v^2}\right) \quad (4.3)$$

$$p(z_k|x_k) = \frac{1}{\sqrt{2\pi}\sigma_w} \exp\left(\frac{-(z_k - h(x_k) - a^Tb)^2}{2\sigma_w^2}\right) \quad (4.4)$$

The log-likelihood equation can be written as follows:

$$\begin{aligned}
-2 \ln p(x_n, y_n | \Theta) = c &+ n \ln \sigma_v^2 + \sum_{k=1}^n \frac{(x_k - Fx_{k-1} - G_1 U_k)^2}{2\sigma_v^2} \\
&+ n \ln \sigma_w^2 + \sum_{k=1}^n \frac{-(z_k - h(x_k) - a^T b)^2}{2\sigma_v^2}
\end{aligned} \quad (4.5)$$

The expectation step:

$$\begin{aligned}
Q(\Theta, \Theta^{i-1}) &= E\{-2 \ln p(x_n, y_n | \Theta)\} = c + n \ln \sigma_v^2 \\
&+ \frac{1}{\sigma_v^2} \left(S_{11} - 2FS_{10} + F^2 S_{00} + \sum_{k=1}^n (2Fx_{k-1|n} G_1 U_k + G_1^2 U_k^2 - 2G_1 U_k x_{k|n}) \right) \\
&+ n \ln \sigma_w^2 + \frac{1}{\sigma_w^2} \left[\sum_{k=1}^n (z_k - h(x_{k|n}) - a^T b)^2 + h'(x_k) P_{k|n} h'(x_k) \right]
\end{aligned} \quad (4.6)$$

where h' is $\frac{\partial h(x)}{\partial x} |_{\hat{x}_{k|n}}$

If we apply (4.1) & (4.2) to the BMS problem where the state being tracked (x) is the state of charge then

$$\begin{aligned}
F &= 1 \\
G_1 &= \frac{1}{3600 C_{batt}} \\
U_k &= \Delta_k i_k \\
h(x_k) &= k_0 + \frac{k_1}{x_k} + \frac{k_2}{x_k^2} + \frac{k_3}{x_k^3} + \frac{k_4}{x_k^4} + k_5 x_k + k_6 \ln(x_k) + k_7 \ln(1 - x_k) \\
a^T &= \begin{bmatrix} v d_{k-1} & -v d_{k-2} & i_k & i_{k-1} & -i_{k-1} \end{bmatrix} \\
b &= \begin{bmatrix} \alpha & \beta & R_0 & \check{R}_1 & \check{R}_2 \end{bmatrix}^T \\
h'(x_k) &= -\frac{k_1}{x_k^2} - \frac{2k_2}{x_k^3} - \frac{3k_3}{x_k^4} - \frac{4k_4}{x_k^5} + k_5 + \frac{k_6}{x_k} - \frac{k_7}{1 - x_k}
\end{aligned}$$

see [1] for further information on $v d_{k-1}$, $-v d_{k-2}$, i_k , i_{k-1} , $-i_{k-1}$, α , β , R_0 , \check{R}_1 & \check{R}_2 .

The maximization step:

$$\frac{\partial Q(\Theta, \Theta^{i-1})}{\partial G_1} = \frac{1}{\sigma_v^2} \sum_{k=1}^n (-2U_k x_{k|n} + 2F x_{k-1|n} U_k + 2G_1 U_k^2) = 0 \quad (4.7)$$

$$= \sum_{k=1}^n -U_k x_{k|n} + F x_{k-1|n} U_k = \sum_{k=1}^n -2G_1 U_k^2 \quad (4.8)$$

$$\sum_{k=1}^n G_1 U_k = \sum_{k=1}^n x_{k|n} - F x_{k-1|n} \quad (4.9)$$

$$G_1 = \frac{\sum_{k=1}^n x_{k|n} - F x_{k-1|n}}{\sum_{k=1}^n U_k} \quad (4.10)$$

using 4.7 we can write 4.9 as follows,

$$\frac{1}{3600 C_{batt}} = \frac{\sum_{k=1}^n x_{k|n} - x_{k-1|n}}{\sum_{k=1}^n \Delta_k i_k} \quad (4.11)$$

$$C_{batt} = \frac{\sum_{k=1}^n \Delta_k i_k}{3600 \sum_{k=1}^n x_{k|n} - x_{k-1|n}} \quad (4.12)$$

$$C_{batt} = \frac{\sum_{k=1}^n \Delta_k i_k}{3600(x_{n|n} - x_{0|n})} \quad (4.13)$$

Same approach can be used to find $h(x)$. This part is left as the future work along with testing this algorithm on simulated and real data.

4.1 Bibliography

- [1] B. Balasingam, G. Avvari, B. Pattipati, K. Pattipati, and Y. Bar-Shalom, "A robust approach to battery fuel gauging, part i: Real time model identification," *Journal of Power Sources*, vol. 272, pp. 1142 – 1153, 2014. pages 65

Chapter 5

Conclusion

This thesis composed of three chapters. The first chapter served as an introduction to battery management system (BMS) and its components. It also provided a brief literature review on the current topics of research in the BMS domain.

Chapter 2 introduced a novel scaling approach that can be employed to further enhance the OCV-SOC characterization by reducing the modeling error. In this chapter a novel scaling approach was presented and tested on nine different Li-ion batteries. All the batteries showed similar results with an optimal scaling factor around 1.75. Additionally, the scaling approached showed that it can minimize the state of charge (SOC) modeling error by 9%. Furthermore, the coulomb counting equation were adjusted to count for the scaling factor and a novel extended Kalman filter is presented for online SOC estimation where the SOC estimation error was reduced by around 8% with the scaling approach.

In chapter 3, the SSM problem where a Kalman filter can't be used due to unknown state transition parameters or unknown state and measurement covariance was introduced. The expectation maximization (EM) algorithm was proposed to solve this problem. The EM algorithm was applied on the CWNA model and shown how this model can be used to reduce the complexity of the EM algorithm where a

scalar quantity is being estimated instead of a matrix. Furthermore, the EM algorithm was tested on different scenarios for the scalar case. For all four cases of the SSM the RMSE and estimated log-likelihood were reported and their performance compared. Furthermore, the results show that using the CWNA will at least have the same performance accuracy when used if not even higher accuracy compared to using the general form.

Finally, Chapter 4 presents future work based on this thesis.

Vita Auctoris

NAME:	Mostafa Shaban Mohamed Ahmed
PLACE OF BIRTH:	Assiout, Egypt
YEAR OF BIRTH:	1995
EDUCATION:	Manaret Al Farouk IGCSE, Cairo, Egypt 2013, High School German University in Cairo, Cairo, Egypt 2018, Bachelor of Science Electronics Engineering University of Windsor, Windsor, Ontario 2019, Master of Applied Science Electrical and Computer Engineering