12-22-2014

Advanced Adaptive Prognostication Algorithms with Application to BMS, Automotive, and Aerospace Systems

Bharath Pattipati
University of Connecticut - Storrs, bharath.pattipati@uconn.edu

Advanced Adaptive Prognostication Algorithms with Application to BMS, Automotive, and Aerospace Systems

Bharath Pattipati, Ph.D.

University of Connecticut, 2015

This thesis aims to solve several important problems in engineering. Four fundamental areas of research have been examined: (i) Battery Management Systems; (ii) Battery Fuel Gauging; (iii) Condition-based Maintenance; and (iv) Prognosis in Coupled Systems. The prognostication algorithms developed have been validated on data collected from either real-world or hardware-in-the-loop experiments or both. The approaches proposed are modular and have the potential to be applicable to a wide variety of complex systems, ranging from portable applications to automotive and aerospace systems.

Advanced Adaptive Prognostication Algorithms with Application to BMS, Automotive, and Aerospace Systems

Bharath Pattipati

B.E., M.S. Ramaiah Institute of Technology, 2005

M.S., University of Connecticut, 2009

A Dissertation

Submitted in Partial Fullfilment of the

Requirements for the Degree of

Doctor of Philosophy

at the

University of Connecticut

2015

APPROVAL PAGE

Doctor of Philosophy Dissertation

Advanced Adaptive Prognostication Algorithms with Application to BMS, Automotive, and Aerospace Systems

Presented by

Bharath Pattipati

Major Advisor

Krishna R. Pattipati

Associate Advisor

Yaakov Bar-Shalom

Associate Advisor

Shengli Zhou

Associate Advisor

Balakumar Balasingam

University of Connecticut

2015

This thesis is dedicated to my Mom, *Padmavathi*, Dad, *Mahendra*, and Brother, *Karthik*, for their unwavering love, support, and encouragement over the years

ACKNOWLEDGEMENTS

One of the joys of completing graduate school is to look over the journey past and remember all the friends, family, researchers, and professors who have helped and supported me along this long but fulfilling road.

First and foremost, I would like to convey my profound gratitude and respect to my major advisor and mentor, *Prof. Krishna R. Pattipati*, none of this would have been possible without his time, support, encouragement, attention to detail, and expertise which have guided me throughout my graduate studies. My sincere thanks to *Prof. Yaakov Bar-Shalom* for his advice, guidance, witty remarks, and insightfulness of which only he is capable. I am grateful to *Prof. Shengli Zhou* for his help, support, and advice. To *Dr. Balakumar Balasingam*, a heartfelt thanks for all the help, it was a joy working with him.

To other faculty at the Electrical Department: *Prof. Rajeev Bansal*, *Prof. Peter Luh*, *Prof. Peter Willett*, *Asst. Prof. Ashwin Dani*, *Asst. Prof. Shalabh Gupta*, *Prof. Faquir Jain*, a sincere thanks for all the guidance over the years.

A big thank you to each and every member of the systems and optimization lab (Cyberlab) of past and present, it was a pleasure working with you all. To the current crop of cyberlab minions: *Ali*, *Bala*, *Chen*, *David*, *Devaki*, *Diego*, *Lingyi*, *Manisha*, *Niranjan*, *Pujitha*, *Rajeev*, *Sravant*, and *Vinod*; I will forever treasure the moments we shared and

the chats we had. You made me a better person and the workplace cheerful. A special mention to the prior cyberlab students: *Anuradha*, *Bill*, *Chaitanya*, *Chulwoo*, *Kihoon*, *Satnam*, *Suvasri*, *Woosun*, and *Xu*; I learned a lot from you guys and thank you from the bottom of my heart.

A special thanks to my friends from back home in India: *Arjun*, *Ajeya*, *Tirtho* and other childhood friends. You made me the person I am today.

A very warm thank you to *Missie* for her understanding and true friendship; her boys, *Dakota* and *Thane* for ligting up my life; and her parents, *Butch* and *Lisa*, for their moral support and encouragement.

A special mention to all the staff at the Electrical Department and BECAT: *Mary*, *Dee*, *Paul*, *Celine*, *Karen*, *Marvy*, *Sandy*, and *Tina*. A wholehearted thanks to you all for help with the documentation and paperwork over the years.

I gratefully acknowledge the researchers at the different organizations we have collaborated with on projects during my masters and PhD: *Madhavi*, *Chigusa*, and *Danil* from Toyota Technical Center; *Mark*, *Youssef*, *Yilu*, and *Mutasim* from GM R&D; *Jon Christopherson* from INL; *Mike*, *Travis*, *Brian*, *Tai-sik*, and *Steve* from Fairchild Semiconductor.

And last, but not the least, I thank all the people whom I have interacted with over the years and have made this research endeavor possible.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

21^{st} century engineering problems involve a complex range of challenges and mandate creative robust solutions. Rapid advances in electronics, control, communication, and computing technologies have led to the development of integrated and smart systems spanning automotive, avionics, buildings, manufacturing, consumer electronics, healthcare, emergency response, and defense systems.

Current automotive vehicles contain more than 70 distributed electronic control units (ECUs), 100s of Megabytes of software, 5 or more distinct communication networks, a wide variety of sensors and actuators, and 1000s of data and control signals exchanged in real-time every second.

Although designed to sustain long life, systems degrade in performance due to gradual development of anomalies and unanticipated faults. Failures may range from issues that affect a single hardware or software subsystem, to issues that occur as a result of coupling among multiple subsystems. In addition, system usage and operating conditions (e.g., weather, road surfaces in the context of an automobile) may lead to different failure modes that can affect the performance of a complex system. A ma-

jor goal for safe and reliable operation of systems is to develop robust analytical tools, based on an information ensemble generated from integrated sensor-based (also referred to as data-driven), model-based and knowledge-based approaches, which enable early diagnosis and prognosis (D&P) of incipient faults. The objective is to facilitate condition-based maintenance and timely actions to reduce the probability of failures without significantly compromising quality and performance. Hence, advanced detection, diagnosis and prognosis techniques are needed that can adapt to varying usage and operating conditions to infer and track the faults in complex systems. These, in turn, contribute to life-cycle cost minimization by reducing service and inventory costs, providing effective maintenance through timely repair actions, and by judiciously extending the life of expensive equipment (e.g., engines and batteries in a hybrid electric vehicle).

Battery technology has come a long way since the invention of the first voltaic cell in the 1800s. Because of the increased interest in electric/hybrid electric vehicles, a battery management system (BMS) has become one of the chief components in an automobile. The goals of BMS are to maximize the run-time per discharge cycle, as well as the number of life cycles attainable for the life of the battery. Automotive battery management is very demanding, because it has to work in real time in rapidly varying chargedischarge conditions as the vehicle accelerates and brakes, as well as work in a harsh and uncontrolled environment.

The rapid advancement of portable electronic devices such as smart phones,

notebooks and tablet computers requires the design of a robust and accurate battery management system that integrates one or more of the functions such as cell protection, charge-discharge control, determining the state of charge (SOC), state of health (SOH), and remaining useful life (RUL) of the battery, battery capacity monitoring, remaining run-time estimation, monitoring and storing historical data, etc. The need for the online estimation of such prognostic information with acceptable error under all operating conditions over the entire lifetime of a battery requires radical re-thinking towards developing new approaches. Precise estimation of prognostic information requires the consideration of key factors such as charge-discharge rates, temperature, age, self-discharge, hysteresis, and open-circuit voltage (OCV) recovery effects. The key objectives of this research can be summarized as follows:

1. Adaptive algorithms for accurately estimating the pivotal characteristics of batteries

2. Fault diagnosis, prognosis, condition-based maintenance, and prognostic health management

1.1 List of Publications

1.1.1 Conference Papers

1. **B. Pattipati**, K. Pattipati, J. P. Christopherson, S. M. Namburu, D. V. Prokhorov, and L. Qiao, "Automotive Battery Management Systems", IEEE AUTOTESTCON, pages $581 - 586$, September, 2008. [Cited by 48]

2. **B. Pattipati**, C. Sankavaram, K. Pattipati, Y. Zhang, M. Howell, and M. Salman, "Multiple Model Moving Horizon Estimation Approach to Prognostics in Coupled Systems", IEEE AUTOTESTCON, pages $149-157$, September, 2011. [*Walter E. Petersen Award for Best New Technology Paper*]

3. **B. Pattipati**, K. Pattipati, Y. A. Ghoneim, M. Howell, and M. Salman, "Electronic Returnless Fuel System Fault Diagnosis and Isolation: A Data-Driven Approach", PHM Society, 2013.

4. **B. Pattipati**, B. Balasingam, A. Abdollahi, G. V. Avvari, K. R. Pattipati, and Y. Bar-Shalom, "Integrated Battery Fuel Gauge and Optimal Charger", IEEE AUTOTESTCON, July, 2014.

5. **B. Pattipati**, B. Balasingam, G. V. Avvari, K. Pattipati, and Y. Bar-Shalom, "A Normalized Approach to Open Circuit Voltage Characterization of Lithium-ion Batteries"', International Conference on Renewable Energy Research and Applications (ICRERA), July 2014.

6. C. Sankavaram, **B. Pattipati**, A. Kodali, K. Pattipati, M. Azam, S. Kumar, M. Pecht, "Model-based and data-driven prognosis of automotive and electronic systems", IEEE International Conference on Automation Science and Engineering (CASE), pages $96 - 101$, August 2009.

7. C. Sankavaram, **B. Pattipati**, K. Pattipati, Y. Zhang, M. Howell, M. Salman, "Data-driven fault diagnosis in a hybrid electric vehicle regenerative braking system", IEEE Aerospace Conference, pages $1 - 11$, March 2012.

8. B. Balasingam, **B. Pattipati**, C. Sankavaram, K. Pattipati, and Y. Bar-Shalom, "An EM approach for dynamic Battery Management Systems", pages $2110 - 2117$, Proceedings of 15^{th} International Conference on Information Fusion, Singapore, July 2012.

9. B. Balasingam, G. V. Avvari, **B. Pattipati**, K. R. Pattipati and Y. Bar-Shalom, "A Robust Approach to Online Parameter Estimation in Li-ion Batteries", International Conference on Renewable Energy Research and Applications (ICRERA), July 2014.

10. B. Balasingam, G. V. Avvari, **B. Pattipati**, K. R. Pattipati and Y. Bar-Shalom, "A Robust Approach to Capacity Estimation in Li-ion Batteries", International Conference on Renewable Energy Research and Applications (ICRERA), July 2014.

11. B. Balasingam, G. V. Avvari, **B. Pattipati**, K. R. Pattipati and Y. Bar-Shalom, "A

Robust State of Charge Tracking Approach for Li-ion Batteries", International Conference on Renewable Energy Research and Applications (ICRERA), July 2014.

1.1.2 Journal Papers

1. **B. Pattipati**, C. Sankavaram, K. R. Pattipati, "System Identification and Estimation Framework for Pivotal Automotive Battery Management System Characteristics", IEEE Transactions on Systems, Man and Cybernetics, Part C: Applications and Reviews, vol. 41, issue 6, pages $869-884$, November, 2011. [Cited by 61]

2. **B. Pattipati**, C. Sankavaram, K. Pattipati, Y. Zhang, M. Howell, and M. Salman, "Multiple Model Moving Horizon Estimation Approach to Prognostics in Coupled Systems", IEEE AESS, vol. 28, issue 3, pages $4-12$, April, 2013.

3. **B. Pattipati**, B. Balasingam, G. V. Avvari, K. R. Pattipati, Y. Bar-Shalom, "Open Circuit Voltage Characterization of Lithium-ion Batteries", Journal of Power Sources, volume 269, pages $317-333$, December, 2014.

4. G. V. Avvari, **B. Pattipati**, B. Balasingam, K. R. Pattipati, and Y. Bar-Shalom, "Battery Fuel Gauge Hardware-In-the-Loop Validation on Li-ion Batteries", *submitted to IEEE Transactions on Instrumentations and Measurements*, May 2014.

5. C. Sankavaram, **B. Pattipati**, K. R. Pattipati, Y. Zhang, and M. Howell, "Fault Diagnosis in Hybrid Electric Vehicle Regenerative Braking System", IEEE Acess, vol. 2, pp.$1225-1239$, 2014.

6. B. Balasingam, G. V. Avvari, **B. Pattipati**, K. R. Pattipati, Y. Bar-Shalom, "A

robust approach to Battery Fuel Gauging, Part I: real time model identification", Journal of Power Sources, vol. 272, pages $1142 - 1153$, December, 2014.

7. B. Balasingam, G. V. Avvari, **B. Pattipati**, K. R. Pattipati, Y. Bar-Shalom, "A robust approach to Battery Fuel Gauging, Part II: real time capacity estimation", Journal of Power Sources, vol. 269, pages $949 - 961$, December, 2014.

8. B. Balasingam, G. V. Avvari, **B. Pattipati**, K. R. Pattipati, Y. Bar-Shalom, "A robust approach to Battery Fuel Gauging, Part III: State of Charge tracking", *submitted to Journal of Power Sources*, 2014.

9. A. Abdollahi, N. Raghunathan, **B. Pattipati**, X. Han, B. Balasingam, K. R. Pattipati, Y. Bar-Shalom, B. Card, "Optimal Battery Charging, Part III: Capacity Modeling and Optimal Control Setting for Maximizing the Useful Life of Battery", *submitted to Journal of Power Sources*, May, 2014.

1.1.3 U.S. Patents

1. B. Balasingam, B. French, Y. Bar-Shalom, **B. Pattipati**, K. Pattipati, J. Meacham, T. Williams, G. V. Avvari, T-S Hwang, U.S. Patent Number US $2014/0244193$ A1, Battery State of Charge Tracking, Equivalent Circuit Selection and Benchmarking, August 28^{th}, 2014.

2. B. Balasingam, B. French, Y. Bar-Shalom, **B. Pattipati**, K. Pattipati, J. Meacham, T. Williams, G. V. Avvari, T-S Hwang, U.S. Patent Number US $2014/0244225$ A1, Battery State of Charge Tracking, Equivalent Circuit Selection and Benchmarking, August 28^{th}, 2014.

Chapter 2

Automotive Battery Management Systems

The Battery management system is an integral part of an automobile. It protects the battery from damage, predicts battery life and maintains the battery in an operational condition. The Battery management system performs these tasks by integrating one or more of the functions, such as protecting the cell, thermal management, controlling the charge/discharge, determining the state of charge (SOC), state of health (SOH), and remaining useful life (RUL) of the battery, cell balancing, data acquisition, communication with on-board and off-board modules as well as monitoring and storing historical data. In this chapter, we propose a Battery management system that estimates the critical characteristics of the battery (such as SOC, SOH, and RUL) using a data-driven approach. Our estimation procedure is based on a modified Randles circuit model consisting of resistors, capacitor, and Warburg impedance for Electrochemical Impedance Spectroscopy test data, and a lumped parameter model for Hybrid Pulse Power Characterization test data. The resistors in a Randles circuit model usually characterize the self-discharge and internal resistance of the battery, the capacitor generally represents the charge stored in the battery, and the Warburg impedance represents the diffusion

phenomenon. The Randles circuit parameters are estimated using a frequency-selective non-linear least squares estimation technique, while the lumped parameter model parameters are estimated by the prediction error minimization method. We investigate the use of support vector machines to predict the capacity fade and power fade, which characterize the State of Health of a battery, as well as estimate the State of Charge of the battery. An alternate procedure for estimating the power fade and energy fade from low-current Hybrid Pulse Power characterization test data using the lumped parameter battery model has been proposed. Predictions of Remaining Useful Life of the battery are obtained by support vector regression of the power fade and capacity fade estimates. Survival function estimates for reliability analysis of the battery are obtained using a hidden Markov model trained using time-dependent estimates of capacity fade and power fade as observations. The proposed framework provides a systematic way for estimating relevant battery characteristics with a high-degree of accuracy.

2.1 Introduction

Battery technology has come a long way since the invention of the first voltaic cell in the 1800s. Because of the increased interest in hybrid vehicles, a Battery Management System (BMS) has become one of the chief components in an automobile. The goals of BMS are to maximize both the run-time per discharge cycle, as well as the number of life cycles attainable for the life of the battery [18]. Automotive battery management is very demanding, because it has to work in real-time in rapidly varying charge-discharge

conditions as the vehicle accelerates and brakes, as well as work in a harsh and uncontrolled environment. In addition, it must interface with other on-board systems, such as the engine management, climate controls, communications, and safety systems.

The function of a BMS in hybrid electric vehicles (HEV) and plug-in hybrid's (PHEV) are multifaceted. They include monitoring the conditions of individual cells which make up the battery, maintaining all the cells within their operating limits, protecting the cells from out-of-tolerance conditions, compensating for any imbalances in cell parameters within the battery chain, providing information about the State of Charge (SOC), State of Health (SOH), and Remaining Useful Life (RUL) of the battery, providing the optimum charging algorithm, responding to changes in the vehicle operating mode and so on.

The main motivation for this chapter is to develop a systematic procedure for estimating three critical characteristics of a battery, viz., State of charge (SOC), State of Health (SOH), and Remaining Useful Life (RUL). Important questions such as, how long a battery lasts before it fails, how much charge is left in the battery, performance of a battery to complete a certain task, how much time a battery is charged before use, the different degradation processes affecting the battery, and so on, are different paradigms encapsulated in these critical battery characteristics. Knowing the amount of charge left in a battery compared to when it was fully charged gives the user an indication of how much longer a battery will continue to perform before it needs recharging. The State of Charge (SOC), a measure of remaining capacity in the battery, is used to ensure

optimum control of the charging/discharging process. During the lifetime of a battery, its "health" deteriorates gradually due to irreversible physical and chemical changes which take place with usage and age until eventually the battery is no longer usable or dead. The SOH is an indication of the point which has been reached in the life cycle of the battery and a measure of its condition relative to a fresh battery. The State of Health (SOH) is a measure of a battery's capability to deliver its specified output. This is vital for assessing the readiness of emergency power equipment, and is an indicator of whether maintenance actions are needed. We will employ capacity fade and power fade as measures of State of Health (SOH) of a battery. The ability to accurately predict the Remaining Useful Life (RUL) is the key to proactive, condition-based maintenance of batteries.

2.1.1 Previous Research

Critical battery characteristics and their estimation

Key factors affecting the SOC are the charge-discharge rates, hysteresis, temperature, cell age and self-discharge due to internal resistance of the battery [67]. SOC is not directly measurable and needs to be inferred from other measurements in the battery system. Several methods for estimating the SOC of a battery have been used. Some are specific to particular cell chemistries. Most depend on measuring some convenient parameter, which varies with the SOC. Some of the prominent techniques for estimating the SOC are summarized in Table 2.1 below.

Techniques	Application Fields	Advantages	Drawbacks
Look-up tables	NiCd	Comparison between two batteries (one with known SOC)	*Offline*[1], Sensitive to battery and operating conditions
Current sharing method	Lithium, NiMH, Lead	Easy to implement, Less computation time	Sensitive to battery and operating conditions
Discharge Test	Used for capacity determination at the beginning of life	Easy and accurate; Independent of SOH	*Offline*[1], time intensive, modifies the battery state, loss of energy
Physical Properties of Electrolyte (Density, Concentration, Color etc.)	Lead, ZnBr, Va	*Online*[1]; Information about SOH	Sensitive to temperature and impurities, Error due to acid stratification
Coulomb counting	All battery systems, most applications	Accurate if enough re-calibration points are available and with good current measurements	Sensitive to parasite reactions, needs regular re-calibration points
Open-circuit voltage (OCV)	Lead, lithium, ZnBr	*Online*[1], cheap, OCV prediction	Needs long rest time (current = 0)
Coup de Fouet	Lead	Estimating battery capacity	Sensitive to battery and operating conditions
Linear Model	Lead, photovoltaic	*Online*[1], easy to implement	Needs reference data for fitting parameters
Impedance Spectroscopy	All battery systems	Gives information about SOH and quality	Temperature sensitive,cost sensitive
Internal resistance	Lead, Lithium, NiCd	Gives information about SOH; possibility of online measurements	Good accuracy, but only for a short interval

Artificial Neural Networks	All battery systems	*Online*[1]	Needs training data of a similar battery, Back-propagation algorithms are slow to converge
Fuzzy Logic	All battery systems	*Online*[1]	Definition of membership functions are highly subjective, does not scale to large models
Kalman Filters	All battery systems, photovoltaic, dynamic applications such as HEV	*Online*[1]; dynamic	Difficult to implement the filtering algorithm that considers all features such as non-normalities and nonlinearities

Table 2.1: Summary of Different Techniques for the Determination of SOC along with their Applications, Advantages and Drawbacks [87], [92].

Pop *et al.* [91] proposed a real-time SOC evaluation system for Li-ion batteries combining direct measurement of electromotive force (EMF) during the equilibrium state and coulomb counting during the charge/discharge states.

Kim *et al.* [59] implemented a sliding mode observer for robust tracking under nonlinear conditions based on a simple RC battery model. The performance was validated by the urban dynamometer driving schedule test and the SOC error was less than 3% for most cases. In a more recent work [60], he proposed a dual-sliding-mode ob-

[1] Online refers to the processing of the data piece-by-piece in a sequential manner without having it available from the start, whereas, in Offline the entire data is available for the algorithm.

server consisting of a fast time-varying observer to estimate the SOC, terminal voltage and polarization effects, and a slow time-varying observer to estimate the SOH in terms of capacity fade and resistance deterioration.

Determining the SOH of a battery adaptively is important for optimal energy management and on-board diagnostics for efficient operation of HEVs and PHEVs. Generally, SOH is used to deduce how well the battery system is functioning relative to its nominal (rated) and end (failed) states [129]. The SOH reflects the general condition of a battery and its ability to deliver the specified performance in comparison to a new battery. It takes into account factors, such as charge acceptance, internal resistance, voltage, and self-discharge. Hence, knowing changes in the SOH with time may be viewed as enabling one to assess the increase in irreversible losses that is inherent in the aging of batteries [129].

In practice, SOH is estimated from a single measurement of either the cell impedance or the cell conductance. In pursuit of accuracy, others advocate measuring several cell parameters, all of which vary with the age of the battery, and estimate SOH from a combination of these factors. Examples are capacity, internal resistance, self-discharge, charge acceptance, discharge capabilities, the mobility of electrolyte and cycle counting, if possible. The absolute readings will depend on the cell chemistry involved. Weighting is added to individual factors based on experience, the cell chemistry and the importance of the particular parameter in the application for which the battery is used. If any of these variables provide marginal readings, the end result will be af-

fected. A battery may have a good capacity, but the internal resistance maybe high. In this case, the SOH estimation will be lowered accordingly. Similar demerit points are added, if the battery has high self-discharge or exhibits other chemical deficiencies. The points scored for the cell are compared with the points assigned to a new cell to give a percentage result or figure of merit.

An alternative method of specifying the SOH is to base the estimation on the usage history of the battery rather than on some measured parameter. The number of charge-discharge cycles completed by the battery is an obvious measure, but this does not necessarily take into account any extreme operating conditions experienced by the battery, which may have affected its functionality. It is however possible to record the duration of any periods during which the battery has been subject to abuse from out-of-tolerance voltages, currents or temperatures, as well as the magnitude of the deviations. From this, the SOH can be determined by using a weighted average of the measured parameters.

Bhangu *et al.* [14] employed extended Kalman filters to estimate the variation in bulk capacitance, which characterizes the ability of the battery to store charge over time, indicating the loss of cell capacity and, consequently, a means for monitoring SOH.

Do *et. al.* [36] proposed a real-time identification method based on an extended Kalman filter for observation of the parameters of the modified Randles circuit battery model. He suggested an increase in the number of parallel RC-circuits to model

the Warburg impedance and improve the low frequency performance. However, experimentation on Gen 2 Li-ion cell data via the modified Randles scheme with more than one parallel RC-circuit, leads to ill-conditioning (over-parameterization) of the optimization problem. In addition, extended Kalman filters have divergence issues, problems of initialization, and the covariance matrix tends to be too-optimistic.

Vebrugge *et al.* [128] concluded that a regressed high-frequency resistance (from weighted recursive least-squares), which must be normalized to nominal values and is a function of both temperature and SOC, is not sensitive to erroneous values of SOC and that an adaptive acquisition of the high-frequency resistance of the battery can be used to assess the SOH. This increase in the high-frequency resistance can be anticipated for all commercially viable PHEV and HEV batteries upon extensive cycling.

Goser *et al.* [49] derived a neuro-fuzzy scheme in which a Kohenen Self-Organizing Map (SOM) is used to train the data in a fuzzy system. The SOM is an unsupervised clustering technique that seeks to organize the data. By using a SOM to feed the data into a fuzzy system, the system is made both reactive and adaptive (i.e. it unifies both neural networks and fuzzy systems). The architecture of the system implemented by Goser *et al.* [49] is as follows: The fuzzification is split into two layers, leading to a rule base and then subsequently to defuzzification. This two layer network is then trained in part by using neural network algorithms. Buchmann *et al.* [22] implemented a neuro-fuzzy based battery management system that can track the battery SOH in three minutes. The neural network was trained on fuzzified data, and the outputs are then de-

fuzzified.

Saha *et al.* [110] proposed a Bayesian framework (based on Relevance Vector Machines and Particle Filters) for predicting the SOC, SOH, and RUL, and for providing uncertainty bounds based on correlations between battery performance (capacity) and model parameters (resistance). However, the battery model used for estimating the battery characteristics was simplistic because the low-frequency Warburg impedance was merely represented by a resistance. Moreover, this resistance is in series with the charge-transfer resistance making it unidentifiable i.e., they cannot be uniquely estimated. Furthermore, the Particle filter framework incorporates exponential growth models for predicting the internal battery parameters (resistance), and is CPU-intensive.

Conventional battery management system approaches have mostly focused on addressing the estimation of SOC or SOH with limited attention to RUL. Consequently, we propose a general framework for estimating these critical characteristics (SOC, SOH, and RUL) and validate it using Li-ion battery data.

Key Battery modeling approaches

Several battery models have been proposed in the literature, which are briefly summarized in Table 2.2. An appropriate trade-off between the model complexity and performance is a key aspect in battery modeling. In this chapter, we focus on equivalent circuit battery models consisting of voltage and/or current sources, resistors, capacitors (or constant-phase element, CPE), inductors, and/or Warburg impedance.

Table 2.2: Battery Models and Key Features [53].

Broad Classification	Types of Models	Key Features
Equivalent Circuit Battery Models	Thevenin electrical model	Usually consists of a series resistor and an RC network to predict the battery response to transient signals at a particular SOC, by assuming that the open-circuit voltage [OCV(SOC)] is a constant.
	Impedance based electrical models	Employ the method of electrochemical impedance spectroscopy to obtain an AC-equivalent impedance model in the frequency domain, and then use an equivalent network to fit the impedance spectra.
	Runtime electrical models	Runtime models use a complex circuit to simulate battery runtime and DC voltage response for a constant discharge current in SPICE-compatible simulators.
	Linear electric models	In its general form, the linear battery model consists of an OCV applied across a voltage source and a capacitor. In series with this is a network of 3 capacitors and 3 resistors to model overcharge. In parallel to all these elements is a self-discharge resistor.
	Nonlinear electric models	The circuit consists of a series connection of two parallel Resistor-Capacitor (RC) networks, an OCV in parallel across the RC networks, and a resistor in series with the RC network. The elements of the circuit are functions of voltage, and one of the resistors in the RC networks is temperature-dependent. To account for variations between the charge and discharge states; the series resistance and the resistor in the RC networks connected to the series resistance, are modeled as a parallel combination of two diodes, and the internal resistance is computed as the sum of these two resistances.
Electrochemical Battery Models	Peukert Equation	Discharge current of a battery decreases with increasing "constant current" discharge time.
	Shepherd Model Equation	Describes the electrochemical behavior of the battery directly in terms of voltage and current.Often used in conjunction with Peukert equation to obtain battery voltage and state of charge given power draw variations.
	Modifications to Shepherd Model: Lindstorm Model	Consists of adding terms to describe certain aspects of battery performance.
	Wood Model	Adds an Improved internal resistance calculation. Incorporates secondary equations to describe overcharging and gas generation, along with a self-discharge term.
	Unnewehr Universal Model	Simplification of Shepherd Equation. Models the variation in internal resistance with respect to SOC.
Fractional Discharge Battery Models		Measures the battery incremental state of charge in energy terms. Relationship between power-density and energy density is usually obtained through a series of constant power-discharge tests and modeled as a quadratic function.
Dynamic Lumped parameter Battery Model		Based on a static battery model that describes the instantaneous terminal voltage as a function of steady charge/discharge current.
Other Battery Models	Hydrodynamic Model	Analogy between the charging and discharging of a battery with the hydrodynamic filling and draining of a double tank reservoir.
	Finite Element Type Models	Devised for battery simulations Some divide each cell into a number of finite elements while others use each cell as an individual element.
Models based on tabulated data		Tables of battery performance are used to interpolate battery parameters over a wide range of conditions.
Modeling Specific Factors Affecting Battery Performance	Temperature Models	Impact of temperature on battery capacity is modeled: Parameters of the model are temperature-dependent. Including the temperature coefficient into the general equation for battery capacity.
	State of Charge	Some models use the linear relationship between the OCV and SOC whereas others use SOC as a state in the model.
	Cycle History	Logarithmic plots of Depth-of-discharge versus cycles to failure for various battery types.
	Battery Age	Corrosion is the main component behind decreased battery performance by age.

Electrochemical models [35], [55] are typically time-consuming and are best suited for understanding the physical design aspects of electrodes and electrolyte.

Peukert's equation [37] is only applicable to batteries discharged at constant temperature and constant discharge current. When applied to a battery with a variable discharge rate and changing operating temperature, it generally results in an underestimation of the remaining capacity. Specifically, Lithium-ion batteries, whose capacity is strongly dependent on the battery temperature, which in turn is a function of duty cycle, battery design and environmental conditions, the Peukert equation is not applicable.

Similarly, the Shepherd equation [117] has three major shortcomings. The model parameters could only be calculated from the experimental discharge curves with constant current. Subsequently, the applicability of this model appeared to be limited to temperature range from 10^0C to 30^0C, and finally, the aging effect on the battery model parameters had not been considered. Modifications to the equation have been proposed in the literature to improve certain aspects of battery performance, but limitations still remain.

Other battery models such as hydrodynamic models and finite element type battery models are complex and CPU-intensive requiring high memory and significant computation time.

In this chapter, equivalent-circuit battery models have been employed as they are more intuitive in circuit simulations, robust to operating conditions, provide real-time implementation, and enable the battery nonlinearities to be modeled using circuit

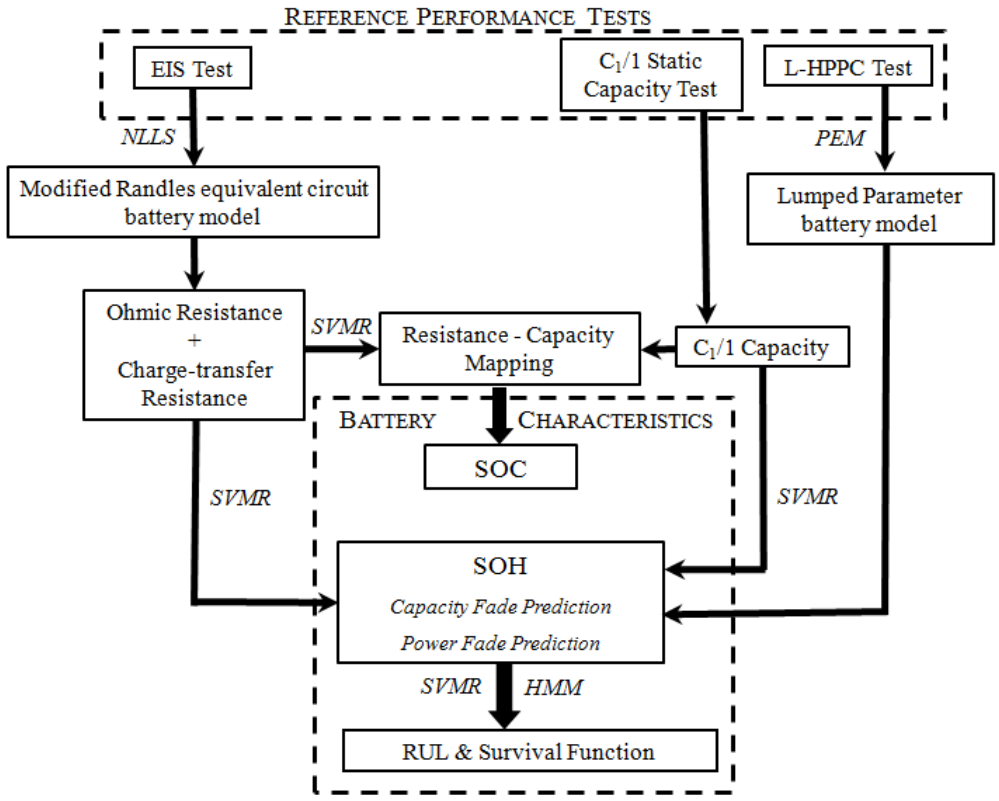


Fig. 2.1: SOC, SOH, and RUL Estimation Framework

parameters such as resistors, capacitors, inductors, etc.

2.1.2 Data for Battery Modeling and Characterization

The validation data used in this chapter had been collected as part of the Advanced Technology Development (ATD) program, where performance testing was conducted on second-generation lithium-ion cells (i.e., Gen 2 cells) [54]. The 18650-size Gen 2 cells, with baseline and variant chemistry, were cycle-life tested at Idaho National Laboratory (INL), only baseline cells data was used in this chapter. Reference Performance Tests (RPTs) are a set of tests performed at periodic intervals during life testing

to establish the condition and rate of performance degradation of batteries under test. These tests should be performed *prior to the beginning of life (BOL) testing, at defined periodic intervals,* and *at the end of life (EOL) testing* of all batteries undergoing either cycle life testing or calendar life testing.

Life testing was interrupted every 4 weeks (i.e., 33600 cycle-life profiles) for reference performance tests (RPTs) consisting of $C_1/1$ [2] static capacity test, a low-current hybrid pulse power characterization (L-HPPC) test, and electrochemical impedance spectroscopy (EIS) test, at 60% SOC and 25^0C [54]. The static capacity test measures the battery capacity in ampere-hours at a constant-current discharge rate corresponding to the manufacturer's rated C_1 capacity in ampere-hours. Discharge is terminated on a manufacturer-specified discharge voltage limit. After the static capacity tests, all cells were subjected to electrochemical impedance spectroscopy (EIS) test. EIS was used to determine impedance changes in the electrode-electrolyte interface. EIS measurements were conducted by discharging the cells from a fully-charged state to the specified open circuit voltage (OCV) corresponding to 60% SOC, followed by an 8- to 12- hour rest at OCV (to reach electrochemical equilibrium), and then the impedance is measured over a frequency range of 10 kHz to 0.01 Hz [43], [90]. The EIS measurements consisted of 61 data points in the frequency range; computed using a constant AC voltage (5mV) and measuring the current response.The Hybrid Pulse Power Characterization

[2] $C_1/1$ Rate: The current corresponding to the rated capacity in ampere-hours for a 1-hour discharge. For example, if the battery's rated 1-hour capacity is 5 Ah, then the $C_1/1$ is 5 A [67].

(HPPC) Test is intended to determine the dynamic power capability over the battery's useable charge and voltage range using a test profile that incorporates both discharge and regen pulses [43]. Characterization testing was conducted at 25^0C with $C_1/1$ static capacity tests to establish performance parameters, such as capacity, resistance, power and energy. These tests consist of a constant-current discharge from a fully charged state using a fraction of the rated capacity defined at the 1-hour rate. The Baseline cell degradation generally increased with increasing test temperature, SOC, and test time. The chemistry of Gen 2 baseline cell used in the data analysis of this chapter may be found in [54]. The battery ratings and limitations used for testing are shown in Table III.

2.1.3 Scope and Organization of the Chapter

Our approach to estimating the SOC, SOH and RUL based on EIS data employs a modified Randles circuit model of a battery. This model consists of a resistor at high frequency, a parallel RC circuit for modeling the charge transfer phenomenon at medium frequencies, and a Warburg impedance to model the diffusion phenomenon at low frequencies. The circuit parameters are estimated from the EIS data using non-linear least squares (NLLS) estimation techniques. In order to overcome the *ill-conditioning* associated with the NLLS problem, we divide the estimation process into high-frequency region, charge-transfer region, and diffusion region. The long-term temporal variations of battery resistance are modeled by an auto-regressive support vector machine. Ex-

Table 2.3: Gen 2 Baseline Cells Ratings and Limitations [54].

$C_1/1$ rated capacity	1.0 Ah
$C_1/1$ nominal capacity (average)	0.961 Ah
Cell nominal weight (average)	38.5 g
Battery Size Factor	553
Electrode area	846.3 cm^2
Temperature	
Operating Range	-20^0C to $+60^0C$ (discharge)
Storage	$10^0C \pm 3^0C$
Voltage Limits	
Minimum discharge voltage	3 V (18-sec pulse), 3 V (continuous)
Maximum charge voltage	4.3 V (10-sec pulse), 4.1 V(continuous)
HPPC Calculation Voltages	
Maximum (V_{MAX})	4.1 V
Minimum (V_{MIN})	3 V
Current Limits	
Maximum discharge current	8 A (18-sec pulse), 2 A (continuous)
Maximum charge current	8 A (10-sec pulse), 1 A (continuous)

ploiting the linear correlation between the battery resistance and $C_1/1$ capacity, the nonlinear SVM models are used to forecast the battery's SOC, as well as the capacity fade and the power fade. An alternative estimation procedure based on USABC's (U.S. Advanced Battery Consortium) lumped parameter model (LPM) [43] for predicting the power fade and energy fade from low-current Hybrid Pulse Power characterization (L-HPPC) test data is also proposed to validate the estimates from EIS data. Unlike the method in [43], our parameter estimation technique for the LPM model, viz., prediction error minimization (PEM) method, ensures that the relevant circuit parameters are non-negative. The estimates of capacity fade and power fade, in turn, are used to estimate the Remaining Useful Life (RUL) of the battery via moving average SVMR. An important advantage of using SVMR is that the determination of the model parameters corresponds to a convex optimization problem, and hence, the solutions are globally optimal (given the assumptions) and unique along with being sparse and having simple geometric interpretations. Unlike Artificial Neural Networks, the computational complexity of SVMR does not depend on the dimensionality of the input space and are less prone to over-fitting. Survival functions and RUL estimates of the battery are also predicted via a hidden Markov Model (HMM) trained using time-dependent power fade and capacity fade estimates. The HMM provides a further validation of our RUL estimates. The estimation framework proposed in this chapter is summarized in Figure 2.2.

The primary contributions of this chapter are four fold: (i) frequency-selective

NLLS estimation to overcome ill-conditioning associated with the modified Randles circuit parameters from the EIS data, (ii) constrained optimization to ensure that the identified LPM parameters are non-negative, (iii) novel auto-regressive SVMR-based capacity and power fade estimation, and (iv) SVMR and HMM-based RUL estimation.

The chapter is organized as follows. Section II discusses our approaches for identifying the battery model parameters. Section III describes procedures for estimating the SOC, SOH and RUL characteristics of batteries based on the models identified in Section II. Section IV validates our estimation approach on second-generation lithium-ion cell (i.e. Gen 2 cell) data collected at the Idaho National Laboratory (INL). Section V concludes with a summary and future research directions.

2.2 Battery Modeling

2.2.1 Identification of impedance-based battery model

In a previous work, Kuhn *et al.* proposed a modified Randles scheme [63] shown in Figure 2.2 as an equivalent circuit model of a battery.

Here, the Warburg impedance Z_W represents the diffusion phenomenon; R_{HF} (high-frequency resistance) denotes the electrolyte and connection resistances, and the $R_{tc}||C_{dl}$ parallel circuit models the charge-transfer phenomenon. Hence, the frequency-dependent impedance expression for the above circuit becomes:

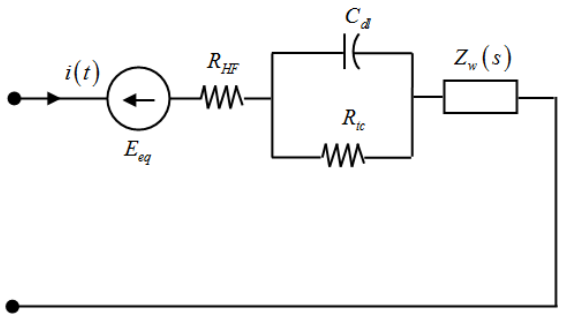


Fig. 2.2: Modified Randles Circuit Model

$$Z(\omega) = R_{HF} + \frac{R_{tc}}{1 + j\omega R_{tc} C_{dl}} + Z_W(\omega) \tag{2.1}$$

Typically, Warburg impedance is theoretically expressed as a non-integer function [64].

$$Z_W(\omega) = \gamma \omega^{1/2} (1 - j) \tag{2.2}$$

where γ is a parameter, which depends on the electrochemical phenomenon.

Nyquist plots of resistance versus (negative) reactance as a function of frequency are used to depict impedance changes in the electrode-electrolyte interface using equivalent circuit models. As shown in Figure 2.3, the Nyquist curve generally shows a semi-circle of radius r governed by the charge transfer phenomenon, and a 45^0 slope (theoretically) where the diffusion phenomenon is predominant. These profiles tend to grow as a function of cell age and are particularly sensitive to the chosen battery chemistry, the nature of the solid electrolyte interface, and the temperature. The main difficulty in modeling a battery is the diffusion phenomenon at low frequencies.

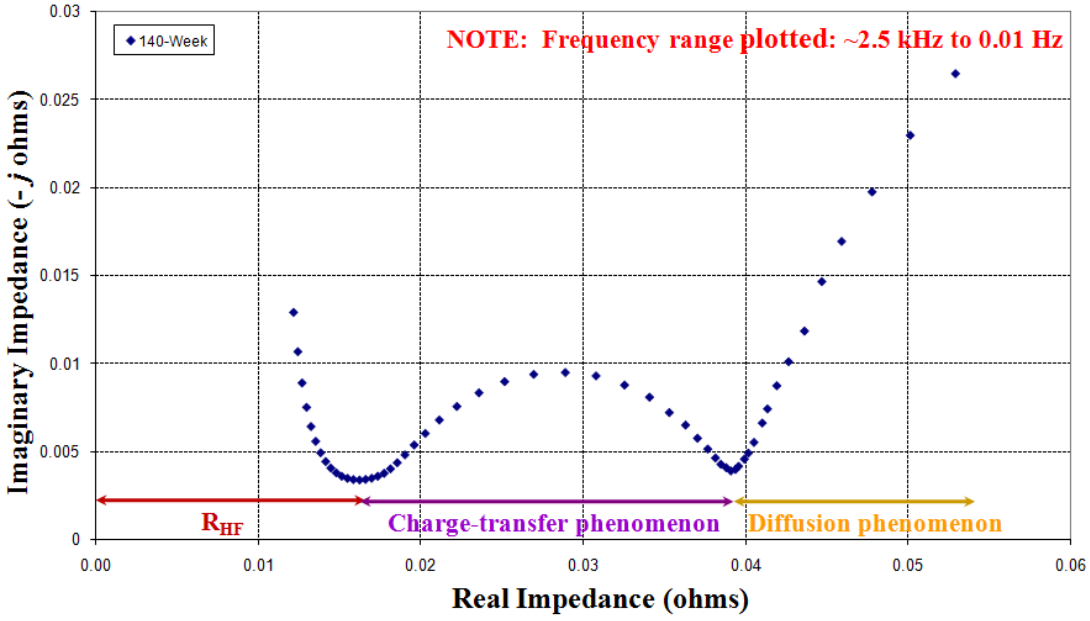


Fig. 2.3: Nyquist plot of Battery Impedance

In this chapter, we implemented a nonlinear least squares estimation technique using the *lsqnonlin* function in the Optimization Toolbox of MATLAB® to identify the parameters of the equivalent circuit of the battery cell shown in Figure 2.2. The

optimization uses trust region method and is based on the interior-reflective Newton method [32], [31]. The non-integer function presented in equation 2.2 was used to model the Warburg impedance.

The optimization algorithm does not converge to a solution (optimal parameter estimate) sometimes due to severe ill-conditioning. Hence, we decoupled the EIS data according to the three different frequency regions of the Nyquist plot (namely high-frequency region, charge-transfer region, and diffusion region) shown in Figure 2.3 and then, implemented non-linear least squares on each independent data-set to obtain the parameter estimates (R_{HF} characterizes the high-frequency region, R_{tc} and C_{dl} characterize the charge-transfer phenomenon, and γ represents the diffusion phenomenon). The three frequency regions are chosen based on the frequency samples from the EIS data as follows:

High-frequency Region $\rightarrow R_{HF} \rightarrow 10000\text{Hz} - 125.89\text{Hz}$

Charge-transfer Region $\rightarrow R_{tc}||C_{dl} \rightarrow 100\text{Hz} - 0.12589\text{Hz}$

Diffusion Region $\rightarrow Z_W \rightarrow 0.1\text{Hz} - 0.01\text{Hz}$

Hence, the inputs of the parameter estimation procedure are (1) frequency, (2) real impedance, and (3) imaginary impedance. The outputs are the parameters of modified Randles circuit, viz.,R_{HF} , R_{tc}, C_{dl}, and Z_W.

2.2.2 Identification of Lumped Parameter battery model

A model that encompasses the complex non-linear behavior of a battery, and describes all facets of its performance over the entire life and over any energy storage cycle,

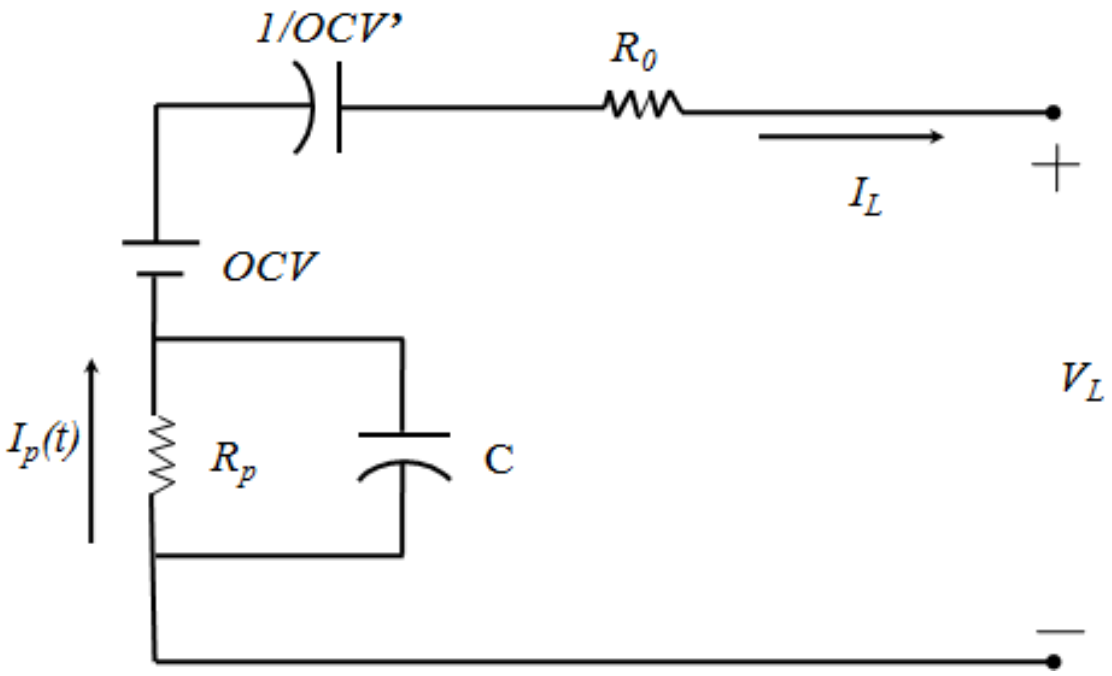


Fig. 2.4: Lumped Parameter Battery Model

contains parameters that are difficult (or impossible) to estimate from the available test data. Hence, USABC proposed a battery model which is linear at a given point in life based on a repeatable test cycle, such as the Hybrid Pulse Power Characterization (HPPC) test [90]. The USABC lumped parameter model [43] is shown in Figure 2.4.

The parameters of the model are: *OCV* = Ideal voltage representing open-circuit battery voltage, R_0 = Battery internal ohmic resistance, R_P = Battery internal polarization resistance, *C* = Shunt capacitance, $\tau = R_P C$ = Polarization time constant, I_L = Load current (positive for discharge and negative for charge), I_P = polarization current = solution of the differential equation $\frac{dI_P}{dt} = \frac{(I_L - I_P)}{\tau}$, initial condition, $I_P\,(0^-) = 0$,

$\frac{1}{OCV'}$= Capacitance that accounts for variation in *OCV* with the time integral of load current I_L. Thus,

$$V_L = -OCV' \left[\int_0^t I_L dt \right] + OCV - R_P I_P - R_0 I_L \tag{2.3}$$

$$= \text{Battery terminal voltage}$$

The objective is to estimate the lumped battery model parameters OCV, OCV', R_0, R_P and τ using the L-HPPC test data. However, since the L-HPPC test repeats each pulse profile at several fixed depth of discharge values, the parameters are estimated at each depth-of-discharge (DOD). Formulation of the linear lumped parameter model in state-space representation is as follows:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -1/\tau & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 1 \\ 1/\tau \\ 0 \end{bmatrix} I_L$$

$$V_L = \begin{bmatrix} c_1 & c_2 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + d\, I_L \tag{2.4}$$

$$x_1(t) = \int_0^t I_L(\sigma) d\sigma; x_2(t) = I_P(t), x_3(t) = OCV_0,$$

$$c_1 = OCV', c_2 = -R_P, d = -R_0, \tau = R_P C$$

The load current, I_L, and terminal voltage, V_L, for one pulse (one DOD) of L-HPPC test (shown in Figure 2.5) are inputs to the system identification procedure.

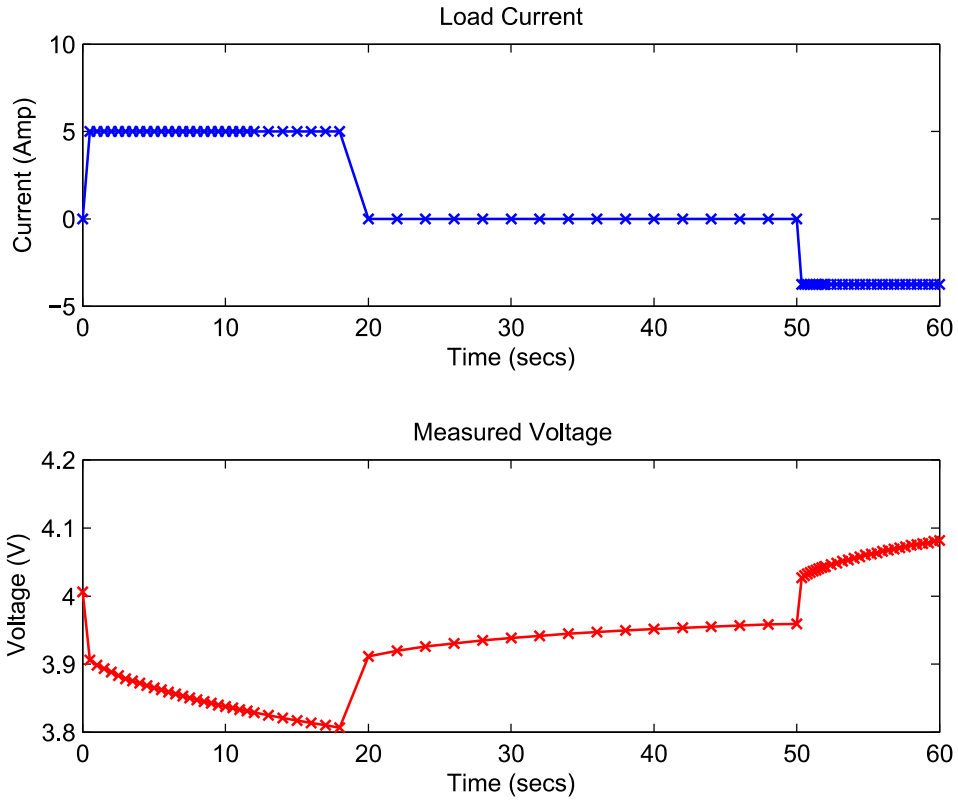


Fig. 2.5: Load Current and Terminal Voltage for One-Pulse of L-HPPC data

The model parameters are estimated using the iterative *Prediction-Error Minimization* (PEM) method (Appendix B). The algorithm implementation is similar to the autoregressive moving-average algorithm, with modifications to the computation of prediction errors and gradients, and the cost function (sum of squares of measurement prediction errors) minimization is implemented in such a way that the variables are constrained to be *non-negative*. The R^2*-statistic* obtained was approximately 99.5% for all pulses. The estimated and measured terminal voltage and the R^2*-statistics* for one week (i.e. 8 pulses) of the L-HPPC data are shown in Figure 2.6.

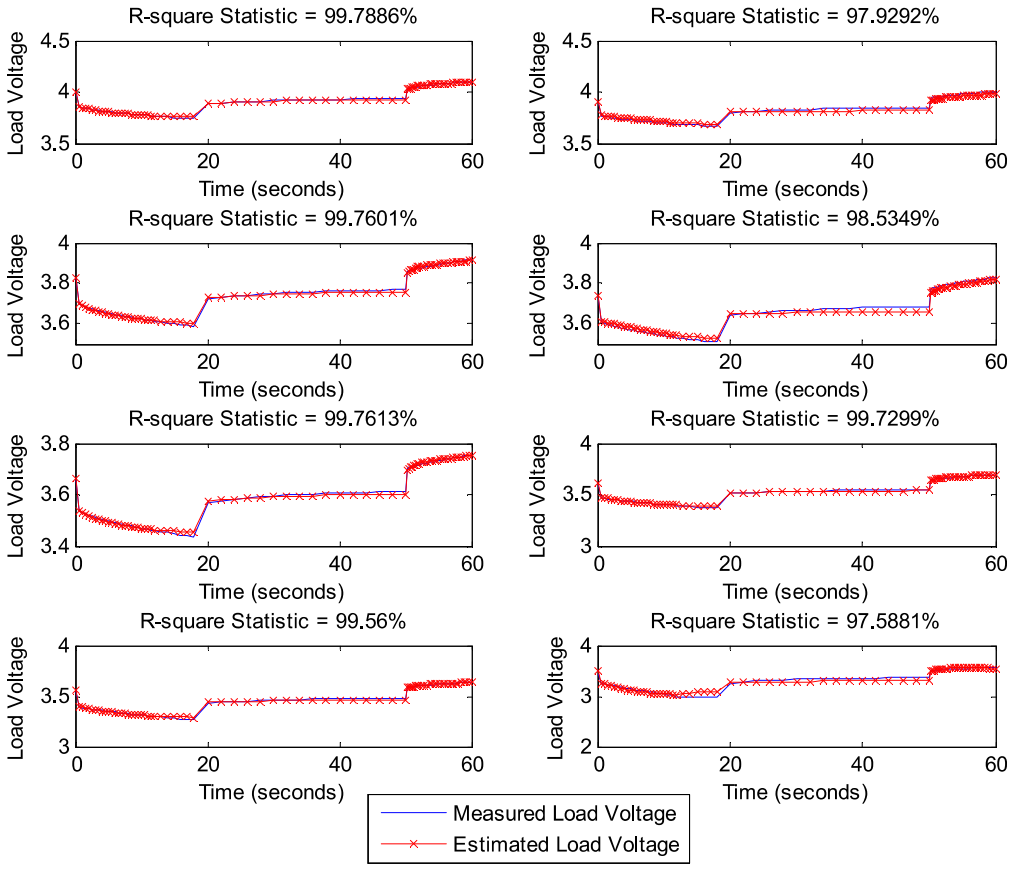


Fig. 2.6: Estimated and Actual Terminal Voltage for L-HPPC data

2.3 Estimation of Battery Characteristics

2.3.1 SVMR Prediction

The circuit parameters vary with time. We model the temporal variations of total resistance $R(k) = (R_{HF}(k) + R_{tc}(k))$ in week *k* via an auto-regressive SVMR of length *L* (typically, *L*=2). Specifically,

$$R(k) = f(\{R(i)\}_{i=k-L}^{k-1}) \tag{2.5}$$

We also use moving average SVMRs for predicting the capacity fade from $C_1/1$ capacities, and RUL from capacity and power fade estimates.

2.3.2 State of Charge (SOC)

State of Charge (SOC) is defined as the available capacity in a battery expressed as a percentage of the actual (or estimated) rated capacity. This is normally referenced to a constant-current discharge at a $C_1/1$ rate. That is,

$$\text{SOC}\,(\%) = \frac{\text{estimated capacity} - \text{capacity removed}}{\text{estimated capacity}} \times 100 \tag{2.6}$$

where capacity removed $=\int_0^t I_L(\tau)d\tau$. For constant-current discharge, capacity removed = $I_L.t$, where, I_L= discharge current in Amperes and t = time in hours.

The sum of high-frequency (or ohmic) resistance and charge-transfer resistance is *correlated* with the $C_1/1$ capacities from the $C_1/1$ static capacity test data. SVMR

[16] is then used to make future predictions of the resistance using the model in 2.5, and consequently, the estimated capacity is obtained from these correlation plots. The capacity removed depends on the battery usage.

2.3.3 State of Health (SOH)

State of Health (SOH) is the ability of a cell to store energy, source and sink high currents, and retain charge over extended periods, relative to its initial or nominal capabilities [14]. It is a *figure of merit* that describes the degree of degradation of a battery, and gives a quantitative measure that replaces fuzzy statements, such as 'fresh', 'aged', 'old', and 'worn out' [41]. In this chapter, we characterize the SOH of a battery by its *power fade* and *capacity fade*.

Power Fade

The loss of cell power due to an increase in cell impedance during aging is known as power fade [67]. In this chapter, we determine the power fade of the battery from the EIS test. The actual power from the EIS test is calculated as:

$$P = \frac{V^2}{R} \tag{2.7}$$

where V is the voltage (5mV for EIS test) and R is the total resistance, $(R_{HF} + R_{tc})$, obtained from EIS data using nonlinear least squares or predicted via SVMR using 2.5. Hence, power fade is computed as follows:

$$PowerFade = 1 - \left(\frac{Power(k)}{Power(0)}\right) = 1 - \frac{R(0)}{R(k)} \tag{2.8}$$

where $\mathrm{Power}(0)$ is the power at the beginning-of-life (BOL) and $\mathrm{Power}(k)$ is the power at the desired time (week). Consequently, power fade at the chosen week is obtained via SVMR predictions of the total resistance using equation 2.8.

An alternate procedure for estimating the power fade and energy fade from Hybrid Pulse Power characterization (HPPC) test data is shown in Figure 2.7.

A detailed procedure for determining the Available Energy and Available Power, along with the resistance characteristics, open-circuit voltage (OCV) and the pulse power capability, from the L-HPPC data has been discussed in [54] and [43]. The Power fade and Energy fade from the beginning of life (BOL) to some later point in time is calculated using the equations,

$$\mathrm{PowerFade}(\%) = \left(1 - \frac{\text{Available Power}\,(k)}{\text{Available Power}\,(0)}\right) \times 100 \tag{2.9}$$

$$\mathrm{EnergyFade}(\%) = \left(1 - \frac{\text{Available Energy}\,(k)}{\text{Available Energy}\,(0)}\right) \times 100 \tag{2.10}$$

where, time *'0'* refers to the BOL reference performance test (RPT) and time *'k'* refers to a RPT performed at a later time where the power fade and energy fade are to be determined.

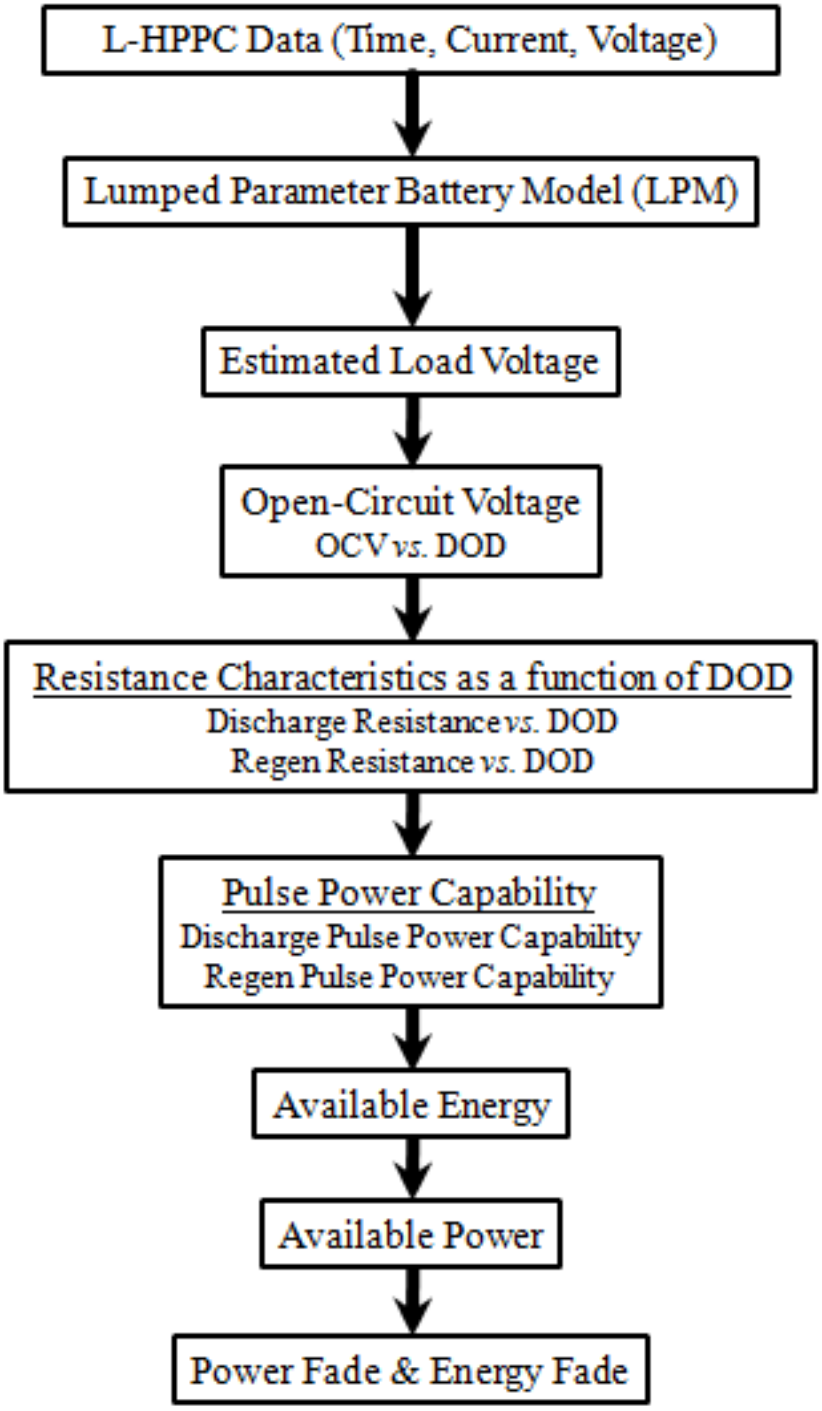


Fig. 2.7: Block diagram for Estimating Power Fade and Energy Fade from L-HPPC test data using LPM Model

Capacity Fade

The gradual loss of capacity of a secondary battery with cycling is known as capacity fade [67]. Capacity fade is also the percent loss in $C_1/1$ discharge capacity [90].

$$\text{CapacityFade}\,(\%) = \left(1 - \frac{\text{Capacity(k)}}{\text{Capacity}(0)}\right) \times 100 \tag{2.11a}$$

$$\text{CapacityFade}\,(\%) = \frac{\alpha(R(k) - R(0))}{\beta} \times 100 \tag{2.11b}$$

where $Capacity(0)$ is the BOL Capacity, and $Capacity(k)$ is the Capacity at the desired time (week), k. Hence, Capacity fade at the chosen week can be computed by SVMR prediction of the $C_1/1$ capacities from the static capacity test. Given that the capacity is approximately linearly related to the total resistance, i.e., $C(k) = \beta - \alpha(R(k) - R(0))$, it can also be written as equation 3.82.

The capacity fade of a battery is estimated by predicting the $C_1/1$ capacities using an auto-regressive SVMR:

$$C(k) = g(\{C(i)\}_{i=k-L}^{k-1}) \tag{2.12}$$

Once the capacity is estimated, the capacity fade is computed from equation 3.82. Alternately, one can compute the capacity fade from equation 3.82 by appealing to the linear correlation between the capacity and battery resistance.

2.3.4 Remaining Useful Life (RUL) via SVMR

In this chapter, we predict the RUL of the battery using a moving average SVMR for different thresholds on capacity fade and power fade. Formally, at week k, we estimate

$$RUL(k) = h(\{P(i), C(i)\}_{i=1}^{k}) \tag{2.13}$$

where $C(k)$ and $P(k)$ are the capacity fade and power fade, respectively, and $RUL(k)$ is the remaining useful life at time k.

2.3.5 Estimation of Survival Function and RUL using HMMs

Survival function is defined as the probability that the system operates without fail beyond a specified time instant *'t'*, given that it is operational at time *'0'*. It is also known as the reliability function and is given by,

$$S(t) = P(T > t) = \int_{t}^{\infty} f(\tau)d\tau \tag{2.14}$$

where T is a continuous random variable denoting *'time-to-failure'* (TTF) having pdf $f(t)$. Here, we estimate the survival functions of batteries using hidden Markov models (HMMs), which, in turn, is used to predict the remaining useful life of batteries.

An HMM is characterized by the number of states (N), observations ($\underline{x}$), state transition probability matrix (A), emission probability matrix (B) and initial state probability vector ($\underline{\pi}$). HMM parameters are generally denoted as $\lambda = (A, B, \underline{\pi})$, where λ

represents the parameter set of HMM. A brief description of HMMs is provided in Appendix C. Capacity fade and power fade are the observations of the hidden Markov model. Since the observations $\underline{x}(t)$ at time t (e.g., week t) are continuous, they are modeled as Gaussian mixture with *m* components having means $\{\mu_i\}$, covariances $\{\Sigma_i\}$, and mixture priors $\{w_i\}$. Formally,

$$b_i(\underline{x}) = \sum_{i=1}^{m} N(\underline{x}; \mu_i, \Sigma_i)\ .\ w_i \tag{2.15}$$

The HMM parameters λ are learned from the training data using the Baum-Welch algorithm [97], [96], [12]. The number of states of HMM are chosen based on how well the HMM predicts the power fade and capacity fade as determined by the coefficient of determination (R^2-statistic) which in turn depends on the squared error loss of the capacity fade and power fade estimates. Formally, the predicted observations at time step '$t+1'$ can be computed recursively via

$$\begin{aligned}
&p\left(\underline{x}(t+1)|\underline{\mathrm{x}}^t\right) = \sum_{s(t+1)} p\left(\underline{x}(t+1), s(t+1)|\underline{\mathrm{x}}^t\right) \\
&= \sum_{s(t+1)} p\left(\underline{x}(t+1)|s(t+1)\right) \sum_{s(t)} p\left(s(t+1)|s(t)\right) p\left(s(t)|\underline{\mathrm{x}}^t\right) \\
&= \tfrac{1}{p(\underline{\mathrm{x}}^t)} \sum_{s(t+1)} p\left(\underline{x}(t+1)|s(t+1)\right) \sum_{s(t)} p\left(s(t+1)|s(t)\right) p\left(\underline{\mathrm{x}}^t, s(t)\right) \\
&= \tfrac{1}{p(\underline{\mathrm{x}}^t)} \sum_{s(t+1)} p\left(\underline{x}(t+1)|s(t+1)\right) \sum_{s(t)} p\left(s(t+1)|s(t)\right) \alpha(s(t)) \\
&= \tfrac{1}{p(\underline{\mathrm{x}}^t)} \sum_{j=1}^{N} p\left(\underline{x}(t+1)|s(t+1)=j\right) \sum_{i=1}^{N} a_{ij}\alpha_i(t)
\end{aligned} \tag{2.16}$$

where $\underline{\mathrm{x}}^t = [\underline{x}(1), \underline{x}(2), ..., \underline{x}(t)]$ is the set of observations until time $t(1 \leq t \leq K)$, $\alpha_i(t) = p(\underline{\mathrm{x}}^t, s(t) = i)$ is the forward variable, and '*s*' is the hidden state.

An estimate of state probabilities ($\underline{\pi}(t)$) at the current time instant *'t'* can be computed from the HMM model via the forward algorithm or the Viterbi algorithm [97]. Here, we determined the j^{th} component of ($\underline{\pi}(t)$) from the Viterbi algorithm using,

$$\pi_j(t) = \frac{\delta_j(t)}{\sum_{n=1}^{N} \delta_n(t)} \tag{2.17}$$

where $\underline{\delta}_j(t)$ is the probability that the observations up to time *'t'* end in state *'j'* and is given by,

$$\begin{aligned} \delta_j(t) &= \max_{s(1),s(2),\ldots,s(t-1)} P\left(s(1)\, s(2) \ldots s(t) = j, \underline{x}(1)\, \underline{x}(2) \ldots \underline{x}(t) | \lambda\right) \\ &= \max_{1 \le i \le N} \left[\delta_i(t-1) a_{ij}\right] b_j(\underline{x}(t)) \end{aligned} \tag{2.18}$$

We can also estimate ($\underline{\pi}(t)$)'s from the normalized forward variables (via forward algorithm) using

$$\alpha_j(t) \;=\; \max_{1 \le i \le N} \left[\alpha_i(t-1) a_{ij}\right]\; b_j\left(\underline{x}(t)\right) \tag{2.19}$$

and normalizing the forward variables as in 2.17. Once the state probabilities are obtained, the estimation of survival (or reliability) function is computed using,

$$S(t+n) = \underline{\pi}(t)\, A^n \underline{e}_f \tag{2.20}$$

where $\underline{e}_f$ is a vector of size *'N'* with 0's in failing states and 1's in the operational states. The mathematical derivation of equation 2.20 is given in Appendix D.

2.4 Results and Discussion

Second-generation lithium-ion cells with baseline chemistry were chosen as the validation data in this chapter. Life testing was interrupted every 4 weeks (data was collected for a total of 140 weeks) for reference performance tests (RPTs) consisting of static capacity test, a low-current hybrid pulse power characterization (L-HPPC) test, and electrochemical impedance spectroscopy (EIS) test. Five baseline cells (8, 9, 11, 14, and 15) out of a test matrix of 30 cells have been chosen for our experiments because of time-period of collection of data (140 weeks). The other cells in the test matrix were removed earlier (at different time-periods) as they were considered "dead" (i.e. approximately, power fade = 50% and capacity fade = 30%).

2.4.1 Parameter Estimation Results

Estimates of the modified Randles circuit parameters, R_{HF}, R_{tc}, C_{dl}, and Warburg parameter, γ, for different battery cells are obtained as a function of time (weeks) using frequency-selective nonlinear least squares from the EIS test data. The time-dependent behavior of ($R_{HF} + R_{tc}$) will be used to estimate the degradation of battery cells. The degradation of some of the battery cells as a function of time (weeks) is shown below (C_{dl} and Warburg parameter, γ, are not shown here since they show negligible change over the aging process, and are not used in our estimation framework).

The battery's internal impedance parameter identified from the EIS test chosen for correlation analysis is the total internal resistance of the battery (i.e., $R =$

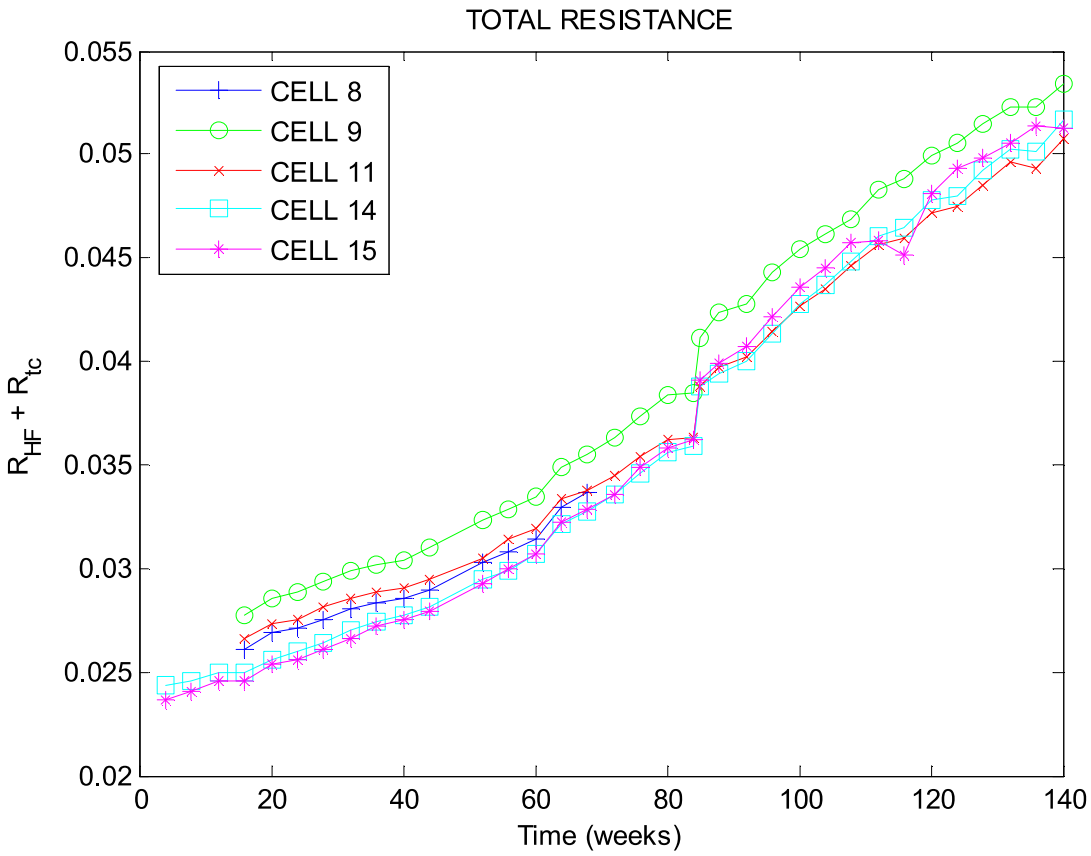


Fig. 2.8: Total resistance versus battery age in number of weeks

$R_{HF} + R_{tc}$). Figure 8 shows the variation of resistance as a function of weeks. The SVMR model (with $C = 1000$ and $\sigma = 0.001$) with $L = 2$ for the total resistance parameter R showed excellent R^2-fit values of 97% for as many as $15 - 20$ week-ahead predictions. Hence, SVMR proved to be an ideal algorithm for future predictions of the circuit parameters. This implies that EIS tests can be done as part of routine vehicle maintenance (e.g., oil changes).

The $C_1/1$ discharge capacities were extracted from the $C_1/1$ static capacity test data. Figure 9 shows the average $C_1/1$ discharge capacity for different Baseline cells

that were aged at 60% SOC.

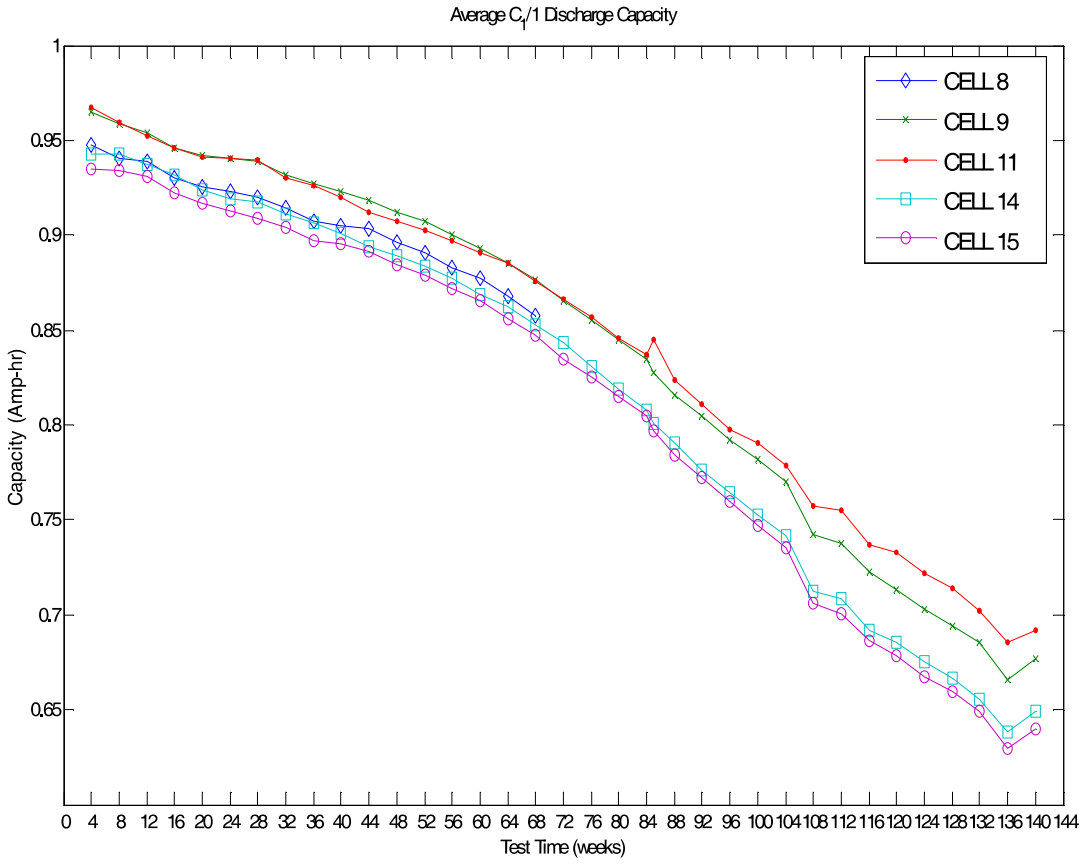


Fig. 2.9: Average $C_1/1$ discharge capacity for Baseline cycle-life cell groups

Figure 10 shows the high degree of *linear correlation* between the $C_1/1$ discharge capacity and the internal resistance parameter R for different cells. For the linear model $C(k) = \beta - \alpha(R(k) - R(0))$, typical values for α and β for the various cells were $\alpha \in [9.52, 11.03]$ and $\beta = C\,(0) \in [0.93, 1]$. Hence, linear model parameters of the correlations plots for different temperatures and SOC's can be stored as part of a look-up table for instantaneous computation and prediction of internal resistance and

battery capacity.

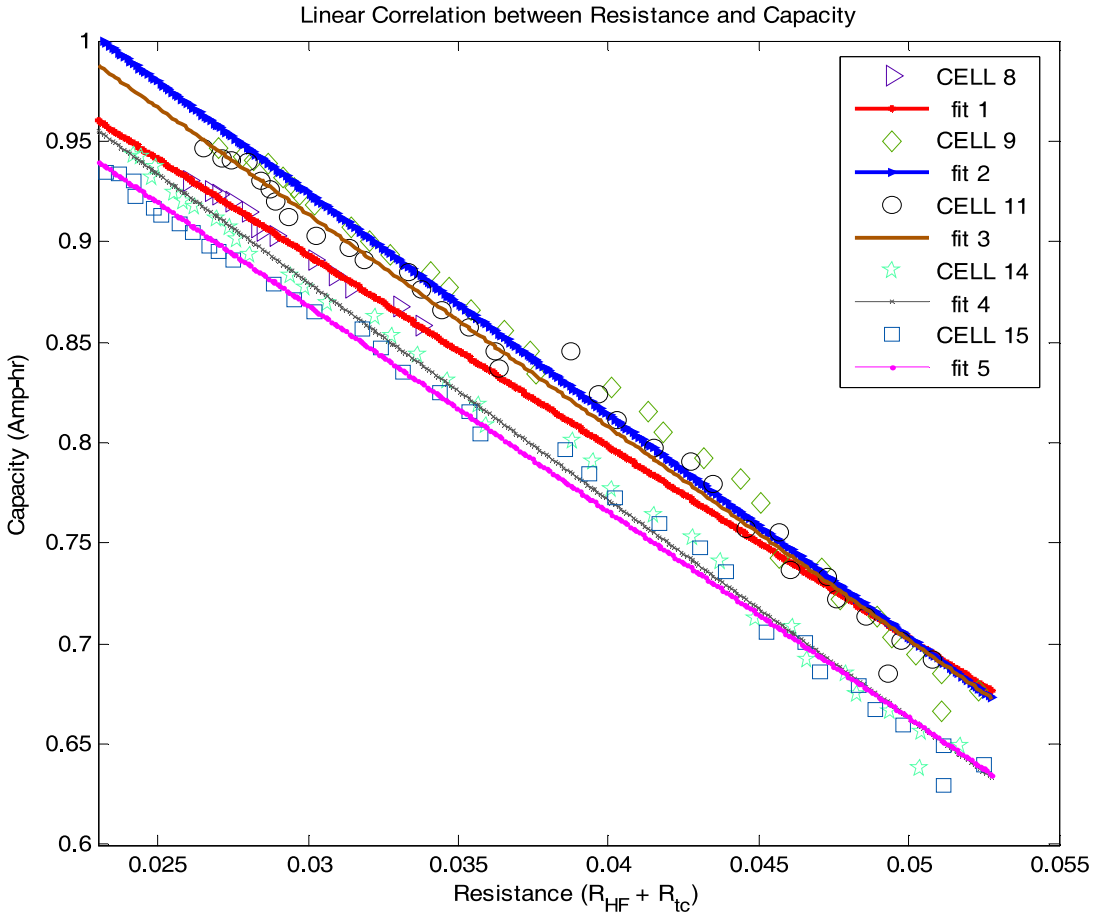


Fig. 2.10: Correlation between resistance and battery capacity

2.4.2 State of Charge (SOC) Results

The resistance predictions are used to obtain the corresponding $C_1/1$ discharge capacity from the linear correlation plots in Figure 2.10. Subsequently, the SOC is obtained using equation 2.6. Figure 2.11 shows the variation of the SOC for a constant-current discharge of baseline cell 8 from beginning-of-life (BOL) to end-of-life (EOL). Conse-

quently, SVMR could also be used to predict SOC even under nonlinear driving conditions with good accuracy.

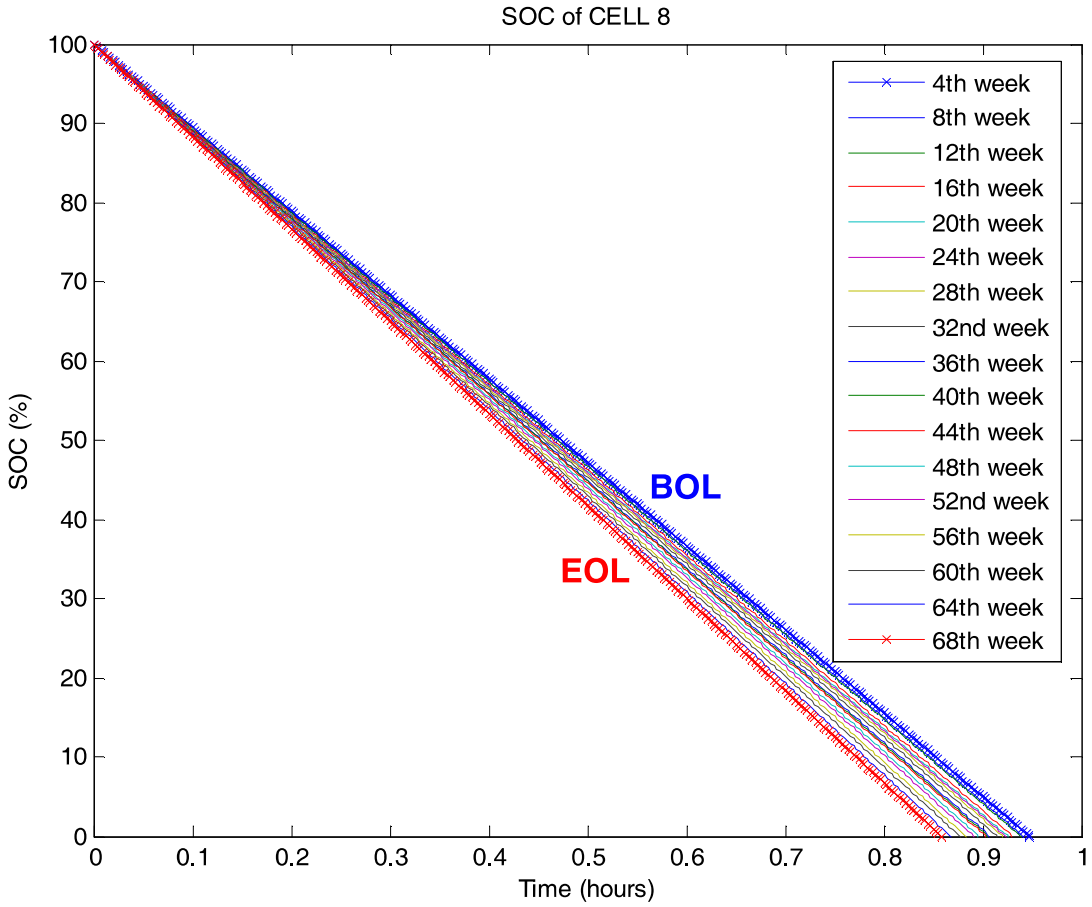


Fig. 2.11: Percentage SOC as a function of time for Baseline Cell 8

2.4.3 State of Health (SOH) Results

The rate of capacity loss (capacity fade) generally increases with increasing temperature and SOC. The SVMR model (with $C = 100$ and $\sigma = 0.0015$) with L=2 for the $C_1/1$ capacities showed excellent R^2-fit values of 95% for as many as 10 week-ahead predictions. Figure 2.12 shows the capacity fade of different baseline-chemistry cells obtained using equation (11).

The power fade of different cells predicted using the approaches described in section 3 is shown in Figure 2.13. The power fade computed from EIS data (using equation (8)), and L-HPPC data (using equations (9) and (10)) showed similar results, with the EIS predictions providing an underestimate by approximately 2% for all cells. Hence, battery health monitoring in terms of its power fade and capacity fade can be obtained using an autoregressive SVMR, which enables condition-based maintenance of the batteries.

2.4.4 Remaining Useful Life (RUL) Results

Figure 2.15 shows the RUL for an *end-of-life* (EOL) criterion of approximately 50% power fade and 30% capacity fade, at which time the battery is assumed to have failed operationally.

As seen in the Figure, the RUL of the battery decreases with time and the EOL criterion is based on the application-dependent capacity fade and power fade. The SVMR can be trained for various thresholds of capacity and power fade thresholds, and

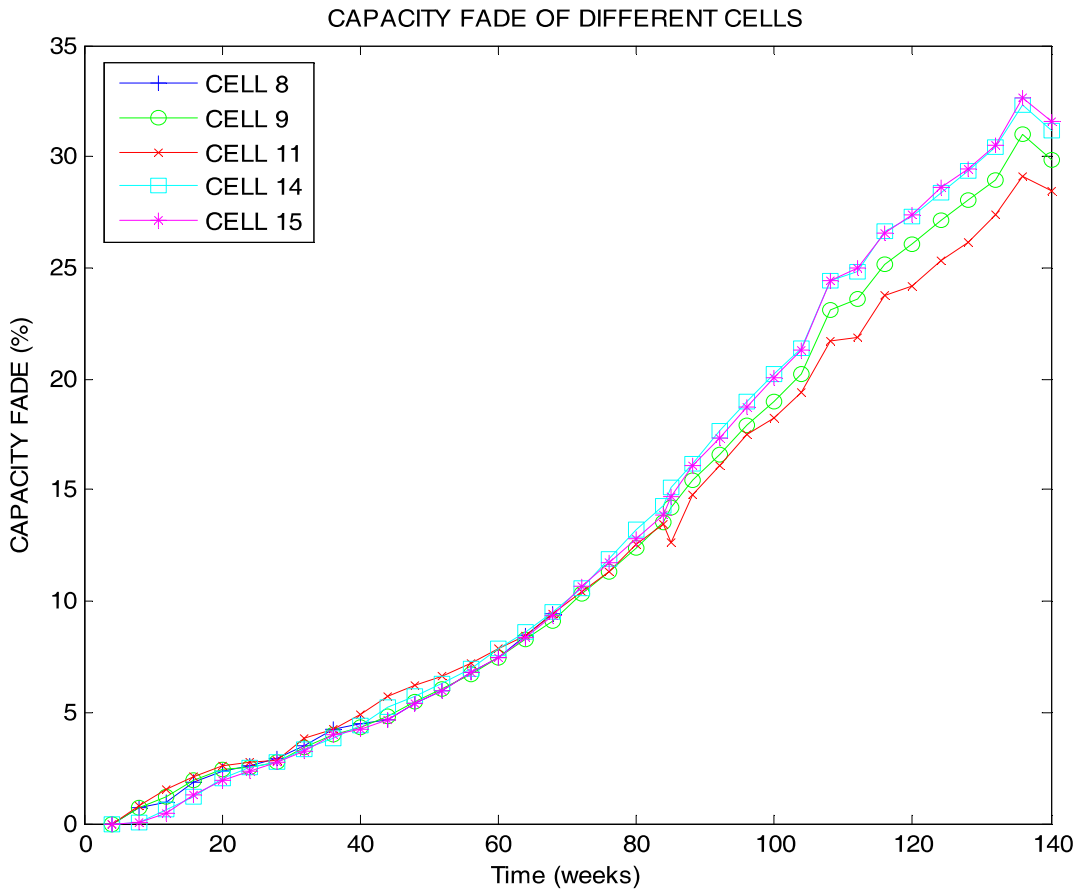


Fig. 2.12: Capacity fade prediction for different cells

consequently, the RUL can be obtained for a specific context-dependent threshold. The SVMR regularization parameter was found to be 100 (i.e. $C = 100$) and σ of the RBF kernel function was found to be 1, for RUL predictions.

2.4.5 Survival function and Reliability analysis Results

An 8-state HMM model is trained using capacity fade and power fade of a battery as observations and the survival function is estimated for the cells. A two component

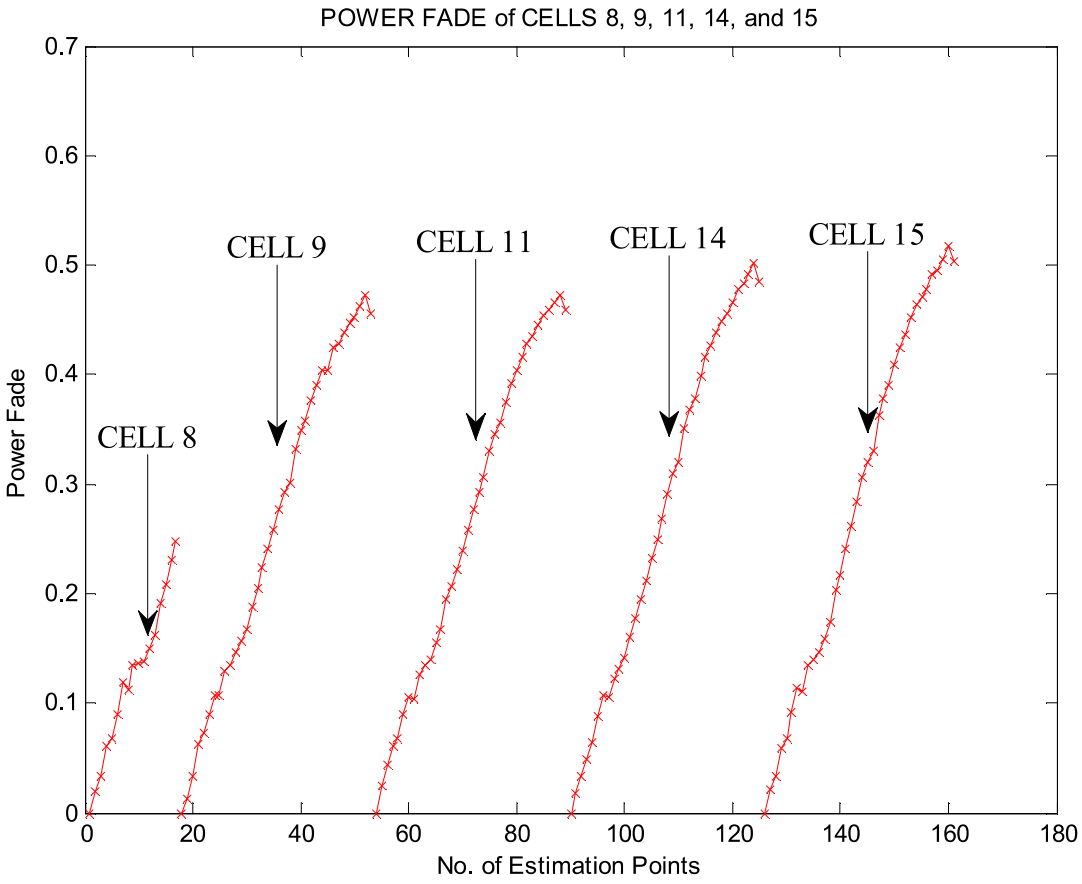


Fig. 2.13: Power fade estimation for different cells

Gaussian mixture model was used as the emission model. Figure 2.14 shows an estimated survival function from hidden Markov models. It is a monotonically decreasing function that depicts the reliability of a battery at any time *'t'*. The R^2-fit was about 99.91%.

The expected remaining life at any time *'t'* can be computed from the survival function by defining a threshold on reliability. Mathematically, it is written as,

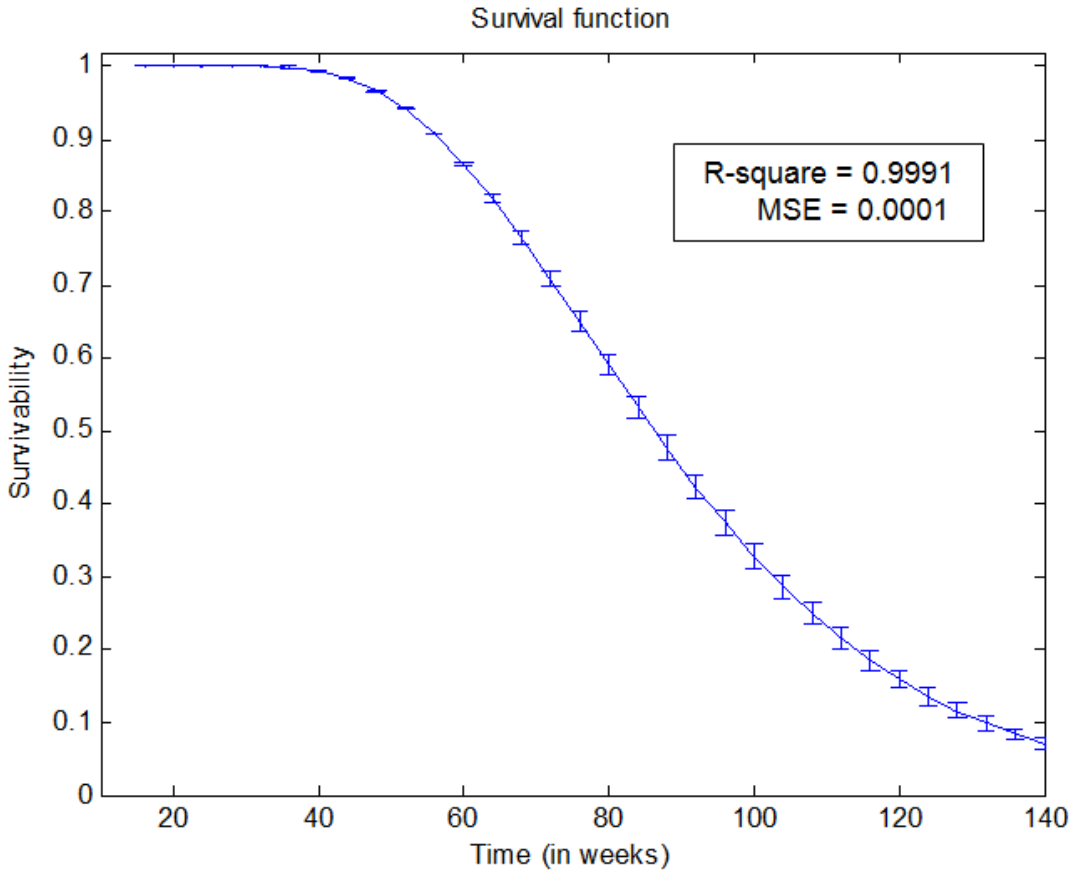


Fig. 2.14: Survival Function Estimation via HMM

$$RUL(t) = \arg\min_{n} \left\{ \left(\underline{\pi}(t) A^n \underline{e}_f \right) \leq \varepsilon_0 \right\} \tag{2.21}$$

where ε_0 denotes the threshold of functional failure. Given that a 50% power fade and a 30% capacity fade constitutes a functional failure of the battery (i.e., EOL), ε_0 is approximately 0.2578. Figure 2.15 shows the expected remaining life of the battery with ε_0 as 0.2578. Consequently, SVMR and HMM based RUL predictions are quite similar with SVMR providing better estimates (based on R^2 and MSE measures) and

can be used for prognostic health management of the battery module. Nevertheless, HMM's provide consistently comparable results and outperform SVMR towards the end of life (EOL) of the battery (i.e. it gives estimates with smaller variance for the 136^{th} and 140^{th} week predictions than SVMR).

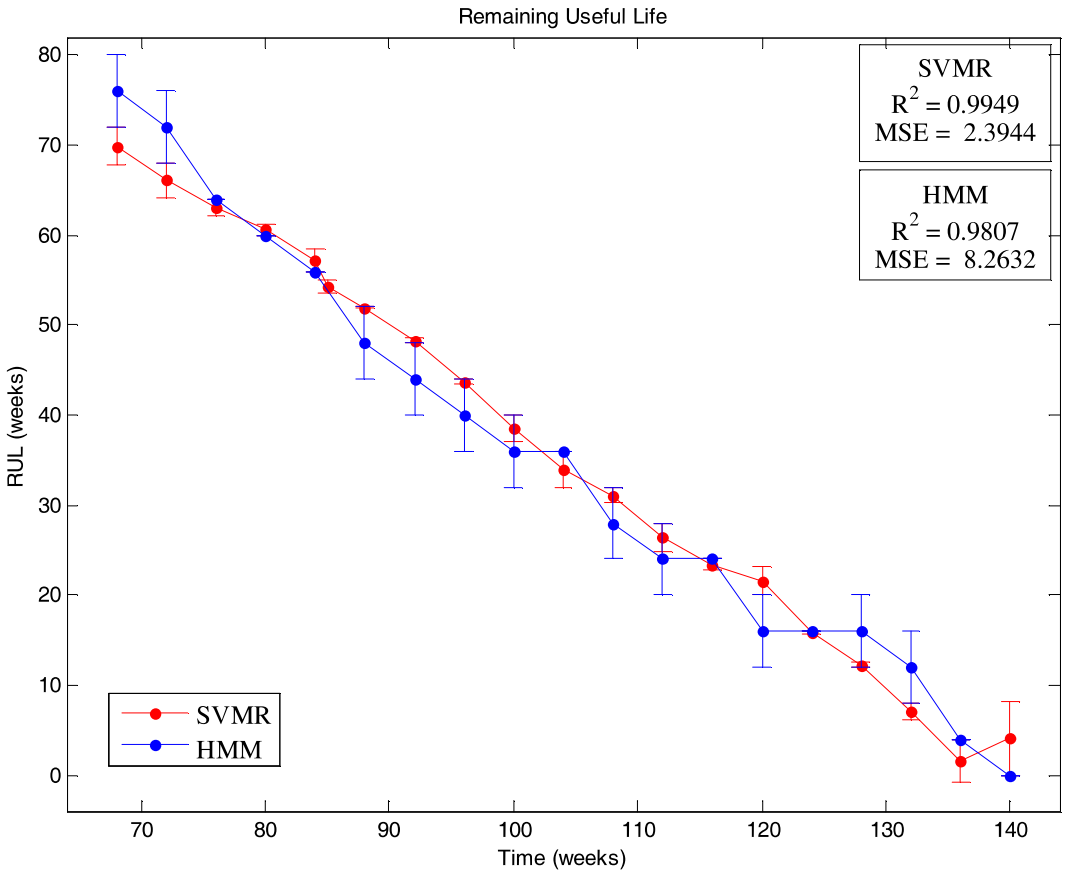


Fig. 2.15: Remaining Useful Life Prediction via SVMR and HMM

2.5 Conclusions

Next generation battery management systems will feature online tracking and monitoring of pivotal battery characteristics, such as SOC, SOH, RUL, survival function, area specific impedance (ASI), and power density to facilitate efficient diagnostic and prognostic maintenance of batteries. Here, we proposed a framework for estimating and predicting these salient battery performance measures. Three commonly used reference performance tests were used for our analysis, viz., $C_1/1$ static capacity test, electrochemical impedance spectroscopy (EIS) test, and the low-current hybrid pulse power characterization (L-HPPC) test. Our approach to estimating the SOC, SOH, and RUL is based on a modified Randles equivalent circuit model of a battery. This model consists of a high frequency resistance, a parallel RC circuit for modeling charge transfer phenomenon at medium frequencies, and Warburg impedance to model the diffusion phenomenon at low frequencies. The circuit parameters are estimated from the EIS data using non-linear least squares (NLLS) estimation techniques. The ill-conditioned parameter estimation problem has been addressed by decoupling the high-frequency, charge-transfer, and diffusion regions of the Nyquist plot. We proposed an autoregressive SVMR using an ε-sensitive loss function for predicting the resistance. The $C_1/1$ capacity was obtained from the $C_1/1$ static capacity test. Hence, the SOC can be obtained using the resistance-capacity mapping or from the $C_1/1$ prediction via an autoregressive SVMR. Consequently, a framework was proposed to predict the capacity fade and power fade, which characterize the SOH of a battery. An alternate procedure

for estimating the power fade and energy fade from low-current Hybrid Pulse Power characterization (L-HPPC) test data using the lumped parameter model (LPM) is also proposed. The RUL predictions were made by setting different application-dependent thresholds on capacity fade and power fade. Survival function and RUL estimation of the battery was performed using HMM model predictions as well. The proposed framework for the Gen 2 Li-ion battery data showed good accuracy for estimating the important battery characteristics.

Consistent estimates of residual useful life (RUL) via two different techniques (SVMR, HMM) provide partial cross-validation, as well as demonstrate the accuracy of RUL estimates. In addition, our use of steady-state EIS and transient HPPC data to separately estimate power fade and capacity fade provides another cross-validation of our approach.

Further investigation is required on the validation of the estimation framework for real-time driving schedule conditions and dependence of circuit parameters on temperature and cycling conditions, as well as different battery chemistries. In addition, a systematic evaluation of different algorithms for the estimation of essential battery characteristics on benchmark battery data is suggested as a future area of research.

Chapter 3

Integrated Battery Fuel Gauge and Optimal Charger

3.1 Introduction

Modern-day battery management systems (BMS) have become an integral part of several key automotive, aerospace and portable applications. The functionality of a BMS is multifaceted, ranging from cell protection, charge-discharge control, determining the pivotal battery characteristics such as state of charge (SOC), state of health (SOH), and remaining useful life (RUL), to battery capacity monitoring, remaining run-time estimation, monitoring and storing historical data, etc. The goals of BMS are to maximize the run-time per discharge cycle, as well as the number of charge-discharge cycles attainable during the life of the battery. The need for the online estimation of such prognostic information with acceptable error under all operating conditions over the entire lifetime of a battery requires developing new approaches.

Figure 3.1 shows a conceptual block diagram of a BMS consisting of a battery fuel gauge (BFG) and an optimal charger. Some of the key requirements in designing the next-generation lithium-ion BMS, especially those used in portable devices, are

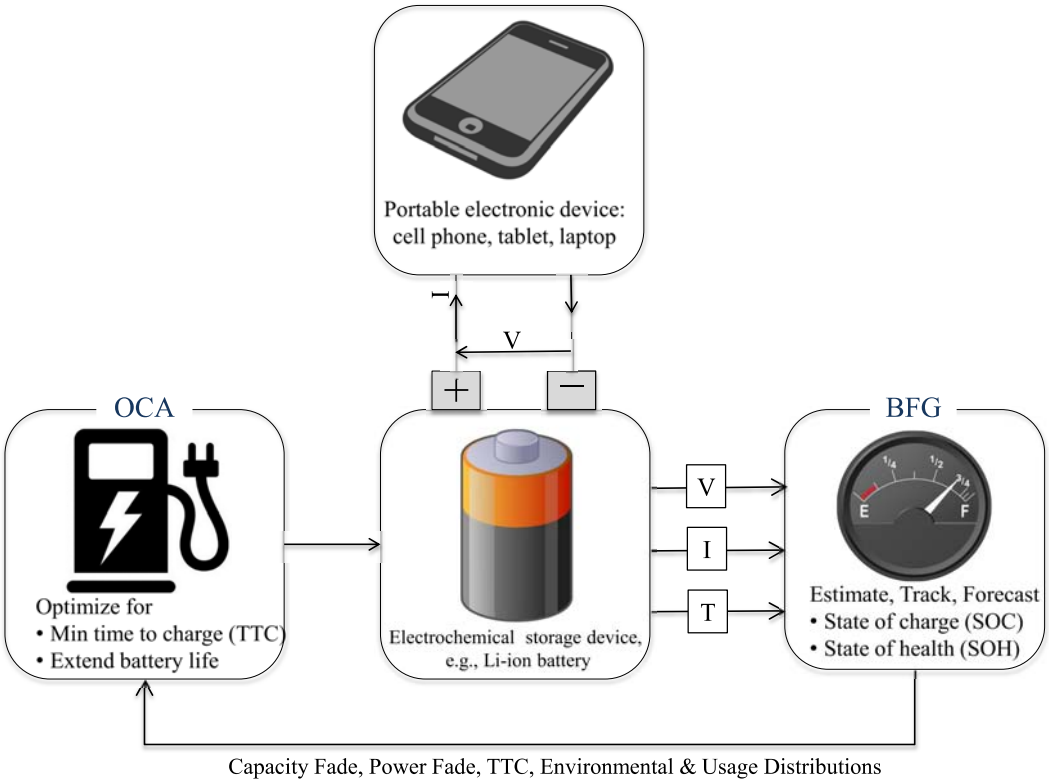


Fig. 3.1: Conceptual Block Diagram of a BMS

high accuracy, fixed-point computation implementation, ability to work with limited memory, and adaptation to short-term and long-term battery changes. Precise estimation of the characteristics of lithium-ion batteries requires the consideration of key chemical phenomena as listed below.

1. *Hysteresis Effect:* The term hysteresis is derived from the Greek *hustereia*, which means "to arrive late". As illustrated in Figure 3.2, the *major hysteresis loop* corresponds to a full cell charge and discharge (Point A to point B while charging, and B to A while discharging), and the *minor hysteresis loops* are encountered

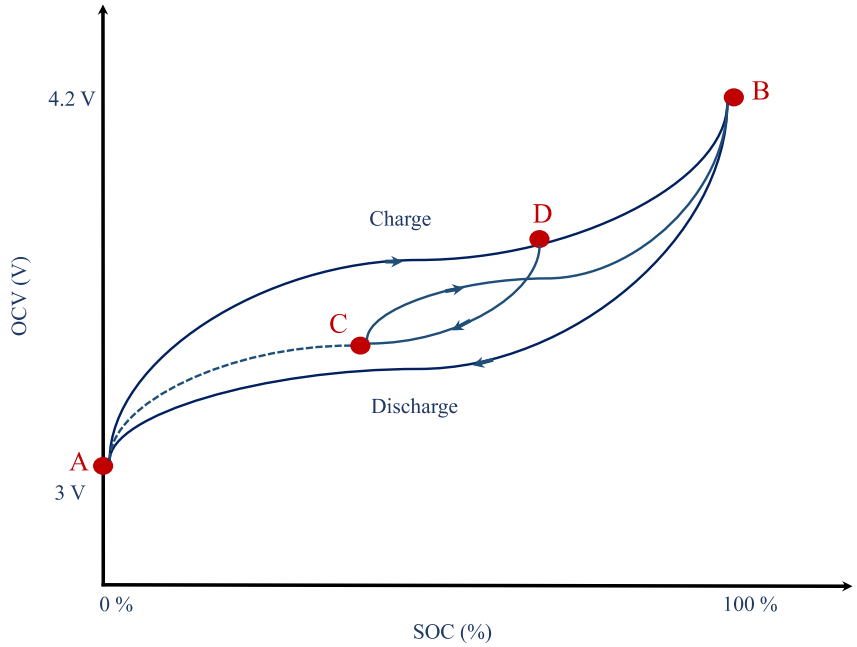


Fig. 3.2: Illustration of Major and Minor Hysteresis Loops

when a partial charge is followed by a partial discharge (Point D to point C, and point C to point B), and vice versa. The polarization does not immediately flip sign upon a current reversal, but slowly decays from one leg of the major hysteresis loop to the other. Hysteresis is a function of current and SOC, and is found to be considerably larger at low temperatures (especially for Li-ion Polymer Batteries) [89]. The hysteresis in Li-ion batteries is due to the thermodynamic entropic effects, mechanical stress, and microscopic lattice distortions within the active electrode materials during Lithium insertion/extraction [107]. Cells with two-phase active materials have been shown to produce pronounced open-circuit voltage (OCV) hysteresis effect, where an electrode's potential after Li insertion

is lower (against Li/Li^+ reference) than after Li extraction, even with the same stoichiometric Li content [11, 118].

2. *Relaxation Effect:* The relaxation effect is another basic characteristic of the battery that represents the slow convergence of the battery's OCV to its equilibrium point after hours of relaxation following charging/discharging. It is a phenomenon caused by the double layer charging/discharging [107]. This may be viewed as a low-pass filter on the current [89].

3. *Memory Effect:* Memory effect refers to batteries gradually losing their usable capacity owing to a reduced working voltage due to repeated recharging after being only partially discharged. Memory effects are well known to users of nickel-cadmium and nickel-metal-hydride batteries, and typically, Li-ion batteries have no memory effect. However, $LiFePO_4$ was recently observed to display memory effects after partial charge and discharge cycles. Memory effect was enhanced at lower SOC; it did not disappear even after long rest time of 24 hours [115]. Also, $LiMnPO_4$ and $LiNiPO_4$, which are other candidates for next-generation Li-ion batteries, should show this memory effect because they have the same olivine structure and two-phase reaction as $LiFePO_4$.

4. *Two-phase Transition/Reaction Effect:* Due to the occurrence of at least two different material phases within the electrodes, electrochemical potential remains almost constant during lithium insertion and extraction. This means that a volt-

age plateau occurs during charging or discharging of the battery cell [107]. This makes SOC estimation ill-conditioned.

5. *Lithium-ion Plating:* Plating occurs when the battery is driven with a potential large enough to promote reduction of Li^+ ion to solidify at the surface of the anode. Plating does impact capacity, but is a greater concern for battery reliability/safety. It is possible, through dendrite growth, to short the anode and cathode.

6. *Solid-Electrolyte Interface:* Solid Electrolyte Interface (SEI) is a skin-like material that develops on the anode and that impedes current flow (resistance); it consumes Lithium in a non-reversible fashion.

The proposed integrated BFG and optimal charger consists of many novel elements; The BFG algorithms is based on a novel *voltage drop* model which enables robust SOC tracking, parameter and capacity estimation [7, 8]. The optimal charging algorithms compute closed-loop charging current waveforms for various scenarios based on novel battery aging models and optimal control theory [1–3].

The chapter is organized as follows. An overview of the battery fuel gauge (BFG) and optimal charging algorithms (OCAs) is presented in Sections 3.3 and 3.4. Section 3.5 discusses the battery management system (BMS) based on the BFG and OCA. Section 3.6 presents the results of the BFG and OCA on HIL data collected across different commercially available batteries and varying operating temperatures. Finally, Section 3.7 concludes with a summary.

3.2 Open Circuit Voltage Characterization of Lithium-ion Batteries

3.2.1 Introduction

Rechargeable Lithium-ion batteries are widely used in electric vehicles, personal/wearable electronic devices, household appliances, aerospace equipment, grid storage, and so on. Estimating the state of charge (SOC) and state of health (SOH), known as battery fuel gauging [7–9], is an important function in equipment powered by rechargeable batteries. The OCV is a measure of the electromotive force (EMF) of the battery, which is known to have a monotonic relationship with the SOC of the battery, hence, estimating SOC must have been a straightforward, voltage-look-up process. However, due to the variable internal resistance, capacity, hysteresis and relaxation effects of the battery, the BFG involves many components as outlined in Figure 3.3. The focus of the chapter is on estimating the parameters of the OCV-SOC characterization.

Existing OCV modeling approaches can be broadly classified into chemistry-based and Current-Voltage based approaches. In chemistry-based approaches [27], the OCV of each electrode (anode and cathode w.r.t some reference) is expressed as a function of the utilization of the electrode (the lithium concentration in the electrode normalized by the maximum possible concentration) or the SOC of each electrode. It is generally assumed that this anode and cathode SOC varies linearly with the cell SOC [98]. Subsequently, the difference between the OCV of the anode and cathode gives the OCV of the complete cell. High current rates (i.e., near the rated maximum) have been

shown to affect the macroscopic processes in a way that the OCV hysteresis vanishes for Li-ion cells, which regularly show OCV hysteresis after low current application [105]. Roscher et. al [106] conducted OCV (full and partial charge-discharge cycle) tests on Lithium ion phosphate ($LiFePO_4$) batteries to characterize the hysteresis and recovery effects. The final OCV model is constructed by concatenating the actual SOC, the recovery factor and the hysteresis factor. Our focus in this chapter is on Current-Voltage based OCV characterization.

The Current-Voltage based OCV-SOC characterization can be summed by two simple steps:

(i) Collect pairs of $\{\mathrm{OCV}, \mathrm{SOC}\}$ values, spanning the entire range of $\mathrm{SOC} \stackrel{\Delta}{=} [0, 1]$

(ii) Use the above data to estimate the parameters of the function $\mathrm{OCV} = f(\mathrm{SOC})$ for a hypothesized function f.

There are challenges in both of the above steps. The objective of this chapter is to detail them and discuss ways to address them.

Figure 3.4 shows the equivalent circuits of a battery when it is experiencing dynamic current versus when it is rested [7]. When the battery is experiencing dynamic current, the voltage difference between OCV, $V_\circ(s[k])$, where, $s[k]$ is the SOC at the discrete time k, and the measured terminal voltage $v[k]$ becomes higher. When the current $i[k]$ becomes zero, the battery starts to "relax", i.e., the capacitors $C_1, \ldots, C_N$ start to discharge through $R_1, \ldots, R_N$, respectively. The battery is said to be in a "rest state"

when $i[k] = 0$ for longer than the multiple (typically ≥ 5) of the largest time constant R_iC_i in the equivalent circuit model, i.e., until all the capacitors $C_1, \ldots, C_N$ are fully discharged. However, the OCV is still unreachable at the terminal even after the battery is rested, due to the hysteresis voltage $h[k]$.

The hysteresis, also known as "voltage pull" – a derivative expression, depends on the direction, magnitude and duration of the preceding current as well as on the SOC. Hysteresis is known to result from thermodynamical entropic effects, mechanical stress, and microscopic distortions within the active electrode materials, which perform a two-phase transition during lithium insertion/extraction [105]. Thus, knowing the exact value of hysteresis is impractical. However, since the direction of the hysteresis voltage is opposite during charging and discharging, the OCV (at a certain SOC) can be approximately estimated by averaging the measured terminal voltage during charging and discharging when the entire experiment is performed using the same magnitude of current. Usually, a smaller current[1] is required in order to reach the OCV at both ends, i.e., to measure OCV from $V_\circ(0)$ to $V_\circ(1)$.

The SOC of the battery for OCV-SOC characterization is computed through Coulomb counting, which keeps track of the amount of Coulombs extracted from/inserted into the battery and computes the SOC as a ratio of remaining Coulombs and battery capacity. Hence, the knowledge of battery capacity is crucial for accurate OCV-SOC char-

[1] Using high current in OCV experiments will prevent reaching full battery (SOC = 1) or empty battery (SOC= 0).

acterization. Usually, the OCV-SOC characterization is performed on a new battery, hence, manufacturer specified capacity (also known as "rated capacity") or Coulomb counting from a fully charged battery to a fully empty battery at very low current values will serve as a fairly accurate estimate of the initial battery capacity.

However, the battery capacity is known to fade over time; hence, the BFG[2] needs to account for the capacity fade over time. At this point, the OCV-SOC characterization made using incorrect capacity will become more and more unsuitable. We will elaborate on this point later in this chapter.

The conclusion from the existing literature is that the OCV characterization changes with battery temperature. Hence, many existing approaches suggested computing OCV parameters at intermediate temperatures spanning possible battery usage. The OCV parameters at any other temperature are then obtained through interpolation. In these characterizations, the SOC is computed relative to a fixed (nominal) capacity of the battery, which, based on the manufacturer specifications, holds true only at the "nominal temperature" of the battery.

Hence, one of the focal points of this research, backed by data collected from 34 battery cells each at up to 16 different temperatures spanning $-25^{\circ}\mathrm{C}$ to $50^{\circ}\mathrm{C}$ at equal steps, is to investigate if the knowledge of battery capacity at different temperatures plays a role in OCV-SOC characterization. We computed the OCV characterization

[2] A BFG needs the battery capacity information in the "Coulomb counting equation" or process model, see [9].

curves at each temperature and analyzed the differences. Further, rather than using the rated capacity of the battery, we computed the battery capacity through slow discharge from full to empty at each temperature. The experiments were repeated on slightly older batteries as well. Our conclusion is that the OCV characterization curve remains the same, regardless of the temperature or age[3] of the battery, as long as the correct battery capacity is used in the characterization. The implication of this conclusion is significant: There is no need to repeat OCV characterization experiments at different temperatures — doing so at one temperature is sufficient for SOC tracking at all temperatures of battery operation as long as the capacity is estimated dynamically and continuously [7–9].

Once the pairs of $\{\mathrm{OCV}, \mathrm{SOC}\}$ values are collected, the next step is to model these data given later in (3.1) as a function. From a mobile computing perspective, the following constraints are relevant in selecting such a function:

- Fewest possible parameters
- Ease of computation and inversion.

Nowadays, rechargeable Li-ion batteries are used in many small electronic devices (e.g., smart watches, Google glass, etc.)and BFG is a desired capability in these devices, however, it must be able to function with limited amount of memory and computational needs. Since many of the BFG algorithms, such as the ones in [7–9], require

[3] Our analysis on aged batteries was done using a limited amount of data, hence, the conclusion about the aged batteries must be considered preliminary.

the computation of OCV and its derivative for a certain SOC and finding the SOC given a certain OCV (inversion of (3.1)), the selected OCV characterization function needs to be easily computable and invertible. The functions that we eventually select involve logarithms – we employ a computationally efficient, numerically stable approach to implementing logarithms through a Padé approximation.

The novel contributions and results of this chapter can be summarized as follows:

1. *Normalized OCV modeling.* We show that the OCV characterization is the same for any temperature; hence, doing so at one convenient temperature is adequate for BFG at all temperatures.

2. *New OCV-SOC models.* We analyze many models for OCV-SOC characterization and propose novel models that are suitable for mobile applications, i.e., with modest memory and computational complexity.

3. *Novel approaches for OCV model parameter estimation.* We propose several approaches for OCV parameter estimation.

4. *Performance comparison of OCV-SOC modeling as it relates to BFG.* We employ several performance evaluation metrics in order to assess the suitability of models in BFG. First, the required computational complexity of each model is summarized. Then, OCV modeling accuracy is compared at a certain complexity.

The rest of the section is organized as follows: A summary of existing and some

newly proposed OCV-SOC models are described in Section 3.2.2. Section 3.2.3 discusses the different OCV-SOC model parameter estimation techniques. The novel normalized OCV modeling is summarized in Section 3.2.4. In Section 3.2.5, we elucidate the different metrics for the selection of an OCV-SOC model for the BFG. Section 3.6 evaluates the different models based on HIL data collected across different batteries and varying operating temperatures. Finally, Section 3.7 concludes with a summary and future research directions.

3.2.2 OCV-SOC Models

The OCV of the battery has a monotonic relationship to the SOC. This relationship, usually modeled as a function, is the backbone of a BFG. In a general form, it is written as

$$V_{\circ}(s) = f(s) \tag{3.1}$$

where $V_{\circ}(s)$ denotes the OCV and $s \in [0, 1]$ denotes the SOC. In this section, we summarize important OCV-SOC functions.

Linear Regression-Based Models

A linear OCV model satisfies

$$V_{\circ}(s) = \mathbf{p}(s)^T \mathbf{k} \tag{3.2}$$

where $\mathbf{p}(s)^T$ is a row vector of linear/non-linear functions of s and $\mathbf{k}$ is the OCV parameter vector. It must be noted that the above expression is linear in the parameter

vector $\mathbf{k}$, not necessarily in s.

We consider the following linear-in-parameter models in our analysis with the objective of finding an appropriate model that satisfies the restrictions in the number of parameters and computational complexity while providing sufficient modeling accuracy:

$$\mathbf{p}_\circ(s)^T = \begin{cases} \begin{bmatrix} 1 & \frac{1}{s} \end{bmatrix} & \text{(i) Shepherd model [124] [77]} \\ \begin{bmatrix} 1 & s \end{bmatrix} & \text{(ii) Unnewehr universal model [124] [126]} \\ \begin{bmatrix} 1 & \ln(s) & \ln(1-s) \end{bmatrix} & \text{(iii) Nernst model [124] [122, 130]} \\ \begin{bmatrix} 1 & s & \frac{1}{s} & \ln(s) & \ln(1-s) \end{bmatrix} & \text{(iv) Combined model [88]} \\ \begin{bmatrix} 1 & s & \frac{1}{s} & \frac{1}{s^2} & \frac{1}{s^3} & \frac{1}{s^4} & \ln(s) & \ln(1-s) \end{bmatrix} & \text{(v) Combined+3 model [9]} \\ \begin{bmatrix} 1 & s & \dots & s^{L_1} & s^{-1} & \dots & s^{-L_2} \end{bmatrix} & \text{(vi) Polynomial model [121]} \\ \begin{bmatrix} 1 & e^s & \dots & e^{s^{L_1}} & e^{-s} & \dots & e^{-s^{L_2}} \end{bmatrix} & \text{(vii) Exponential model} \\ \begin{bmatrix} T_0(s) & T_1(s) & T_2(s) \dots & T_{L-1}(s) \end{bmatrix} & \text{(viii) Chebyshev model} \end{cases} \tag{3.3}$$

$$\mathbf{k}_\circ = \begin{bmatrix} K_0 & K_1 & K_2 & \dots K_M \end{bmatrix}^T \tag{3.4}$$

where, $M+1$ is the length of the parameter vector $\mathbf{k}$ (for example $M=1$ for the Shepherd Model and $M = L_1 + L_2$ for the polynomial model), L is the order of the Chebyshev polynomial and T_l is the l^{th} Chebyshev polynomial.

Next we discuss some of the above models.

Combined+3 Model The "Combined+3 model", first used by the authors in [7–9], was formed by including three additional terms $\frac{1}{s^2}$, $\frac{1}{s^3}$ and $\frac{1}{s^4}$ to the existing combined model. These terms are introduced to capture the sharp decline (or "knee") in the OCV

as the SOC reaches zero. It is noticed that, such sharp knees are found in the majority of the batteries except for a few battery chemistries. For batteries without such knee shape in the OCV-SOC characterization, the Combined model alone will be adequate to accurately capture the OCV curve.

Chebyshev Model The Chebyshev model is proposed in this chapter due to its numerically stable characteristics in fixed point implementations and its known property of minimizing the maximum error (i.e., uniform error across all SOC values) [93] [4].

Let us define $s_c \in$ [-1,1] as

$$s_c = 2s - 1 \tag{3.5}$$

The function $V_\circ(s)$ can be approximated in terms of the Chebyshev polynomial as

$$V_\circ(s) \approx \sum_{i=0}^{L-1} K_i T_i(\bar{s}_{ck}) \tag{3.6}$$

where

$$K_0 = \frac{1}{L}\sum_{k=1}^{L} V_\circ(\bar{s}_{ck}) \tag{3.7}$$

$$K_i = \frac{2}{L}\sum_{k=1}^{L} V_\circ(\bar{s}_{ck})\, T_i(\bar{s}_{ck}) \tag{3.8}$$

and

$$\bar{s}_{ck} = \cos\frac{\pi(2k-1)}{2L}, \quad k = 1, 2, ..., L \tag{3.9}$$

$$T_i(\bar{s}_{ck}) = \cos\frac{i(2k-1)\pi}{2L}, \quad i < L \tag{3.10}$$

Clenshaw Recurrence for the Evaluation of Chebyshev Model Once the coefficients of the Chebyshev expansion are known, then $V_\circ(s)$ for any given $s \in [0,1]$ can be computed using the backward recurrence relations

$$\begin{aligned} y_{L+1} &= 0 && (3.11)\\ y_L &= 0 && (3.12)\\ y_{L-1} &= K_{L-1} - y_{L+1} + 2y_L s_c && (3.13)\\ &\vdots \\ y_1 &= K_1 - y_3 + 2y_2 s_c && (3.14)\\ y_0 &= K_0 - y_2 + 2y_1 s_c && (3.15)\\ V_\circ(s) &= (K_0 - y_2) + y_1 s_c = y_0 - y_1 s_c && (3.16) \end{aligned}$$

This numerically stable approach to the evaluation of functions expressed in Chebyshev series is known as Clenshaw's recurrence.

Padé Approximation of Natural Logarithm For OCV-SOC models that include natural logarithm, there needs to be an approximate way to implement them in smaller micro controllers and hardware logic circuits. We use the following expression, which gives a very accurate approximation of natural logarithm through Padé approximation [58]

$$\ln_{\text{pade}}(1-x) = \frac{0.01812x^5 - 0.30555x^4 - 1.30555x^3 - 2x^2 + x}{0.00396x^5 - 0.11904x^4 + 0.83333x^3 - 2.22222x^2 + 2.5x - 1} \tag{3.17}$$

The $\ln_{\text{pade}}$ above can be implemented efficiently using Horner's method as follows:

$$\ln_{\text{pade}}(1-x) = \frac{((((137x-2310)x+9870)x-15120)x+7560)x}{((((30x-900)x-6300)x-16800)x+18900)x-7560} \tag{3.18}$$

In Figure 3.5, the $\ln_{\text{pade}}$ approximation is compared with $\ln$ implemented using 5^{th} order Taylor series approximation and the $\ln$ implementation by Matlab. It shows that the Padé approximation is more accurate than Taylor series approximation of the same order[4]

Nonlinear Regression-Based Models

In this section, we review three non-linear models presented in the literature and present a new non-linear OCV-SOC model using a rational approximant.

For nonlinear modeling, we designate

$$V_{\circ,n}(s, \boldsymbol{\kappa}_\circ) \triangleq V_\circ(s) \tag{3.19}$$

where

$$\boldsymbol{\kappa}_\circ = [\kappa_0 \quad \kappa_1 \quad \dots \quad \kappa_M] \tag{3.20}$$

is the nonlinear OCV parameter vector.

[4] Indeed, higher order Taylor series approximations, up to more than 15^{th} order, were tested and the conclusion is that the Taylor series approximation did not improve the accuracy with respect to (3.18).

$$V_{\circ,n}(s,\boldsymbol{\kappa}_\circ) = \begin{cases} \kappa_0 + \kappa_1 s + \kappa_2\left(1-e^{-\kappa_3 s}\right) + \kappa_4\left(1-e^{-\frac{\kappa_5}{1-s}}\right) & \text{(a) Double exponential model [51]} \\ \kappa_0 - \frac{\kappa_1}{s} + \kappa_2 e^{-\kappa_3(1-s)} & \text{(b) Nonlinear exponential model 1 [81]} \\ \kappa_0 e^{-\kappa_1 s} + \kappa_2 + \kappa_3 s + \kappa_4 s^2 + \kappa_5 s^3 & \text{(c) Nonlinear exponential model 2 [28]} \\ \frac{\sum_{i=0}^{m}\kappa_i s^i}{1+\sum_{j=1}^{n}\kappa_j s^j}, m \geq 0, n > 0 & \text{(d) Rational approximant} \end{cases} \tag{3.21}$$

Rational Approximant: In (3.21), we propose a model based on rational approximant which gives very good modeling accuracy. The model based on rational approximant is proposed based on Padé approximation ideas of [58]. The coefficients $\kappa_0, \kappa_1, \ldots, \kappa_{m+n}$ of the rational approximant can be uniquely determined. This method is also known as m/n order rational approximant.

Hybrid or Piecewise Linear Models

It was noticed that using a single function to model the entire shape of $V_\circ(s)$ requires more parameters to reach a certain accuracy. The same level of accuracy can be obtained with fewer parameters through hybrid modeling. The objective is to simplify the models as much as possible, hence, only linear regression-based models are considered in this approach.

The intuition behind hybrid modeling is to approximate the OCV models as a few linear curves. For example, for some batteries, starting from a fully charged state, the OCV decreases gradually (and linearly) with SOC until a certain (low) SOC. After that, the OCV drops steeply (and linearly) with decreasing SOC. This particular battery can be represented using two straight lines: one for the gradual drop and the other for

the steep drop. Some other OCV-SOC characteristic curves could be approximated by three (or more) different models. We consider only two of the following general form:

$$V_{\circ}(s) = \begin{cases} \mathbf{p}_{\circ}^{i}(s)^{T}\mathbf{k}_{\circ}^{i} & \text{if } s \geq \zeta \\ \mathbf{p}_{\circ}^{j}(s)^{T}\mathbf{k}_{\circ}^{j} & \text{if } s < \zeta \end{cases} \tag{3.22}$$

where each of $\mathbf{p}_{\circ}^{i}(s)$ and $\mathbf{p}_{\circ}^{j}(s)$ denote one of the eight different OCV models in (3.3). In this chapter, we chose the combined model for $\mathbf{p}_{\circ}^{i}(s)$ and a 2^{nd} order polynomial model for $\mathbf{p}_{\circ}^{j}(s)$. It is possible to have $i = j$ with the exception that the corresponding parameters $\mathbf{k}_{\circ}^{i}$ and $\mathbf{k}_{\circ}^{j}$ respectively, have to be different.

When $\mathbf{p}_{\circ}^{i}(s)$ and $\mathbf{p}_{\circ}^{j}(s)$ represent two straight lines, the minimum number of parameters will be five: two parameters each for the straight lines and ζ.

OCV-SOC Table

An alternate approach to OCV modeling is to store values of OCV and SOC as a table. Table 3.5 shows such an OCV-SOC table made of 11 SOC points. It will become apparent after Section 3.2.4 that, only one OCV-SOC table is all that is required for fuel gauging at all temperatures. Generally, the BFG algorithm requires the derivative of the OCV function to be used in an extended Kalman filter-based SOC tracking approach; hence, the table needs to store the derivatives as well.

Table 3.5 was formed by uniformly sampling SOC. A nonlinear sampling can be employed for better accuracy. For example, selecting the number of samples in a "region" proportional to $\frac{dV_{\circ}(s)}{ds}$ will be more accurate than a table made of samples

selected linearly across $s \in [0, 1]$.

3.2.3 OCV Parameter Estimation Techniques

In this section, we describe approaches for estimating the parameters of the models described in Section 3.2.2. For accurate estimation, we need the $\{V_\circ(s), s\}$ pairs spanning $s \in [0, 1]$. We collect this data 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 $C/40$)[5].

Figure 3.6 shows the equivalent circuit of a battery when it is charged/discharged with a constant rate. First, we define the SOC at a given time as

$$s[t_k] \triangleq s \quad \text{at time } t_k \tag{3.23}$$

Considering the voltage measurement errors, the measured voltage is written as

$$z_v[t_k] \;\; = \;\; v[t_k] + n_v[t_k] \tag{3.24}$$

where $n_v[t_k]$ is the voltage measurement noise which is modeled as white Gaussian with standard deviation (s.d.) σ_v. Since the current during the OCV experiment is usually kept constant through highly accurate programmable charge/load devices, we assume that there is no noise involved in $i[t_k]$.

During the OCV experiment, the terminal voltage can be written as

$$z_v[t_k] = V_\circ(s[t_k]) + h[t_k] + i[t_k]R_0 + n_v[t_k] \tag{3.25}$$

[5] C/N is a common representation for the discharge/charge rate of a battery, where C denotes the battery capacity in Ah and N is in hours.

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

$$h[t_k] \propto i[t_k] \tag{3.26}$$

Hence, (3.25) can be rewritten as

$$z_v[t_k] = V_\circ(s[t_k]) + i[t_k]R_{0,h} + n_v[t_k] \tag{3.27}$$

where the *effective resistance*

$$R_{0,h} = R_0 + R_h \tag{3.28}$$

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

Linear Model Parameter Estimation

The parameters of the linear OCV-SOC model in (3.2) can be written as,

$$z_v[t_k] = \underbrace{\begin{bmatrix}\mathbf{p}_\circ(s[t_k])^T & i[t_k]\end{bmatrix}}_{\mathbf{p}[t_k]^T} \underbrace{\begin{bmatrix}\mathbf{k}_\circ \\ R_{0,h}\end{bmatrix}}_{\mathbf{k}} + n_v[t_k] \tag{3.29}$$

By considering a batch of N voltage observations, (3.29) can be written as

$$\mathbf{v} = \mathbf{P}\mathbf{k} + \mathbf{n} \tag{3.30}$$

where

$$\mathbf{v} = [v[t_1] \quad v[t_2] \quad \dots \quad v[t_N]]^T \tag{3.31}$$

$$\mathbf{P} = [\mathbf{p}[t_1] \quad \mathbf{p}[t_2] \quad \dots \mathbf{p}[t_N]]^T \tag{3.32}$$

$$\mathbf{n} = [n[t_1] \quad n[t_2] \quad \dots \quad n[t_N]]^T \tag{3.33}$$

$$\mathbf{k} = [\mathbf{k}_\circ \quad R_{0,h}]^T \text{an} \quad (\mathrm{M}+2) \quad \text{vector} \tag{3.34}$$

The least squares estimate of the parameter vector is given by

$$\hat{\mathbf{k}} = \left(\mathbf{P}^T\mathbf{P}\right)^{-1}\mathbf{P}^T\mathbf{v} \tag{3.35}$$

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

$$\hat{V}_\circ(s) = \mathbf{p}_\circ(s)^T\hat{\mathbf{k}} \tag{3.36}$$

where $\hat{\mathbf{k}}$ is formed by the first $M+1$ (3.4) elements of $\hat{\mathbf{k}}$.

In Figure 3.7, the estimated values of $R_{0,h}, R_0$, and R_h are shown using data collected on a Samsung battery with serial number EB575152 at different temperatures.

Total Least Squares Parameter Estimation

In contrast to the classical least squares (LS) problem, in which the solution to the linear system of equations $\mathbf{Pk} \approx \mathbf{v}$ assumes noise only in the observation vector $\mathbf{v}$, the total least squares (TLS) problem assumes that the elements of both $\mathbf{P}$ and $\mathbf{v}$ are contaminated by noise. This leads to the following formulation

$$\hat{\mathbf{k}}_{\mathrm{TLS}} = \arg\min_{\mathbf{k},\hat{\mathbf{P}},\hat{\mathbf{v}}} \left\| \begin{bmatrix} \mathbf{P} & \mathbf{v} \end{bmatrix} - \begin{bmatrix} \hat{\mathbf{P}} & \hat{\mathbf{v}} \end{bmatrix} \right\|_F \tag{3.37}$$

subject to

$$\hat{\mathbf{P}}\mathbf{k} = \hat{\mathbf{v}} \tag{3.38}$$

where $\begin{bmatrix}\mathbf{P} & \mathbf{v}\end{bmatrix}$ and $\begin{bmatrix}\hat{\mathbf{P}} & \hat{\mathbf{v}}\end{bmatrix}$ are matrices of equal size. The matrix $\mathbf{P}$ is noisy due to errors in computed SOC, as discussed in Section 3.2.4. Further discussion of the TLS approach and its method of solution can be found in [52, 72, 73].

Tikhonov Regularized Total Least Squares: In order to compute stable solutions to ill-conditioned problems, where singular values of data matrix $\mathbf{P}$ gradually decay to zero, Tikhonov's regularization method was employed in TLS, where a damping is added to each SVD component of the solution, consequently filtering out the the components corresponding to the small singular values. Golub et. al. [48] presented the Tikhonov regularized TLS approach, its regularizing properties, computational aspects and numerical examples.

Nonlinear Model Parameter Estimation For nonlinear models, we rewrite (3.39) in the following form

$$\mathbf{v}(\boldsymbol{\kappa}) \;=\; \mathbf{v}_{\circ}(\boldsymbol{\kappa}_{\circ}) + \mathbf{i}R_{0,h} + \mathbf{w} \tag{3.39}$$

where

$$\boldsymbol{\kappa} \;=\; [\boldsymbol{\kappa}_{\circ} \quad R_{0,h}]^T \tag{3.40}$$

$$\mathbf{v}_{\circ}(\boldsymbol{\kappa}_{\circ}) \;=\; \begin{bmatrix} V_{\circ,n}(s[t_1], \boldsymbol{\kappa}_{\circ}) & \dots & V_{\circ,n}(s[t_N], \boldsymbol{\kappa}_{\circ}) \end{bmatrix}^T \tag{3.41}$$

$$\mathbf{i} \;=\; [i[t_1] \quad i[t_2] \quad \dots, \quad i[t_N]] \tag{3.42}$$

and $\mathbf{w}$ is the noise vector.

The coefficients of the nonlinear regression-based models were computed using the Matlab optimization toolbox function for nonlinear least squares *lsqnonlin*. The nonlinear least squares problem solves the following problem

$$\hat{\boldsymbol{\kappa}} \quad = \quad \arg\min_{\boldsymbol{\kappa}} \|\mathbf{v} - \hat{\mathbf{v}}(\boldsymbol{\kappa})\| \tag{3.43}$$

Hybrid Model Parameter Estimation

It is well known that OCV is a monotonically increasing function of SOC. Consequently, we imposed the inequality constraint $\frac{dV_{\circ}(s)}{ds} \geq 0$ in order to obtain an OCV model with monotonic estimates of OCV with respect to SOC.

Furthermore, in hybrid models, the OCV model is assumed to follow a particular function until $s = \zeta$ and then assumes a different model structure. In order to realize such a model structure, the least squares problem is implemented with additional equality constraints as shown below:

$$\hat{\mathbf{k}}_1 \quad = \quad \arg\min_{\mathbf{k}_1} \left\| \left(\mathbf{P}^i(s)\mathbf{k}_1 - \mathbf{v} \right) \right\| \quad s \in [0, \zeta] \tag{3.44}$$

subject to

$$\frac{d\mathbf{P}^i\hat{\mathbf{k}}_1}{ds} > 0 \tag{3.45}$$

$$\mathbf{P}^i(s)\hat{\mathbf{k}}_1\Big|_{s=\zeta} - \mathbf{P}^j(s)\hat{\mathbf{k}}_2\Big|_{s=\zeta} = 0 \tag{3.46}$$

$$\frac{d\mathbf{P}^i(s)\hat{\mathbf{k}}_1}{ds}\Big|_{s=\zeta} - \frac{d\mathbf{P}^j(s)\hat{\mathbf{k}}_2}{ds}\Big|_{s=\zeta} = 0 \tag{3.47}$$

$$\frac{d^2\mathbf{P}^i(s)\hat{\mathbf{k}}_1}{ds^2}\Big|_{s=\zeta} - \frac{d^2\mathbf{P}^j(s)\hat{\mathbf{k}}_2}{ds^2}\Big|_{s=\zeta} = 0 \tag{3.48}$$

$$\hat{\mathbf{k}}_2 = \arg\min_{\mathbf{k}_2} \left\| \left(\mathbf{P}^j(s)\mathbf{k}_2 - \mathbf{v}\right) \right\| \quad s \in (\zeta, 1] \tag{3.49}$$

subject to

$$\frac{d\mathbf{P}^j\hat{\mathbf{k}}_2}{ds} > 0 \tag{3.50}$$

$$\mathbf{P}^i(s)\hat{\mathbf{k}}_1\Big|_{s=\zeta} - \mathbf{P}^j(s)\hat{\mathbf{k}}_2\Big|_{s=\zeta} = 0 \tag{3.51}$$

$$\frac{d\mathbf{P}^i(s)\hat{\mathbf{k}}_1}{ds}\Big|_{s=\zeta} - \frac{d\mathbf{P}^j(s)\hat{\mathbf{k}}_2}{ds}\Big|_{s=\zeta} = 0 \tag{3.52}$$

$$\frac{d^2\mathbf{P}^i(s)\hat{\mathbf{k}}_1}{ds^2}\Big|_{s=\zeta} - \frac{d^2\mathbf{P}^j(s)\hat{\mathbf{k}}_2}{ds^2}\Big|_{s=\zeta} = 0 \tag{3.53}$$

The constrained least squares algorithm software in the optimization toolbox of Matlab is used for the optimizations above.

3.2.4 Normalized OCV Modeling

So far in OCV model parameter estimation the SOC is assumed to be known, for example, the SOC in (3.27) is computed through Coulomb counting:

$$s[t_k] = s[t_{k-1}] + \frac{\eta \Delta_k}{3600C}\left(\frac{i[t_{k-1}] + i[t_k]}{2}\right) \tag{3.54}$$

where

$$\Delta_k = t_k - t_{k-1} \tag{3.55}$$

is the sampling interval, and

$$\eta = \begin{cases} \eta_c \leq 1 & \text{during charging} \\ \eta_d = 1 & \text{during discharging} \end{cases} \tag{3.56}$$

is the efficiency and C denotes the battery capacity (in Ampere-hours), which plays a major role in OCV characterization of the battery, as we describe below.

There are two different ways of characterizing OCV based on the assumption of the battery capacity:

$$C = \begin{cases} C_{\text{nom}} & \text{Nominal OCV modeling} \\ C_{\text{batt}} & \text{Normalized OCV modeling} \end{cases} \tag{3.57}$$

where C_{batt} is the total battery capacity (clearly defined in a subsequent paragraph).

We define

1. *Nominal OCV modeling*, when the same battery capacity $C = C_{\text{nom}}$ is used at all temperatures for computing the SOC in (3.27) and

2. *Normalized OCV modeling*, when the computed total battery capacity $C = C_{\text{batt}}$ at each temperature is used for computing the SOC in (3.27).

In Figure 3.9, we show the total capacities, computed by $C/30$ discharge from full to empty, of cells listed in Table 3.3 at different temperatures. The capacity vs. temperature is observed to take a concave form in general, i.e., the maximum total capacity is achievable only at a particular temperature.

Now, let us define the following terms related to battery capacity:

- **Maximum/full-charge battery open circuit voltage** $(\text{OCV}_{\text{max}})$**:** OCV_{max} refers to the OCV of the battery when it is fully charged, i.e.,

$$\text{OCV}_{\text{max}} = V_{\circ}(s)|_{s=1} \tag{3.58}$$

  It must be noted that due to the internal impedance of the battery and hysteresis, the measured cell voltage of the battery will be higher than OCV_{max} right after the battery is fully charged.

- **Minimum/empty battery open circuit voltage** $(\text{OCV}_{\text{min}})$**:** Similar to (3.58)

$$\text{OCV}_{\text{min}} = V_{\circ}(s)|_{s=0} \tag{3.59}$$

  where, due to the internal impedance of the battery and hysteresis, the measured cell voltage of the battery will be lower than OCV_{min} right after the battery is fully discharged.

- **Rated battery capacity, C_{r}:** Rated capacity is the manufacturer specified (labeled) capacity of the battery. Manufacturer specified capacity should be treated as an approximate measure of the true value of battery capacity at the specified temperature.

- **Total capacity, C_{batt} or $C_{\mathrm{batt}}(T)$:** Total battery capacity is the maximum amount of Coulombs that can be discharged from a battery at cell temperature T, starting from a fully charged battery corresponding to its $\mathrm{OCV} = \mathrm{OCV}_{\mathrm{max}}$ until the battery is fully discharged, i.e., until the battery terminal reaches $\mathrm{OCV} = \mathrm{OCV}_{\mathrm{min}}$ using an infinitesimal[6] load current.

- **Nominal capacity[7], C_{nom}:** Nominal capacity is the maximum amount of Coulombs that can be discharged from a battery at any cell temperature. In other words, nominal capacity is the maximum total capacity of the battery. The temperature at which the total battery capacity is maximum is defined as the **nominal**

[6] The series resistance of the battery prevents measured terminal voltage from reaching $\mathrm{OCV}_{\mathrm{min}}$ and hence fully discharging the battery. A smaller load current makes sure the battery is fully emptied as much as possible. The gradient of the OCV-SOC curve is very high (see Table 3.5 and a sample OCV-SOC plot in Figure 3.14) close to an empty battery, which helps minimize the errors in total capacity estimation by slow discharging.

[7] Most of the battery literature treats nominal capacity as exactly similar to the rated capacity. We introduce this slightly altered definition based on observations in the literature where the nominal capacity is treated as the maximum possible capacity. For example, in [134], the nominal capacity of a power plant is the maximum possible power output. Nominal capacity is technically achievable, but, not always practical. In the battery example, the nominal capacity is only achievable at the nominal temperature.

temperature of the battery, i.e.,

$$\hat{T}_{\text{nom}} = \arg\max_{T} C_{\text{batt}}(T) \tag{3.60}$$

$$C_{\text{nom}} = C_{\text{batt}}(\hat{T}_{\text{nom}}) \tag{3.61}$$

The total capacity C_{batt} increases with temperature up to the nominal temperature. Beyond that, the total capacity starts to decline with increasing temperature. The above definition implies that the nominal temperature of the battery varies from battery to battery. It must be noted that the manufacturer specified nominal temperature (and nominal capacity) are only approximate. Hence, for accurate BFG, the nominal temperature of the battery must be dynamically estimated — we will present those details in a future publication.

Table 3.1 lists the differences between nominal and normalized OCV modeling.

The conclusion is that the normalized OCV modeling has many advantages over nominal OCV modeling.

Figure 3.10 shows the quadratic model fitting of the total capacity estimates against temperature using data collected from 3 different cells of the same chemistry and type of battery. Once the model is fitted, the nominal temperature and nominal capacity of the battery can be estimated based on (3.60) and (3.61).

The nominal temperatures are estimated for all the cells listed in Table 3.3 and Figures 3.11 summaries the estimated nominal temperatures as a histogram. Based on these (limited number of) observations, the most cells had a nominal temperature of

Table 3.1: Comparison of Nominal vs. Normalized OCV modeling

Nominal OCV modeling	**Normalized OCV modeling**
The SOC at any temperature is relative to the fixed, nominal capacity of the battery	The SOC is relative to the total capacity of the battery at the given temperature
There is no need to know the total capacity at a temperature in order to compute the SOC	Need to know the total capacity in order to compute SOC at a particular temperature
Gives accurate (durable) OCV parameters only with a new battery	OCV characterization can be done using an aged battery as well
Capacity estimation becomes erroneous with an aged battery	Capacity estimation works well even for an aged battery

$22.5°\mathrm{C}$ and the average nominal temperature is $23.5°\mathrm{C}$. It must be noted that in most of the batteries in Table 3.3, the manufacturer specified nominal temperature is $25°\mathrm{C}$. Hence, for superior accuracy, a BFG must estimate (and track) the Capacity of the battery.

Figure 3.12 shows an example of normalized OCV modeling at different temperatures. It can be seen that regardless of the measured terminal voltages at different temperatures, the corresponding OCV curves are nearly identical. This is further illustrated later, using Figure 3.13.

Figure 3.13 shows the comparison of OCV-SOC curves obtained from nominal (top) and normalized (bottom) OCV characterization. The nominal OCV modeling used $C_{\mathrm{nom}} = 1.5\mathrm{Ah}$ in computing SOC at all temperatures and the normalized model-

ing used the computed total capacity at each temperature in order to compute SOC. It can be observed that in normalized OCV modeling, the curves computed at all temperatures are nearly identical.

Figure 3.14 shows the OCV-SOC characteristic curves, obtained on a new as well as aged battery at different temperatures. First, the OCV data was collected from a newly bought battery at $-25^{\circ}\mathrm{C}$, $-15^{\circ}\mathrm{C}$, $-5^{\circ}\mathrm{C}$, $15^{\circ}\mathrm{C}$, $25^{\circ}\mathrm{C}$ and $35^{\circ}\mathrm{C}$ and $45^{\circ}\mathrm{C}$. Then, the battery was left for "self aging" for approximately one year. Then, the OCV data was collected from this aged battery at $-20^{\circ}\mathrm{C}$, $-10^{\circ}\mathrm{C}$, $0^{\circ}\mathrm{C}$, $10^{\circ}\mathrm{C}$, $20^{\circ}\mathrm{C}$ and $30^{\circ}\mathrm{C}$, $40^{\circ}\mathrm{C}$ and $50^{\circ}\mathrm{C}$[8]. It can be seen that no significant difference of patterns was found with respect to the age and temperature of the battery. *Consequently, a single normalized OCV-SOC curve in conjunction with an adaptive battery capacity estimation algorithm can be used in the BFG to obtain accurate SOC estimates for varying usage conditions over the lifetime of a battery*.

3.2.5 OCV-SOC Model Metrics For Selection of the Best Model

In this section, we develop a systematic approach to select the best approach for OCV models. This is done by computing several *error metrics* and *model evaluation metrics*, described below, and then by ranking the models based on *Borda counts.*

[8] There was no particular reasons for the selection of these temperatures. At first, it was decided to do the tests in $10^{\circ}\mathrm{C}$ intervals and later it was decided to to the tests (on aged batteries) at slightly different temperature.

OCV Prediction Error Metrics

The following four error metrics are used for predicting $\mathbf{v}$, defined in (3.31) to evaluate the models:

Best Fit

$$\mathrm{BF}(\%) = \left(1 - \frac{\|\hat{\mathbf{v}} - \mathbf{v}\|}{\|\mathbf{v} - \bar{\mathbf{v}}\|}\right) \times 100 \tag{3.62}$$

R^2 **Fit**

$$\begin{aligned} \mathrm{R}^2(\%) &= \left(1 - \frac{\|\hat{\mathbf{v}} - \mathbf{v}\|^2}{\|\mathbf{v} - \bar{\mathbf{v}})\|^2}\right) \times 100 \\ &= \left(\frac{\|\hat{\mathbf{v}} - \bar{\mathbf{v}}\|^2}{\|\mathbf{v} - \bar{\mathbf{v}})\|^2}\right) \times 100 \\ &= [\mathrm{BF}\,(2 - \mathrm{BF})] \times 100 \end{aligned} \tag{3.63}$$

Max Error

$$\mathrm{ME} = \max_i \{|\mathbf{v}_i - \hat{\mathbf{v}}_i|\} \tag{3.64}$$

Root-Mean Square (RMS) Error

$$\mathrm{RMS} = \frac{\|\mathbf{v} - \hat{\mathbf{v}}\|}{\sqrt{N - M}} \text{ or } \sqrt{MSE} \tag{3.65}$$

where N is the number of data points, M is the number of parameters, $\hat{\mathbf{v}}$ is the predicted value of $\mathbf{v}$ using the estimated parameters, for example, for linear models

$$\hat{\mathbf{v}} = \mathbf{P}\hat{\mathbf{k}} \tag{3.66}$$

and

$$\bar{\mathbf{v}} = \frac{1}{N}\sum_{i=1}^{N}\hat{\mathbf{v}}(i) \tag{3.67}$$

Model Evaluation Metrics

Model evaluation metrics consider the trade off between the number of model parameters and the number of data points. The following four metrics are important ones.

Akaike's Information Criterion (AIC) If the models are fitted using Least Squares, then [23] suggests the following analog of AIC:

$$\mathrm{AIC} = N\ln\left(\frac{S_2}{N}\right) + 2\left(M+1\right) \tag{3.68}$$

where

$$S_2 \;=\; \sum_{i=1}^{N}\mathbf{e}_i^2 \tag{3.69}$$

with

$$\mathbf{e} \;=\; \mathbf{v} - \hat{\mathbf{v}} \tag{3.70}$$

In the above S_2 is the sum of the squares of errors (SSE), $\mathbf{e}_i$ is the i^{th} element of the residual vector $\mathbf{e}$ and M is the number of parameters in the OCV model. *The better the model, the lower the AIC.*

Akaike's Information Criterion 2 (AIC2) A second version of AIC, given below, is useful when when $N >> M$

$$\mathrm{AIC2} = \ln\left[\mathcal{L}_f\left(1+\frac{2p}{N}\right)\right] \tag{3.71}$$

where the loss function is defined as

$$\mathcal{L}_f = \frac{\mathbf{e}^T\mathbf{e}}{N} \tag{3.72}$$

Akaike's Final Prediction Error (FPE)

$$\text{FPE} = \mathcal{L}_f \left[\frac{1+\frac{M}{N}}{1-\frac{M}{N}}\right] \tag{3.73}$$

Bayesian Information Criterion (BIC) The derivation of BIC assumes equal priors on each model and noninformative priors on the parameters, given each model. The goal of the BIC is to find the best (i.e. highest posterior probability) model for prediction.

$$\text{BIC} = 2\left(L_N\right) + \left(M+1\right)\ln N \tag{3.74}$$

The negative log-likelihood given the pdf of the residuals (assuming normal or Gaussian) conditioned on the parameters $\mathbf{k}$ and the s.d. of residuals σ is given by

$$L_N = -\ln\left\{L\left(\mathbf{k};\mathbf{e}\right)\right\} = \sum_{i=1}^{N}\left\{\left(\frac{\mathbf{e}_i^2}{2\sigma^2}\right) + 0.5\ln\left(2\pi\sigma^2\right)\right\} \tag{3.75}$$

where, L_N is the negative log-likelihood, L is the likelihood, $\mathbf{k}$ is the parameter vector which minimizes L_N, and σ is the s.d. of the residuals $\mathbf{e}$.

Minimum Description Length (MDL)

$$\text{MDL} = \mathcal{L}_f\left[1 + \frac{M\ln N}{N}\right] \tag{3.76}$$

Combining Rankings

The Borda count [34] is an easy, intuitively appealing, and powerful method for combining different rankings. In this chapter, we use Borda's count method to combine the rankings of different model selection metrics. The Borda count is originally a voting method in which each voter gives a complete ranking of all possible alternatives. The highest ranked alternative or OCV Model (in an n-way vote) gets n votes (or points) and each subsequent alternative gets one less vote (i.e. number 2 gets $n-1$ votes and number 3 gets $n-2$ votes and so on). Then, for each alternative, all the votes (or points) are added up and the alternative (i.e. OCV Model) with the highest number of votes wins the election. The OCV Models are then ranked in descending order of votes (or points) to give the overall Borda ranking.

Computational Complexity Metrics

Most of the present day applications of BFG need to operate with limited amount of memory and computations. This metric aims to quantify the

1. Memory: Number of variables required to define the OCV-SOC characterization
2. Computations: The computational complexity in finding the OCV for a specific SOC.

In Table 3.2, we list the number of parameters to be stored in memory as well as the computational requirements for calculating OCV for a given SOC in terms of the num-

ber of additions (add), number of floating point multiplications (mult) and the number of logarithmic computations (ln).

Table 3.2: Memory and Computational Requirement of OCV-SOC Models

Model	Required Memory	Computation: SOC $\rightarrow$ OCV
Shepherd model (3.2.2)	2	$1\,\text{add} + 1\,\text{mult}$
Unnewehr model (3.2.2)	2	$1\,\text{add} + 1\,\text{mult}$
Nernst model (3.2.2)	3	$2\,\text{add} + 2\,\text{mult} + 2\,\ln$
Combined model (3.2.2)	5	$4\,\text{add} + 4\,\text{mult} + 2\,\ln$
Combined+3 model (3.2.2)	8	$7\,\text{add} + 7\,\text{mult} + 2\,\ln$
Polynomial model (3.2.2)	$L+1$	$L\,\text{add} + L\,\text{mult}$
Exponential model (3.2.2)	$L+1$	$L\,\text{add} + L\,\text{mult} + L\,\exp$
Chebyshev model (3.2.2)	L	$1\,\text{add} + 2\,\text{mult}$
Double exponential (3.2.2)	5	$5\,\text{add} + 3\,\text{mult} + 2\,\exp$
Nonlinear exponential 2 (3.2.2)	6	$1\,\text{add} + 2\,\text{mult}$
Rational approximant (3.2.2)	2	$1\,\text{add} + 2\,\text{mult}$
Hybrid Models (3.2.2)	variable	model dependent

3.2.6 HIL Data Analysis

The *OCV characterization data* was gathered by hardware-in-the-loop experiments conducted on 9 different batteries at 16 different temperatures from -25^0C to $+50^0C$

Table 3.3: Hardware-in-the-loop Battery OCV Characterization Data Overview

Battery Supplier	Battery Type	Number of Batteries	Rated Capacity (mAh)
Samsung	EB575152VA	4	1500
Samsung	EBL1A2GBA	4	1650
Samsung	EB555157VA	2	1750
Samsung	EB504465VA	4	1500
Samsung	AB463651	4	960
Samsung	L1G6LLAGS	2	2100
Nokia	BP-4L	4	1500
LG	LGIP-530B	2	1100
RIM	F-S1	2	1270
RIM	M-S1	4	1500
Huawei	HB4Q1	2	1670

in steps of 5^0C using a Tenney thermal chamber and an Arbin BT-2000 cell test station. The battery manufacturers, battery types, number of batteries used in the experiment, and their rated capacities are summarized in Table 3.3.

For each cell, the data is collected in the following manner: First, the cell is rested at the test temperature at full charge. Then, the cell was discharged at $C/30$ rate until fully discharged to the minimum voltage (3 V) followed by a short rest period.

Subsequently, the cell was charged at $C/30$ rate until fully charged (4.2 V). The low rates were used to minimize the cell dynamics. Some optional charge/discharge pulses (approximately $C/3$ rate) were used to determine (initialize) the cell resistances and time-constants. The OCV was computed by averaging the cell voltage as a function of SOC under discharge and charge. This eliminates the presence of hysteresis and ohmic resistance effects in the final OCV function.

Figure 3.15 shows a sample OCV model obtained for a particular battery cell (cell 4 of Samsung EB575152) at $50°\mathrm{C}$. The modeling error at a particular SOC is computed by subtracting modeled OCV from the average of measured voltages (during charging and discharging).

In Figure 3.16, we give a comprehensive summary of OCV modeling showing the errors of different OCV modeling approaches, across different batteries and temperatures. First, the average value of the OCV modeling error of a each battery cell at each temperature is computed by averaging the OCV modeling error across all SOC values (i.e., by averaging the OCV modeling error shown in the bottom plot of Figure 3.15). Then, in Figure 3.16, the average error across all the battery cells listed in Table 3.3 is shown as a box plot[9] for each temperature. The Chebyshev model must be noticed for its smallest error across all temperatures and for its consistency across different batteries as indicated by the (compact) sizes of boxes and the lack of outliers across all

[9] Using $boxplot$ function in Statistics Toolbox of Matlab. On each box, the central mark is the median, the edges of the box are the 25^{th} and 75^{th} percentiles, the whiskers extend to the most extreme data points not considered outliers, and outliers are plotted individually.

Table 3.4: Model Selection Metrics Rankings

OCV Model	Rankings									
	AIC	**AIC2**	**FPE**	**BIC**	**MDL**	**RMSE**	R^2	**BF**	**Max Error**	**Borda**
Chebyshev Model	1	1	1	1	1	1	1	1	4	1
Rational Approximant	2	2	2	2	2	2	2	2	5	2
Polynomial (4th order)	3	3	4	3	4	3	4	3	1	3
Hybrid Model	5	5	3	4	3	4	3	4	2	4
Combined Model	6	6	5	6	5	5	5	5	7	5
Exponential Model	7	7	6	5	6	6	6	6	6	6
Nonlinear Exponential Model 2	8	8	7	7	7	7	7	8	9	7
Combined+3 Model	4	4	12	11	12	7	11	7	3	8
Nernst Model	9	10	8	8	8	9	8	9	10	9
Unnewehr Universal Model	11	11	9	10	9	10	9	10	11	10
Double Exponential Model	10	9	10	9	10	11	12	12	12	11
Shepherd Model	12	12	11	12	11	12	10	11	8	12

temperatures.

The results of the model evaluation metrics and error metrics have been combined and presented in Table 3.4. The Chebyshev model performs the best among all the models based on the metrics presented in the chapter.

3.2.7 Conclusions

This chapter presented several approaches and analysis related to the open circuit voltage characterization of Li-ion batteries with a special emphasis on mobile and wearable electronic device applications. Memory and computational power becomes scarce (due

to prioritization) for BFG in these applications, yet, the BFG capabilities such as prediction of time to shut down and remaining useful life are considered desirable features to have.

Offline characterization of open circuit voltage is crucial for BFG, especially in memory and computing power constrained electronic applications. We include several existing OCV modeling functions and many newly proposed OCV modeling approaches in a performance analysis in terms of OCV modeling accuracy, computational complexity and memory requirements.

It was discovered that significant savings in memory can be achieved by the proposed normalized OCV modeling, in which the SOC at each temperature is defined in terms of the temperature specific capacity. As a result, we found that, the OCV-SOC characterization remains practically the same regardless of the temperature or age of the battery. This finding is crucial for online battery capacity estimation, i.e., the normalized OCV parameters will serve as a means for accurate, online battery capacity estimation, such as the one presented in [8].

Apart from the benefits in mobile applications, the normalized OCV modeling approach reported in this chapter has impact on other applications such as in electric vehicles and energy storage grid where accurate capacity estimation is critical.

Table 3.5: A Sample OCV-SOC Table that Requires 11 Memory Space

s	$V_\circ(s)$	$\frac{dV_\circ(s)}{ds}$
0.0	3.0519	44.4073
0.1	3.6594	0.6830
0.2	3.7167	0.8131
0.3	3.7611	0.4807
0.4	3.7915	0.4393
0.5	3.8275	0.6055
0.6	3.8772	0.8113
0.7	3.9401	0.9777
0.8	4.0128	1.0915
0.9	4.0923	1.1822
1.0	4.1797	1.3405

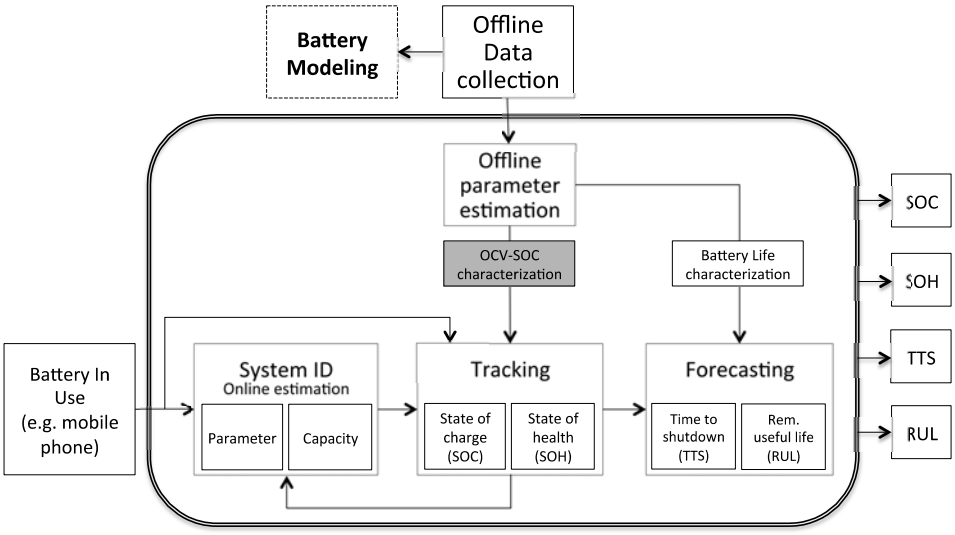
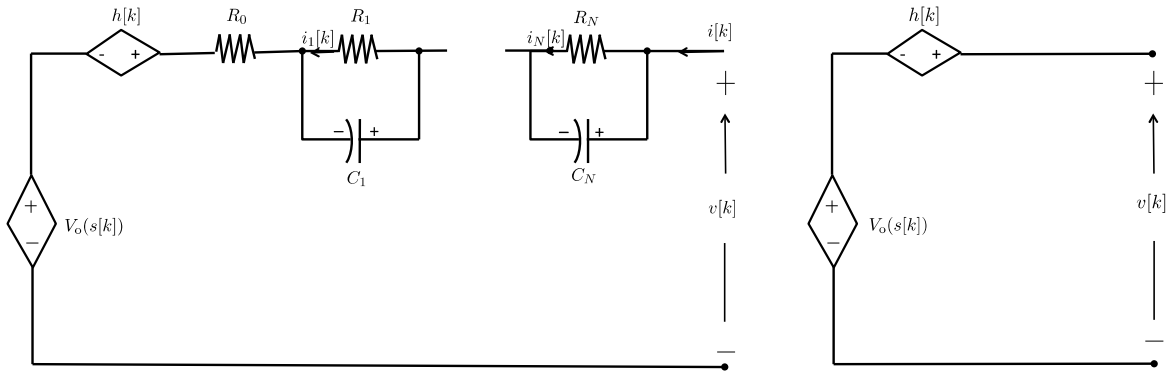


Fig. 3.3: Elements of a BFG. The focus of this chapter, OCV-SOC characterization is shaded in grey. Details of System ID block can be found in [7, 8] and the details of SOC tracking block can be found in [9].



(a) Battery under dynamic current (b) Rested battery

Fig. 3.4: Equivalent circuit of a battery when it is experiencing dynamic current vs. when it is rested. The measurable voltage across battery terminals is indicated by $v[k]$ and the current through the battery is $i[k]$. The currents through $R_1, \ldots, R_N$ are indicated by $i_1[k], \ldots, i_N[k]$, $h[k]$ denotes hysteres and R_0 denotes the series resistance.

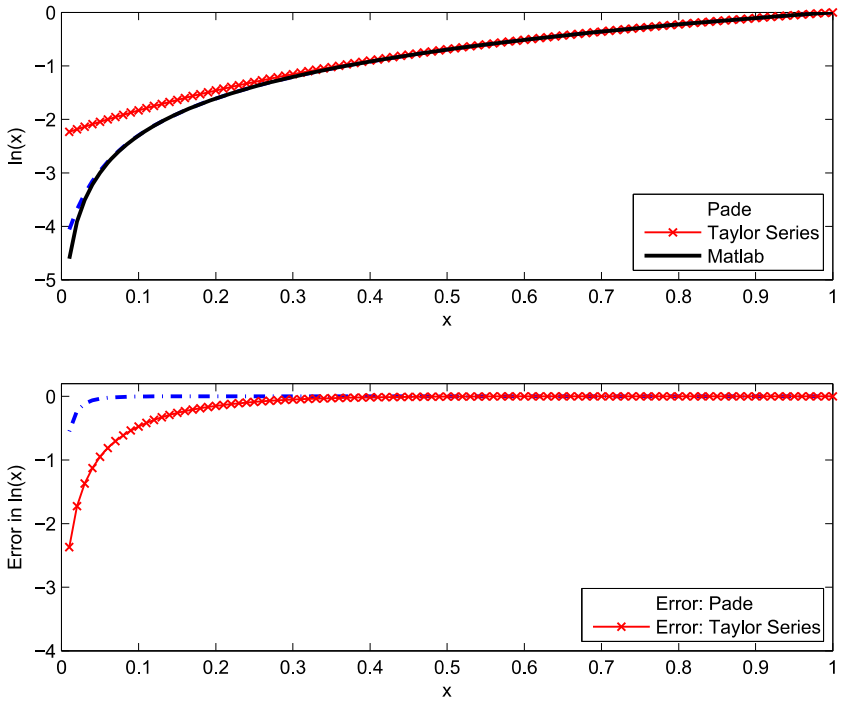


Fig. 3.5: Computing $\ln$ through Padé approximation. Top plot shows $\ln(x)$ computed by Padé approximation (3.18), Taylor series approximation and Matlab (version 2011b). The error plots in the bottom are relative to the Matlab version.

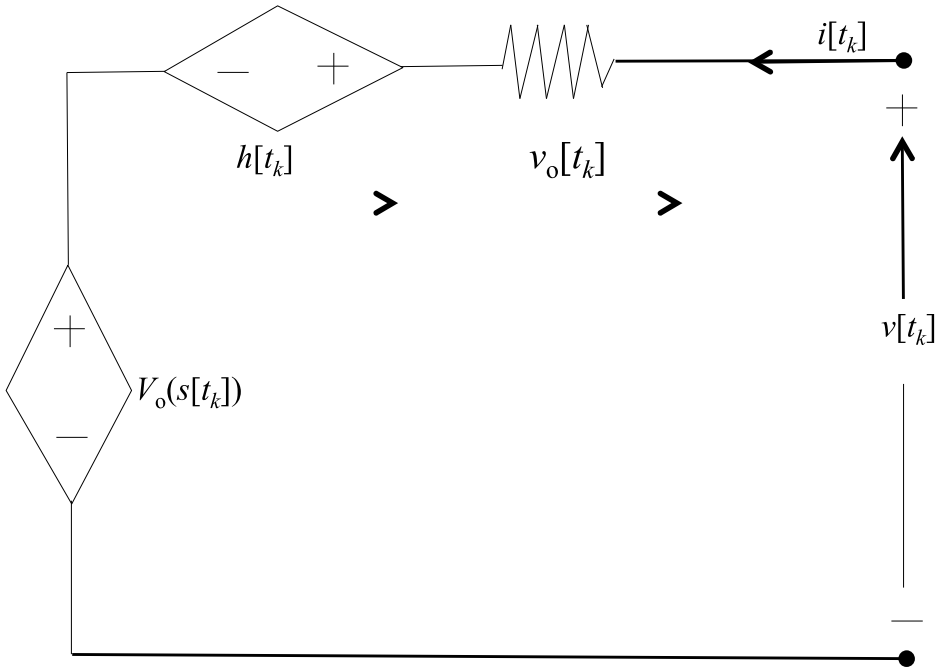


Fig. 3.6: Equivalent circuit of a battery when it experiences constant current. When the current is constant for a long time, the internal capacitors of the battery becomes saturated and the battery equivalent circuit becomes just a resistor.

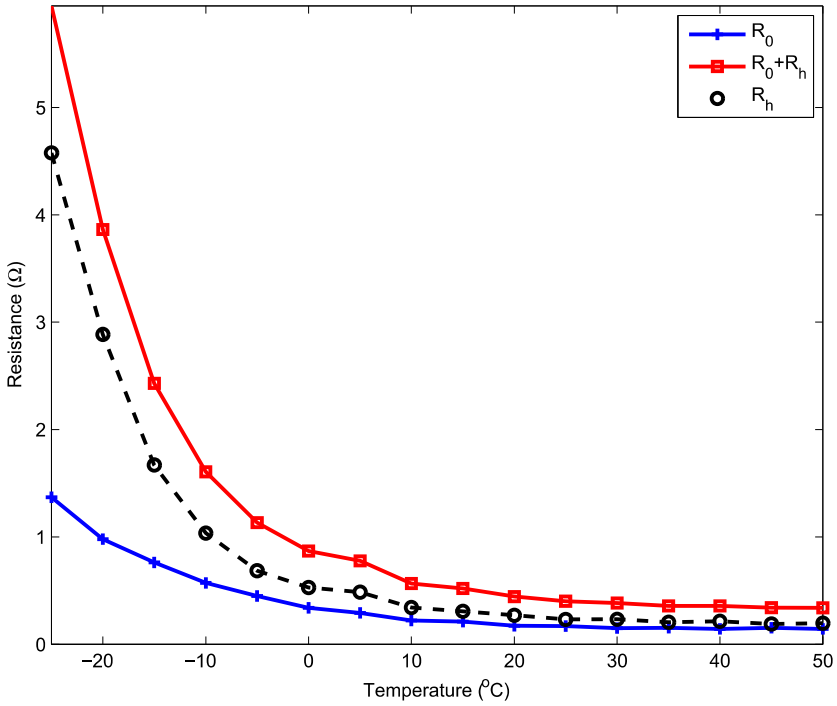


Fig. 3.7: Estimated resistance values. *Top:* The estimated value of effective resistance $R_{0,h}$ is compared with the series resistance R_0, which was computed by applying a discharge pulse on the battery. *Bottom:* Estimated constant-current hysteresis equivalent resistance, $R_h = R_{0,h} - R_0$, which shows a similar temperature dependence as R_0.

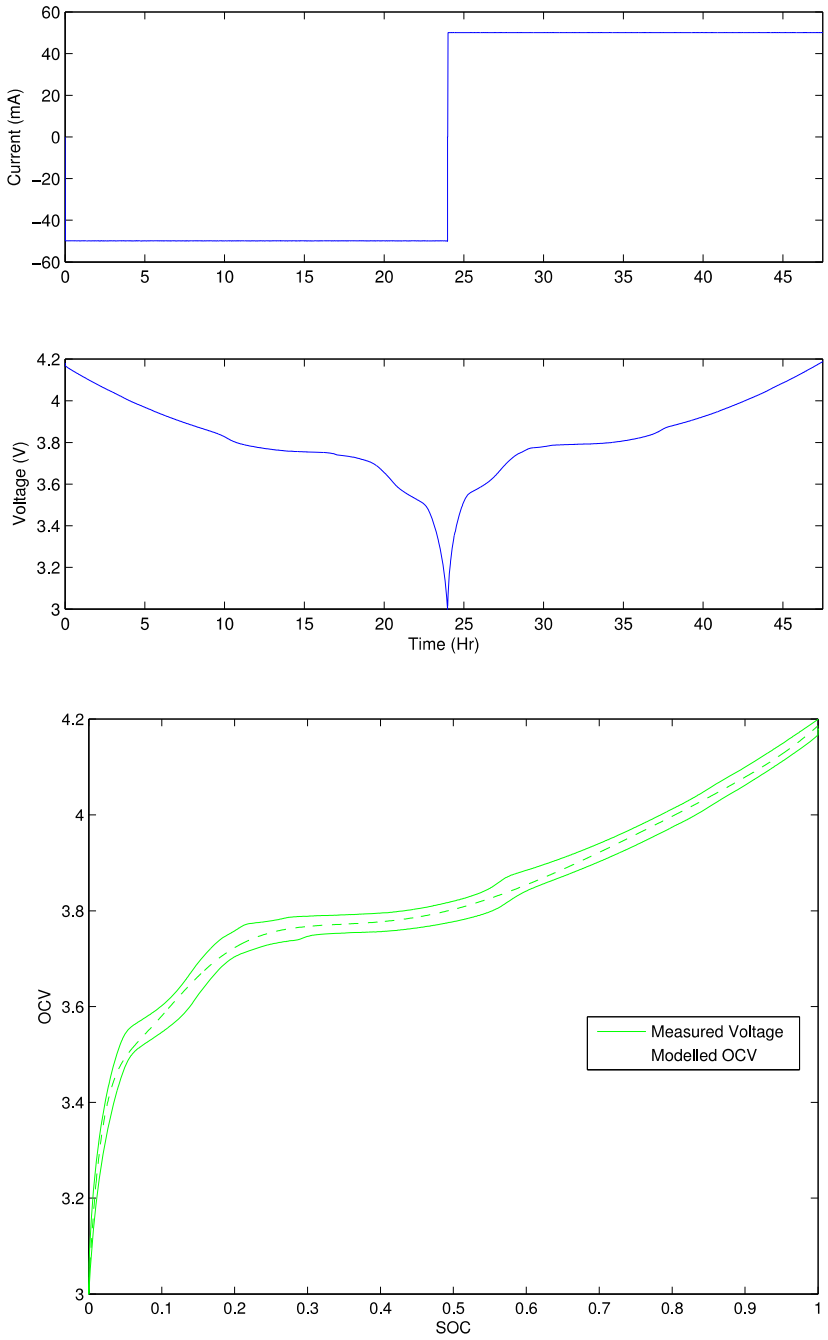


Fig. 3.8: OCV characterization data and sample OCV plot. Top two plots indicate the applied current $i[t_k]$ and measured voltage $z_v[t_k]$. The plot at the bottom shows the same $z_v[t_k]$ in the middle plot above against SOC, instead of time. The SOC was computed by Coulomb counting, explained later in (3.54). The OCV is computed using (3.36).

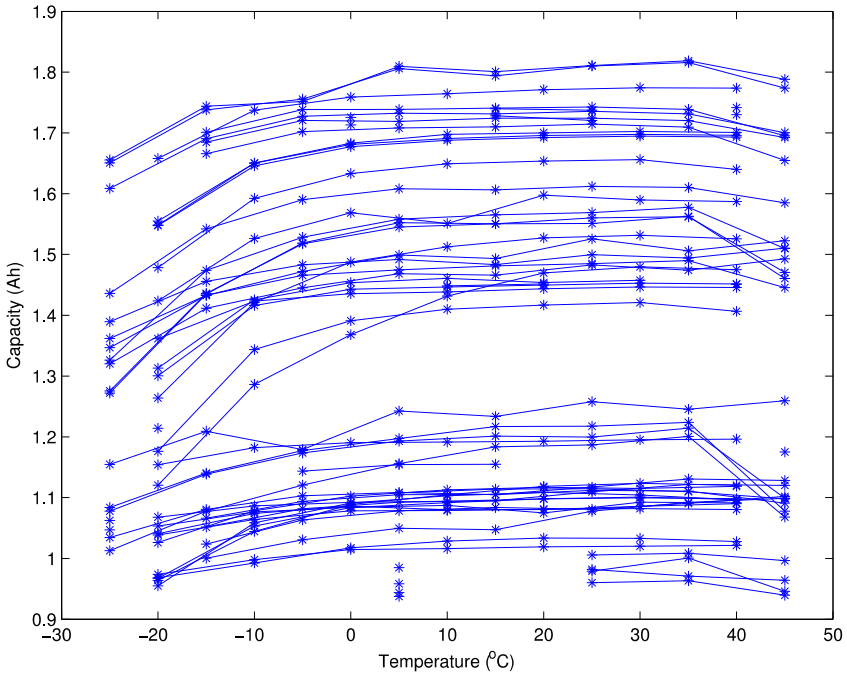


Fig. 3.9: Estimated battery capacities at different temperatures. The capacities are computed by $C/30$ discharge. Capacity measurements from 34 battery cells are plotted against temperature. Markers indicate the measurement and the lines connect the measurements that correspond to a certain cell.

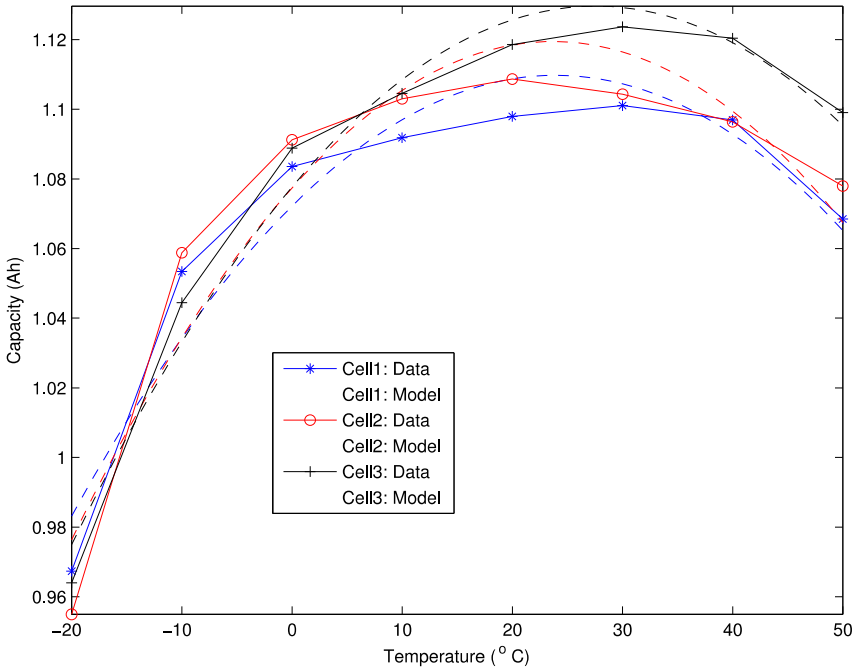


Fig. 3.10: A quadratic model fitting on total capacity estimates: Each measurement plot in Figure 3.9 is fitted with a quadratic model. Here we demonstrate quadrature model fitting on 3 different cells of the same chemistry and type. The corresponding model capacity is plotted in dashed line of the same color. The estimated nominal temperature (3.60) $\hat{T}_{\rm nom}$ that corresponds to the maximum capacity is shown in dash-dot line of the same color.

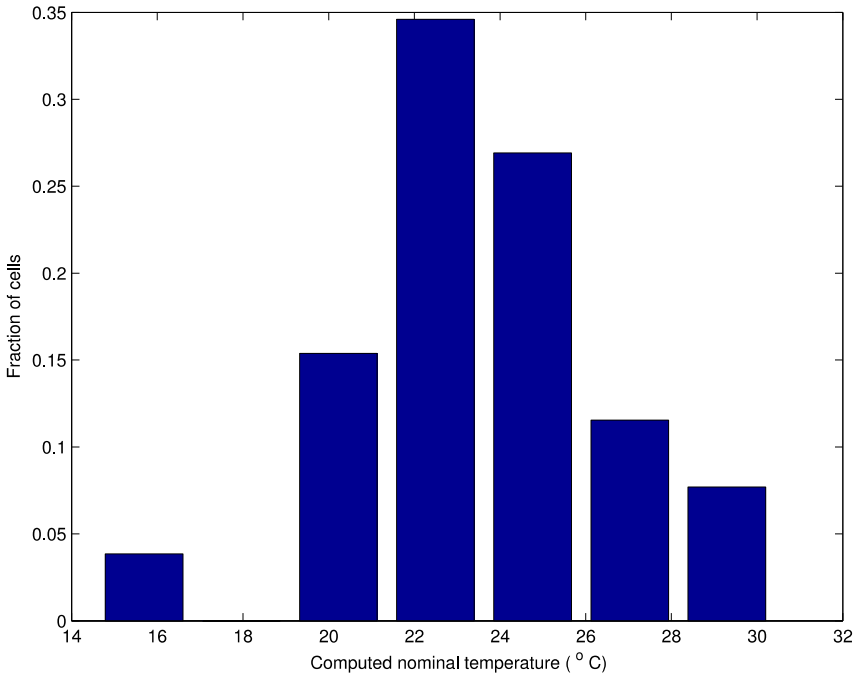


Fig. 3.11: Analysis of estimated nominal temperature : Estimated nominal temperature of 34 battery cells of different manufacturers is used for this experiment.

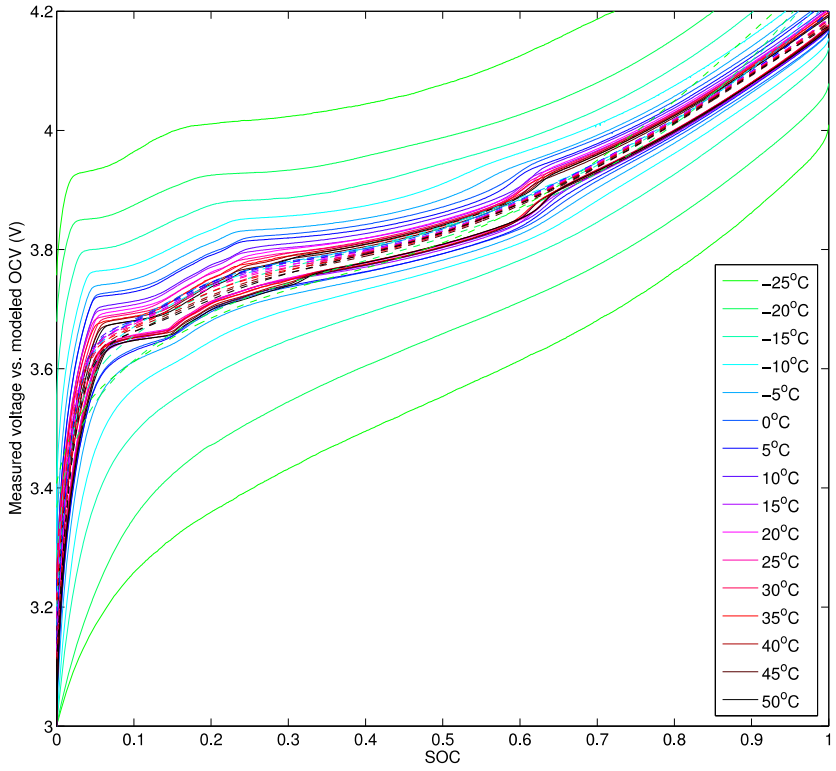


Fig. 3.12: Normalized OCV modeling. Measured terminal voltage is shown in solid and computed OCV is shown in dashed lines.

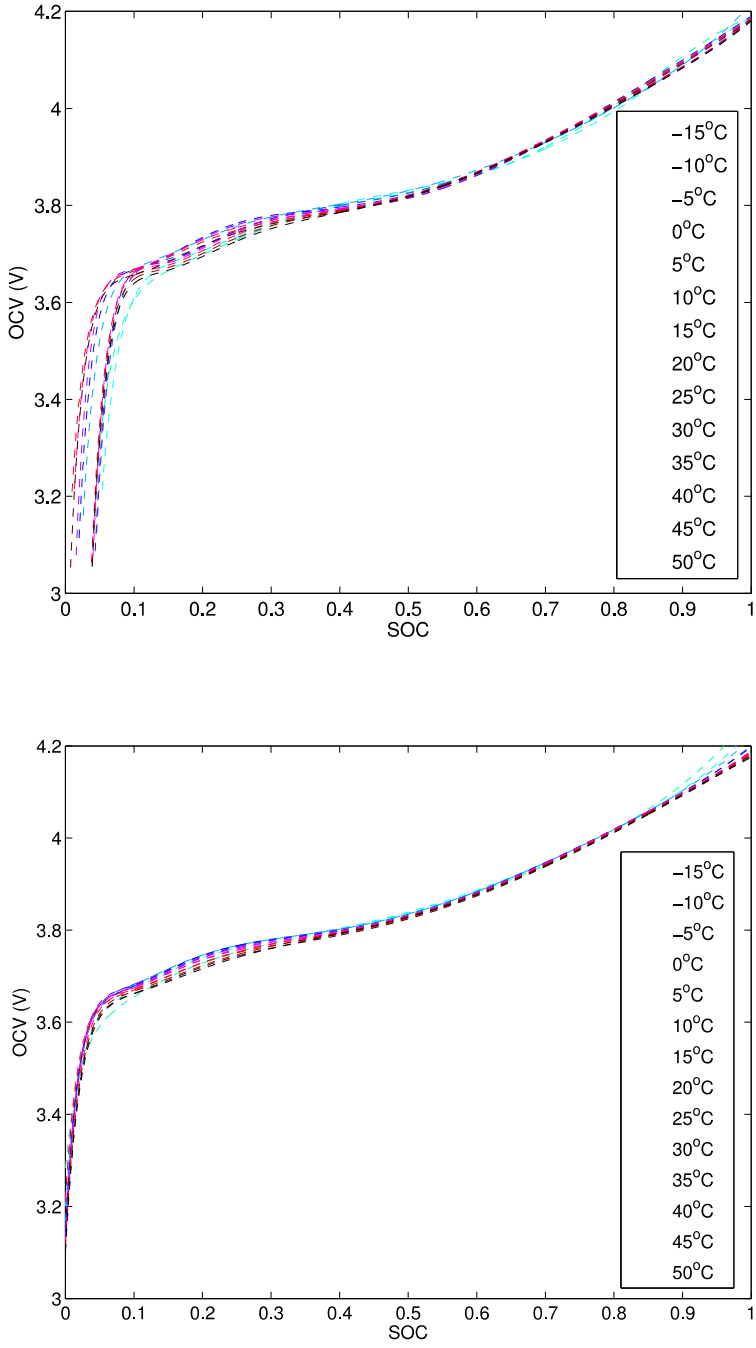


Fig. 3.13: Nominal vs. normalized OCV modeling. Top: Nominal OCV modeling uses $C_{\text{nom}} = 1.5\text{Ah}$ in computing SOC at all temperatures. **Bottom:** Normalized OCV modeling computes C_{batt} at each temperature before computing SOC.

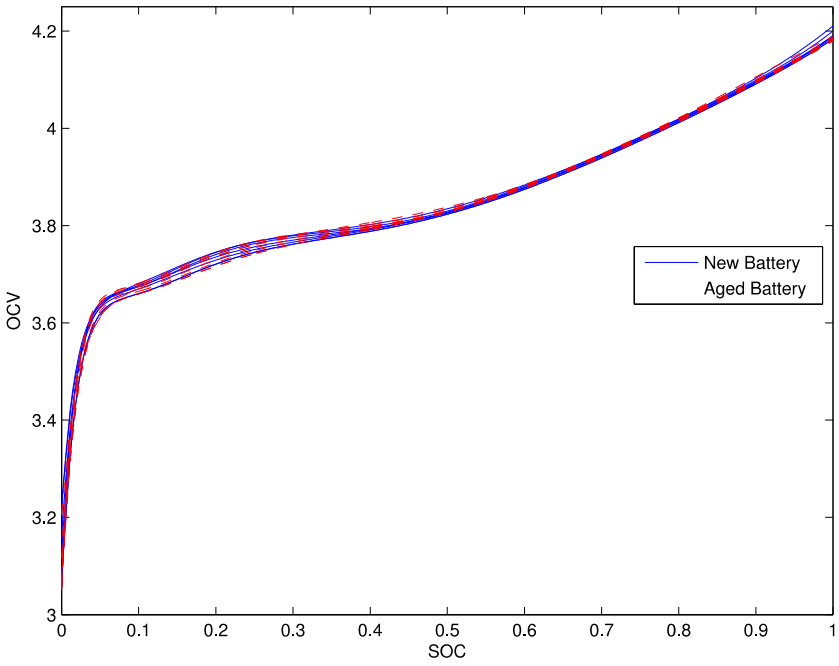


Fig. 3.14: OCV-SOC characterization curves of new vs. aged batteries. New battery curves are plotted in solid blue and aged battery curves are plotted in dashed red. Different curves of the same type correspond to temperatures ranging from $-25°\mathrm{C}$ to $50°\mathrm{C}$. *The new and aged battery OCV curves overlap.*

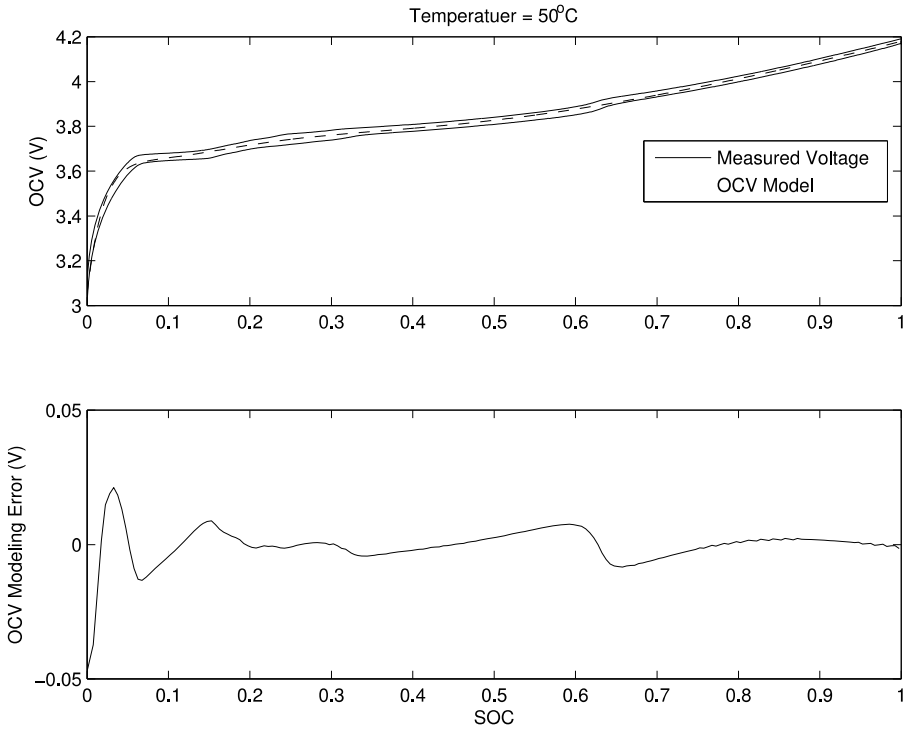
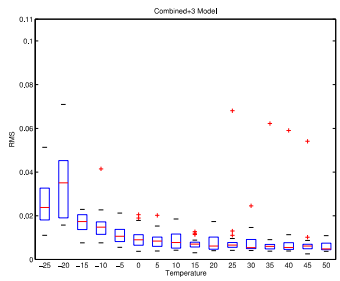
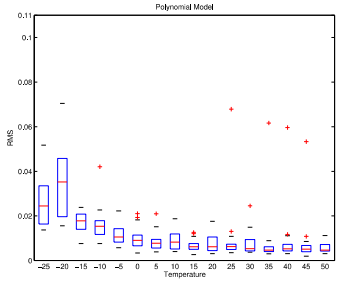


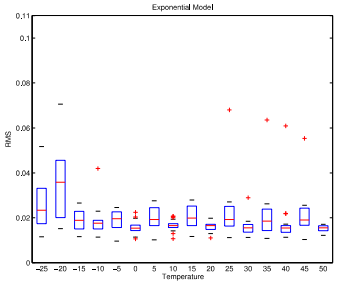
Fig. 3.15: Sample OCV modeling error. *Top:* Measured voltage and modeled OCV. *Bottom:* OCV modeling error.



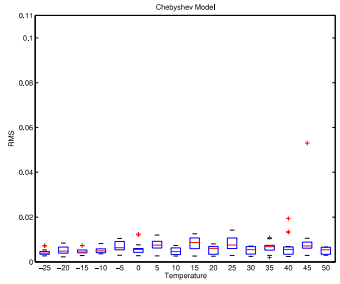
(a) Combined+3



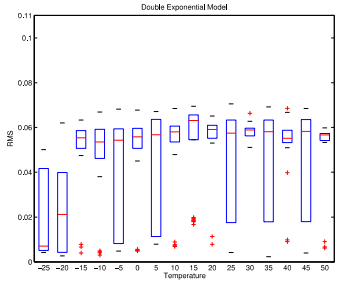
(b) Polynomial



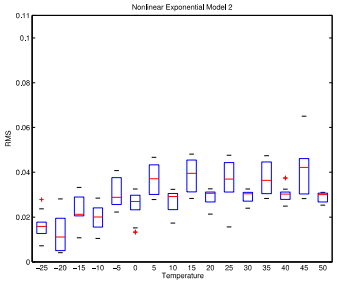
(c) Exponential



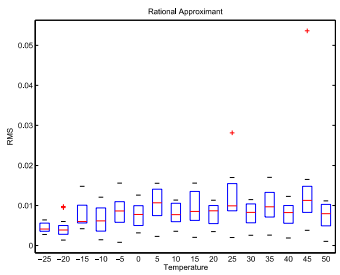
(d) Chebyshev



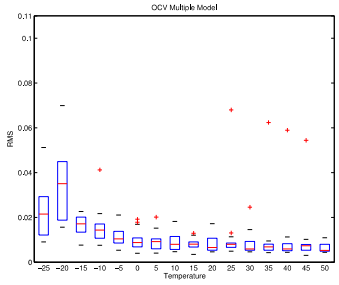
(e) Double Exponential



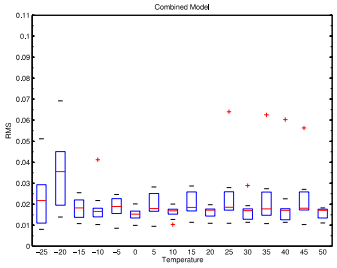
(f) Nonlinear Exponential 2



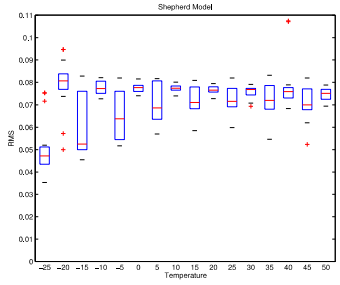
(g) Rational Approximant



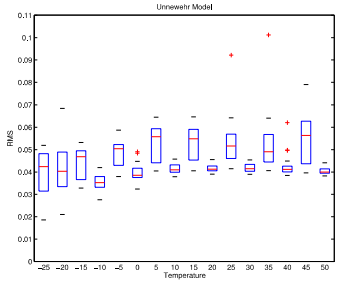
(h) Hybrid



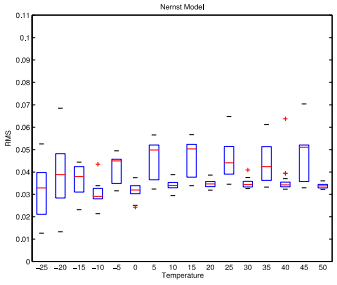
(i) Combined



(j) Shepherd



(k) Unnewehr



(l) Nernst

Fig. 3.16: RMS error across different battery cells. RMS value of the OCV modeling error across different battery cells is given as a box plot at each temperature.

3.3 Battery Fuel Gauge (BFG)

Figure 3.17 shows a block diagram of the BFG. We briefly describe each of the blocks below [7–9].

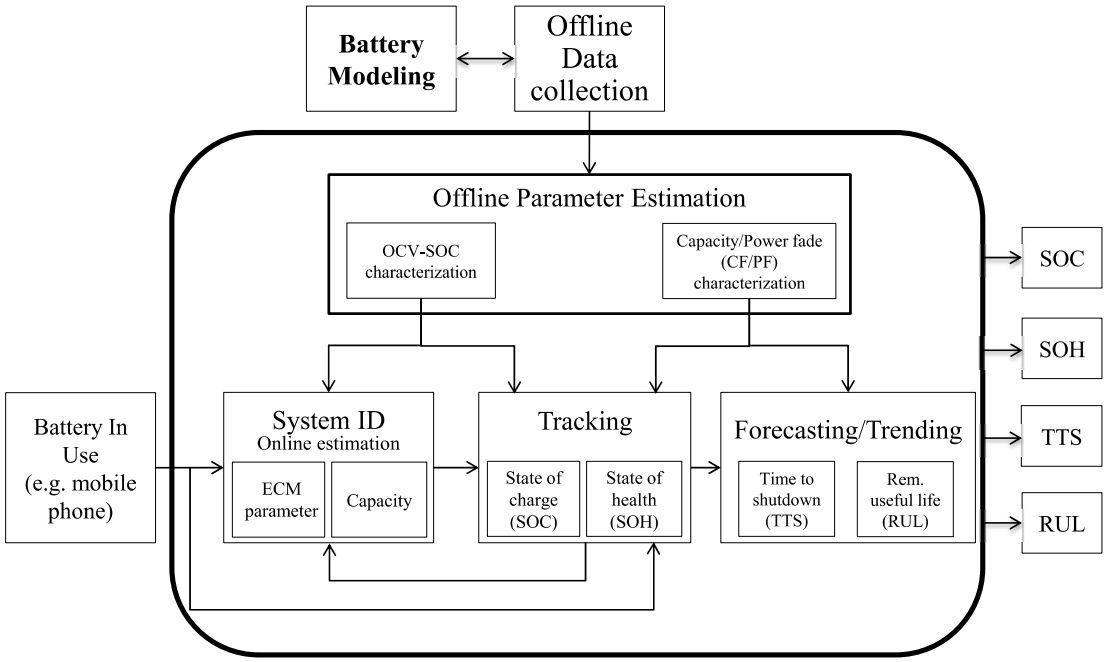


Fig. 3.17: Important Blocks of the Battery Fuel Gauge

- *Offline data collection.* The understanding of the behavior of a battery is achieved through a combination of physical, electrochemical model-based explanations and through data. For example, a particular battery cell having a certain maximum voltage and the fact that there exists a monotonic relationship between the OCV and SOC of the battery can be explained based on electrochemical understanding; the modeling of OCV-SOC relationship is achieved through data [84].

As far as the BFG algorithm development is concerned, the data collection plays an important part in understanding different models, such as, OCV-SOC, capacity fade (CF) and power fade (PF) models, that are required for accurate fuel gauging. It must be noted that data collection and battery modeling go hand in hand until adequate models are defined and the concomitant parameters estimated.

- *Battery modeling.* There are different paradigms in battery modeling as it applies to fuel gauging, such as equivalent electrical circuit model (ECM) based approaches [7–9], electrochemical modeling based approaches [113] and data driven approaches [26, 46, 65]. Our BFG is based on ECM; hence, our discussions and descriptions are related to ECM modeling based BFGs, where the following five aspects of the battery behavior are studied and corresponding models are developed.

    1. *OCV.* The characteristics of OCV-SOC relationship were extensively studied in the literature (see [84] and the references therein). It is understood that the OCV-SOC relationship is stable over temperature changes and aging, given the temperature and age-dependent capacity. It must be noted that, even though there exists a clearly understood OCV-SOC characteristic for which the parameters can be accurately estimated, the SOC cannot be simply computed based on the measured voltage; the *dynamic behavior of the battery* has to be accounted for before computing SOC based on

measured voltage, current and temperature.

2. *Battery capacity.* Battery capacity denotes the total amount of Coulombs that can be discharged from the battery. It is a key battery parameter in computing the SOC online.

3. *Dynamic behavior of the battery.* The measured terminal voltage of the battery is not always equal to the OCV. The *voltage drop* between the OCV and the battery terminal is modeled through ECM (also see Figure 3.18). In addition, different discrete states of battery operation, such as charge, discharge, rest and reset states should be considered.

4. *Capacity fade (CF).* Battery capacity is known to degrade as a result of aging, temperature and depth of discharge (DoD); frequent DoD values closer to both 0 and 1 negatively affect capacity [76]. Unlike OCV modeling, which requires data from just one charge-discharge cycle, CF modeling requires extensive data, at least several tens (or even hundreds) of cycles; the CF modeling requires continuous data from new-to-dead battery under various temperature, aging and DoD scenarios.

5. *Power fade (PF).* Similar to CF, the ECM parameters change as a result of temperature, aging and DoD patterns a battery undergoes. In particular, the internal resistances increase as a result of aging and this limits the amount of power that can be drawn from the battery, resulting in *power fade.* In addition, it must be noted that the available power varies instantaneously as

a result of temperature changes and this adds an extra layer of difficulty in modeling.

- *Offline parameter estimation.*
  - *OCV model parameters.* For OCV parameter estimation, slow discharge data is required for an entire discharge-charge cycle. Several models, such as linear, polynomial, exponential and combinations of these, have been proposed in the literature. A summary of OCV modeling approaches maybe found in [84].
  - *CF parameters.* There has been significant interest in developing a capacity fade model. Several data analyses and attempts at creating a CF model can be found in [3, 66, 76, 82, 83, 86, 99, 100, 104, 112].
  - *PF parameters.* Some examples of PF modeling and parameter estimation are found in [20, 21, 86, 114, 131].
- *Online parameter estimation.* Given the OCV parameters of the battery, the ECM parameters and the battery capacity can be estimated online. Indeed, there is some recent work [30, 50, 136] that attempts to estimate the OCV parameters online as well.
- *Tracking.*

  - *SOC.* A clear understanding of the OCV-SOC models led to the development of SOC tracking as a recursive Bayesian estimation problem that can be solved through non-linear filters, such as the extended Kalman filter (EKF) [10]. The Coulomb counting model represents the instantaneous changes in SOC and serves as the process (plant) model; the OCV+ECM model establishes a relationship between the measured voltage and SOC, serving as the observation model for SOC tracking.
  - *SOH (PF and CF).* Unlike SOC, the SOH changes slowly; hence, there is no need to track the SOH at the same frequency as SOC. Several models developed in [3, 66, 76, 82, 83, 99, 100, 104, 112] present CF and PF in terms of the number of cycles. Unlike OCV-SOC relationship, the CF/PF-cycle relationship is much more complicated; the cycle cannot be uniquely defined; the effects of temperature exposure, depth of discharge (DoD) significantly affect the model; the patterns of temperature exposure and DoD are not uniquely defined. Hence, online estimated capacity and ECM parameter values are well suited for battery fuel gauging. It must be noted that the online estimated capacity and ECM parameters can be improved with the help of CF/PF models as well.

- *Forecasting / Trending.*

- *Time to Shutdown (TTS).* In practical battery applications, the batteries are fitted with protective circuitry that switch off the device once a particular shut down voltage is reached. Predicting TTS requires the knowledge of ECM parameters; usually the TTS prediction is made for a particular load, e.g., TTS for voice, TTS for video, etc.
- *Remaining Useful Life (RUL).* Similar to TTS, the RUL predicts the remaining cycles until a certain SOH (measured in terms of upper bounds on CF and PF) is reached.

The objective of the BFG is to simultaneously estimate the ECM parameters (see Figure 3.18) and estimate the SOC and SOH of the battery. The ECM battery model considers the effects of SEI growth and plating, in the form of Capacity fade and Power fade, for estimating the state of health (SOH) of lithium-ion batteries as shown in Figure 3.19.

We show in [7–9] that BFG can be reduced into a model parameter estimation and state filtering problem for the following state-space model.

$$\begin{aligned} x_s[k] &= x_s[k-1] + c_h \Delta z_i[k-1] + w_s[k] \\ z_v[k] &= V_{\mathrm{o}}(x_s[k]) + \mathbf{a}^T[k]\mathbf{b} + n_d[k] \\ &= \mathbf{p}^T(x_s[k])\mathbf{k} + \mathbf{a}^T[k]\mathbf{b} + n_d[k] \end{aligned} \tag{3.77}$$

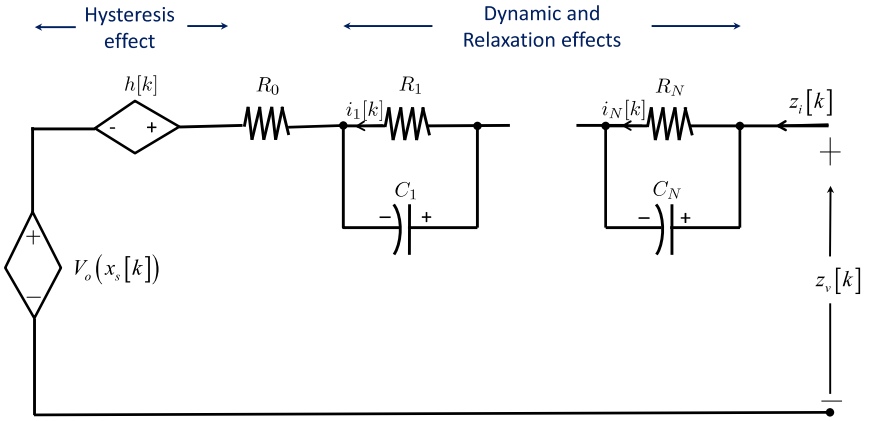


Fig. 3.18: Equivalent Circuit Model (ECM) of the Battery. The OCV is denoted by $V_o(x_s[k])$, $h[k]$ denotes the hysteresis of the battery at time k, and $i_N[k]$ denotes the current through the resistor R_N. The relaxation effect decays when the battery is rested whereas the hysteresis effect remains unaffected by resting the battery.

where $x_s[k]$ is the SOC of the battery, c_h is the Coulomb counting coefficient [8], Δ is the sampling time, $z_v[k]$ is the measured voltage, $z_i[k]$ is the measured current, $w_s[k]$ and $n_d[k]$ are the process and measurement noises (see [9] for description), which are assumed to be zero-mean white with standard deviations $\sqrt{Q}$ and $\sqrt{R}$, respectively, $\mathbf{b}$ corresponds to the dynamic equivalent circuit model parameter vector (which has model dependent dimensions, see [7]), $\mathbf{a}$ is a known vector depending on the dynamic equivalent circuit model used, and the OCV model is given by

$$\begin{aligned}\mathbf{p}^T(x_s[k]) \;=\; \Bigg[1 \;\; \frac{1}{x_s[k]} \;\; \frac{1}{(x_s[k])^2} \;\; \frac{1}{(x_s[k])^3} \;\; \frac{1}{(x_s[k])^4} \\ x_s[k] \;\; \log(x_s[k]) \;\; \log(1-x_s[k])\Bigg]\end{aligned} \tag{3.78}$$

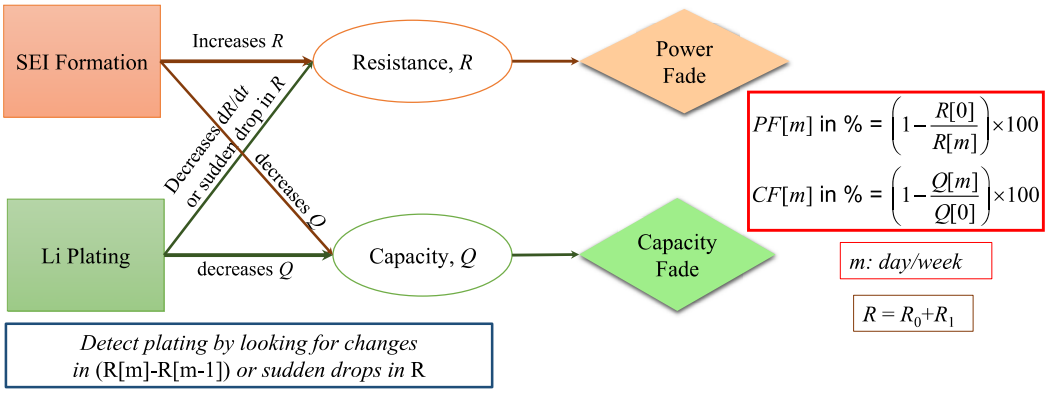


Fig. 3.19: Effects of SEI Formation and Plating. See Figure 3.18, with $N = 1$

and the OCV model parameters are given by

$$\mathbf{k} = \begin{bmatrix} K_0 & K_1 & K_2 & K_3 & K_4 & K_5 & K_6 & K_7 \end{bmatrix}^T \tag{3.79}$$

We discuss ways to estimate the parameters (which includes battery capacity) of the above state-space model in [7, 8]. Figure 3.20 summarizes the BFG algorithm.

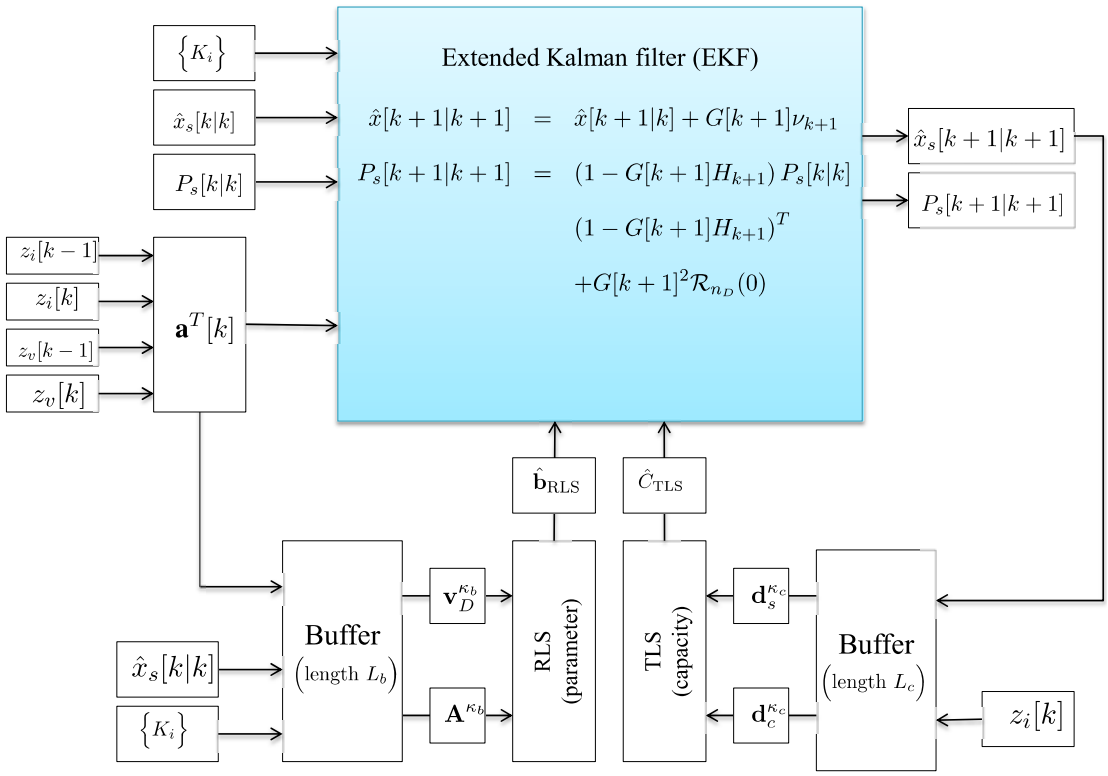


Fig. 3.20: Block Diagram of the Proposed Online SOC Tracking Approach. The buffer length is L_b for parameter estimation and L_c for capacity estimation. The EKF Tracking Module works online for every k whereas the parameter estimation module works for every k which is an integer multiple of L_b, and capacity estimation module works for every k which is an integer multiple of L_c where k is the time index. The OCV parameters $K_0, K_1, K_2, K_3, K_4, K_5, K_6, K_7$ are indicated as $\{K_i\}$. The BFG estimates all the required model parameters and battery capacity required for SOC tracking except for the OCV parameters (that are estimated offline) and voltage and current measurement error standard deviations σ_v, σ_i that come from the calibration of measurement instrumentation circuitry. The recursive least squares (RLS) blocks do not require any external initial conditions – just setting $\lambda = 1$ provides a robust LS estimate. Here, $\hat{\mathbf{b}}_{\mathrm{RLS}}[\kappa_b]$ and $\hat{C}_{\mathrm{TLS}}[\kappa_c]$ are RLS and total least squares (TLS) estimates of model parameters and capacity, respectively, where $\kappa_b = \left\lfloor \frac{k}{L_b} \right\rfloor$ and $\kappa_c = \left\lfloor \frac{k}{L_c} \right\rfloor$ are the batch numbers.

3.4 Optimal Charging Algorithm (OCA)

An overview of the optimal charging algorithm is shown in Figure 3.21. In Level I, the charge control variables, i.e., limits on charge voltage and current, are optimized to trade-off between extending battery life and time-to-charge (TTC). In Level II, the optimal current profile is computed based on a trade-off of time-to-charge (TTC), energy losses (EL), and temperature rise index (TRI), given an ECM in conjunction with a thermal model.

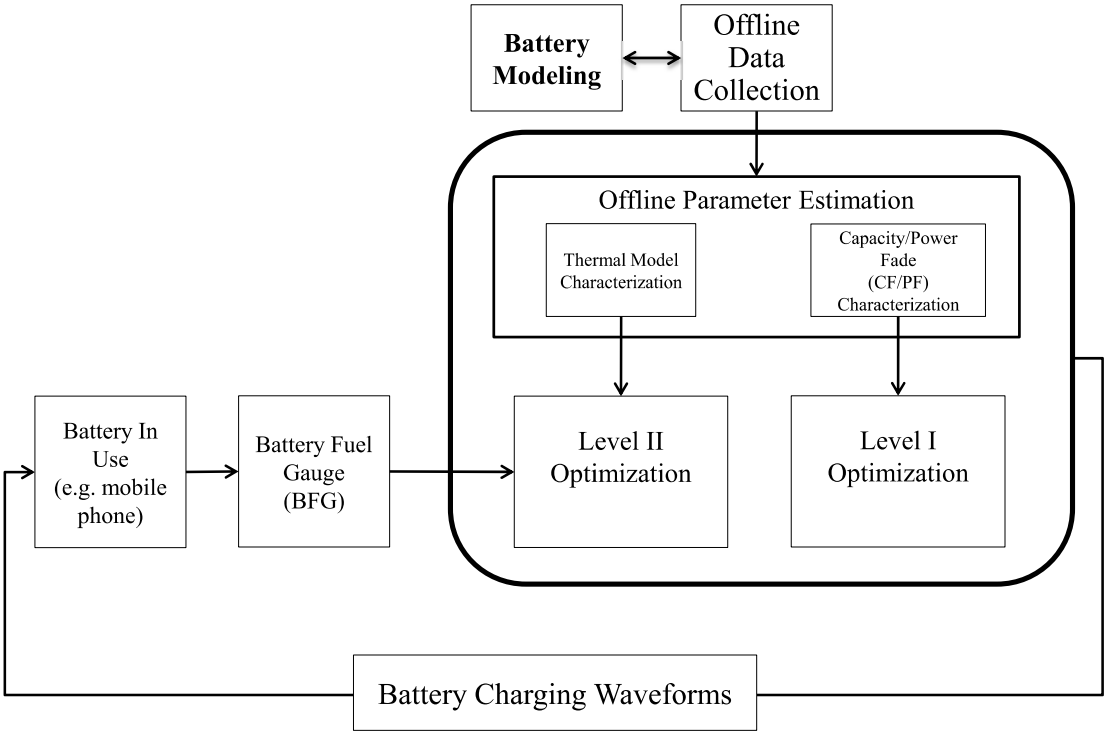


Fig. 3.21: Important Blocks of the Optimal Charging Algorithm

3.4.1 Level I Optimization

We present two models for normalized battery capacity (or, equivalently, capacity fade) as a function of the number of cycles and the two charge control parameters, viz., maximum terminal voltage of the battery, $v_{\max}$, and the maximum charge current, $i_{\max}$. Subsequently, for the optimal charging parameter selection, we propose an approach that relies on a capacity model and the current status of the battery for finding the best voltage and current to achieve a desired cycle life (for example 500 cycles), while maintaining the capacity above a desired threshold of the initial capacity (for example 80%) or equivalently, an upper bound on capacity fade (for example 20%), and attaining the fastest possible TTC [3].

Capacity Fade Modeling

The capacity at cycle n, denoted by $C[n]$, is calculated by trapezoidal integration of the injected current during the charging process, which is also referred to as the *Coulomb counting*. Subsequently, the normalized capacities are defined as,

$$C_{\text{norm}}[n] = \frac{C[n]}{C[1]} \qquad n = 1, 2, \ldots \tag{3.80}$$

These normalized capacities are used for fitting the capacity models. Note that since the capacity model is in normalized form, the capacity fade (CF), which is defined as in (3.81), can also be written as in (3.82).

$$\text{CF}[n] = \frac{C[1] - C[n]}{C[1]} \qquad n = 1, 2, \ldots \tag{3.81}$$

$$\mathrm{CF}[n] = 1 - C_{\mathrm{norm}}[n] \qquad n = 1, 2, \ldots \tag{3.82}$$

1. $\alpha\beta\gamma$ Model: Among the models explored, the $\alpha\beta\gamma$ model with the *least absolute residuals (LAR)* as the criterion of fit was found to capture the trend in the normalized capacity the best.

$$C_{\mathrm{norm}}[n] = \alpha\beta^n n^\gamma \qquad n = 1, 2, \ldots \tag{3.83}$$

The *LAR method* is robust in that it minimizes the sum of the absolute values of residuals, and, therefore, extreme values have substantially smaller influence on the fit.

2. Bi-Exponential Model: The bi-exponential model, which is frequently used in the literature [47], [135], [133] can also be used for capacity fade or normalized capacity modeling.

$$C_{\mathrm{norm}}[n] = \alpha_1\beta_1^n + \alpha_2\beta_2^n \qquad n = 1, 2, \ldots \tag{3.84}$$

The model was fitted using *least squares method*.

3. Control Variable Dependent (CVD) Model: Although the $\alpha\beta\gamma$ model can accurately capture the trend in capacity, its dependency on predefined charging profiles makes it severely restrictive for use in control variable optimization, where we would ideally like to find continuous values for the maximum voltage and maximum current to be applied in the charging process. For this reason, we developed a CVD model, in which the $\alpha\beta\gamma$ parameters are dependent $v_{\max}$, and

$$C_{\mathrm{norm},v_{\max},i_{\max}}[n] = \left(\alpha_1 v_{\max}^{\alpha_2} i_{\max}^{\alpha_3}\right)\left(\beta_1 v_{\max}^{\beta_2} i_{\max}^{\beta_3}\right)^n n^{(\gamma_1+\gamma_2 v_{\max}+\gamma_3 i_{\max})} \qquad n = 1, 2, \quad (3.85)$$

$i_{\max}$. The model chosen for this purpose is given in equation (3.85) of next page.

Optimal Charging Parameter Selection

The goal is to find $v_{\max}$ and $i_{\max}$ so as to preserve the "Least Permissible Normalized Capacity" (LPNC typically 80%) at the end of the "Nominal Cycle Life" (NCL, typically 500 cycles) of battery usage, while minimizing the TTC. We call all such pairs of ($v_{\max}$, $i_{\max}$) satisfying LPNC constraint at the end of NCL as feasible. Suppose we have K different candidate values for $v_{\max}$ and L different candidate values for $i_{\max}$ as follows:

$$v_{\max} \in \{v_1, v_2, \ldots, v_K\} \tag{3.86}$$

$$i_{\max} \in \{i_1, i_2, \ldots, i_L\} \tag{3.87}$$

The optimal charging algorithm is run for different values of v_k (terminal voltage in CV stage), i_l (current), R_0, estimated normalized capacity at the present cycle (C_{PC}), and a specified SOC threshold (s_{th}). Among all the *feasible strategies*, the optimal charge control parameters ($v_{\max}$, $i_{\max}$) are obtained from the strategy with the *fastest TTC*.

3.4.2 Level II Optimization

Based on the cost function and battery model, three different charging strategies are proposed here.

Optimal Constant-Current Constant-Voltage Strategy (CC-CV)

In the first strategy, the overall cost function of the second-level optimization problem is a weighted sum of TTC and EL. The constraints are the maximum permissible control parameters computed from Level I optimization, $i_{\max}$ and $v_{\max}$, the initial and final SOC along with the Coulomb counting SOC equation. Using optimal control theory [1] and an ECM consisting of battery resistance only, the optimal charging strategy was shown to be the well-known CC-CV with the value of the current in the CC stage being a function of the ratio of weighting on TTC and EL, and of the resistance of the battery as shown in equation (3.88).

$$i^*[k] = \sqrt{\frac{\rho_{\mathrm{t}}}{R_0}} \quad k = 0, 1, ..., k_1 - 1 \tag{3.88}$$

where k_1 is the time at which the terminal voltage reaches $v_{\max}$, $\rho_{\mathrm{t}} = w_{\mathrm{t}}/w_{\mathrm{E}}$, w_{t} and w_{E} are weights on the TTC and EL cost functions, respectively. The resistance of the battery, i.e., R_0, is obtained from the BFG. The CV stage with the terminal voltage fixed at $v_{\max}$ follows the CC stage.

Approximate Constant-Current Constant-Voltage Strategy (ACC-CV)

An extension of the CC-CV strategy was then obtained [1] by deriving an approximate CC-CV (ACC-CV) strategy for a cost function consisting of TTC, EL, and TRI, where the value of current in the CC stage is a function of the ratio of weighting on TTC and EL, ρ_{t}, resistance of the battery, R_0, and the effective thermal resistance in $\mathrm{K/W}$ (Kelvin/Watt), R_{Eff}, from the battery thermal model.

The optimal current profile was then derived, analogous to the optimal current profile of (3.88), to be as follows:

$$i^*[k] \approx \sqrt{\frac{\rho_{\mathrm{t}}}{R_0(1+\rho_{\mathrm{T}}R_{\mathrm{Eff}})}} \quad k = 0, 1, ..., k_1 - 1 \tag{3.89}$$

This is the approximate CC-CV charge current profile (ACC-CV), which considers the temperature rise as part of the charging objective function.

Linear Quadratic Constant-Voltage Strategy (LQ-CV)

In this subsection, we present the linear quadratic optimal charging strategy for a Li-ion battery [2] based on battery model with a single RC circuit, i.e. $N = 1$ in Figure 3.18 (although the formulation can be easily extended to any equivalent circuit model by extending the state space) and a combination of different cost functions including: TTC, EL, sum of the squares of the differences of the SOC from the final desired SOC (to have more flexibility in controlling the TTC), and TRI. The constraints as before are the maximum permissible terminal voltage and current from Level I. The optimal solution needs numerical computations and cannot be obtained in a closed form. As

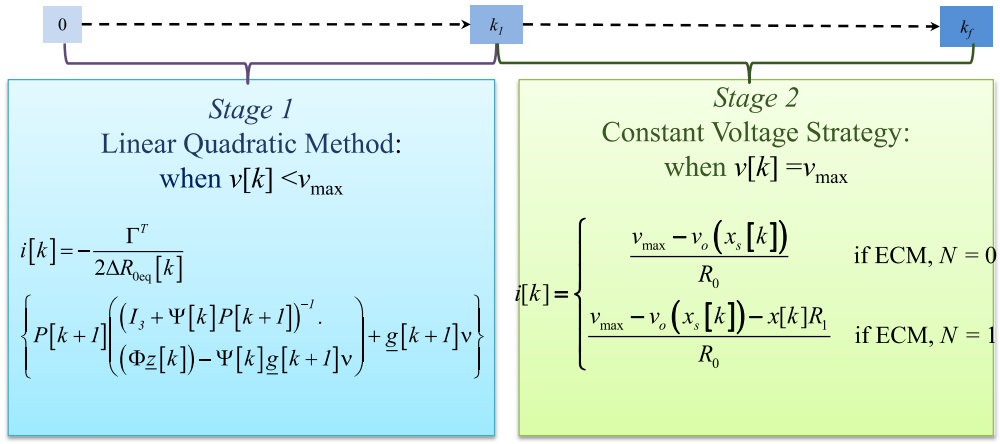


Fig. 3.22: Two Stage LQ-CV Charging Strategy

shown in Figure 3.22, the optimal Linear Quadratic-Constant Voltage (LQ-CV) strategy is composed of two stages: the *first stage* is the solution of a *linear-quadratic (LQ) problem (LQ stage)*, and the *second stage* starts when the terminal voltage of the battery reaches the maximum allowable voltage ($v_{\max}$) and, thereafter, the terminal voltage remains constant (*CV stage*).

For a given k_1 value corresponding to terminal voltage reaching $v_{\max}$, the optimal current profile in stage 1 (the LQ stage) is obtained [2] as in equation (3.90) of next page, where ν is given by:

$$i[k] = -\frac{\Gamma^T}{2\Delta R_{0\mathrm{eq}}[k]}\left(P[k+1]\left((I_3+\Psi[k]P[k+1])^{-1}\left(\Phi\underline{z}[k]-\Psi[k]\underline{g}[k+1]\nu\right)\right)+\underline{g}[k+1]\nu\right) \tag{3.90}$$

$$\nu = \frac{x_s[k_1] - x_{s_d}[k_f] - \underline{g}^T[0]\underline{z}[0]}{\omega[0]} \tag{3.91}$$

$\underline{z}[0]$ is the initial state, and $\underline{g}[0]$ and $\omega[0]$ are calculated by solving the following backward set of optimal control recursions in equation (3.93) of next page. It is assumed that the charging process takes $k_f\Delta$ seconds, to attain a specified final desired SOC of $x_{s_d}[k_f]$ and

$$\begin{aligned} R_{0\text{eq}}[k] &= R_0 + R_{0\text{T}}[k] \\ R_{1\text{eq}}[k] &= R_1 + R_{1\text{T}}[k] \end{aligned} \tag{3.92}$$

where $R_{0\text{T}}[k]$ and $R_{1\text{T}}[k]$ are the heating equivalent resistances, respectively.

$$\begin{aligned} &P[k] = 2Q[k] + \Phi^T P[k+1]\left(I_3 + \Psi[k]P[k+1]\right)^{-1}\Phi; P[k_1] = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \\ &\underline{g}[k] = \Phi^T\left(I_3 + P[k+1]\Psi[k]\right)^{-1}\underline{g}[k+1]; \underline{g}[k_1] = \begin{bmatrix} 1 & 0 \end{bmatrix}^T \\ &\omega[k] = \omega[k+1] - \underline{g}^T[k+1]\left(I_3 + \Psi[k]P[k+1]\right)^{-1}\Psi[k]\underline{g}[k+1]; \omega[k_1] = 0 \end{aligned} \tag{3.93}$$

The state vector is defined as follows:

$$\underline{z}[k] = \begin{bmatrix} x_s[k] - x_{s_d}[k_f] \\ x[k] \end{bmatrix} \tag{3.94}$$

where $x_s[k]$ is the current SOC and $x_{s_d}[k_f]$ is the desired final SOC, $x[k] = \alpha x[k-1] + (1-\alpha)i[k-1]$ is the current through R_1 and with Δ as the sampling interval,

$$\alpha = \exp(-\frac{\Delta}{R_1 C_1}) \tag{3.95}$$

The dynamics of $\underline{z}[k]$ with its initial and final states could be written as follows:

$$\underline{z}[k+1] = \Phi \underline{z}[k] + \Gamma i[k] \tag{3.96}$$

$$\Phi = \begin{bmatrix} 1 & 0 \\ 0 & \alpha \end{bmatrix} \tag{3.97}$$

$$\Gamma = \begin{bmatrix} c_h, 1-\alpha \end{bmatrix}^T \tag{3.98}$$

$$\underline{z}[0] = \begin{bmatrix} x_s[k_0] - x_{s_d}[k_f], 0 \end{bmatrix}^T \tag{3.99}$$

$$\underline{z}[k_f] = \begin{bmatrix} 0, \text{free} \end{bmatrix}^T \tag{3.100}$$

$$Q[k] = \begin{bmatrix} \rho_{\text{s}} & 0 \\ 0 & \Delta R_{1\text{eq}}[k] \end{bmatrix} \tag{3.101}$$

where ρ_{s} is the weight of the cost function term of the sum of the squares of differences of the SOC from the final desired SOC.

As the voltage hits the boundary value of $v_{\max}$ at time k_1, and remains constant onwards, the current in stage 2 is already determined by the constrained dynamics of the system (CV stage), where the initial values of the currents in stage 2 are equal to the final values of currents in stage 1.

Other charging strategies which can be used in Level II are *pulse charging strategies* (i.e. pulse current charging and pulse voltage charging) and as shown in Figure 3.23 *compensator-based charging strategies* (such as series compensator, PID compensator, and pole placement).

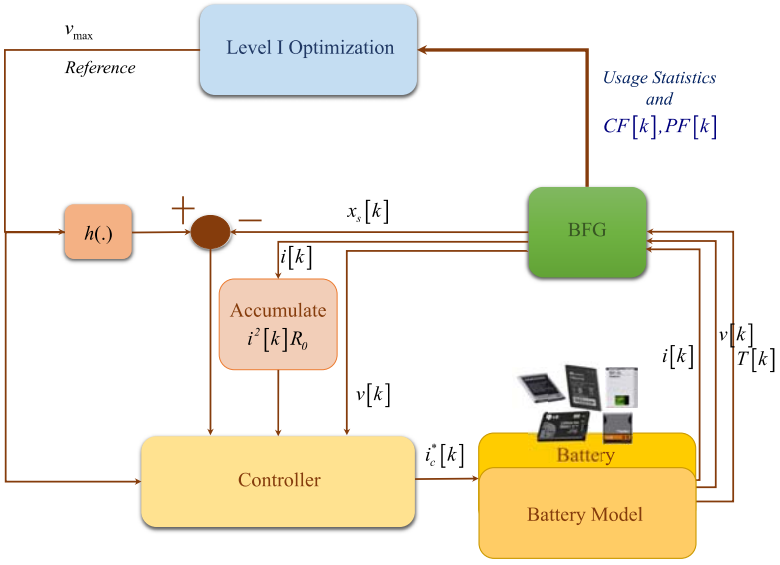


Fig. 3.23: Integrated Closed-loop Charger and BFG

3.5 Battery Management System through Integrated BFG and OCA

A detailed block diagram of our closed-loop integrated BMS, consisting of a BFG and an OCA, is shown in Figure 3.24. The integrated system not only performs accurate estimation of the state of the batteries, such as the SOC, SOH and RUL, but also computes the optimal charging current based on trade-offs between cycle life, TTC, EL, and a TRI. The state estimates from the battery fuel gauge form the sufficient statistics for the optimal charging problem. The *user-defined* set-points in Figure 3.24 are

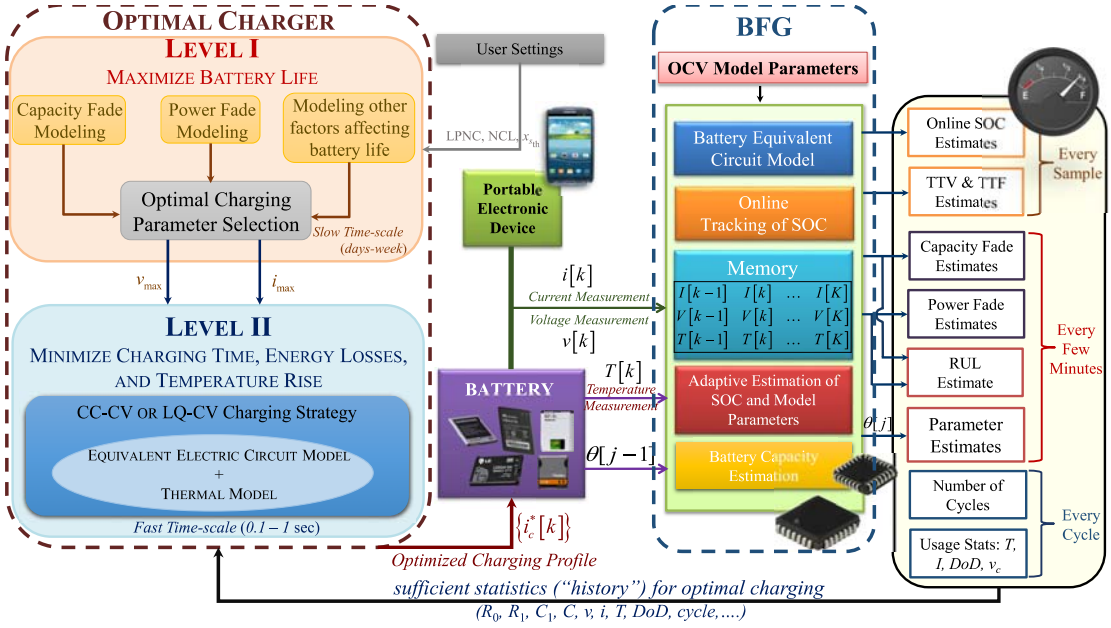


Fig. 3.24: Integrated BFG and OCA Approach

LPNC and NCL, and $x_{\mathrm{s_{th}}}$ (may be different from the desired final SOC after charging, for example 50%). The inputs to the optimization algorithm are current cycle number (PC), C_{PC}, and R_0 which are obtained from the BFG. The outputs of the algorithm are

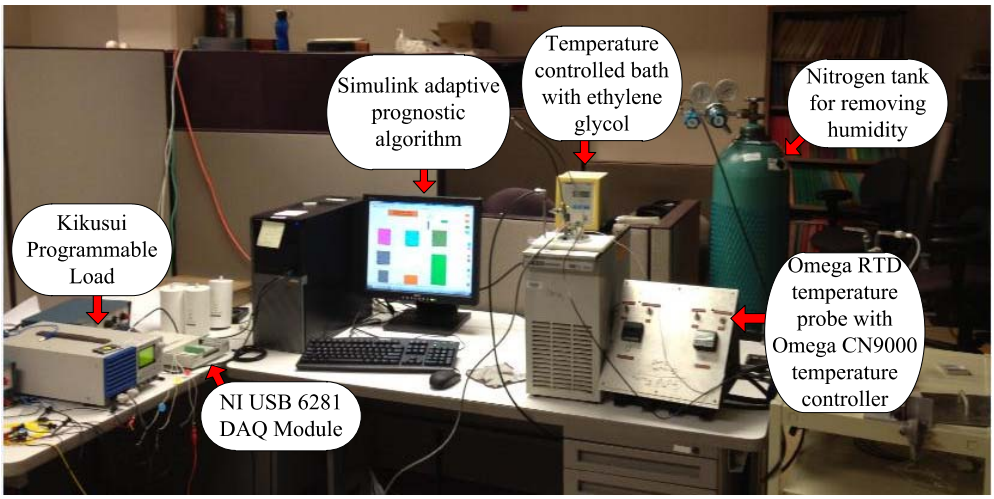


Fig. 3.25: HIL Setup for BFG Validation

$i^*_{\max}$ and $v^*_{\max}$, which are used as $i_{\max}$ and $v_{\max}$ in the level II algorithms [1], [2]. The optimal charge current profile is the output of level II algorithm and is an input to BFG.

3.6 Hardware-in-the-loop Testing and Results

3.6.1 BFG HIL Experiments

Figure 3.25 shows the hardware set up used in the BFG evaluation experiments.

The lithium-ion cells placed inside custom-built Aluminum casts which sit inside a temperature controlled bath filled with ethylene glycol (-25^0 C to 45^0 C). An Omega RTD temperature sensor probe ($PR-10$) is calibrated with Omega CN9000 temperature controller for battery temperature measurement. Nitrogen gas (99.99% purity)is slowly pumped into the Aluminum cast to the remove moisture on batteries. A programmable load box, KiKuSui PLZ-4WA, along with a LabView based graphical

user interface (*GUI*) was used to automate the KiKuSui for custom loads using virtual instrument software architecture (VISA); a standard language for instrument programming. Data acquisition was performed using National Instruments (NI) DAQ 6281 at a sampling frequency of 10 Hz and stored in comma separated value (CSV) format for future simulations and analysis. The BFG is modeled in Simulink on a PC with an operating system (Microsoft Windows or Max OSX) and does not guarantee real-time execution of the BFG. The MPC 50 pin I/O interface board along with xPC Target is used for real-time implementation of the BFG algorithm as shown in Figure 3.26. A Maxim 4069 is used for accurate current sensing.

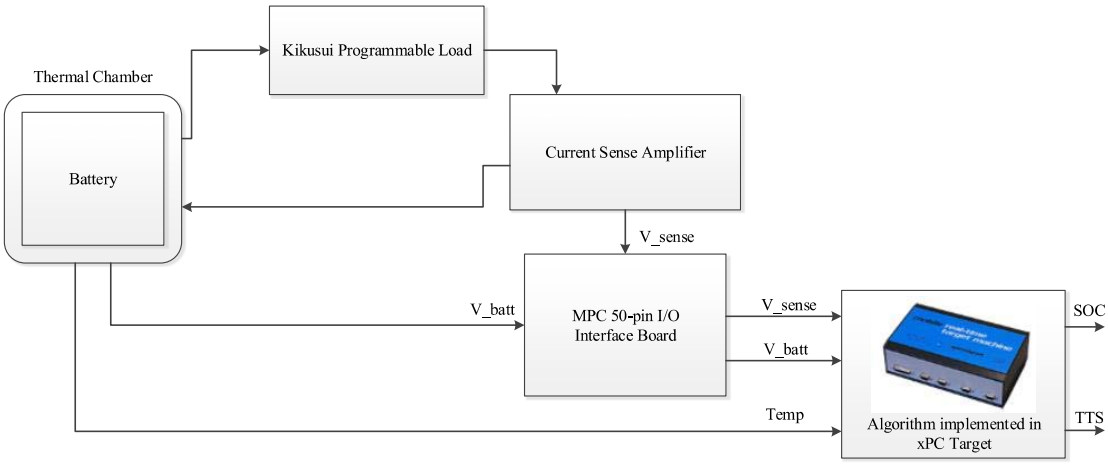


Fig. 3.26: Block Diagram of HIL Setup using xPC Target

A single evaluation method alone is not sufficient to validate a BFG algorithm; evaluation of BFG algorithm must be done by computing multiple performance evaluation metrics [6]. We computed the following three different BFG evaluation metrics

1. *Coulomb counting metric.* This metric computes the error between BFG reported

SOC and the SOC computed by Coulomb counting based on the battery capacity that is measured during/right after the experiment through a slow discharge.

2. *OCV metric.* The OCV metric is computed during intermittent points during experiment; the battery is brought to rest and the SOC is computed based on a look-up procedure in the OCV curve and it is compared to the SOC reported by BFG.

3. *TTV metric.* The TTV metric is computed as the error between computed and actual time it takes to reach a certain terminal voltage.

A summary of the results from the BFG evaluation with three commercially available batteries (GS3, Nokia and LG) at different temperatures ranging from $-20°\mathrm{C}$ to $40°\mathrm{C}$ is presented in Tables 3.6, 3.7 and 3.8.

Table 3.6: CC Metric

RMS Error (in %) of FG reported SOC related to Coulomb counting									
	$-20°C$	$-15°C$	$-10°C$	$-5°C$	$0°C$	$10°C$	$20°C$	$30°C$	$40°C$
EB555157VA	0.44	0.86	0.44	0.86	0.35	0.02	0.75	0.10	0.15
LG LGIP	0.39	0.52	0.90	0.11	0.66	0.53	0.28	0.25	0.49
Nokia	0.23	0.42	0.59	0.94	0.51	0.14	0.87	0.88	0.80

Table 3.7: OCV-SOC Metric

OCV-SOC Error (%)									
	$-20°C$	$-15°C$	$-10°C$	$-5°C$	$0°C$	$10°C$	$20°C$	$30°C$	$40°C$
EB555157VA	4.76	1.22	0.38	1.22	0.08	-0.59	2.41	0.02	-3.56
LG LGIP	-0.58	-0.06	2.40	0.38	0.61	-0.32	-0.98	-0.38	2.20
Nokia	3.15	5.29	0.26	2.97	1.49	4.19	0.23	1.99	1.12

Table 3.8: TTV Metric

TTV Error (%)									
	$-20°C$	$-15°C$	$-10°C$	$-5°C$	$0°C$	$10°C$	$20°C$	$30°C$	$40°C$
EB555157VA	8.7	17.7	1.9	17.7	7.0	8.9	12.2	6.0	4.6
LG LGIP	1.4	0.7	3.8	3.8	3.0	3.8	3.7	6.9	1.5
Nokia	1.5	0.7	4.0	6.1	14.2	1.6	4.6	5.3	4.5

3.6.2 OCA HIL Results

HIL experiments were conducted on several commercially available lithium-ion batteries: Samsung EB575152 (four cells), Samsung EB504465 (four cells), Samsung AB463651 (two cells), Nokia BP-4L (four cells), LG LGIP (two cells), using LabVIEW and Keithley 2650A Source-Measure Unit (SMU).

Level I Optimization Results

- *Capacity Fade Modeling*. Figure 3.27 shows the result of fitting $\alpha\beta\gamma$ and BE models to aging test data sets with length 40, 100, and 160 cycles. It is seen that $\alpha\beta\gamma$ model is consistent with the experimental data even when just 40 samples are used in fitting the model. Also when the number of model-fitting data samples increases, $\alpha\beta\gamma$ model becomes more accurate in projecting to future cycles. Figure 3.28 shows a comparison of the $\alpha\beta\gamma$, CVD, and BE models in captur-

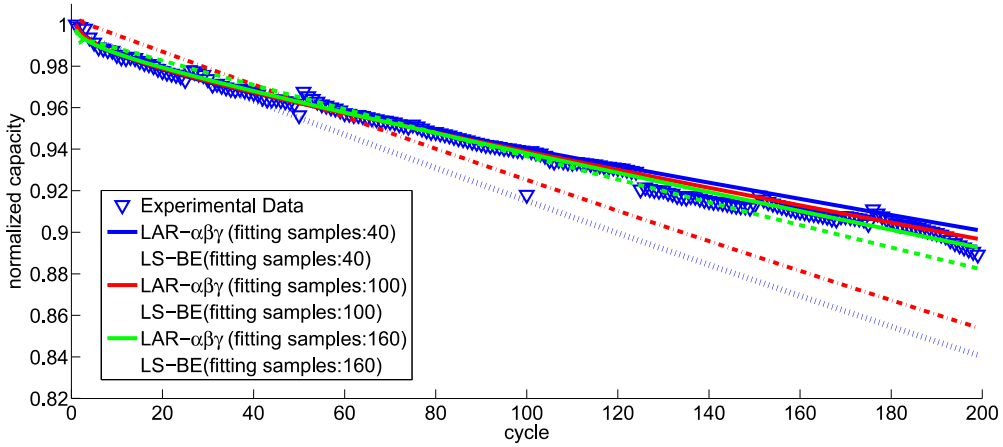


Fig. 3.27: Comparison of $\alpha\beta\gamma$ model and BE model with different fitting data lengths

ing the trend in capacities from different aging experiments. The CVD and $\alpha\beta\gamma$ models are indistinguishable and perform consistently better than the BE model.

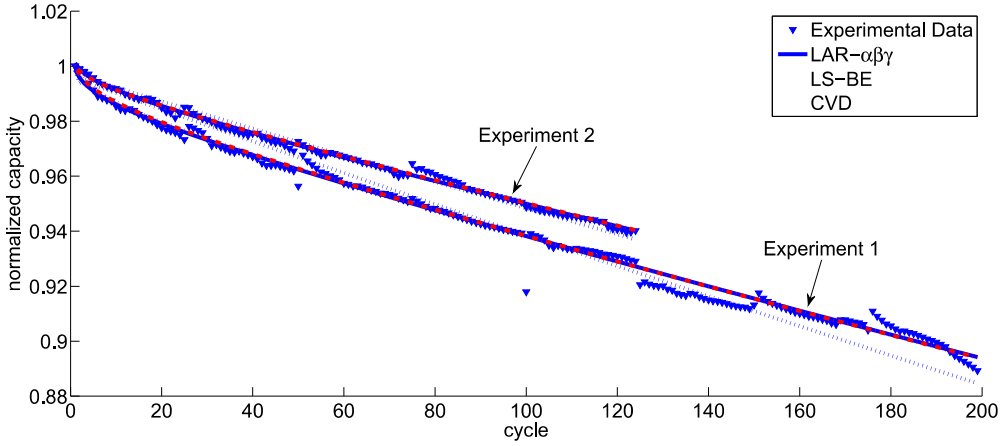


Fig. 3.28: Comparison of $\alpha\beta\gamma$, BE, and CVD models

Level II Optimization Results

- *Optimal CC-CV*. In Figure 3.29 we apply different levels of current and the simulation is run until the terminal voltage reaches $v_{\max}$ and after that a constant voltage of $v_{\max}$ is applied until the battery is charged to $x_{s_d}[k_f]$. At lower levels of current, the CC stage will take a longer time and the terminal voltage reaches the threshold voltage of $v_{\max}$ at a later time. The optimal current profile (i.e., profile 3) has the lowest cost function and either increasing or decreasing the current in the CC stage, results in an increase in the cost function.

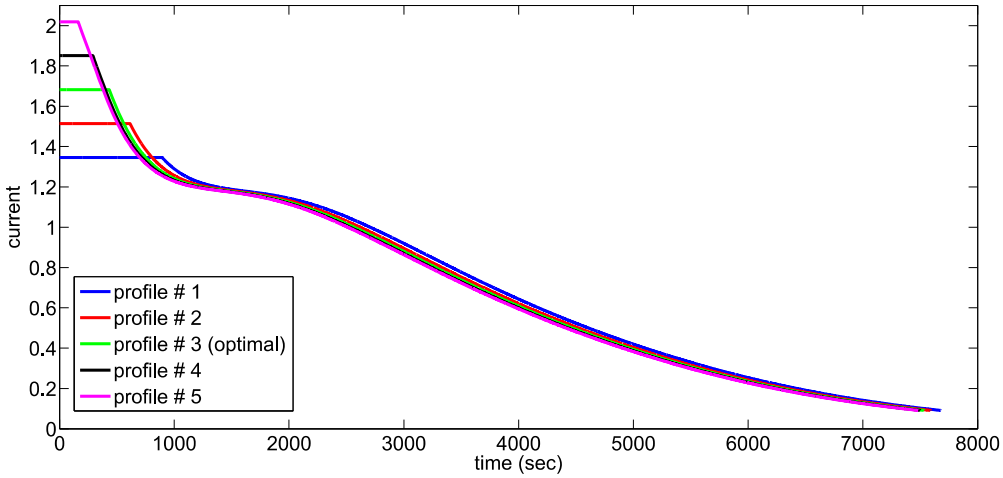


Fig. 3.29: CC-CV Current Profiles (including the optimal profile)

- *Approximate CC-CV*. Here, we consider the effect of TRI on optimal charging where the cost function is a weighted sum of TTC (seconds), EL (Joules) and TRI (Kelvin seconds) resulting in an ACC-CV current profile. As we can see in Figure 3.30, the weight on TRI results in a reduction of current.

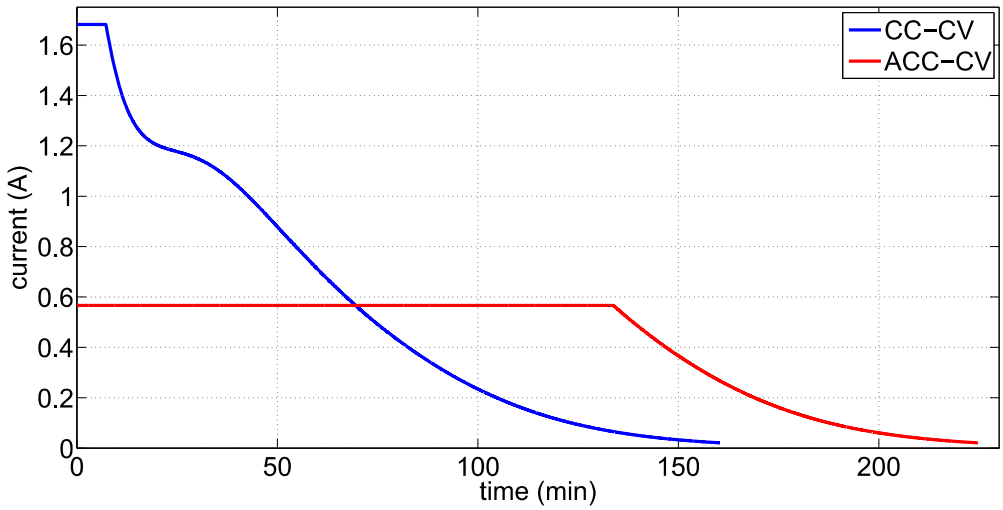


Fig. 3.30: Optimal vs. Approximate CC-CV Current Profiles

- *LQ-CV*. Figure 3.31 shows that low values of ρ_{t} result in low values of current in stage 1 (LQ stage). In other words, a low ρ_{t} puts less emphasis on charging time and more emphasis on the energy losses; hence, it results in low level of current which provides low energy loss and higher charging efficiency. On the other hand, by increasing ρ_{t}, more emphasis is placed on the charging time. Note that we observed similar behavior in CC-CV and ACC-CV; however, here the difference is that the current in stage 1 is not constant, rather it is obtained by numerically solving a linear quadratic problem.

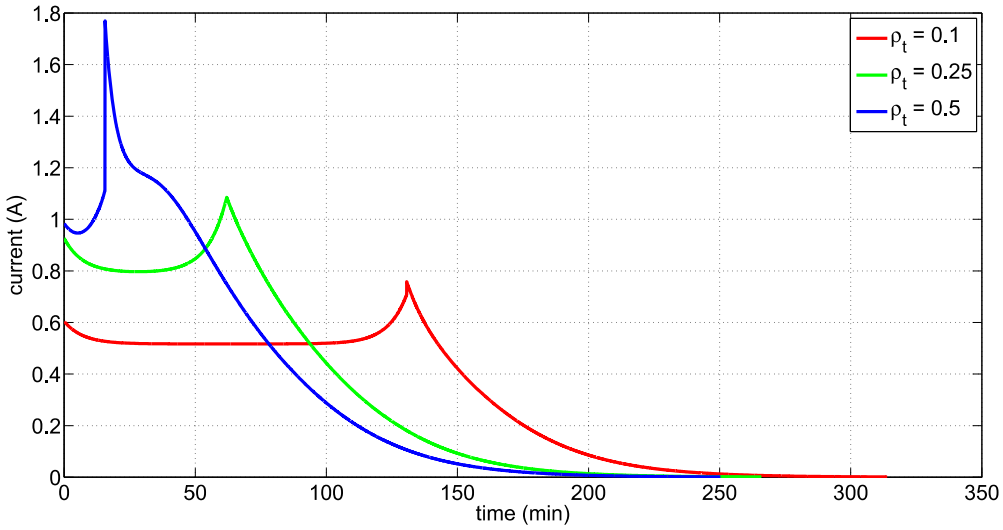


Fig. 3.31: LQ-CV Current Profiles for different values of ρ_{t}

Figure 3.32 shows the optimal current profile for three different weight sets. The blue curve represents the optimal profile when $\rho_{\mathrm{t}} = 0.5, \rho_{\mathrm{s}} = 0, \rho_{\mathrm{T}} = 0$. Here there is no penalty on the temperature rise and the squared difference of the SOC and the final desired SOC ($x_{s_d}[k_f]$). As we change ρ_{T} from 0 to 0.5, the emphasis

on the temperature rise causes the level of the current to decline. This can be seen from the comparison of the red and blue curves in Figure 3.32. The green curve shows the optimal current profile when we set ρ_s to 0.1 as well. Here, the penalty on the squared difference of the SOC and the final desired SOC ($x_{s_d}[k_f]$) causes an increase in the current level.

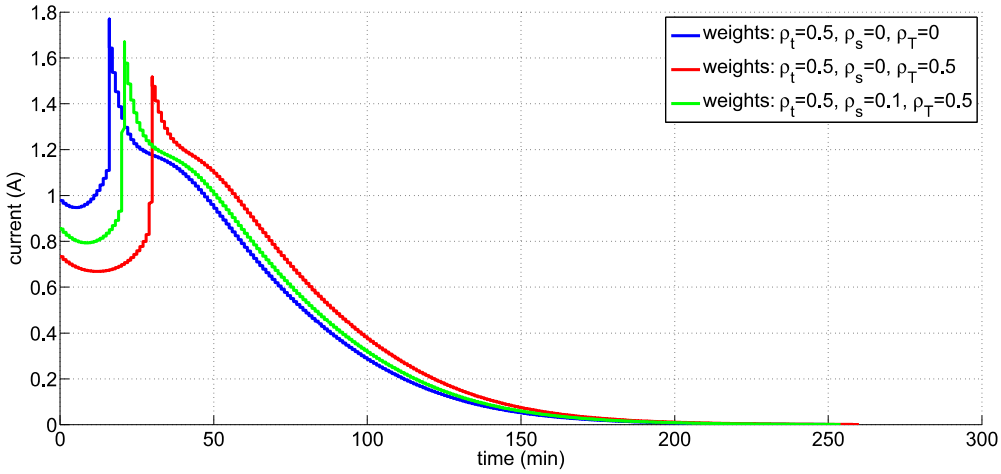


Fig. 3.32: Effect of Different Weights on the Optimal Current Profile

- *Analysis of Different Commercial Batteries*. Fig. 3.33 shows the result of both CC-CV and LQ-CV charging profiles for the TTC versus efficiency for different types of batteries. The CC-CV charging strategy results are represented by "filled" markers. As can be seen, both LQ-CV and CC-CV charging profiles predict almost the same pattern of efficiency performance and charging time. For example, both strategies predict highest TTC for cells 3 and 4 of Samsung EB504465, and Nokia BP-4L has high efficiency for both strategies.

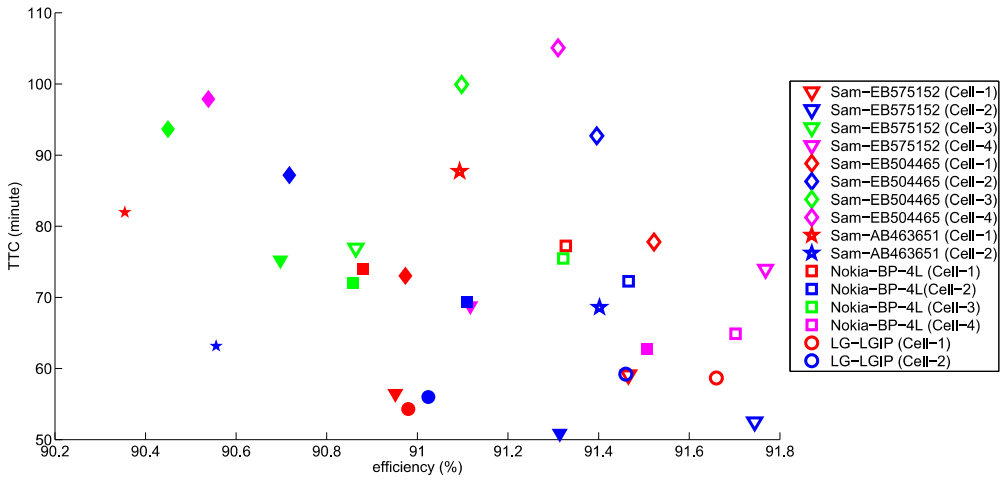


Fig. 3.33: Time to Charge versus Efficiency of Different Battery Types at 25°C

3.7 Conclusion and Discussions

The chapter presented a novel battery management system based on an integrated battery fuel gauge and optimal charger. The algorithms have been validated on commercially available lithium-ion batteries at different temperatures.

Chapter 4

Fault Diagnosis, Prognosis, and Condition-based Maintenance

The Electronic Return-less Fuel System (ERFS) manages the delivery of fuel from the fuel tank to the engine. The pressure in the fuel line is electronically controlled by the fuel system control module by speeding up or slowing down the fuel pump. This allows the system to efficiently control the amount of fuel provided to the engine when compared to vehicles equipped with a standard fuel system wherein the fuel pump continuously runs at full speed. A failure in the fuel system that impacts the ability to deliver fuel to the engine will have an immediate effect on system performance. Consequently, improved reliability and availability, and reduction in the number of walk-home situations require efficient fault detection, isolation and prognosis of the ERFS system. This chapter develops and implements data-driven fault detection, isolation and severity estimation algorithms for the ERFS. The HIL Fuel System Rig and a Chevrolet Silverado truck were used to collect and analyze the fuel system behavior under different fault conditions. Several data-driven classifiers, such as support vector machines, K-nearest Neighbor, Discriminant analysis, Bayes classifier, Partial- least squares, Quadratic and Linear classifiers, were implemented on a limited set of data for

both training and testing. Regression techniques, such as Partial least squares regression and Principle component regression, are used to estimate the severity of faults [85]. Subsequently, we performed vehicle fleet warranty data analysis, developed prognostic modeling approaches, and implemented different remaining useful life prediction algorithms. Consequently, prognostic decision-making blocks and an overall CBM strategy for the ERFS were designed. The resulting solution approach has the potential to be applicable to a wide variety of systems, ranging from automobiles to aerospace systems.

4.1 Introduction

Electronic Return-less Fuel Systems (ERFS) are fast replacing the traditional mechanical fuel delivery systems to transport fuel from the vehicles fuel tank to the fuel rails and fuel injectors. In the ERFS system, the Fuel System Control Module (FSCM) regulates the pressure on the fuel lines to a desired pressure command from the Engine Control Module (ECM) based on the required engine speed by varying the pulse-width-modulation (PWM) control of the fuel pump. A fuel filter and a pressure regulator may be positioned on the respective intake and outlet sides of the fuel pump. Filtered fuel is thus delivered to a fuel rail, where it is ultimately injected into the engine cylinders. An ERFS includes a sealed fuel tank and lacks a dedicated fuel return line. The regulation of the fuel rate to the injectors improves the fuel economy and eliminates liquid recirculation to the fuel tank. The fuel economy is improved by reducing the electrical load on the alternator and by reducing the rail pressure under most operating

conditions. With return-less systems, there is no return line and no circulation of fuel back to the fuel tank from the engine. Consequently, there is no heating of the fuel in the tank and no increase in fuel vapor pressure from driving the vehicle. This reduces the risk of excessive pressure build up inside the fuel tank, vapor leaks, and potential improvements in air/fuel ratio control, and vehicles emission performance. Diagnostic and prognostic methods have mainly evolved upon three major paradigms, viz., model-based [29], data-driven, and knowledge (experience)-based approaches. The model-based approach uses a mathematical representation of the system and thus incorporates a physical understanding of the system into the monitoring scheme. A major advantage of the physics-based model is that the model bears certain behavioral resemblance to the actual system, which can be very useful in the design of a diagnostic procedure. However, models developed from first principles are seldom used for fault diagnosis in automotive industry mainly because of their complexity. In addition, automotive system dynamics are often nonlinear, which renders the design of fault diagnosis procedures difficult. However, with the advances in computing and an improved understanding of automotive systems, the design of model-based diagnosis schemes is expected to be integrated into the concurrent engineering design process. Model-based methods use statistical estimation techniques based on consistency checks (often termed residuals, deltas) generated using observers (e.g., Kalman filters, reduced-order unknown input observers, interacting multiple models, particle filters) and parity relations (dynamic relations among measured variables) to track the component degradations. A data-driven

approach to fault diagnosis and prognosis is preferred when system models are not available (e.g., when subsystem vendors do not share models for competitive reasons), but instead system monitoring data is available [80]. Here, failure prognosis involves forecasting of system degradation and time-to-failure based on state awareness gleaned from monitored data. Neural network and statistical classification methods are illustrative of this approach. The fault scenarios must span the universe of system faults for data-driven approaches to be effective. Mathematical models may be derived (estimated or identified) from data as well. Data-driven models include static models and dynamic models. Static models include linear and polynomial models, and look-up tables. Dynamic models include dynamic linear and nonlinear system models. Knowledge-based systems are based on the methods and techniques of artificial intelligence. The core components of these systems are the knowledge base and the inference mechanisms. Examples of knowledge-based systems are: rule-based systems, case-based reasoning systems, and graphical models [70]. Examples of graphical models include: signed directed graphs, multi-signal flow graphs, Petri nets, and Bayesian networks [71]. Conventional diagnostic techniques for a vehicle fuel system typically rely on knowledge of a prior failure condition. For example, when servicing a vehicle, the maintenance technician may determine that the fuel pump requires repair or replacement by direct testing and/or review of a recorded diagnostic trouble (error) code. This reactive diagnosis may not occur until vehicle performance has already been compromised. A proactive approach which tracks degradations in a fuel system is more advantageous

than a reactive approach, particularly when used with emerging vehicle designs utilizing an ERFS. In this chapter, the fault detection and isolation problem of EFRS is characterized and some basic definitions are given. The main idea of fault diagnosis is to determine if there is any fault or abnormal behavior is present in the system, and to localize (isolate) the fault. In order to detect and localize the fault, a diagnosis system is needed. The diagnosis systems exploits the known signals, i.e. input signals such as control signals, and measured output signals from the system under diagnosis, to infer the fault.

The problem of fault diagnosis can be divided into several sub-problems. Here, we focus on three:

- Fault Detection: To determine if a fault is present in the system and usually the time when the fault has occurred.
- Fault Isolation: Determination of the location of the fault, i.e. which component or components have failed.
- Fault Severity (Estimation): Determination of the size and possibly time-varying behavior of a fault.

The three sub-problems are closely nested, and many algorithms cover several of them. The focus of this chapter is to develop data-driven fault isolation, and severity estimation algorithms based on neural network and statistical pattern recognition techniques exemplified by Support Vector Machines (SVM) [127], [45], [119], K-

Nearest Neighbor (KNN), Principal Component Analysis (PCA) [56], Partial Least Squares (PLS) [19], Gaussian Mixture Models (GMM), Discriminant Analysis, and so on [17], [39], and validate them based on fault injection in the HIL bench and the Chevrolet Silverado truck. We also estimate the severity of the isolated fault by PLS and principal component regression. The techniques chosen in the chapter are based on popularity, range of complexity, robustness, data structure, and to assess the difficulty of the classification and regression problem.

Subsequently, we performed vehicle fleet warranty data analysis, developed prognostic modeling approaches, and implemented different remaining useful life (RUL) prediction algorithms. Consequently, prognostic decision-making blocks and an overall CBM strategy for the ERFS were designed.

Some of the challenges faced during this research are as follows:

- Warranty data at present only consists of incipient failures (sudden failures)
- Many cases with no data before warranty
- Lack of prognostic failure data
- Premature component replacement (if possible, recommend procuring the near-failure/replaced fuel system for prognostic analysis)
- Missing/Insufficient vehicle data
- Fuel pump manufacturer variability

As part of the fleet data collection, 11 parameter identifiers (PIDs) were collected for the fuel system under different sampling modes via the telematics module (figure 4.1):

1. Control Module Voltage
2. Estimated Motor Resistance
3. Fuel Pump Power Module Sensed Battery Voltage
4. Adjusted Fuel Pump Current
5. Fuel Pump Input Power
6. Fuel Pump Output Power
7. Fuel Pressure Relative to Atmosphere
8. Desired Fuel Supply Pressure
9. Fuel Pump Module Output PWM Duty Cycle
10. Fuel Pump Duty Cycle Long-term Correction
11. Fuel Pump State of Health

The chapter is organized as follows. Section 2 presents the overall framework for real-time fault detection and diagnosis of fuel delivery systems, the neural network and statistical pattern recognition techniques, the results of these classification and regression techniques for fault isolation and severity estimation on real data collected

Fig. 4.1: Fleet Data Collection

from the Chevrolet Silverado truck and the HIL rig and finally, the implementation of these data-driven techniques, embedded software in Simulink, which can be used for real-time fault isolation and severity estimation. Section 3 presents the fleet prognosis, fault Isolation, and condition-based maintenance (CBM) framework and results. Subsequestly, section 4 discusses the remaining useful life estimation approaches for both fleet and individual vehicles. Section 5 consists of prognostic decision-making approaches and an overall CBM decision block for the fuel system. Finally, section 6 concludes with a summary.

4.2 Real-Time Fault Detection, Isolation and Severity Estimation of Fuel Delivery Systems

4.2.1 FDI Framework for Fuel Delivery Systems

The Fault Detection and Diagnosis (FDD) process consists of an *offline* training phase and an *online* testing phase. Figure 4.2 depicts the block diagram of a real-time FDD scheme for the fuel delivery system.

During the steady-state detection, a model-based detector based on residuals, parity equations, regression, and parameter estimation techniques is implemented on the ECU,and detects the fault and estimates the state of health (SOH) during real-time operation of the vehicle. The nominal residuals for system operation are obtained during the offline phase via HIL rig experiments, and testing and validation is performed on the Chevrolet Silverado truck data, and the faults detected in real-time conditions based on these nominal conditions.

In the *offline* phase, steady-state sensor data from different fault classes is used to train two fault classifiers, the Support Vector Machines (SVM), and the K-Nearest Neighbor (KNN). Partial Least Squares (PLS), and Principal Component Regression (PCR) estimators were also trained to assess the fault severities after fault isolation. The trained classifiers and their corresponding parameters and/or weights are exported to the online module for real-time FDD. An optimal sensor selection block is used to select the significant sensor suite for maximum diagnosability.

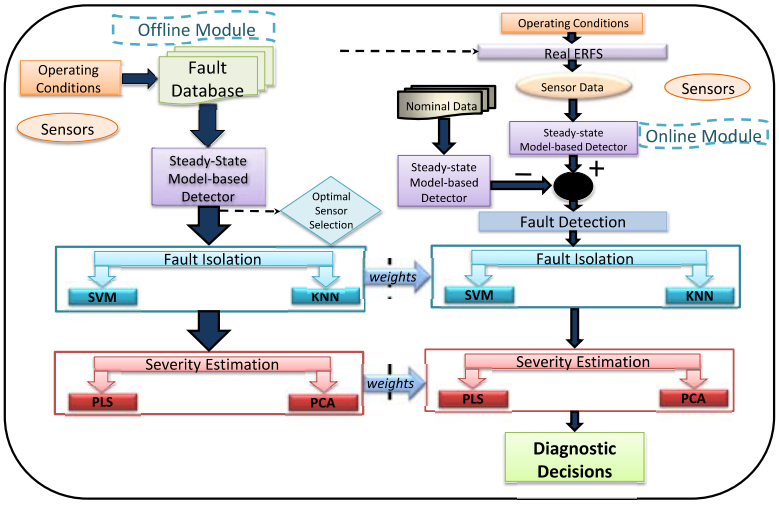


Fig. 4.2: Framework for real-time fault detection and diagnosis of fuel systems

The *online* FDD phase consists of three steps: fault detection, fault isolation or classification, and fault identification or severity estimation. In the fault detection step, the steady-state model based detector analyses the residuals generated from the steady-state measurements of faulty and nominal systems. Upon detection of a fault, trained classifiers (SVM and KNN) are used for the online categorization of faults. In the next step, the PLS and PCA estimators corresponding to the isolated fault are used to determine its severity.

4.2.2 Fault Universe for HIL and In-vehicles Experiments

The fuel pump is an electronically controlled closed-loop system that maintains a desired fuel system pressure ($\approx$ 400 KPa for the Chevrolet Silverado truck) and provides fuel flow on-demand to the engine under all operating conditions. The five critical fuel

Table 4.1: Fault universe for Chevrolet Silverado truck

Fault	Fault Type	Component
F1.1	Pressure Sensor Bias (Slew 9.3 on pressure box)	Fuel Line
F1.2	Pressure Sensor Bias (Slew 9 on pressure box)	Fuel Line
F2.1	Winding Fault (1 Ω resistance added)	Pump/Motor
F2.2	Winding Fault (0.66 Ω resistance added)	Pump/Motor

pump faults considered in this chapter are listed in tables 4.1 and 4.2. The faults in table 4.1 correspond to those in Chevrolet Silverado truck and the faults in table 4.2 are for the HIL Rig.

Altogether, fault injection experiments were performed with a commonly occurring motor/fuel pump fault, 2 sensor faults (pressure and current sensors), a pump module fault, and a fuel line fault. The fuel pressure and current sensors are located anywhere between the fuel pump and fuel rail, and the pressure and current sensor bias faults are often difficult to isolate, especially between each other, as current bias shows up as pressure bias and vice-versa. As the fuel pump degrades with age, the motor winding resistance increases and consequently, the pump PWM increases to supply the same desired pressure. A positive and negative pressure sensor bias results in the pump drawing less and more current respectively to compensate for the sensor errors. The Filter plugged fault is a result of the pump filter being blocked or clogged, and the

Table 4.2: Fault universe for HIL rig

Fault	Fault Type	Component
F1	Winding/Commutator Fault	Pump/Motor
F2	Pressure Sensor Bias Fault	Fuel Line
F3	Current Sensor Bias Fault	Fuel Line
F4	Filter Plugged	Pump Module
F5	Fuel Leakage Fault	Fuel Line

Table 4.3: Severity levels of each fault for HIL rig

Winding Fault	Pressure Sensor Bias Fault	Current Sensor Bias Fault	Filter Plugged	Fuel Leakage Fault
0.3158Ω	50 KPa	1 A	25% closed	25% closed
0.4Ω	100 KPa	1.5 A	50% closed	70% closed
0.5Ω	-50 KPa	2 A	80% closed	80% closed
0.66Ω	-100 KPa	3 A	100% closed	100% closed
0.75Ω				
1Ω				
1.2Ω				
1.5Ω				
2Ω				
3Ω				

effect of a leakage in the fuel line is represented by the fuel leakage fault.

The 2 faults in table 4.1 were conducted at 2 different severity levels using a pressure and resistance box, respectively. However, since the HIL Rig allows for more flexibility, the winding fault was conducted at 10 different severity levels, and the pressure sensor bias fault, current sensor bias fault, filter plugged fault and fuel leakage faults were conducted at 4 severity levels as summarized in Table 4.3. The severity levels experiments of the winding fault, pressure and current sensor bias faults were conducted by adding resistances (resistance box), adding (positive) and subtracting (negative) pressure (pressure box), and injecting current, to the pump resistance, pressure, and current correspondingly. The filter plugged and fuel leakage fault experiments were conducted by restricting the fuel flow using valves (flow restrictor in figure 4.4). The severity levels were chosen to represent the degradation of a fuel pump from low (operating normally) to high (end-of-life).

4.2.3 Fuel System Hardware-in-the-loop Rig

A HIL system was designed as a means for validating the diagnostic algorithms, analyze the fuel system behavior under different operating conditions, and compare the physics-based system models to the actual system. The HIL rig was controlled by a lab machine and its performance parameters were linked to a user-interface (display screen) via CAN, to warn customers of likely vehicle failure/breakdown. A schematic of the Chevrolet Silverado based HIL rig is shown in Figure 4.3.

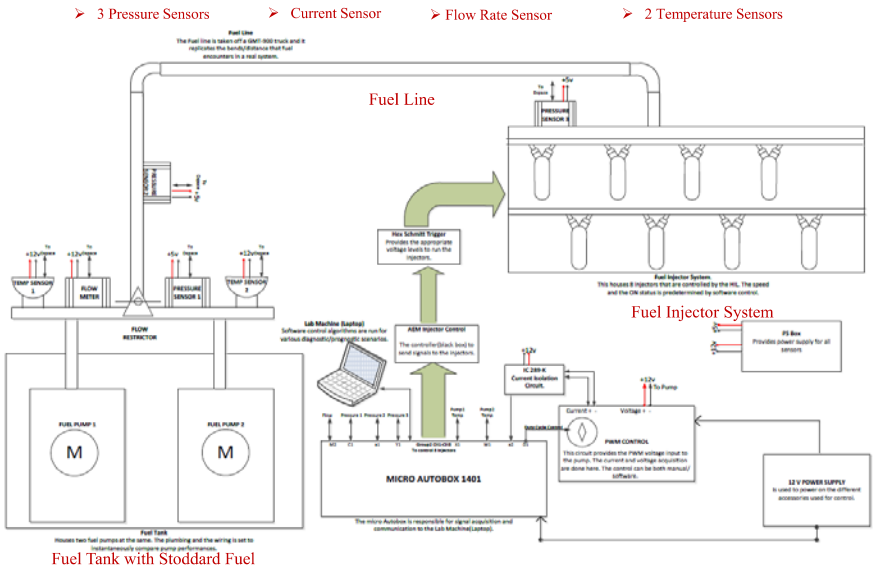


Fig. 4.3: Chevrolet Silverado based Hardware-in-the-loop rig

The fuel tank assembly houses two pumps, one for reference (for e.g. healthy pump) and the other (e.g. faulty pump) for applying different faults and subsequently, comparing the two pumps simultaneously under various diagnostic scenarios. Each pump has its own shut-off valve, when the other pump is in operation. The entire system has a control valve that enables fuel circulation in the loop, which comes in handy to study pump dynamics. Each pump is fitted with 2 thermocouples which act as temperature sensors for monitoring the temperature and providing warning in case of overheating.

Fault simulations were run on HIL Rig using a drive profile obtained from the Chevrolet Silverado test vehicle. A Simulink-dSpace model of the fuel system was used to extract the sensor and parameter identifier (PID) data (current, voltage, pressure, flow, and PWM) from the HIL Rig as shown in figure 4.4. The desired engine speed and

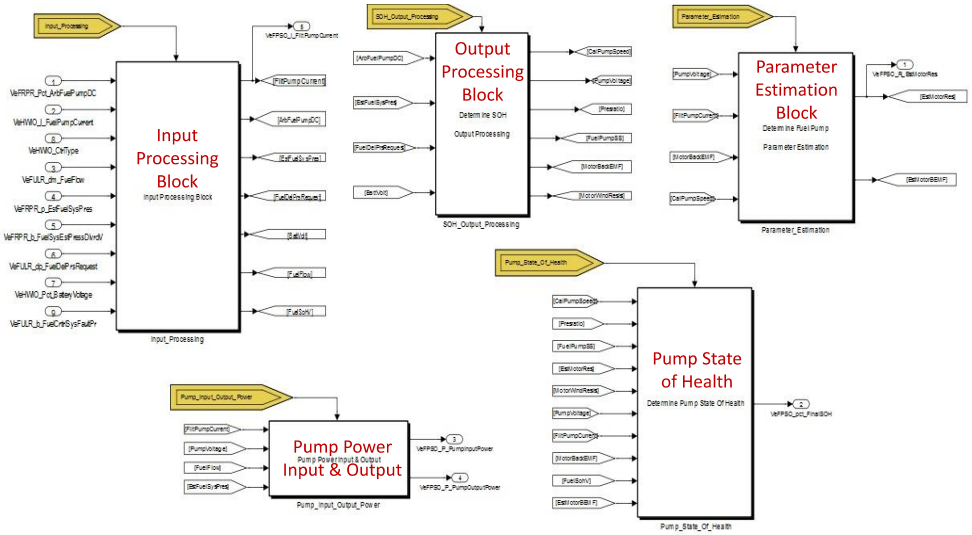


Fig. 4.4: Simulink-dSpace ERFS model

pressure profiles for the drive cycle are presented in figures 4.5 and 4.6, respectively.

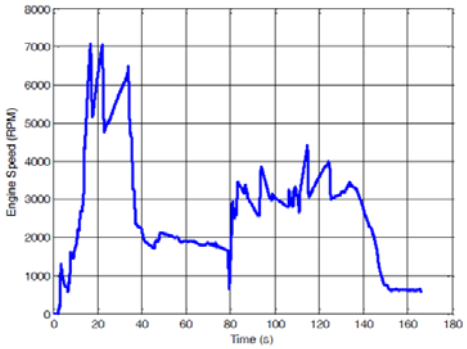


Fig. 4.5: Desired engine speed for the test drive cycle

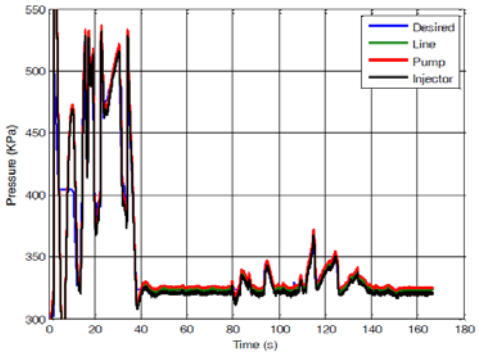


Fig. 4.6: Desired pressure for the test drive cycle

4.2.4 Fault Isolation and Severity Estimation of Fuel Delivery System

Fault Isolation on the Chevrolet Silverado Truck Data

Parameter identifier (PID) data was collected from a Chevrolet Silverado truck under *idle* and *normal* conditions.

The PIDs collected are listed below.

1. Current
2. Pressure
3. Flow
4. PWM
5. Current variance
6. Desired Pressure
7. Engine Speed

Table 4.4: Features for fault isolation

Power Out (Pressure x Flow x Pump Efficiency)	Power In (Voltage x Current x PWM)	PWM	Current	Flow	Pump Pressure

8. Vehicle Speed
9. Pump Pressure
10. Pump Voltage
11. Pump Efficiency
12. Status

The features used for fault isolation are presented in table 4.4.

The fault universe, listed in Table 4.1, was used to define the fault classes for the classification algorithms as follows:

- Class 1: No Fault
- Class 2: Pressure sensor bias (Slew9.3 on the pressure box)
- Class 3: Pressure sensor bias (Slew9 on the pressure box)
- Class 4: Winding Fault (1 ohm resistance added)
- Class 5: Winding Fault (0.66 ohm resistance added)

The classification results under the two driving conditions of the truck are presented in tables 4.5 and 4.6.

Table 4.5: Classification/fault isolation accuracy (5x2 cross-validation) under *idle* conditions

Rank	Classifier	Accuracy
1	SVM	100%
2	QDA	100%
3	Fisher Discriminant Analysis Linear Quadratic	 99.9813% 100%
4	GMM	99.6747%
5	PLS	92.3520%

Table 4.6: Classification/fault isolation accuracy (5x2 cross-validation) under *normal* driving conditions

Rank	Classifier	Accuracy
1	QDA	99.84%
2	Fisher Discriminant Analysis Linear Quadratic	 98.9711% 99.8356%
3	SVM	98.2556%
4	GMM	95.1846%
5	PLS	81.2649%

The classification accuracies can be further improved using preprocessing techniques such as auto-scaling, mean-centering, PCA and PLS.

The classification task under idle conditions is much easier than the normal driving conditions. As seen in table 4.5, both SVM and discriminant analysis (linear discriminant analysis and quadratic discriminant analysis) perform well while classifying faults under idling conditions (or steady-state operating conditions) of the truck. However, the classification is reasonably good even under normal operating conditions. SVM consistently performs well with no false alarms under both operating conditions of the truck and hence, was selected as one of the techniques for fault isolation in the data-driven software.

Fault Isolation and Severity Estimation on the HIL Rig Data

The PIDs listed below were directly used as features for the data-driven fault isolation and severity estimation.

1. Current
2. Voltage
3. Pressure
4. Flow
5. PWM

The fault classes used for isolation are as follows.

- Class 1: No Fault
- Class 2: Current Bias Fault
- Class 3: Pressure Bias Fault
- Class 4: Winding Resistance Fault
- Class 5: Fuel Leak
- Class 6: Filter Plugged

Table 4.7 presents the fault Isolation results for the HIL Rig. SVM, and KNN showed the highest accuracy of correct classification rate (>99%). On the other hand, the Bayes and PLS classifiers showed the lowest accuracy. After a fault is detected and isolated, the severity estimation of the fault is needed in some cases. We used partial least squares regression (PLSR) and principal component regression (PCR) to estimate the severity of the isolated fault.

Simulations were run on the HIL Rig to collect data for each severity level. The test drive cycle was run for each severity level of each failure model and PIDs were collected using the Simulink-dSpace model of the ERFS system. Table 4.3 presented the different severity levels for each fault class. The average percent error for each severity level is computed as follows:

$$\left| \frac{\text{Actual severity level} - \text{Average estimated severity level}}{\text{Actual severity level}} \right| \times 100 \qquad (4.1)$$

Table 4.7: Classification/fault isolation accuracy (5x2 cross-validation) for HIL rig data

Rank	Classifier	Correct Classification Rate (%)	Overall False Alarm (%)
1	SVM	99.7028%	0.2972%
2	k-Nearest Neighbor		
	$k = 1$	99.5218%	0.4782%
	$k = 2$	99.5218%	0.4782%
	$k = 3$	99.4565%	0.5435%
3	Fisher Discriminant Analysis		
	Linear	85.2393%	14.761%
	Quadratic	81.3819%	18.618%
4	Bayes Classifier with GMM Model	82.0410%	17.959%
5	PLS	81.2871%	18.713%

Table 4.8: Average errors for each severity level for PLSR

Winding Fault	Pressure Sensor Bias Fault	Current Sensor Bias Fault	Filter Plugged	Fuel Leakage Fault
8.9049%	0.9262%	8.9693%	2.50146%	0.2821%
47.17%	4.2832%	3.3192%	8.25643%	0.4683%
5.441%	2.04771%	0.7428%	9.3655%	0.0952%
7.5665%	0.12258%	3.5819%	11.4154%	0.1155%
5.224%				
27.978%				
6.645%				
16.962%				
6.55%				
15.5783%				

Tables 4.8 and 4.9 show the average percent error for each severity level for both PLSR and PCR.

The R^2 results are presented in table 4.10. The fit accuracy doesnt provide as good an insight into the problem of severity estimation as the average percentage errors due to the fact that it looks for strictly the same value as the truth and provides a comparison between the true (Y) and estimated values ($\hat{Y}$).

$$\mathrm{R}^2(\%) = \left(1 - \frac{\left\|Y - \hat{Y}\right\|_2^2}{\|Y - mean(Y)\|_2^2}\right) \times 100 \tag{4.2}$$

Table 4.9: Average errors for each severity level for PCR

Winding Fault	Pressure Sensor Bias Fault	Current Sensor Bias Fault	Filter Plugged	Fuel Leakage Fault
9.4538%	0.8987%	8.8356%	1.7446%	0.3537%
47.4694%	4.117%	3.2657%	8.7698%	0.4622%
5.8208%	2.338%	0.5541%	9.3293%	0.1133%
7.6480%	0.4139%	3.3568%	11.6222%	0.1119%
5.43909%				
28.0482%				
6.56998%				
16.864%				
6.573%				
15.5545%				

Table 4.10: R^2 fit results for different regression methods

Faults	Regression Techniques	
	PLSR	PCR
Current Bias	92.6863%	92.6285%
Pressure Bias	98.1168%	98.1251%
Winding Fault	91.9874%	91.99%
Fuel Leak	99.9035%	99.9033%
Filter Plugged	89.4918%	89.6038%

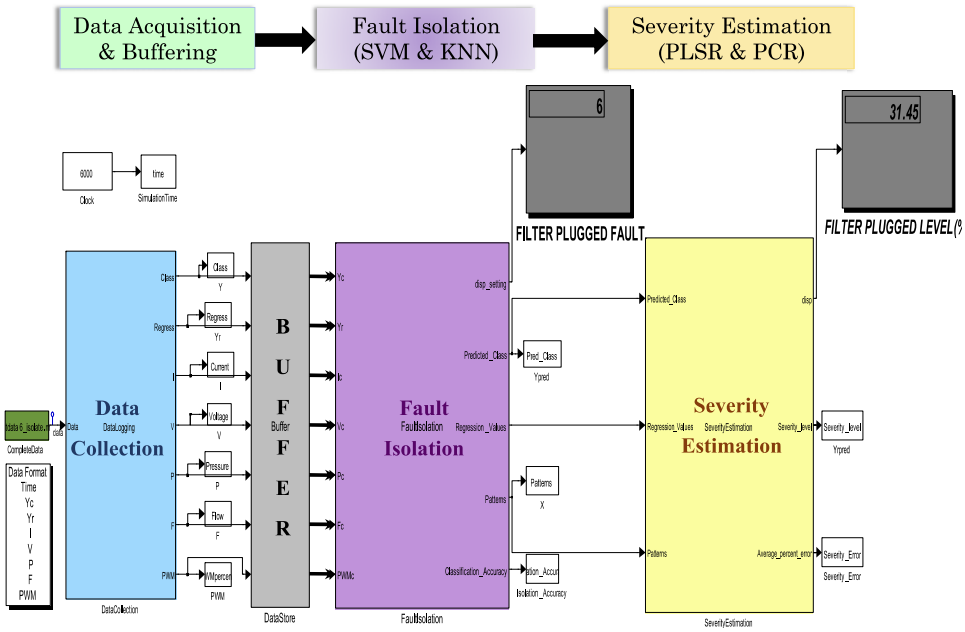


Fig. 4.7: Data-driven fault isolation and severity estimation software

The overall data-driven fault isolation and severity estimation software based on figure 4.2 was implemented in Simulink/MATLAB environment using Embedded MATLAB functions as shown in figure 4.7. The Data Acquisition and Buffering Block simulates real-time data storage of the sensor and PID data (current, voltage, pressure, and flow). Once a preset number of samples (e.g. 1000) are stored in the database, the fault isolation block consisting of SVM and KNN is triggered. As soon as the fault is isolated, the severity estimation block consisting of regression techniques (PLS and PCA) are triggered, and the severity level of the fault is estimated. The parameters for SVM, KNN, PLS and PCA are obtained in the training phase offline.

4.3 Fleet Prognosis, Fault Isolation, and CBM framework

The fleet analysis in this chapter had been performed on data from a fleet of 595 vehicles of different types.

4.3.1 Vehicle Fleet PID Distributions and Statistics

The statistics such as minimum, maximum, mean, and variance of different sensors or PIDs were collected for a fleet of vehicles in nominal condition by fitting a normal (or Gaussian) distribution to the distribution of fleet features as shown in figure 4.8. Subsequently, these statistics were used in prognostication algorithms for fault isolation, remaining useful life prediction, prognostic decision-making, and condition-based maintenance of fuel systems.

4.3.2 Fleet Prognosis, Fault Isolation, and CBM framework

In figure 4.9, we present the CBM decision-making approach with application to the ERFS based on data from a fleet of vehicles. The approach combines decisions from the fault isolation block, remaining useful life (RUL) prediction block, and prognostic decision-making (PDM) block to adopt the final maintenance action plan and subsequently, inform the customer via telematics. Three commonly occurring fuel system failure modes (motor/resistance fault, positive and negative pressure sensor bias faults) and four maintenance actions (no action, minimal maintenance, preventative maintenance, and corrective maintenance) have been considered. The fault isolation block

comprises of mode-based test-outcome evaluation with the true dependency-matrix (D-Matrix) learned from HIL and in-vehicle tests. RUL is obtained based on multiple model predictions (hidden Markov model [78], least squares Support Vector Machines Regression (LS-SVMR) [120], polynomial, and double exponential models) via three different approaches, namely, estimated end-of-life (EOL), mean Sojourn time, and hidden Markov model (HMM) most-likely state approaches. A knowledge-based approach, Markov Decision Process (MDP) [5] and Reinforcement Learning (RL) based approaches have been implemented for the prognostic decision-making block (PDM). Odometer-based averaging was used to capture the Fleet prognostic trends. On entering the VIN numbers and collecting the PIDs of Current, Resistance and Long-term Integral Correction Factor (LTICF)[1], we extract the trending information from them using exponential smoothing or robust local regression smoothing techniques. Subsequently, the prognostic features are fed into the mode-decision block to decide the mode-of the feature i.e. whether it is increasing, decreasing, or constant. Based on offline HIL and in-vehicle testing, we calculate the True Dependency Matrix (discussed in next section) i.e. relationship between the Faults and Tests, and based on the test-outcomes from the mode-decision block, we isolate the fault by computing the minimum Euclidean distance or maximum correlation metrics.

The online framework was implemented using windows, viz., 100 samples/measurements are processed at each run of the algorithms, and once the RUL and prognostics decisions

[1] filtered integral corrective action

Table 4.11: General Feature Patterns or Fault Signatures from HIL and In-vehicle Testing

Faults	Tests					
	Current	Resistance	LTICF	Input Power	Applied Voltage	PWM Adjusted
Normal	Constant	Constant	Constant	Constant	Constant	Constant
Motor/Resistance Fault	Decreases (*small*)	Increases	Increases	Increases	Increases	Increases
Pressure Sensor Bias (Positive)	Decreases (*large*)	Increases	Decreases	Decreases	Decreases	Decreases
Pressure Sensor Bias (Negative)	Increases	Decreases (*small*)	Increases	Increases	Increases	Increases

are computed, the past measurements are thrown-out and subsequently, 100 recent measurements are used and outputs updated accordingly. This improves the computational complexity of the prognostication algorithms. Another approach for complexity reduction, improving speed and memory requirements would be to employ parallelization of algorithms.

4.3.3 Fuel System Dependency Matrix

In this subsection, we derive the final D-matrix (Tables 4.11, 4.12, 4.13) with features which provide maximum diagnosability and prognostication of the fuel system. We consider 3 features (current, resistance, and long-term integral correction factor) and 4 failure modes (normal, motor resistance fault, and positive and negative pressure sensor bias faults) for our analysis. The framework in figure 4.9 can be easily extended to incorporate other features and failure modes.

Table 4.12: Denoting Feature Modes with Constant $= 0$, Increasing $= 1$, Decreasing $= -1$

Faults	Tests					
	Current	Resistance	LTICF	Input Power	Applied Voltage	PWM Adjusted
Normal	0	0	0	0	0	0
Motor/Resistance Fault	-1	1	1	1	1	1
Pressure Sensor Bias (Positive)	-1	1	-1	-1	-1	-1
Pressure Sensor Bias (Negative)	1	-1	1	1	1	1

Table 4.13: True D-Matrix obtained by removing redundant tests with same fault signatures

Faults	Tests		
	Current	Resistance	LTICF
Normal	0	0	0
Motor/Resistance Fault	-1	1	1
Pressure Sensor Bias (Positive)	-1	1	-1
Pressure Sensor Bias (Negative)	1	-1	1

4.3.4 Clustering Results

This section (figure 4.10) presents the prognostic feature clusters (VIN odometer-based averaging of features in similar modes i.e. either constant, increasing or decreasing) of current, resistance, and long-term integral correction factor (LTICF) under different failure modes.

4.3.5 Multiple Model Predictions

Once clustered, the prognostic features are predicted using different models as shown in figure 4.11. The red predictions indicate HMM, dark green predictions are performed via LS-SVMR, orange shows polynomial model predictions, and maroon presents double exponential model predictions. The black bands in the figures indicate the feature EOL thresholds based on fleet statistics.

4.4 Remaining Useful Life Prediction

4.4.1 Fleet of Vehicles

Estimated End-of-Life (EOL) Approach

In this section, we discuss the estimated end-of-life approach where, we predict the EOL using the multiple models based on an EOL threshold determined from fleet statistics. The steps are as follows:

1. Compute the mean and standard deviation of estimated EOL based on predictions

of different features and modeling approaches

$$t \rightarrow \text{current time}, \varepsilon \rightarrow \text{fleet statistics based threshold } (\pm 6\sigma_{Fleet}) \tag{4.3}$$

$$EOL_{f_{mn}}(t) = \begin{cases} \text{Predicted Time}(\sigma_{feature} > \varepsilon), \text{if feature is increasing} \\ \text{Maximum Time (assumed)}, \text{if feature is constant} \\ \text{Predicted Time}(\sigma_{feature} < \varepsilon), \text{if feature is decreasing} \end{cases} \tag{4.4}$$

$$\hat{\mu}_{EOL}(t) = mean\left(EOL_{f_{11}}(t), EOL_{f_{12}}(t), ..., EOL_{f_{mn}}(t)\right) \tag{4.5}$$

$$\hat{\sigma}_{EOL}(t) = std\left(EOL_{f_{11}}(t), EOL_{f_{12}}(t), ..., EOL_{f_{mn}}(t)\right) \tag{4.6}$$

$$\text{where}, f_{11}, f_{12}, f_{13}, ..., f_{mn} \rightarrow N \text{ features predicted using } M \text{ prognostic models} \tag{4.7}$$

2. Compute RUL by subtracting the current time from estimated mean EOL

$$RUL_{mean}(t) = \left[\hat{\mu}_{EOL}(t) - t\right] \tag{4.8}$$

$$RUL_{std}(t) = \hat{\sigma}_{EOL}(t) \tag{4.9}$$

Figure 4.12 presents the estimated end-of-life for the fleet of vehicles under different failure modes.

As seen in the figure, nominal vehicles have the maximum average EOL, whereas negative bias vehicles have the minimum average EOL.

Mean Sojourn-time Approach

The steps for computing the RUL from the mean Sojourn-time approach have been presented below [125].

1. Compute the mean Sojourn-time or mean stay time in each system health state

$$\mu\left(D\left(s_i\right)\right) = \frac{1}{\Omega}\sum_{w=1}^{\Omega} D\left(s_{iw}\right) \tag{4.10}$$

2. Compute the Sojourn-time standard deviation for each system health state

$$\sigma\left(D\left(s_i\right)\right) = \sqrt{\frac{1}{\Omega}\sum_{w=1}^{\Omega}\left[D\left(s_{iw}\right) - \mu\left(D\left(s_i\right)\right)\right]^2} \tag{4.11}$$

   $i \rightarrow$ State index, $D\left(.\right) \rightarrow$ Visit duration (Days or Odometer Reading), $w \rightarrow$ Visit index, $\Omega \rightarrow$ Total number of visits.

3. The current state is the most persistent health state in the last observations. The observations factor l is set to 10.

   State sequence = $(s_1, s_2, ..., s_t)$, last states = $(s_{t-l}, ..., s_{t-2}, s_t)$, $l \rightarrow$ past observations factor, $t \rightarrow$ current time.

4. Find critical path from the current state to failure state. The last health state (i.e. 10) is considered the failure state.

   - Minimal Path: Greedy Search Algorithm (*Pessimistic*)
   - Longest Path: Path traversing all states from current to failure state (*Optimistic*)

5. The RUL is then computed based on critical paths from step 4.

$$RUL_{upper} = \sum_{i=\text{currentstate}}^{N} \left[\mu\left(D\left(s_i\right)\right) + cf.\sigma\left(D\left(s_i\right)\right)\right] \tag{4.12}$$

$$RUL_{mean} = \sum_{i=\text{currentstate}}^{N} \mu\left(D\left(s_i\right)\right) \tag{4.13}$$

$$RUL_{lower} = \sum_{i=\text{currentstate}}^{N} \left[\mu\left(D\left(s_i\right)\right) - cf.\sigma\left(D\left(s_i\right)\right)\right] \tag{4.14}$$

$\forall i \in$ state in active critical path, cf $\rightarrow$ confidence coefficient.

From figure 4.13, we can see that the RUL of failure mode vehicles degrades considerably faster than nominal mode vehicles.

Hidden Markov Model-based: Most Likely States

We also implemented a hidden Markov model approach for RUL predicted as shown below [17]:

1. First, convert the HMM model to a left-to-right HMM to represent the fuel system health state evolution (as the health of a system degrades forward in time) by,

   (a) Setting the initial state probability vector $\pi_1 = 1 \text{and} \pi_i = 0, i \neq 1$, so that every sequence is constrained to start at state 1, and,

   (b) Setting state transition probability matrix such that

   •

$$p_{ij} = 0, \text{if} j < i \tag{4.15}$$

- Constrain the transition matrix so that large state changes do not occur

$$p_{ij} = 0, \text{if} j > i + \Delta \tag{4.16}$$

(c) In our algorithm, transition probabilities are all set to zero except for those that keep state index k the same or increment by 1, i.e. $\Delta = 1$

2. Subsequently, compute the most likely health state evolution as follows:

(a) The posterior distribution of hidden states is calculated using the *Forward-Backward Algorithm*

$$\gamma\left(s_n\right) = p\left(s_n|Z\right) = \frac{\alpha\left(s_n\right)\beta\left(s_n\right)}{p\left(Z\right)} \tag{4.17}$$

$$\alpha\left(s_n\right) = p\left(z_1, ..., z_n, s_n\right) \tag{4.18}$$

$$\beta\left(s_n\right) = p\left(z_{n+1}, ..., z_n|s_n\right) \tag{4.19}$$

(b) Then, the optimal health state sequence (Individually most-likely sequence of states) is found using

$$X_t^* = \underset{s_i \in S}{\arg\max}\, \gamma_t\left(i\right), 1 \leq t \leq T \tag{4.20}$$

3. Finally, the RUL is computed as the time required (in days or odometer reading) by the fuel system to go from the current health state to failure state.

The health state evolution for the different features have been presented in figure 4.14.

Figure 4.15 shows the RUL in kilometers for small fleet vehicles in different failure modes. The positive and negative sensor bias vehicles degrade much faster than vehicles in nominal mode.

4.4.2 Individual Vehicles: Hierarchical Hidden Markov Model Approach

In this subsection, we present the RUL predicted for individual small fleet vehicles using the hierarchical HMM approach as shown in figure 4.16. On obtaining the PIDs such as current, resistance, and LTICF of individual vehicles, the prognostic features were extracted using a moving average filter. Then, we train a hierarchical HMM consisting of M HMMs for N states of the fuel system [24]. Now, once we acquire the PID data online, the prognostic features are extracted and multiple model predictions are made, and the system health state is decoded based the maximum likelihood among the hierarchical NxM HMMs. If the current health state is not equal to the failure state, we update k and continue L-step prediction until the health state is equal to the failure state, then RUL = k time units [68].

The multiple model predictions for 595 vehicles using 3 different models, LS-SVR (dark green), polynomial (orange), and double exponential (maroon) have been shown in figure 4.17. Like before, the black bands in the figures represent the feature EOL thresholds based on fleet statistics.

Figure 4.18 present the predicted evolution of health states under different models.

The small fleet vehicles RUL prediction for individual vehicles is shown in figure 4.19. The figure incorporates individual vehicle moving average standard deviations in RUL confidence estimates (indicated by dotted lines). The predicted fuel system lifetime mean was found to be 1411.1 days or 3.86 years and the standard deviation was 258.0379 days or 0.707 years.

4.5 Prognostic Decision Making

4.5.1 Knowledge-based Approach

As an extension of figure 4.9, figure 4.20 presents the knowledge-based prognostic decision-making (PDM) process. In knowledge-based PDM, we select the maintenance action in each system state based on experience or knowledge, and there is no optimization involved. The feature multiple model predictions are fed to knowledge-based PDM block and based on the outputs the *action matrix* is computed from different features and models. Subsequently, the majority vote is taken to form the final CBM decision.

The assumed fuel system degradation model for knowledge-based PDM is shown in figure 4.21. This knowledge is used to compute the action matrix.

In figure 4.22, we present some results from the knowledge-based PDM on the fleet vehicles data. The action matrix consists of maintenance actions for different features and prediction models. Here we show a 4 window ahead prediction, where each window is approximately $15,000$ KMs ahead prediction. Finally, based on a majority vote for each window, the final optimal maintenance action (final decision) given by the

knowledge-based PDM block has been shown.

4.5.2 Markov Decision Process Approach

The Markov Decision Process (MDP) approach [94] is a 4-tuple consisting of the system health states, maintenance actions, transition probability model and the rewards model as shown in figure 4.23. It is fully observable, viz., the new state resulting from executing an action is known. A policy is a mapping from the system health States to maintenance actions. The goal of MDP is to provide an optimal maintenance policy to take in each particular system health state that minimizes the average expected costs or maximizes the expected sum of discounted rewards, in the long run and hence, saving warranty costs.

Three different algorithms have been implemented [25](policy Iteration, value Iteration and *Q*-learning) and the final decision is based on a minimum (conservative or risk averse decision) of all policies and is fed to CBM decision block. We consider 4 maintenance Actions [5]:

1. No action, when the system is operating normally.

2. Minimal maintenance, when the system requires a minor fix like sensor replacement.

3. Preventative maintenance where the component failure is imminent, and

4. Corrective maintenance, when the system has failed and requires replacement.

The observations are the prognostic features such as current, motor resistance and LTICF. The rewards matrix or cost matrix is the expected immediate reward received after transition from one state to another and comprises of the cost of maintenance, cost of failure, cost of stay in particular state and the cost of inspection.

MDP Notations

1. State Space Model S: States of MC (states of fuel system) S_f: Failed States, e.g. last state

2. Actions A, i.e. No Action (NA), Minimal Maintenance (MM), Preventative Maintenance (PM), Corrective Maintenance (CM)

3. Cost Structure

   $C_M(s, a)$: Cost of maintenance action, $a \in A$

   $C_F(s)$: Cost of each failure, $s \in S_f$

   $C_I(s)$: Cost of inspection in state $s \in S$

   $C_S(s)$: Cost per unit time stay in state $s \in S_f$

4. $Q_{ij}(a)$: Probability that maintenance action $a \in A$ performed in state i brings the system to state j

MDP Implementation Details

The MDP algorithm implementation assumes the following parameter values:

1. Fuel System Health States, $S = 10$

2. Failure State, $S_f = 10$

3. Cost of Maintenance Actions

   (a) Cost of NA, $C_M(NA) = \$0$

   (b) Cost of MM, $C_M(MM) = \$100$

   (c) Cost of PM, $C_M(PM) = \$500$

   (d) Cost of CM, $C_M(CM) = \$1000$

4. Cost of Failure, $C_F = \$500$

5. Cost of Inspection, $C_I = \$100$

6. Cost per unit time stay in state, $C_S = \$10$

7. Probability that maintenance action $a \in A$ performed in state i brings the system to state j

   (a) If NA is performed, there is no change in deterioration

$$Q_{ij}(NA) = \begin{cases} 1 \text{ for i} = \text{j} \\ 0 \text{ for i} \neq \text{j} \end{cases} \tag{4.21}$$

   (b) If MM is performed, 1 stage of deterioration is reduced

$$Q_{ij}(MM) = \begin{cases} 1 \text{ for i} = \text{j} + 1 \\ 0 \text{ for i} \neq \text{j} + 1 \end{cases} \tag{4.22}$$

(c) If PM or CM is performed, the system reaches perfect or initial state

$$Q_{ij}\left(PM\right) = Q_{ij}\left(CM\right) = \begin{cases} 1 \text{ for j} = 1 \\ 0 \text{ for j} \neq 1 \end{cases} \tag{4.23}$$

MDP Algorithm

The MDP algorithm steps are as follows:

1. Based on the current system health state s, select N equally distributed states over a possible range for next health state (s)

2. Solve Markov Chain for $p_{ij}\left(t\right)$ and $\tau_{ij}\left(t\right)$.

$$p_{ij}\left(t\right) \rightarrow P\left\{\text{state } j \text{ at time } t | \text{ initial state is } i\right\} \tag{4.24}$$

$$\tau_{ij}\left(t\right) \rightarrow E\left\{\text{ time spent in state } j \text{ in } \left(0, t\right) | \text{ initial state is } i\right\} \tag{4.25}$$

3. For $s, s \in S$, $a \in A$, compute $P(s, s, a)$

$$P\left(s, s', a\right) = \sum_{k \in S} Q_{sk}\left(a\right) \tau_{ks'}\left(t\right) \tag{4.26}$$

4. For $s, s \in S$, $a \in A$, compute $Y(s, a)$ i.e. Expected transition time

$$Y\left(s, a\right) = \sum_{s' \in S_f} \sum_{k \in S} Q_{sk}\left(a\right) \tau_{ks'}\left(t\right) \tag{4.27}$$

5. For $s, s \in S$, $a \in A$, compute expected cost or rewards matrix $R(s,a)$

$$\begin{aligned}\text{Overall Expected Cost}: \ \mathrm{R(s,a)} \ = \ & \overbrace{C_M(s,a)}^{\text{Cost of Maintenance}} + \sum_{s' \in S_f} P(s,s',a) \ \overbrace{C_F(s')}^{\text{Cost of Failure}} \\ & + \sum_{s' \in S_f} Q_{sk}(a)\, \tau_{ks'}(t) \quad \overbrace{C_S(s')}^{\text{Cost of Stay in State}} \\ & + \sum_{s' \in S} P(s,s',a) \quad \overbrace{C_I(s')}^{\text{Cost of Inspection}} \end{aligned} \tag{4.28}$$

6. Using $P(s,s,a)$, $Y(s,a)$, $R(s,a)$, compute *optimal policy*:

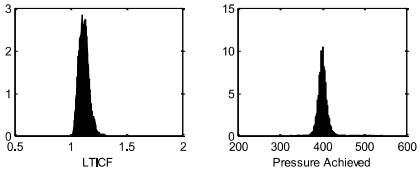
   (a) Policy Iteration Algorithm (figure 4.24) [13]

   (b) Value Iteration Algorithm (figure 4.25) [13]

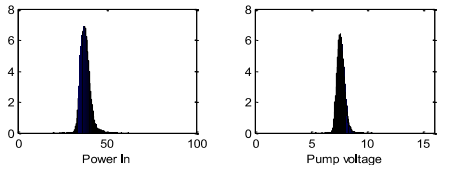
   (c) *Q*-learning Algorithm: Reinforcement Learning Technique (figure 4.26)

7. Final Policy (risk-averse):

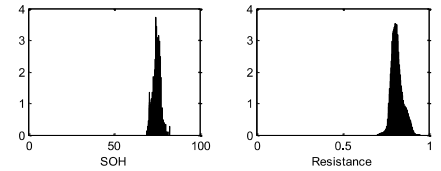
$$\min\left\{\pi^*_{PI}(s), \pi^*_{VI}(s), \pi^*_{QL}(s)\right\} \tag{4.29}$$



Feature/STATS	MIN	MAX	MEAN	VAR
LTICF	1.0485	1.2530	1.1474	0.0014
Pressure Error	0	144	19.9296	210.1523



Feature/STATS	MIN	MAX	MEAN	VAR
Input Power	22.4918	76.8627	39.7519	10.9310
Pump Voltage	5.6	11.7529	7.822	0.1422
Current	0.0998	7.0607	5.0770	0.0495



Feature/STATS	MIN	MAX	MEAN	VAR
SOH	6.6178	83.8407	75.8550	8.5624
Resistance	0.6496	0.8678	0.8036	5.0865×10^{-4}
Desired Pressure	285	543	401.4203	673.4217

Fig. 4.8: Illustrative Vehicle Fleet PID Distributions and Statistics

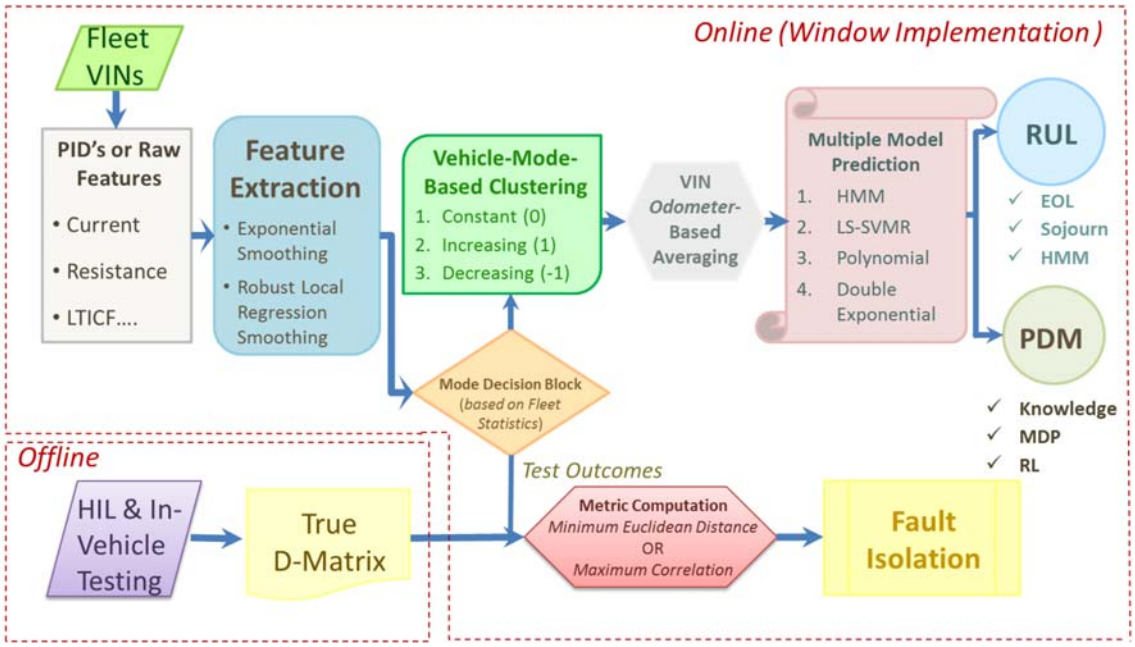
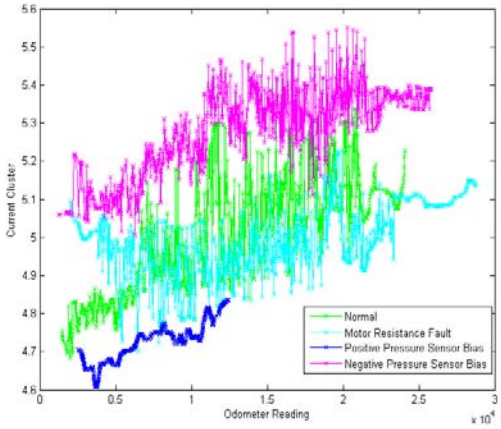
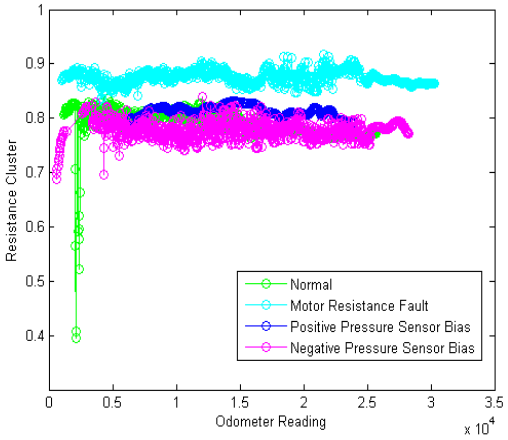


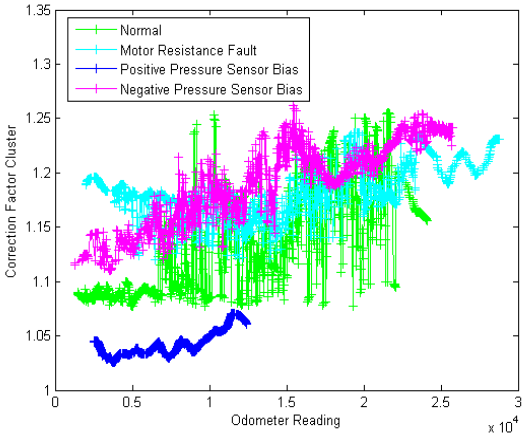
Fig. 4.9: Fleet Prognosis, Fault Isolation, and CBM framework



(a) Current Cluster

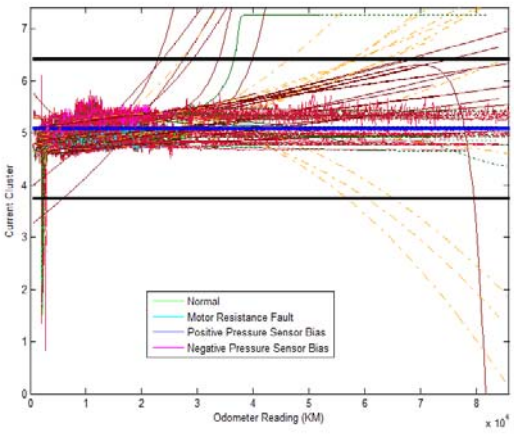


(b) Resistance Cluster

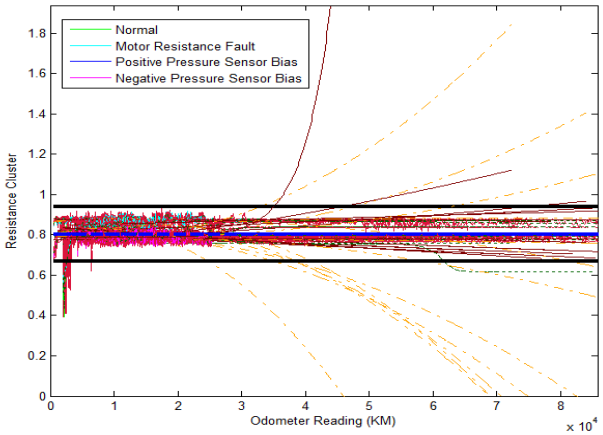


(c) LTICF Cluster

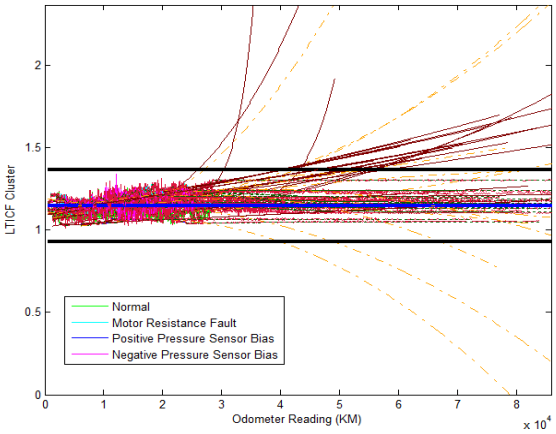
Fig. 4.10: Clustering Results



(a) Current Cluster Predictions



(b) Resistance Cluster Predictions



(c) LTICF Cluster Predictions

Fig. 4.11: Multiple Model Prediction Results

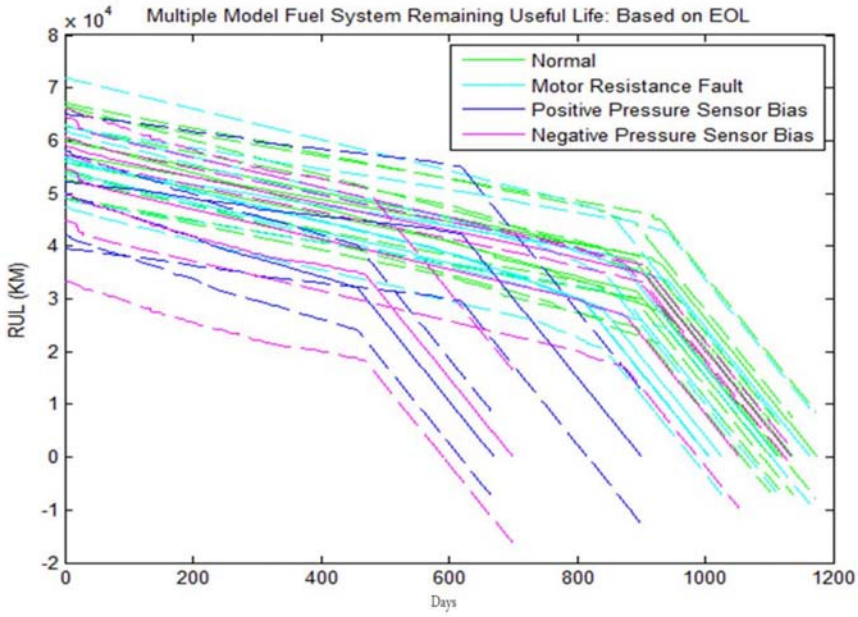


Fig. 4.12: Estimated EOL for different failure modes

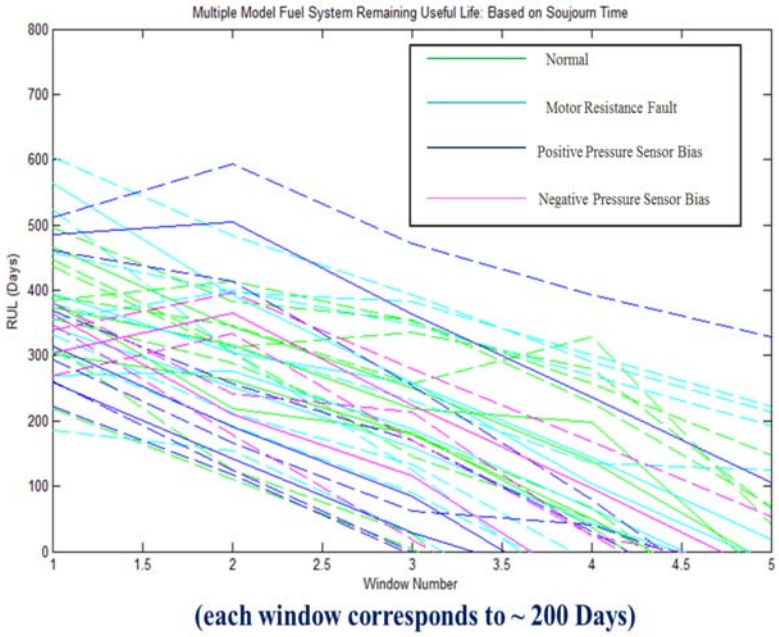
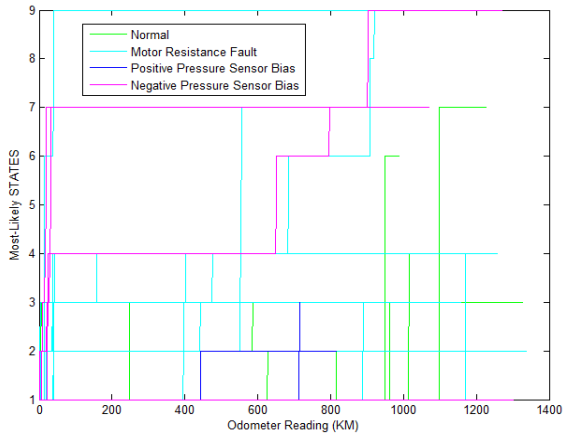
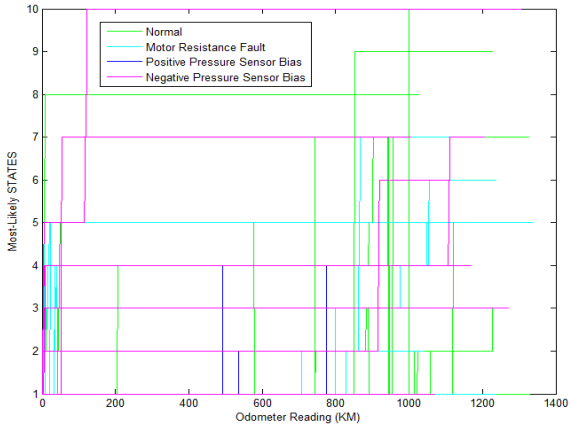


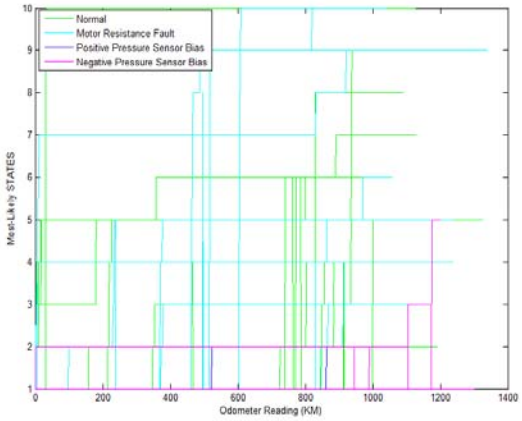
Fig. 4.13: Estimated RUL using Sojourn-time Approach



(a) Current Health States Evolution



(b) Resistance Health States Evolution



(c) LTICF Health States Evolution

Fig. 4.14: Feature Health State Evolution

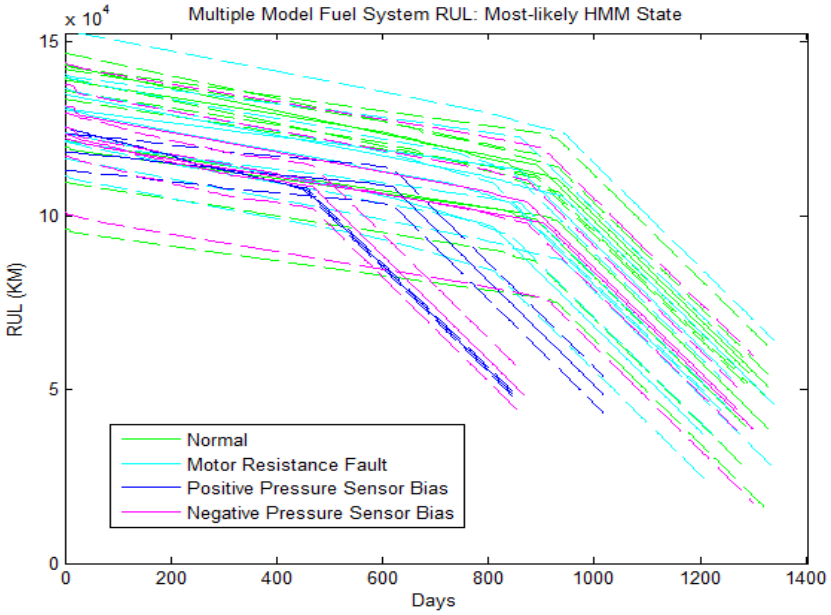


Fig. 4.15: Estimated RUL using HMM most-likely health state evolution

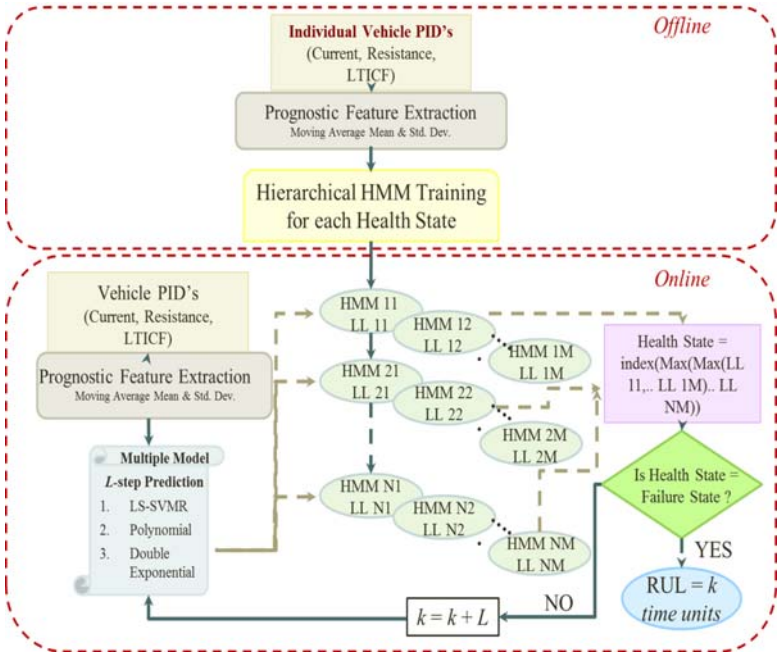
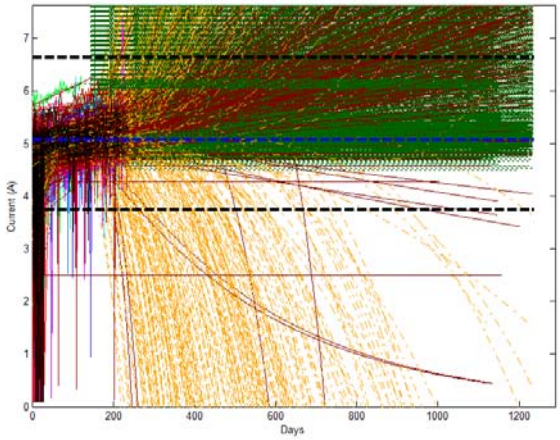
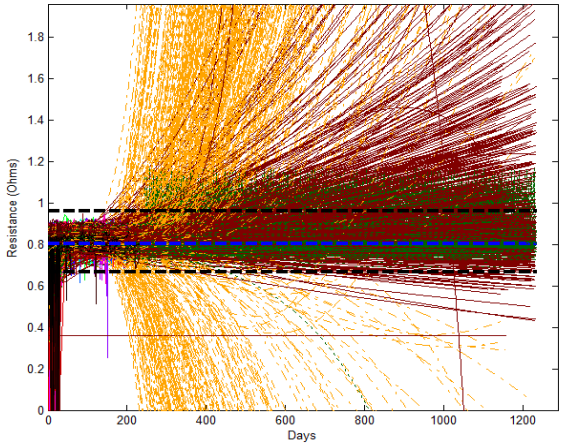


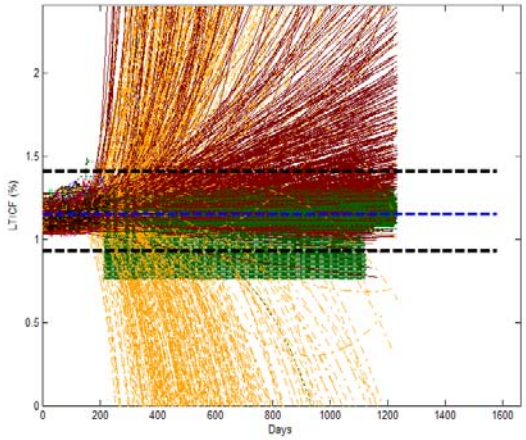
Fig. 4.16: Hierarchical HMM RUL prediction approach



(a) Current Multiple Model Predictions

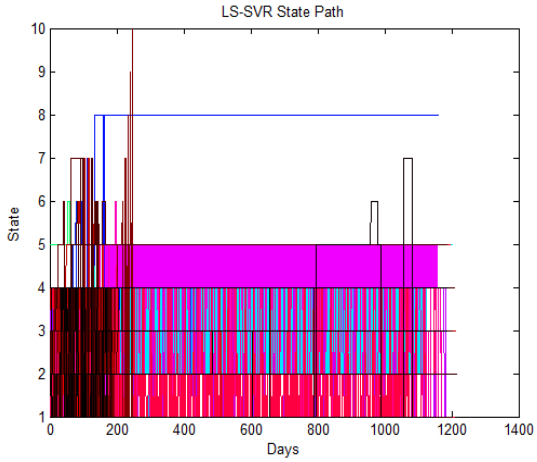


(b) Resistance Multiple Model Predictions

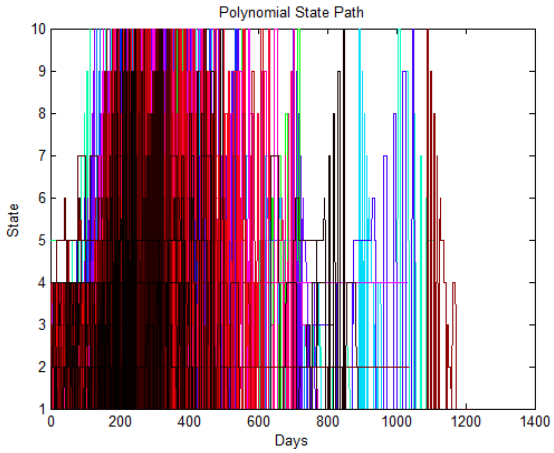


(c) LTICF Multiple Model Predictions

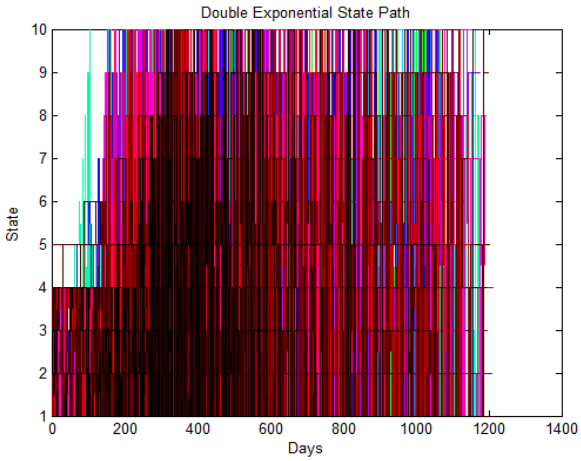
Fig. 4.17: Feature Multiple Model Predictions



(a) LS-SVR based Health State Evolution



(b) Polynomial based Health State Evolution



(c) Double Exponential based Health State Evolution

Fig. 4.18: Multiple Model Health State Evolution

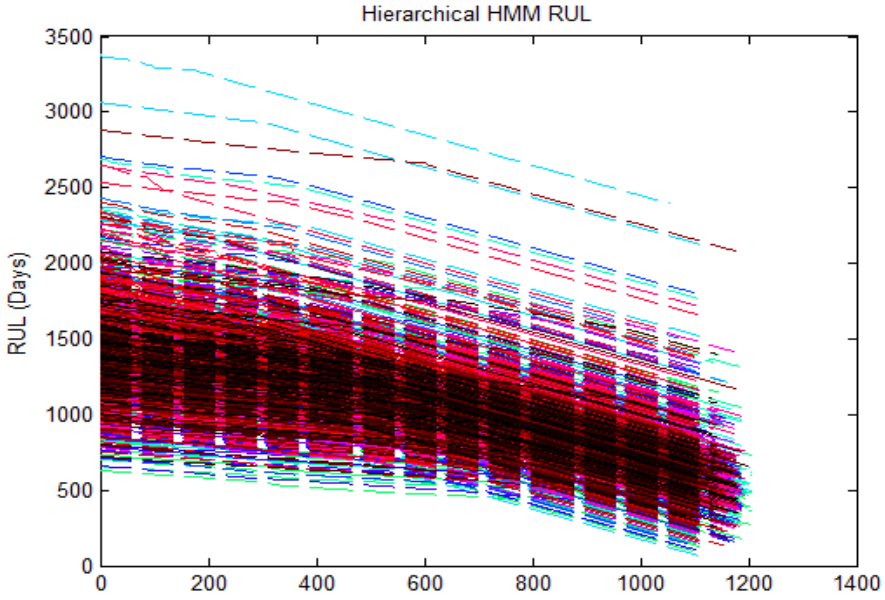


Fig. 4.19: Hierarchical HMM Individual Vehicle RUL Predictions

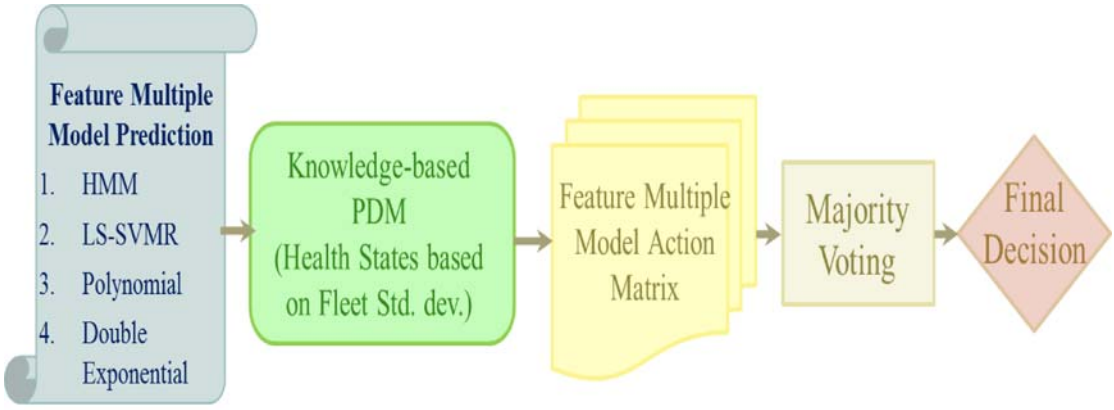


Fig. 4.20: Knowledge-based PDM Approach

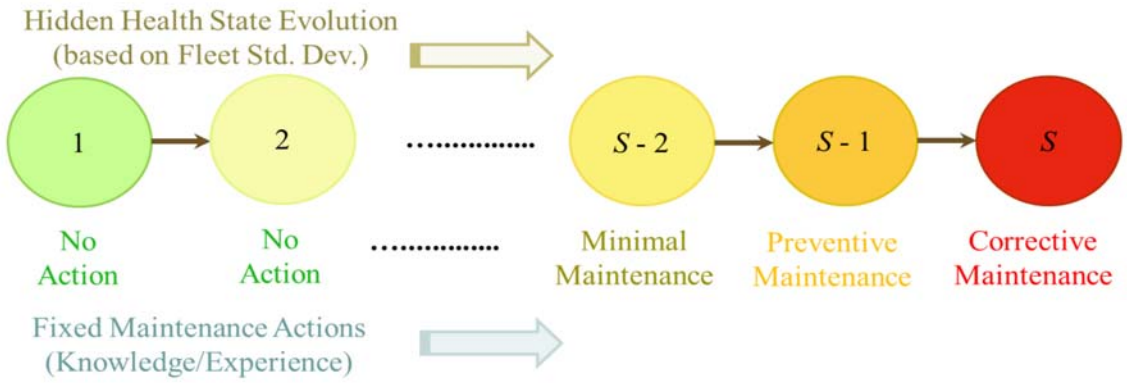


Fig. 4.21: Fuel system degradation model for knowledge-based PDM

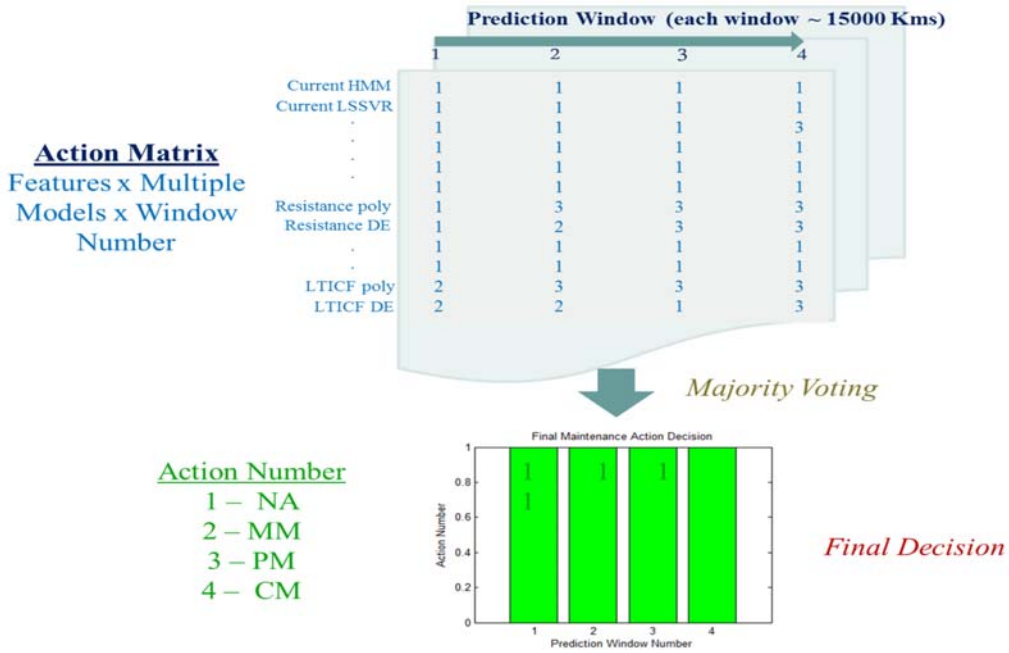


Fig. 4.22: Knowledge-based PDM Result

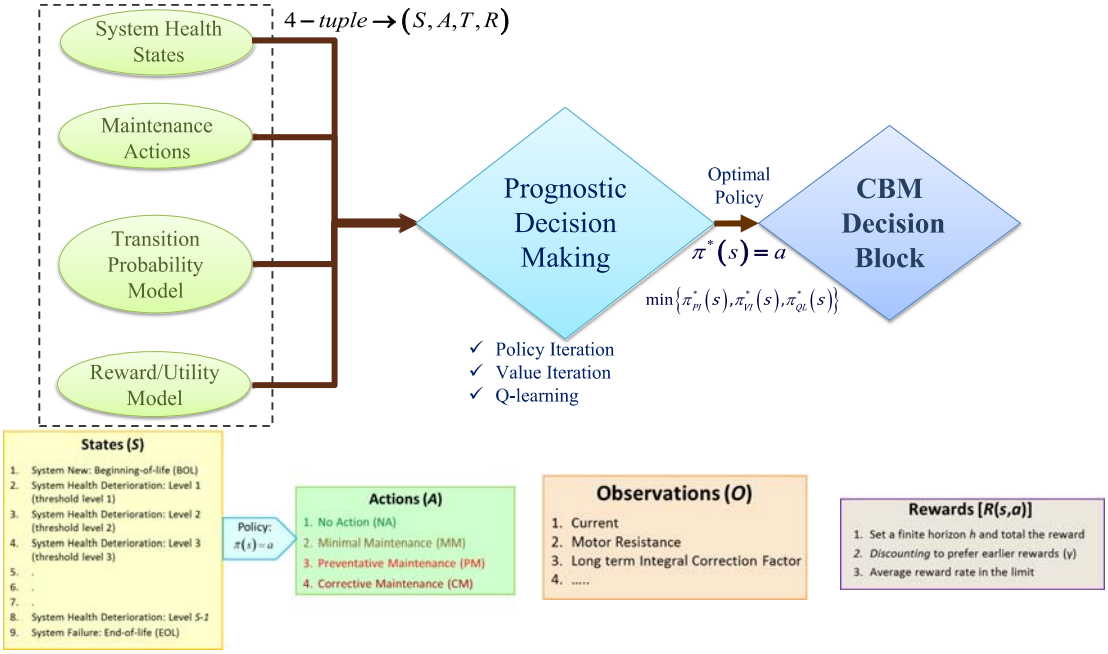


Fig. 4.23: MDP Approach

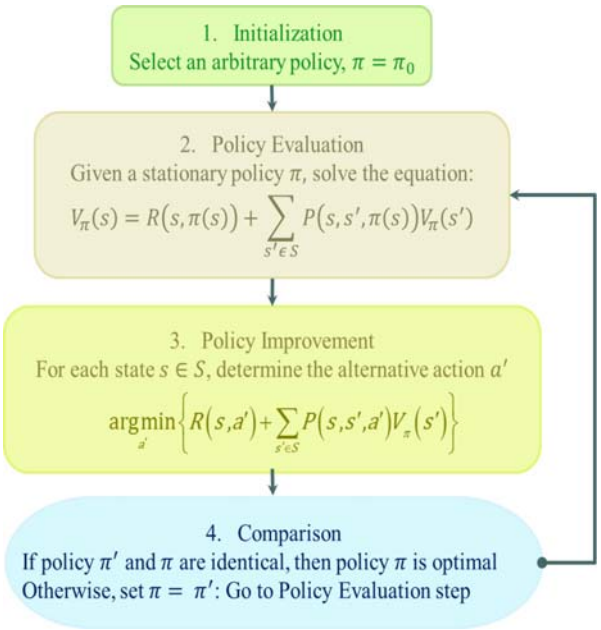


Fig. 4.24: Policy Iteration Algorithm

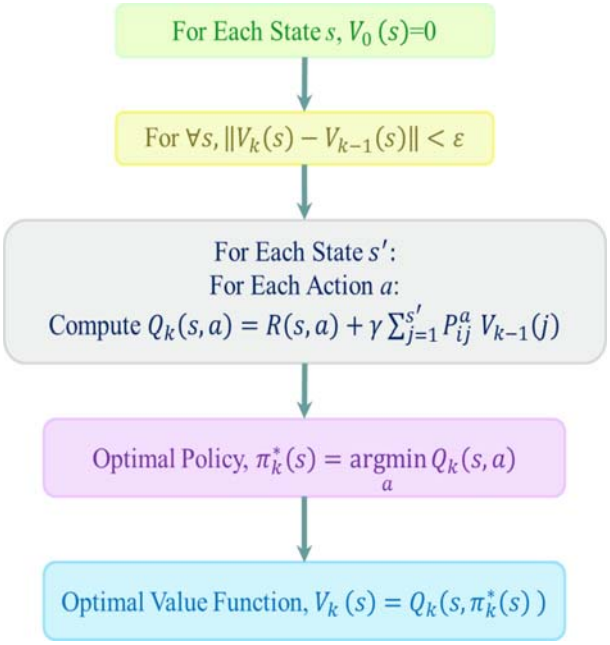


Fig. 4.25: Value Iteration Algorithm

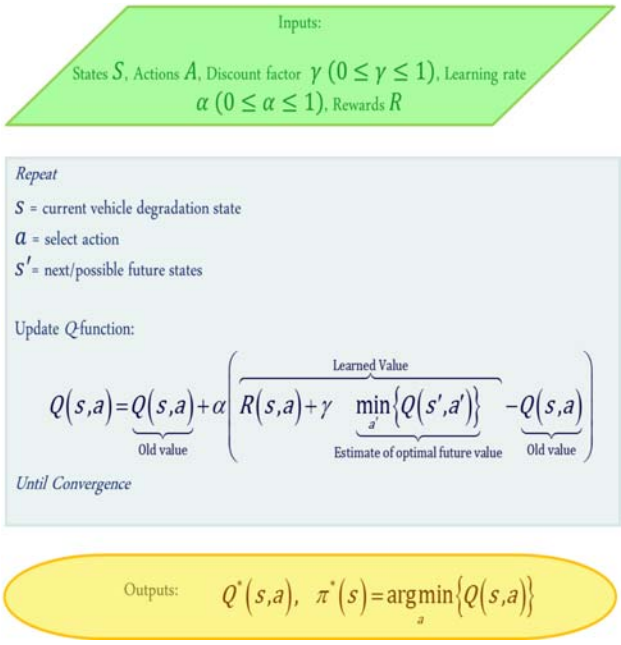


Fig. 4.26: Q-learning Algorithm

The MDP result for a fleet of vehicles is presented in figure 4.27. The overall policy is computed by taking the majority vote among different feature-multiple model predictions. MDP computes the optimal maintenance policy for each health state. We have assumed a 10-state degradation model for the fuel system and the optimal MDP policy for each system state has been shown below.

In figure 4.28 we present the overall CBM decision block for the fuel system. The fault isolation library consists of model-based and data-driven blocks (and maybe knowledge-based block). The RUL and survival function library comprises of the multiple model approach, hierarchical HMM approach and survival function estimation techniques such as multiple model moving horizon estimation approach and the soft DMFD. The PDM library consists of the knowledge-based approach, MDP, partially

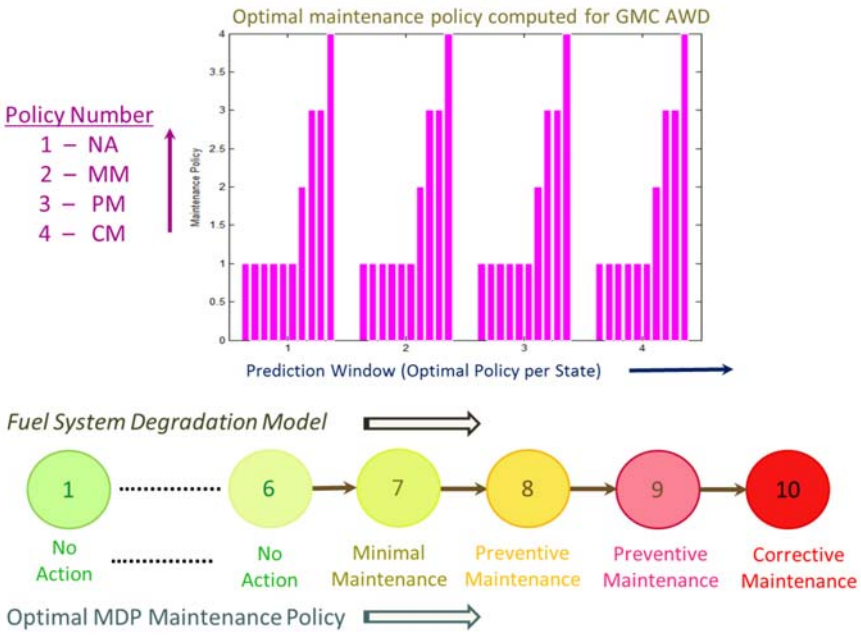


Fig. 4.27: MDP PDM Approach Result on a fleet of vehicles

observable Markov decision process (POMDP) and reinforcement learning techniques. The Condition-based decision block fuses the decisions made by each individual block and makes an overall decision which can then be informed to the customer via telematics, if required. The warranty data provides validation of the CBM performance and is fed-back to improve the decisions.

4.6 Summary

The contributions of this research can be summarized as follows:

- Developed data-driven Fault Diagnosis and Isolation (FDI) approaches based on neural network and statistical pattern recognition techniques for Automotive ERFS

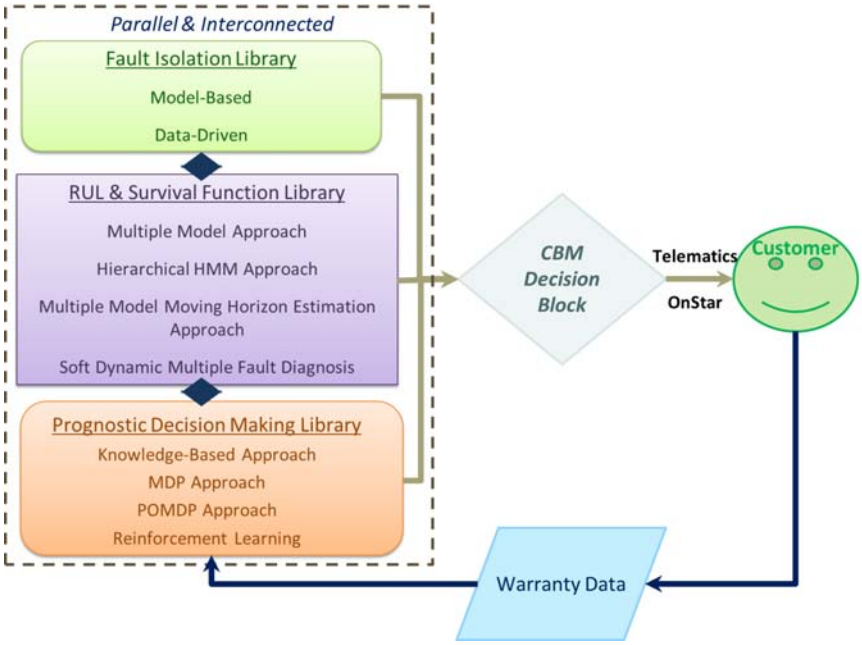


Fig. 4.28: Fuel System Roadmap: Overall Decision Block

- HIL testing for rapid prototyping of ERFS Validated proposed FDI framework using HIL and in-vehicle experiments
- Designed and developed an embedded data-driven FDI software for ERFS
- Fleet Prognosis and Fault Isolation Algorithms
- Multiple Model and Hierarchical HMM approaches for Fleet and Individual Vehicle RUL Prediction
- Prognostic Decision making algorithms based on expertise/knowledge and Markov Decision Process
- Fuel System Overall CBM Decision Block layout fusing different frameworks

Chapter 5

Multiple Model Moving Horizon Estimation Approach to Prognostics in Coupled Systems

The key objectives of this chapter are to analyze and implement a novel moving horizon model predictive estimation scheme based on constrained nonlinear optimization techniques for inferring the survival functions and residual useful life (RUL) of components in coupled systems. The approach employs a data-driven prognostics framework that combines failure time data, static and dynamic (time-series) parametric data, and the Multiple Model Moving Horizon Estimation (MM-MHE) algorithm for predicting the survival functions of components based on their usage profiles. Validation of the approach has been provided based on data from an electronic throttle control (ETC) system. The proposed prognostic approach is modular and has the potential to be applicable to a wide variety of systems, ranging from automobiles to aerospace.

5.1 Introduction

The ideas of Moving Horizon Estimation (MHE) date back to the early 1990s [75]. The fundamental philosophy of MHE is to estimate the temporal evolution of states by solving a (nonlinear) least squares problem, while penalizing the deviation between the measurements and predicted outputs of the system. The observations considered for optimization lie in a fixed finite length time window. The MHE problem is appealing due to the ability to incorporate nonlinearities and constraints on states and disturbances. The viability and stability of MHE for state estimation of linear and nonlinear systems, along with its ability to explicitly incorporate constraints, was shown to perform better than other strategies, such as extended Kalman filtering and output error linearization. The convergence of MHE algorithms depend on the accuracy with which the old data is approximated by the arrival cost (analogous to cost-to-arrive concept in deterministic dynamic programming). Hence, the state estimate is determined online by solving a finite horizon optimization problem. As new measurements become available, the old measurements are discarded from the estimation window, and the finite horizon state estimation problem is re-solved to determine the new estimate of the state. Two major advantages of a moving horizon approach over an increasing horizon or batch estimation approach are that it solves a problem of fixed size at each time epoch and summarizes the past data by the arrival cost. Hence, MHE algorithms are suitable for practical implementation because they amount to problems of finite dimension and have the capability to incorporate nonlinearities and constraints on states and

disturbances [42]. Prognosis deals with early fault diagnosis and remaining useful life estimation. It plays a pivotal role in condition-based maintenance (CBM) of contemporary systems. The prognostic methods can be broadly classified into model-based and data-driven approaches [29]. Model-based techniques assume an accurate mathematical model of the system, and track the residuals using observers (such as Kalman filters, interacting multiple models, particle filters) and parity relations (dynamic consistency checks among measured variables) [69]. On the other hand, data-driven techniques are derived from routinely monitored system operating data, and are based on statistical regression techniques (such as nonlinear least squares, partial least squares, principal component analysis, support vector machines) to graphical models (Bayesian models, hidden Markov models), and signal analysis [116]. The time-series based approaches to prognosis are component-centric and do not make use of widely available data in archived databases of equipment, such as historical usage patterns, error codes, observed failure modes, repair and inspection intervals, environmental factors, skill levels of personnel, and status parameters collected periodically or at the onset of error codes. Consequently, the time-series based prognostic health management approaches are both incomplete and inaccurate for coupled systems with cross-subsystem fault propagation. On the other hand, the classical survival theory-based approaches rely on Weibull and other nonlinear regression models to infer time-to-failure, and these estimates are used to optimize the time-to-maintain or time-to-repair/replace; these techniques do not consider condition indicators of equipment and cross-subsystem fault propagation. Con-

sequently, the survival theory-based techniques result in large variability in the time-to-failure estimates. Evidently, the two disparate methodologies need to be reconciled and unified under a common modeling framework that combines archived failure time data, and static and dynamic parametric data [111]. The chapter focuses on a data-driven prognostics approach for estimating the states of component survival functions and cluster weights of an electronic throttle control (ETC) system given noisy test measurements from a mixed component cluster. The main contributions of this chapter can be summarized as follows:

1. Novel Multiple Model Moving Horizon Estimation (MM-MHE) algorithm for predicting the survival functions of components based on their usage profiles.

2. A comprehensive prognostic framework based on offline and online data for remaining useful life prediction.

The chapter is organized as follows. Section 5.2 presents a general model predictive estimation model and previous work in this area. Section 5.3 presents a unified prognostics framework that combines the failure time data, as well as static and dynamic parametric data based on Cox proportional hazards model (PHM) [33, 61]. Section 5.4 describes the multiple-model moving horizon estimation algorithm used for predicting the survival functions and cluster weights. The experimental results are presented in section 5.5. Section 5.6 concludes with summary and future research directions.

5.2 Model Predictive Estimation

Model Predictive Control (MPC) is a form of control in which the current control action is obtained online, at each sampling instant, by solving a finite horizon open-loop optimal control problem, using the current state of the system as the initial state. The process yields an optimal control sequence and the first control in this sequence is applied to the system [74]. An important advantage of this approach is its ability to cope with equality and inequality constraints on controls and states. The essence of MPC is to optimize, over the manipulable inputs, forecasts of the system behavior. Hence, MPC usually comprises of the following three fundamental ideas:

1. Explicit use of a model to predict the process output over a future time horizon;

2. Calculation of a control sequence to optimize a performance index over this horizon, and

3. The application of the first control signal of the sequence and moving the horizon by one step towards the future.

   The moving horizon MPC approach has been presented in Figure 5.1.

1. The process model computes the predicted future outputs $\hat{y}\left(k+j|k\right), j = 1, ..., N-1$ for the prediction horizon N at each time instant k. These depend on the known values up to instance k (past inputs and outputs), including the current output (or initial condition) $y\left(t\right)$ and on the future control signals $u\left(k+j|k\right), j = 0, ..., N-1$, to be calculated.

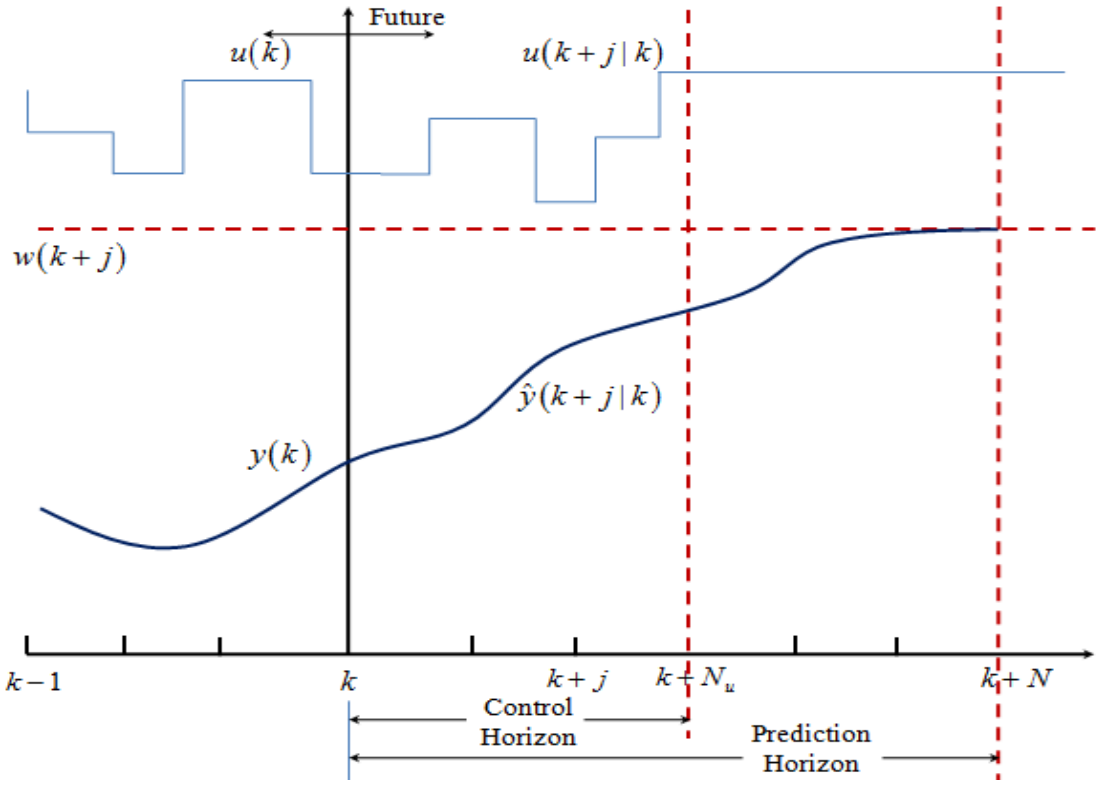


Fig. 5.1: Moving Horizon Strategy of MPC [40]

2. The sequence of future control signals is computed to optimize a performance criterion over the prediction horizon.

3. Only the current control signal $u\,(k|k)$ is applied to the system. At the next sampling instant, $y\,(k+1)$ is measured and step 1 is repeated and all sequences are brought up to date. Hence, $u\,(k+1|k+1)$ is calculated using the receding horizon concept.

The process models can be either linear or nonlinear. The predicted state estimates are computed by minimizing the predicted error or a finite horizon cost function.

The MPC concept has a long history. In the 1970s engineers at shell oil developed their independent MPC technology with an initial application in 1973 [44]. Originally developed to meet the specialized control needs of power plants and petroleum refineries, MPC technology can now be found in a wide variety of application areas including chemicals, food processing, automotive, and aerospace applications. Several publications provide a good introduction to theoretical and practical issue associated with MPC technology. Rawlings et al. [102] present an excellent tutorial aimed at control practitioners. An overview of commercially available linear and nonlinear MPC technology, its applications and brief history is provided by Qin et. al. [95]. Rao et. al. [101] investigated MHE as an online constrained optimization strategy for estimating the state variables of constrained discrete-time systems, and demonstrated the superior performance and modeling accuracy of MHE compared to other strategies, such as extended Kalman filtering and output error linearization. Samar et. al. [108, 109] proposed a model-based diagnosis approach based on MHE for estimating the evolution of fault parameters online. The proposed approach was validated by application to rocket flight control and fault diagnosis in an unmanned aerial vehicle (UAV). Authors [103] have also proposed a combination of powerful state estimation techniques such as particle filters and MHE for detecting and tracking multi-modal densities which are common in chemical process control applications.

5.3 Prognostic Framework

As shown in Figure 5.2, three types of data are considered in our prognostic framework:

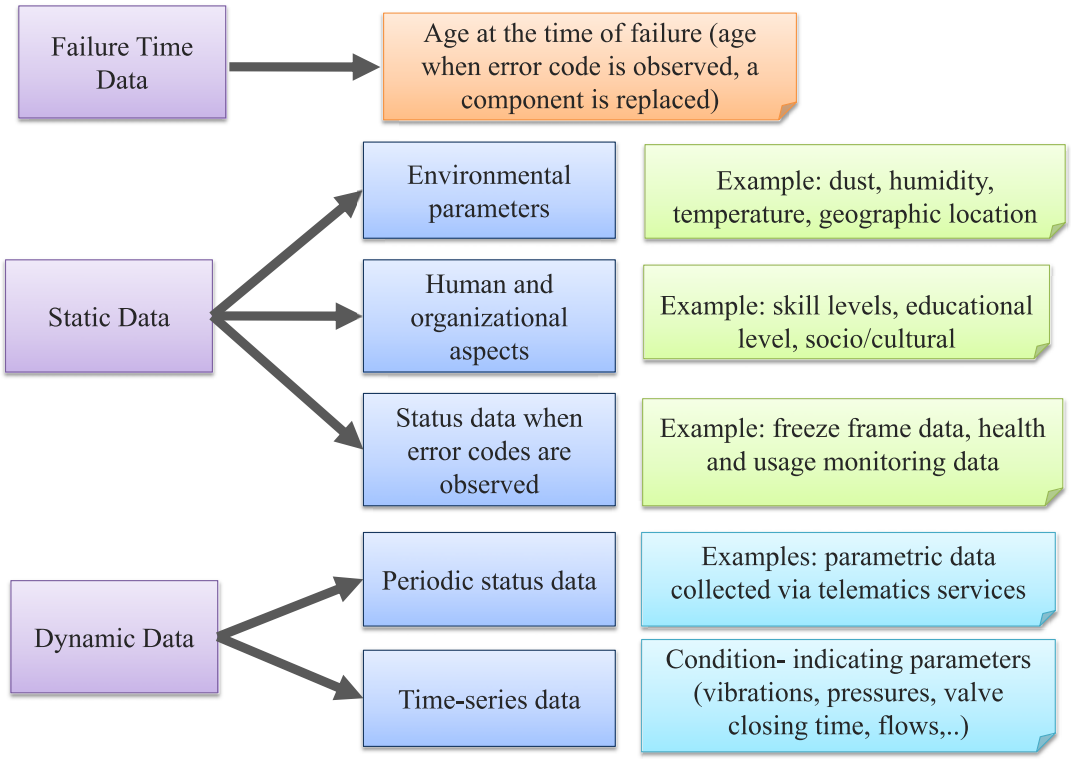


Fig. 5.2: Prognostic Data Categorization

1. Archived failure data (or Type I data): age of the equipment at the time of failure, i.e., age when an error code or symptom is observed, or a component is replaced;

2. Static environmental and status parameter data (or Type II data); and

3. Dynamic data (or Type III data): time-series data and periodic status data.

The proposed data-driven prognostic framework is shown in Fig. 5.3. It is divided into two phases, namely, the training and model learning phase (*offline module), and testing or deployment phase (*online module).

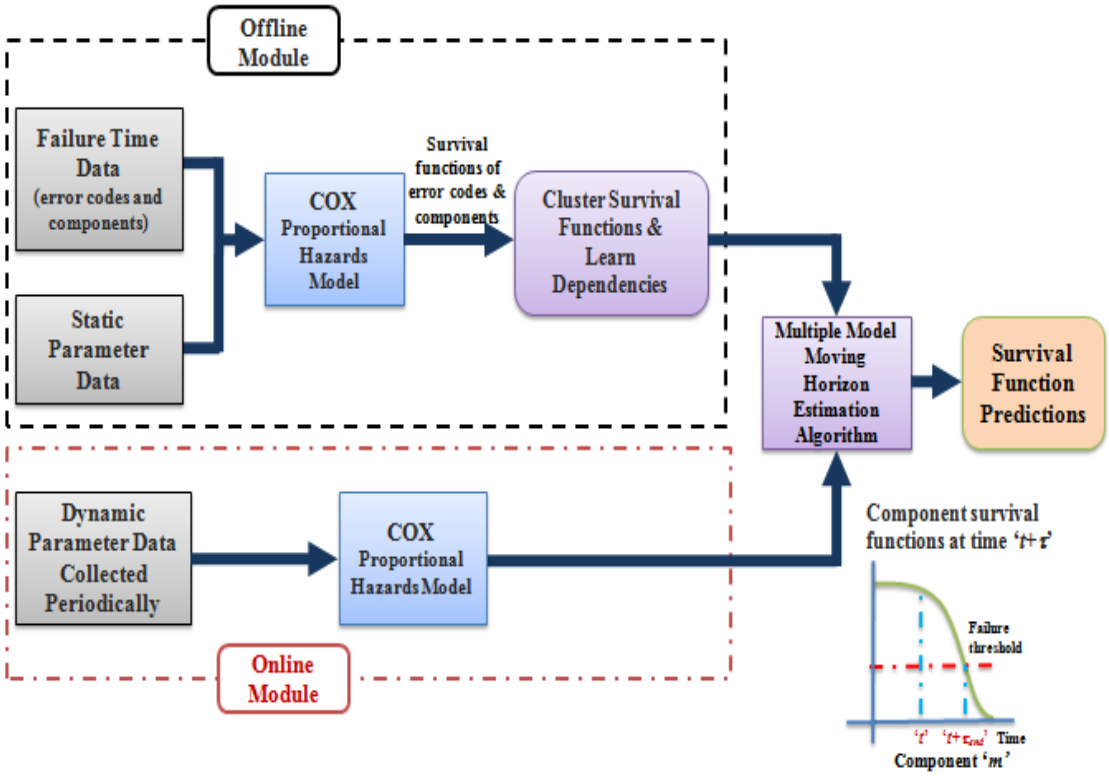


Fig. 5.3: Cox-PHM Multiple Model Prognostics Approach

In the training and validation (or model learning) phase, we compute the static data-modulated component survival functions, error codes, symptoms and any observable test outcomes via Cox proportional hazards model and cluster the survival functions for each component via clustering techniques such as k-means, learning vector quantization (LVQ), Gaussian mixture models (GMM) or hierarchical clustering

[15, 38]. These clusters represent the different usage profiles for the components depending on the usage conditions, environmental factors, etc. The Cox PHM assumes a hazard function of the form:

$$h_i\left(t,\right) = h_0\left(t\right)\exp\left(^T\right) \tag{5.1}$$

where i denotes the component, diagnostic error code, or any failure mode of interest, $\underline{z}$ is a vector of covariates (Type II static data such as freeze frame data in automotive vehicles or health and usage monitoring data), $\underline{r}$ is the vector of regression parameters, and $h_0\left(t\right)$ is the baseline hazard function (with $\underline{z} = 0$). The baseline hazard function can be from any of the standard failure time distributions (e.g., exponential, Weibull, normal, log normal, Gamma, etc.) or it can be nonparametric. The baseline hazard function and regression parameters are estimated via a maximum likelihood method [62, 132]. The component survival functions,$S_i\left(t,\underline{z}\right)$ and the associated failure density functions, $f_i\left(t,\underline{z}\right)$ can be computed from the hazard functions $h_i\left(t,\underline{z}\right)$ as follows,

$$\begin{aligned} S_i\left(t,\right) &= \exp\left(-\int_0^t h_i\left(\tau,\right)d\tau\right) \\ f_i\left(t,\right) &= h_i\left(t,\right)S_i\left(t,\right) \\ &= h_i\left(t,\right)\exp\left(-\int_0^t h_i\left(\tau,\right)d\tau\right) \\ &= -\frac{dS_i(t,)}{dt} \end{aligned}$$

(5.2)

Subsequently, in the testing (or deployment) phase, on obtaining new measurements via online data acquisition systems, the survival functions and cluster weights are estimated via the Multiple Model Moving Horizon Estimation (MM-MHE) algorithm. Once the component survival functions are obtained, the first and second order non-central moments at the time-of-failure T can be computed from the survival functions via 5.3 and 5.4 respectively.

$$E\left(T\right) = \int_{0}^{\infty} S\left(\tau\right) d\tau \tag{5.3}$$

$$Var\left(T\right) = 2\left(\int_{0}^{\infty} \tau S\left(\tau\right) d\tau\right) - \left(\int_{0}^{\infty} S\left(\tau\right) d\tau\right)^{2} \tag{5.4}$$

Hence, the remaining useful life (RUL) of a component at any time epoch t can be computed from the survival function by defining an application dependent threshold on the survival probability as shown below.

$$RUL\left(t\right) = \arg\min_{\tau}\left(S\left(t+\tau\right) \leq \varepsilon_{0}\right) \tag{5.5}$$

where ε_0 denotes the threshold of functional failure.

5.4 Multiple Model Moving Horizon Estimation Approach

The key problem is to estimate the states of the component survival functions and cluster weights (posterior probabilities of discrete states), given noisy test measurements (prognostic indicators) from a mixed component cluster. Mathematically, the dynamics of the survival function can be represented in discrete-time as [57, 79],

$$\begin{aligned} S_i\left(k+1,l\right) &= a_i\left(k,l\right) S_i\left(k,l\right) = A_i\left(k,l\right) S_i\left(0\right) \\ a_i\left(k,l\right) &= \exp\left(-h_i\left(k,l\right)\Delta\right) \\ A_i\left(k,l\right) &= \left(\prod_{r=0}^{k-1} a_i\left(r,l\right)\right) \end{aligned} \tag{5.6}$$

where Δ is the time step in days. $S_i\left(k\right)$ and $h_i\left(k\right)$ are the survival function and hazard rate of component $i = \{1, 2, ..., m\}$ at time epoch $k\Delta$. The cluster number is given by l and each component may have different clusters represented by L_i, where $l = \{1, 2, ..., L_i\}$ for component i. This implies that we need to infer the discrete-state usage profile from the prognostic indicators.

The measurements are generated based on a nonlinear function (negative logarithm) of the test survival functions (soft test outcomes), which represent a nonlinear function of mixed component survival functions with noise to mimic observations from an unknown user profile as shown below.

$$
\begin{aligned}
y_j(k) &= -\ln T_j(k) \\
&= -\sum_{i=1}^{m} \ln \left\{ \begin{array}{c} \left(\sum_{l=1}^{L_i} S_i(0) A_i(k,l) \alpha_i(l) \right) (PD_{ij} - PF_{ij}) \\ + (1 - PD_{ij}) \end{array} \right\} \\
&+ v_j(k)
\end{aligned} \tag{5.7}
$$

where, $v_j(k) \approx N(0, \sigma^2)$. Equation (7) corresponds to a noisy-OR dependency between the test outcome and the set of components monitored by that test [111]. The noise variance $\sigma^2 = |y_j(k)|^2 \delta$ (δ is varied from 0.01-5.12 in our experiments to assess the robustness of the algorithm to measurement noise. This corresponds to a signal-to-noise ratio range of 20 dB to 7 dB). Hence, R_v is a diagonal n x n measurement noise covariance matrix. The survival function of test $j = \{1, 2, ..., n\}$ at time $k\Delta$ is denoted as $T_j(k)$. The probability of detection and the probability of false alarm of test $j = \{1, 2, ..., n\}$ for failure in component $i = \{1, 2, ..., m\}$ are denoted as PD_{ij} and PF_{ij}, respectively.

The goal is to accurately predict the component degradation according to the current operating conditions. The cost function for the current window is computed based on a nonlinear least squares approach minimizing the square of the residual between the current measurements and predicted outputs of the system. The arrival cost is calculated by penalizing the deviations based on previous predicted state estimates and covariances. Hence, the overall cost is formulated as a summation of the arrival cost and the current window cost. Consequently, the MM-MHE algorithm predicts the com-

ponent survival functions and cluster weights based on the optimal cost and constraints.

The overall cost for the augmented vector $\underline{x} = \begin{bmatrix} \underline{S} \\ \underline{\alpha} \end{bmatrix}$ of survival functions and clusters is as follows:-

$$\min_{\underline{x}} \left\{ \beta \underbrace{\left(\underline{x}_{K_L} - \underline{\hat{x}}_{K_L}\right)^T \left(\Sigma_{k_L}^{-1}\right) \left(\underline{x}_{K_L} - \underline{\hat{x}}_{K_L}\right)}_{Arrival\ Cost} + \sum_{k=k_L+1}^{k_L+w} {\nu_j}^T R_\nu^{-1} \nu_j \right\} \tag{5.8}$$

where, β is the arrival cost scale factor (tuning parameter) and ν_j denotes the measurement prediction residual which is the difference between the current and predicted measurements. For implementation purpose, the augmented vector $\underline{x}$ is set up as a row vector of $\underline{S}$ and $\underline{\alpha}$, where, $\underline{S} = \begin{bmatrix} S_1(0) & . & . & . & S_i(0) \end{bmatrix}^T$ and

$\underline{\alpha} = \begin{bmatrix} \alpha_{11} & \alpha_{12} .. & \alpha_{1l} & \alpha_{21} & \alpha_{22} ... & \alpha_{il} \end{bmatrix}^T$. In the initial window, the cluster numbers $\underline{\alpha}$ have been assumed to be uniformly distributed for all components, and the component survival functions have been initialized with ones. The constraints on the survival functions and the clusters are,

$$\begin{gathered} \sum_{l=1}^{L_i} \alpha_i(l) = 1 \text{and} \alpha_i \geq 0 \\ 0 \leq S_i(k) \leq 1 \end{gathered} \tag{5.9}$$

The covariance update equation for each window is obtained as follows 5.10:

$$\sum_{K_L}^{-1} = \underbrace{\sum_{0}^{-1}}_{Covariance\ from\ previous\ window} + \sum_{k=0}^{K_L} \begin{bmatrix} H_S^T \\ H_\alpha^T \end{bmatrix} R_v^{-1} \begin{bmatrix} H_S & H_\alpha \end{bmatrix} \tag{5.10}$$

$$= \sum_0^{-1} + \sum_{k=0}^{K_L} \begin{bmatrix} H_S^T R_v^{-1} H_S & H_S^T R_v^{-1} H_\alpha \\ H_\alpha^T R_v^{-1} H_S & H_\alpha^T R_v^{-1} H_\alpha \end{bmatrix}$$

where, HS is an n x n matrix obtained by taking the partial derivatives of the measurement equation with respect to the survival functions as follows,

$$\frac{\partial y_j}{\partial S_i(0)} = h_{ji} = -\frac{\left(\sum_{l=1}^{L_i} A_i(k,l)\alpha_i(l)\right)(PD_{ij} - PF_{ij})}{\left(\sum_{l=1}^{L_i} S_i(0)A_i(k,l)\alpha_i(l)\right)(PD_{ij} - PF_{ij}) + (1 - PD_{ij})} \tag{5.11}$$

The complete H_S matrix 5.12 has been presented below:

$$H_S = \begin{bmatrix} -\frac{\left(\sum_{l=1}^{L_1} A_1(k,l)\alpha_1(l)\right)(PD_{11} - PF_{11})}{\left(\sum_{l=1}^{L_1} S_1(0)A_1(k,l)\alpha_1(l)\right)(PD_{11} - PF_{11}) + (1 - PD_{11})} & \cdots \\ -\frac{\left(\sum_{l=1}^{L_1} A_1(k,l)\alpha_1(l)\right)(PD_{12} - PF_{12})}{\left(\sum_{l=1}^{L_1} S_1(0)A_1(k,l)\alpha_1(l)\right)(PD_{12} - PF_{12}) + (1 - PD_{12})} & \cdots \\ \vdots & \cdots \\ \vdots & \cdots \\ \vdots & \cdots \end{bmatrix}_{n \times m} \tag{5.12}$$

and H_α is an n xL_i matrix for each component i, obtained by taking the partial derivatives of the measurement equation with respect to the clusters as shown in 5.13 and 5.14.

$$\begin{aligned} \frac{\partial y_j}{\partial \alpha_i(l)} &= h_{ji}(l) \\ &= -\frac{(S_i(0)A_i(k,l))(PD_{ij}-PF_{ij})}{\left(\sum_{p=1}^{L_i} S_i(0)A_i(k,p)\alpha_i(p)\right)(PD_{ij}-PF_{ij})+(1-PD_{ij})} \end{aligned} \tag{5.13}$$

Hence, the H_α matrix takes the form,

$$H_{\alpha_i} = \begin{bmatrix} -\frac{(S_i(0)A_i(k,1))(PD_{i1}-PF_{i1})}{\left(\sum_{p=1}^{L1} S_i(0)A_i(k,p)\alpha_i(p)\right)(PD_{i1}-PF_{i1})+(1-PD_{i1})} & . & . & . & . \\ -\frac{(S_i(0)A_i(k,1))(PD_{i2}-PF_{i2})}{\left(\sum_{p=1}^{L1} S_i(0)A_i(k,p)\alpha_i(p)\right)(PD_{i2}-PF_{i2})+(1-PD_{i2})} & . & . & . & . \\ & . & . & . & . \\ & . & . & . & . \\ & . & . & . & . \end{bmatrix}_{n\times L_i} \tag{5.14}$$

The moving horizon or online estimation algorithms require us to keep track of the covariance and update it at each window to compute the arrival cost which summarizes the past information. The covariances are initialized with zeros in the first window.

5.5 Application to Electronic Throttle Control System

Here we apply the MM-MHE approach to an automotive electronic throttle control (ETC) system. The ETC [123], also termed drive-by-wire technology, replaces the conventional mechanical linkage between accelerator pedal and throttle body. The function of an ETC system is to determine the necessary throttle opening using sensors (such as accelerator pedal position, engine RPM, and vehicle speed), and drive the actuator to obtain the required throttle position via a close loop control algorithm in the electronic control module (ECM). The ECM monitors the health of the subsystems by processing parameter identifier data (PIDs) collected from various sensors and generates diagnostic trouble codes (DTCs or error codes) in case of a failure in any component.

The dataset derived from the ETC simulator consisted of 11 error codes (DTCs), 479 status parameters (PIDs) collected at the time of DTC firing, age of the vehicle and the repair/replacement actions (i.e., repair codes (LCs)) performed on the system. A total of 5 different repair codes (replaceable components) were present in our training data. However, due to ambiguity in two of the repair codes they were grouped into a single repair code [111]. Mutual information gain is employed to select the minimal number of PIDs and the top 16 PIDs are selected for our analysis [15, 38, 111].

The survival functions for components and tests are initially learned using the Cox PHM model as described in section III. Then, k-means clustering technique is employed to group the survival functions for LCs as well as DTCs. For the dataset provided there appear to be 3 clusters of survival functions for each repair code. In

order to validate the prognostic framework, the detection and false alarm probabilities of tests are initially learned from the averaged DTC survival function and averaged LC survival function by minimizing the objective function 5.15 shown below.

$$\min_{\{PD_{ij},PF_{ij}\}_{i=1,j=1}^{m,n}} D = \sum_{k=1}^{K}\sum_{j=1}^{n}\{y_j(k)+ \\ + \sum_{i=1}^{m}\ln[S_i(k)(PD_{ij}-PF_{ij})+(1-PD_{ij})]\}^2 \tag{5.15}$$

The nonlinear least squares minimization is implemented using MATLABs optimization toolbox function fmincon to determine the optimal detection and false alarm probabilities $\{PD_{ij}, PF_{ij}\}$.

The Multiple Model Moving Horizon Estimation Algorithm was implemented in MATLAB using fmincon function which finds the minimum of the constrained nonlinear multivariate objective function based on Sequential Quadratic Programming based Quasi-Newton Line Search. The scale factor is a tuning parameter and was found by experimentation to lie between 10^{-6} and 10^{-9} for the ETC dataset. The remaining useful life of a component at any time can be computed by defining an application-specific threshold on the survival probability.

The measurements for the simulations were generated from clusters randomly for each component. For the Figures 5.4-5.8 shown below the clusters numbers were 3, 2, 2, and 1 respectively for the 4 components. The results for moving horizon estimation are presented next. Figure 5.4 shows the estimated and true survival function probabilities, and the measurements with and without noise; and the predicted measurements for the

first window of observations. The results show good accuracy (MSE and R^2) in the very first window due to the monotonically decreasing trend of the component survival functions. Figures 5.5-5.8 show a gradual improvement in the performance of the MM-MHE algorithm with each window.

The survival probabilities and cluster numbers in the final window (Figure 5.8) show good agreement with the truth indicating the robustness and computational accuracy of the MM-MHE algorithm.

The simulation results presented here were generated for a noise level of approximately 8 dB (i.e. $\delta = 0.16$). However, experiments were conducted with δ varying from 0.01 to 5.12 (an SNR range from 20 dB to 7dB). As shown in Fig. 9, the MM-MHE algorithm performs very well, with R^2 approximately 99% or higher for deltas in the range of 0.1 to 0.32, and MSE in the order of 10-3, for all windows, in the presence of significant measurement noise. The plot in Fig. 9 computes the R2 by averaging over all windows and components.

The simulation results presented in this section are based on experiments for a fixed finite length window and the survival function estimates are determined online by solving a finite horizon state estimation problem as new measurements arrive. However, further experimentation showed that the effects of window size and length are negligible, and we can have either fixed or variable length windows (for e.g., based on days).

An illustrative example was run to demonstrate the robustness of the MM-MHE

algorithm to switching clusters and thus, modeling switching modes (similar to that modeled in Interacting Multiple Models) and varying usage conditions. The MM-MHE algorithm was implemented over 5 windows (similar to the previous example). The observations for the 1^{st} window were generated from clusters $3, 2, 2$, and 1. Subsequently, the cluster mode changes to $1, 1, 1,$ and 3 for the 2^{nd} and 3^{rd} windows, and finally, switches back to $3, 2, 2$, and 1 for the 4^{th} and 5^{th} windows. The survival function probabilities from the final window of the moving horizon estimation have been presented in Figure 5.10. As shown in Figure 5.16, the MM-MHE algorithm tracks the clusters with reasonable accuracy over all windows.

5.6 Conclusions and Future Work

The chapter presents a novel Multiple Model Moving Horizon Estimation (MM-MHE) algorithm for online prediction of the component survival functions based on their usage profiles. The framework employs Cox proportional hazards model based on offline and online data for the remaining useful life prediction. The proposed approach has been validated by way of application to data derived from an automotive electronic throttle control system simulator. The MM-MHE algorithm shows excellent performance (R^2 and MSE) in the presence of significant measurement noise over all windows and converges to the correct cluster number.

The future work includes application of this approach to continuous parameter identifier (PID) data and to account for the uncertainty in RUL estimation. In the

near future, by simple transformations, the authors plan on implementing the MM-MHE algorithm for measurements and states between $(-\infty, \infty)$, and considering the effects of process noise and hence, modifying the cost function accordingly. A potential extension of the Cox-PHM framework for prognosis of coupled systems will be to model the coupled survival dynamics as monotone positive linear systems or monotone Markov processes in which the state matrix is a Metzler matrix (i.e., has nonnegative off-diagonal elements).

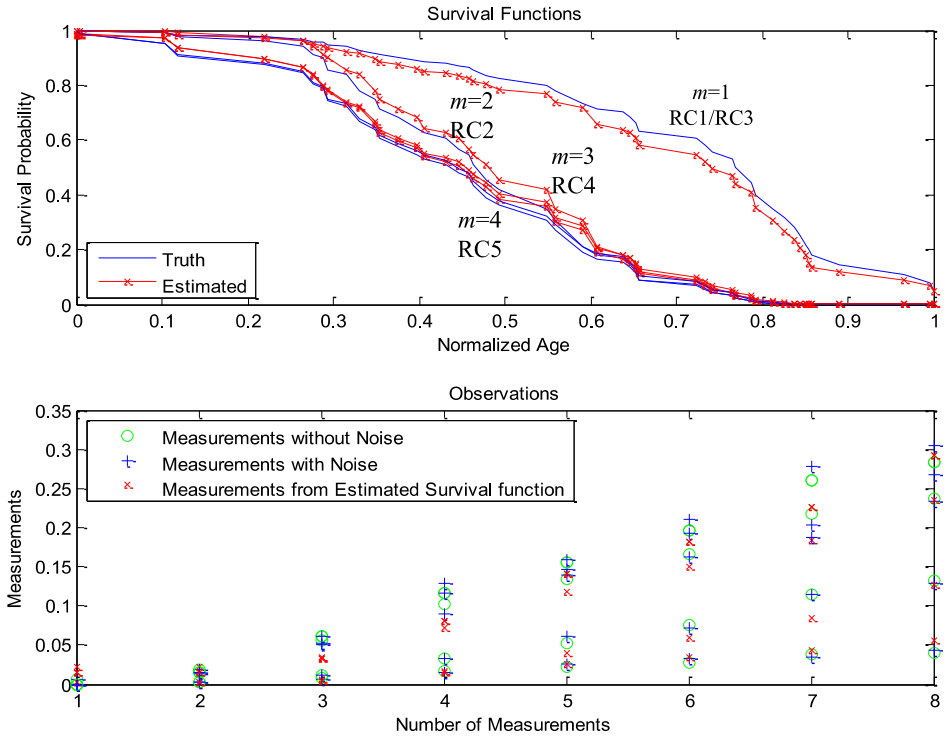


Fig. 5.4: True and Estimated States for Observations 1 to 8.

$$R^2 - statistic(i) = [0.9938; 0.9953; 0.9834; 0.9958]$$

$$MSE(i) = [0.00086; 0.00053; 0.0015; 0.00048]$$

$$\alpha_{il} = \begin{bmatrix} 1\text{e}-03 & 1\text{e}-03 & 0.9980 \\ 1\text{e}-03 & 0.9980 & 1\text{e}-03 \\ 1\text{e}-03 & 0.95722 & 0.04177 \\ 0.9980 & 1\text{e}-03 & 1\text{e}-03 \end{bmatrix}$$

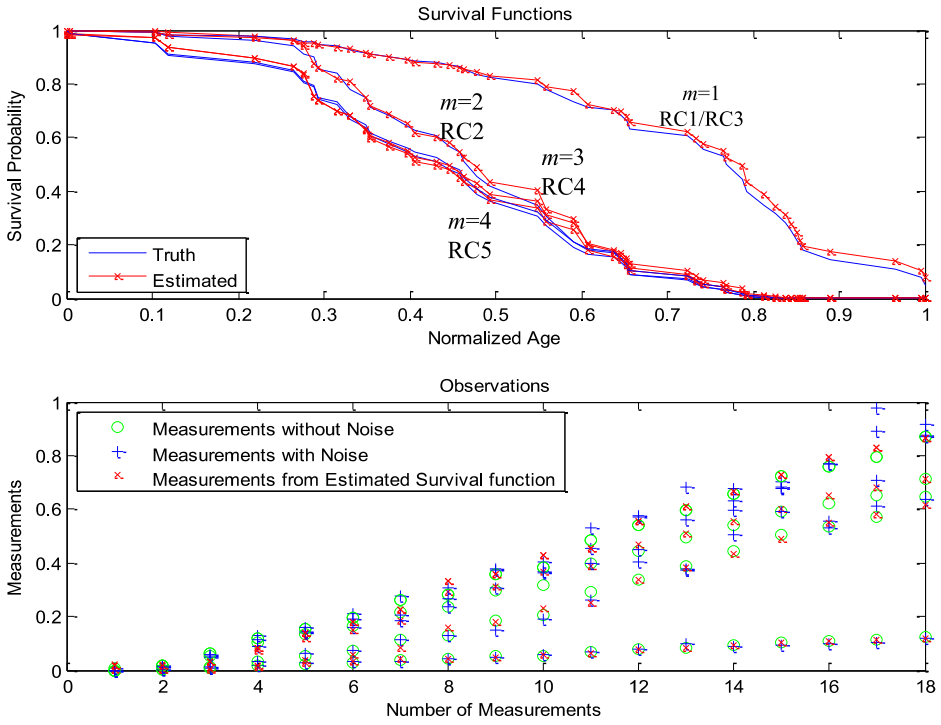


Fig. 5.5: True and Estimated States for Observations 9 to 18.

$$R^2 - statistic(i) = [0.9971; 0.9974; 0.9956; 0.9967]$$

$$MSE(i) = [0.0004; 0.00029; 0.00038; 0.000371]$$

$$\alpha_{il} = \begin{bmatrix} 1\text{e}-03 & 0.001 & 0.9980 \\ 1\text{e}-03 & 0.9980 & 1\text{e}-03 \\ 1\text{e}-03 & 0.9980 & 1\text{e}-03 \\ 0.9620 & 1\text{e}-03 & 0.0369 \end{bmatrix}$$

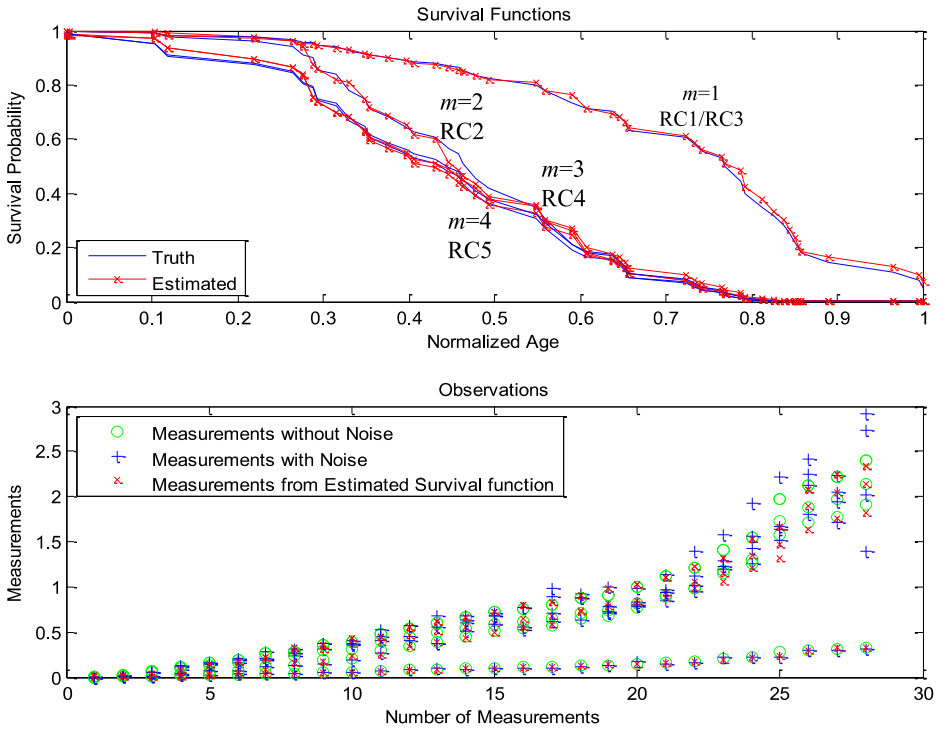


Fig. 5.6: True and Estimated States for Observations 19 to 28.

$$R^2 - statistic(i) = [0.9975; 0.9977; 0.9980; 0.9973]$$

$$MSE(i) = [0.00034; 0.00025; 0.00017; 0.0003]$$

$$\alpha_{il} = \begin{bmatrix} 1\text{e}-03 & 1\text{e}-03 & 0.9980 \\ 1\text{e}-03 & 0.9980 & 1\text{e}-03 \\ 1\text{e}-03 & 0.9927 & 0.00623 \\ 0.9721 & 0.0268 & 1\text{e}-03 \end{bmatrix}$$

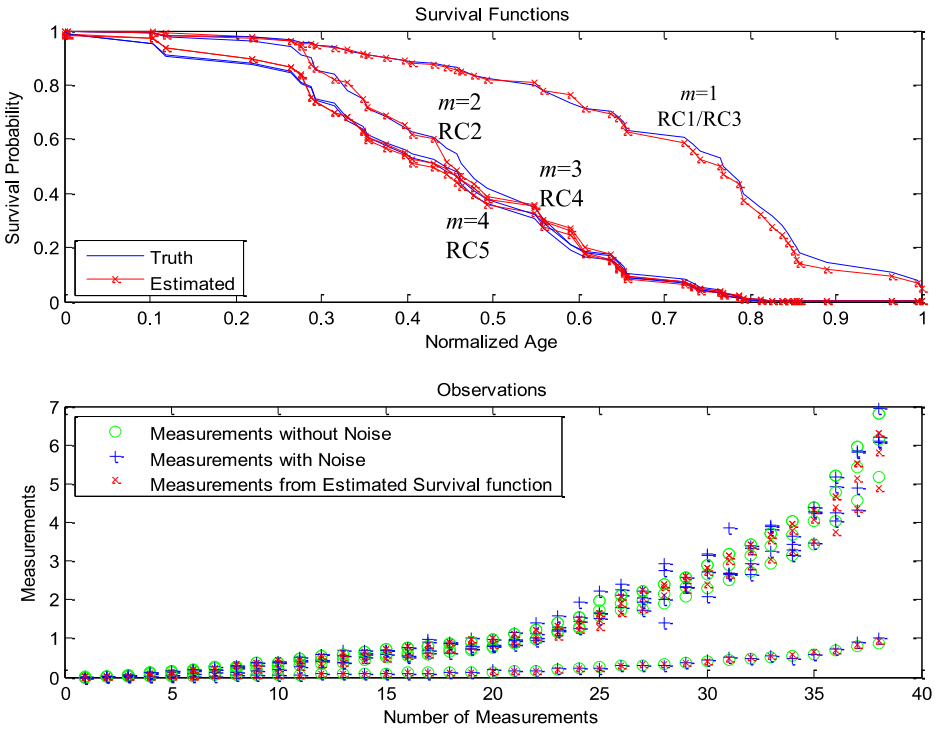


Fig. 5.7: True and Estimated States for Observations 29 to 38.

$$R^2 - statistic(i) = [0.9974; 0.9978; 0.99860; 0.9973]$$

$$MSE(i) = [0.00036; 0.000247; 0.00035; 0.0003]$$

$$\alpha_{il} = \begin{bmatrix} 0.0026 & 0.011 & 0.9963 \\ 1\text{e}-03 & 0.9980 & 1\text{e}-03 \\ 0.009 & 0.9434 & 0.0475 \\ 0.9980 & 1\text{e}-03 & 1\text{e}-03 \end{bmatrix}$$

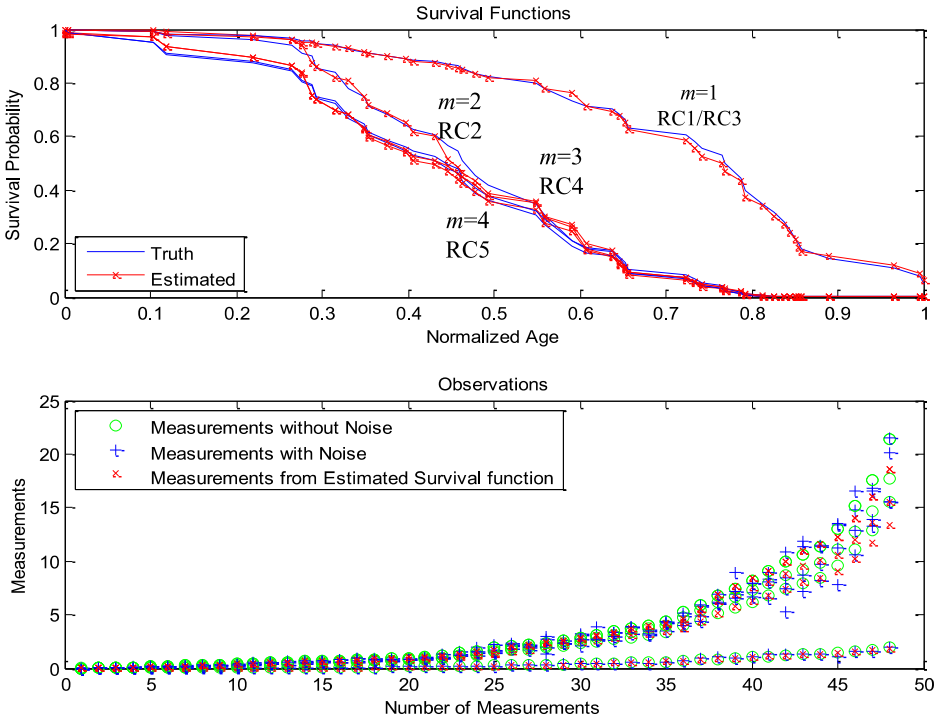


Fig. 5.8: True and Estimated States for Observations 39 to 48.

$$R^2 - statistic(i) = [0.9974; 0.9978; 0.9981; 0.9973]$$

$$MSE(i) = [0.00036; 0.000247; 0.000169; 0.0003]$$

$$\alpha_{il} = \begin{bmatrix} 1\text{e}-03 & 1\text{e}-03 & 0.9980 \\ 1\text{e}-03 & 0.9980 & 1\text{e}-03 \\ 0.007 & 0.99134 & 1\text{e}-03 \\ 0.9980 & 1\text{e}-03 & 1\text{e}-03 \end{bmatrix}$$

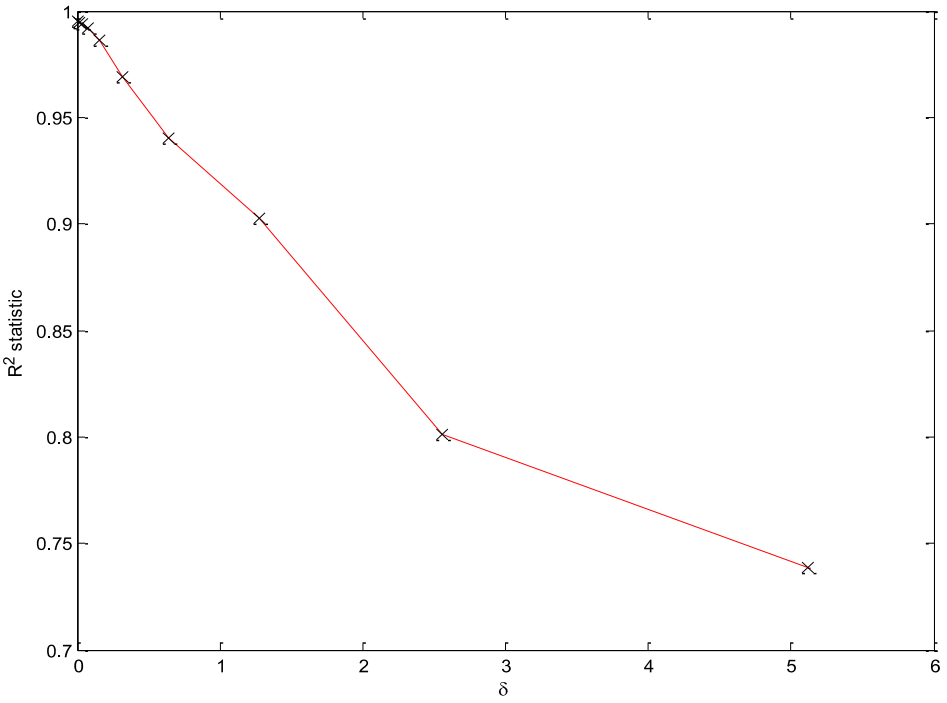


Fig. 5.9: R^2 statistic vs. δ.

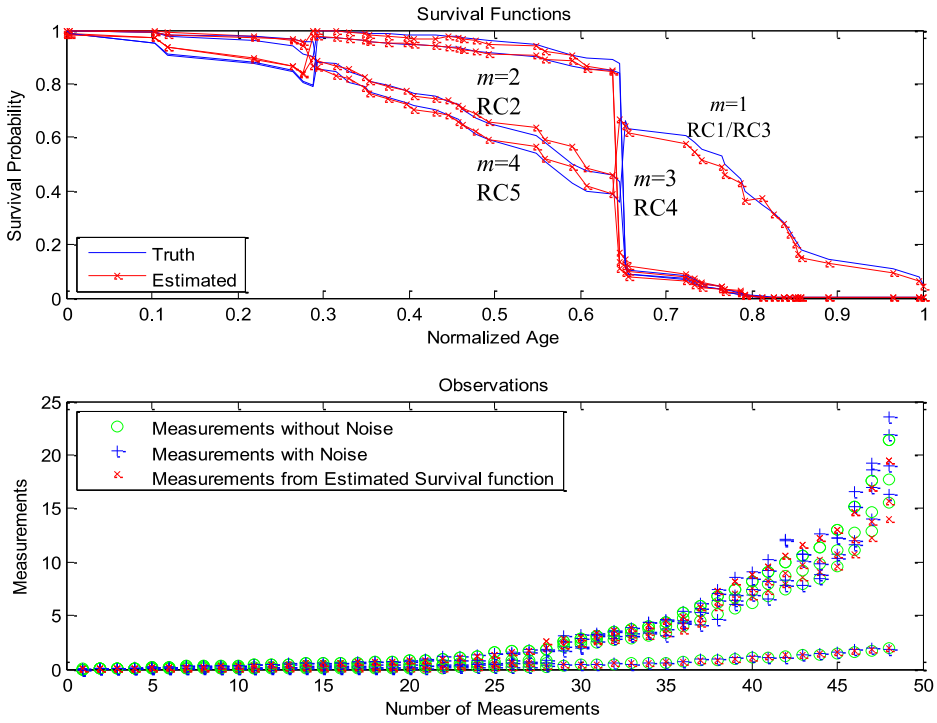


Fig. 5.10: Illustrative Example for Switching Clusters.

$$R^2 - statistic(i) = [0.9499; 0.9818; 0.9661; 0.9427]$$

$$MSE(i) = [0.0108; 0.0026; 0.0025; 0.0111]$$

$$\underbrace{\begin{bmatrix} 0.001 & 0.2462 & 0.7527 \\ 0.001 & 0.998 & 0.001 \\ 0.2209 & 0.778 & 0.001 \\ 0.998 & 0.001 & 0.001 \end{bmatrix}}_{1^{st}\ window\ (1\ to\ 8\ observations)} \rightarrow \underbrace{\begin{bmatrix} 0.998 & 0.001 & 0.001 \\ 0.8103 & 0.1886 & 0.001 \\ 0.001 & 0.7399 & 0.2590 \\ 0.001 & 0.1783 & 0.8206 \end{bmatrix}}_{2nd\ window\ (9\ to\ 18\ observations)} \tag{5.16}$$

$$\rightarrow \underbrace{\begin{bmatrix} 0.998 & 0.001 & 0.001 \\ 0.001 & 0.8529 & 0.146 \\ 0.001 & 0.9802 & 0.0187 \\ 0.0954 & 0.001 & 0.9035 \end{bmatrix}}_{3rd\ window\ (19\ to\ 28\ observations)} \rightarrow \underbrace{\begin{bmatrix} 0.001 & 0.001 & 0.998 \\ 0.001 & 0.998 & 0.001 \\ 0.001 & 0.998 & 0.001 \\ 0.7466 & 0.001 & 0.2523 \end{bmatrix}}_{4th\ window\ (29\ to\ 38\ observations)}$$

$$\rightarrow \underbrace{\begin{bmatrix} 0.3173 & 0.001 & 0.6816 \\ 0.001 & 0.998 & 0.001 \\ 0.001 & 0.998 & 0.001 \\ 0.8949 & 0.001 & 0.10407 \end{bmatrix}}_{5th\ window\ (39\ to\ 48\ observations)}$$

Fig. 5.11: Cluster Tracking via MM-MHE Algorithm.

Chapter 6

Research Impact

The algorithms in this thesis will:

1. Develop next-generation BMSs featuring online tracking and monitoring of pivotal battery characteristics to facilitate efficient diagnostic and prognostic maintenance of batteries

2. Revolutionize battery fuel gauging for robust and accurate estimation of battery SOC, SOH, RUL, TTS, capacity etc.

3. Play a key role in condition-based maintenance (CBM) of contemporary systems

4. Minimize life cycle cost of vehicle systems

5. Minimize system downtime and warranty costs

6. Enhance safety and reliability of vehicular systems

7. Improve personnel and parts management

8. Prevent customer walk-home scenarios

9. Improve customer satisfaction through enhanced vehicle availability

10. Improve component design

Bibliography

[1] A. Abdollahi, X. Han, G. V. Avvari, N. Raghunathan, B. Balasingam, K. R. Pattipati, and Y. Bar-Shalom, "Optimal battery charging, part i: Minimizing time-to-charge, energy loss, and temperature rise for ocv-resistance battery model," *Journal of Power Sources*, May 2014, (under review).

[2] A. Abdollahi, X. Han, N. Raghunathan, B. Balasingam, K. R. Pattipati, and Y. Bar-Shalom, "Optimal battery charging, part ii: Minimizing time-to-charge, energy loss, and temperature rise for ocv-resistance-rc battery model," *Journal of Power Sources*, May 2014, (under review).

[3] A. Abdollahi, N. Raghunathan, B. Pattipati, X. Han, B. Balasingam, K. R. Pattipati, and Y. Bar-Shalom, "Optimal battery charging, part iii: Capacity modeling and control for battery life maximization," *Journal of Power Sources*, May 2014, (under review).

[4] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions*, N. York, Ed. Dover Publications, Inc., 1964.

[5] S. V. Amari, P. Greensburg, L. McLaughlin, and H. Pham, "Cost-effective condition-based maintenance using markov decision processes," *Annual Reliability and Maintainability Symposium*, pp. 464–469, 2006.

[6] G. V. Avvari, B. Pattipati, B. Balasingam, K. Pattipati, and Y. Bar-Shalom, "Battery fuel gauge hardware-in-the-loop validation on li-ion batteries," *submitted to, IEEE Transactions on Instrumentation and Measurement*, June 2014.

[7] B. Balasingam, G. V. Avvari, B. Pattipati, K. R. Pattipati, and Y. Bar-Shalom, "A robust approach to battery fuel gauging, part I: Real time parameter estimation," *accepted, Journal of Power Sources*, June 2014.

[8] ——, "A robust approach to battery fuel gauging, part II: Real time capacity estimation," *accepted, Journal of Power Sources*, June 2014.

[9] ——, "A robust approach to battery fuel gauging, part III: State of charge tracking," *under review, Journal of Power Sources*, January 2014.

[10] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons, 2004.

[11] J. Barker, R. Pynenburg, R. Koksbang, and M. Y. Saidi, "An electrochemical investigation into the lithium insertion properties of lixcoo2," *Electrochimica Acta*, vol. 41, no. 15, p. 24812488, 1996.

[12] L. Baum, T. Petrie, G. Soules, and N. Weiss, "A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains," *Ann. of Math. Stat.*, vol. 41, pp. 164–171, 1970.

[13] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, 3rd ed., U. Nashua, NH, Ed. Athena Scientific, 2005 and 2012, vol. 1 and 2.

[14] B. Bhangu, P. Bentley, D. Stone, and C. Bingham, "Nonlinear observers for predicting state-of-charge and state-of-health of lead-acid batteries for hybrid-electric vehicles," *IEEE Transactions on Vehicular Technology*, vol. 54, pp. 783–794, 2005.

[15] C. M. Bishop, *Neural Networks for Pattern Classification*, Oxford, Ed. Clarendon Press, 1997.

[16] ——, *Pattern Recognition and Machine Learning*, Springer, Ed., 2006.

[17] C. Bishop, *Pattern Recognition and Machine Learning*, U. New York, Ed. Springer, 2006.

[18] E. Bordeaux, "Portable power management - a holistic perspective," *Intersil Corporation*, 2002.

[19] R. Bro, "Multiway calibration. multilinear pls." *Journal of Chemometrics*, vol. 10, no. 1, pp. 47–61, 1996.

[20] M. Broussely, P. Biensan, F. Bonhomme, P. Blanchard, S. Herreyre, K. Nechev, and R. Staniewicz, "Main aging mechanisms in li ion batteries," *Journal of Power Sources*, vol. 146, no. 1, pp. 90–96, 2005.

[21] M. Broussely, S. Herreyre, P. Biensan, P. Kasztejna, K. Nechev, and R. Staniewicz, "Aging mechanism in li ion cells and calendar life predictions," *Journal of Power Sources*, vol. 97, pp. 13–21, 2001.

[22] I. Buchmann, "Artificial intelligence reads battery state-of-health in three minutes," *presented at the Sixteenth Annual Battery Conference*, 2001.

[23] K. P. Burnham and D. R. Anderson, *Model Selection and Inference*, N. York, Ed. Springer-Verlag, 1998.

[24] F. Camci and R. Chinnam, "Health-state estimation and prognostics in machining processes," *IEEE transactions on Automation Science and Engineering*, vol. 7, no. 3, pp. 581–597, July 2010.

[25] I. Chads, G. Chapron, M.-J. Cros, F. Garcia, and R. Sabbadin, "Markov decision processes (mdp) toolbox," Online, June 2013.

[26] M. Charkhgard and M. Farrokhi, "State-of-charge estimation for lithium-ion batteries using neural networks and ekf," *Industrial Electronics, IEEE Transactions on*, vol. 57, no. 12, pp. 4178–4187, 2010.

[27] N. A. Chaturvedi, R. Klein, J. Christensen, J. Ahmed, and A. Kojic, "Algorithms for advanced battery-management systems," *IEEE Control Systems Magazine*, vol. 30, no. 3, pp. 49–68, June 2010.

[28] M. Chen and G. Rincón-Mora, "Accurate electrical battery model capable of predicting runtime and i-v performance," *IEEE Transactions on Energy Conversion*, vol. 21, pp. 504–511, 2006.

[29] L. H. Chiang, E. Russel, and R. Braatz, *Fault Detection and Diagnosis in Industrial Systems*, U. London, Ed. Springer-Verlag, 2001.

[30] Y.-H. Chiang, W.-Y. Sean, and J.-C. Ke, "Online estimation of internal resistance and open-circuit voltage of lithium-ion batteries in electric vehicles," *Journal of Power Sources*, vol. 196, no. 8, pp. 3921–3932, 2011.

[31] T. Coleman and Y. Li, "On the convergence of reflective newton methods for large-scale nonlinear minimization subject to bounds," *Mathematical Programming*, vol. 67, pp. 189–224, 1994.

[32] ——, "An interior, trust region approach for nonlinear minimization subject to bounds," *SIAM Journal on Optimization*, vol. 6, pp. 418–445, 1996.

[33] D. R. Cox and D. Oakes, *Analysis of survival data*. Chapman and Hall, 1984.

[34] J.-C. de Borda, "Memoire sur les elections au scrutin," *Histoire del'Academie Royale des Sciences*, 1781.

[35] D. Dees, V. Battaglia, and A. Belanger, "Electrochemical modeling of lithium polymer batteries," *Journal of Power Sources*, vol. 110, pp. 310–320, 2002.

[36] D. Do, C. Forgez, K. Benkara, and G. Friedrich, "Impedance observer for a li-ion battery using kalman filter," *IEEE Transactions on Vehicular Technology*, vol. 58, pp. 3930–3937, 2009.

[37] D. Doerffel and S. Sharkh, "A critical review of using the peukert equation for determining the remaining capacity of lead-acid and lithium-ion batteries," *Journal of Power Sources*, vol. 115, pp. 395–400, 2006.

[38] R. O. Duda, P. E. Hart, and D. Stork, *Pattern Classification*, N. York, Ed. John Wiley and Sons, 2000.

[39] R. Duda, P. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed., U. New York, Ed. John Wiley and Sons.

[40] C. B. E. F. Camacho, *Model predictive control*, London, Ed. Springer-Verlag, 2004.

[41] G. R. E. Meissner, "The challenge to the automotive battery industry: the battery has become an increasingly integrated component within the vehicle electric power system," *Journal of Power Sources*, vol. 44, pp. 438–460, 2005.

[42] G. Ferrari-Trecate, D. Mignone, and M. Morari, "Moving horizon estimation for hybrid systems," *IEEE transactions on Automatic Control*, vol. 47, no. 10, 2002.

[43] FreedomCAR, "Freedomcar battery test manual for power-assist hybrid electric vehicles," DOE/ID-11069, Tech. Rep., October 2003.

[44] C. E. Garcia, D. M. Prett, and M. Morari, "Model predictive control: theory and practice a survey," *Automatica*, vol. 25, pp. 335–348, 1989.

[45] M. Ge, R. Du, G. Zhang, and Y. Xu, "Fault diagnosis using support vector machine with an application in sheet metal stamping operations," *Mechanical Systems and Signal Processing*, vol. 18, pp. 143–159, 2004.

[46] O. Gérard, J.-N. Patillon, and F. d'Alché Buc, "Neural network adaptive modeling of battery discharge behavior," in *Artificial Neural NetworksICANN'97*. Springer, 1997, pp. 1095–1100.

[47] K. Goebel, B. Saha, A. Saxena, J. Celaya, and J. Christophersen, "Prognostics in battery health management,," *IEEE Instrumentation and Measurement Magazine*, vol. 11, p. 3340, 2008.

[48] G. H. Golub, P. C. Hansen, and D. P. O'Leary, "Tikhonov regularization and total least squares," *SIAM Journal of Matrix Analysis and Applications*, vol. 21, pp. 185–194, 2000.

[49] K. Goser, K. Schuhmacher, M. Hartung, K. Heesche, B. Hesse, and A. Kanstein, "Neuro-fuzzy systems for engineering applications," *IEEE AFRICON 1996*, vol. 2, pp. 759–764, 1996.

[50] H. He, X. Zhang, R. Xiong, Y. Xu, and H. Guo, "Online model-based estimation of state-of-charge and open-circuit voltage of lithium-ion batteries in electric vehicles," *Energy*, vol. 39, no. 1, pp. 310–318, 2012.

[51] Y. Hu, S. Yurkovich, Y. Guezennec, and B. J. Yurkovich, "Electro-thermal battery model identification for automotive applications," *Journal of Power Sources*, vol. 196, pp. 449–457, 2011.

[52] S. V. Huffel, "Total least squares and errors-in-variables modeling: bridging the gap between statistics, computational mathematics and engineering," *COMPSTAT Proceedings in Computational Statistics, Heidelberg: Physika-Verlag*, pp. 539–555, 2004.

[53] T. Inc. Battery modeling for hev simulation. [Online]. Available: http://www.thermoanalytics.com/docs/batteries.html

[54] INL, "Advanced technology development program for lithium-ion batteries: Gen 2 performance evaluation," INL, final report, July 2006.

[55] H. H. J. Newman, K.E. Thomas and D. Wheeler, "Modeling of lithium-ion batteries," *Journal of Power Sources*, pp. 838–843, 2003.

[56] J. E. Jackson, *A Users Guide to Principal Components*, U. New York, Ed. John Wiley and Sons, 1991.

[57] A. K. S. Jardine and A. H. C. Tsang, *Maintenance, Replacement, and Reliability: Theory and Applications*, N. York, Ed. CRC Press, 2005.

[58] K. S. Khattri, "New close form approximations of $\ln(1+x)$," *The Teaching of Mathematics*, vol. 12, no. 1, pp. 7–14, 2009.

[59] I.-S. Kim, "Nonlinear state of charge estimator for hybrid electric vehicle battery," *IEEE Transactions on Power Electronics*, vol. 23, pp. 2027–2034, 2008.

[60] ——, "A technique for estimating the state of health of lithium batteries through a dual-sliding-mode observer," *IEEE Transactions on Power Electronics*, vol. 25, pp. 1013–1022, 2010.

[61] J. D. Klabfleisch and R. L. Prentice, *The statistical analysis of failure time data*, N. York, Ed. Wiley, 2002.

[62] J. P. Klein and M. L. Moeschberger, *Survival analysis techniques for censored and truncated data*, N. York, Ed. Springer-Verlag, 2003.

[63] E. Kuhn, C. Forgez, and G. Friedrich, "Modeling diffusive phenomena using non-integer derivatives: application nimh batteries," *The European Physical Journal Applied Physics*, vol. 25, pp. 183–190, 2004.

[64] D. Landolt, "Corrosion et chimie de surfaces des mtaux," *Trait des matriaux, Presses Polytechniques et Universitaires Romandes*, 1993.

[65] Y.-S. Lee, W.-Y. Wang, and T.-Y. Kuo, "Soft computing for battery state-of-charge (bsoc) estimation in battery string systems," *Industrial Electronics, IEEE Transactions on*, vol. 55, no. 1, pp. 229–239, 2008.

[66] X. Lin, J. Park, L. Liu, Y. Lee, A. Sastry, and W. Lu, "A comprehensive capacity fade model and analysis for li-ion batteries," *Journal of The Electrochemical Society*, vol. 160, no. 10, pp. A1701–A1710, 2013.

[67] D. Linden and T. Reddy, *Handbook of Batteries*, 3rd ed., N. York, Ed. McGraw Hill, 2002.

[68] Z. Liu, Q. Li, X. Liu, and C. Mu, "A hybrid lssvr/hmm-based prognostic approach," *Journal of Sensors*, vol. 13, pp. 5542–5560, 2013.

[69] J. Luo, M. Namburu, K. Pattipati, L. Qiao, M. Kawamoto, and S. Chigusa, "Model-based prognostic techniques," *IEEE Autotestcon Conference*, pp. 330–340, 2003.

[70] J. Luo, H. Tu, K. R. Pattipati, L. Qiao, and S. Chigusa, "Graphical models for diagnosis knowledge representation and inference," *IEEE Autotestcon*, pp. 483–489, 2005.

[71] ——, "Diagnosis knowledge representation and inference," *IEEE Instrumentation and Measurement Magazine*, vol. 9, no. 4, pp. 45–52, 2006.

[72] I. Markovsky and S. V. Huffel, "Overview of total least-squares methods," *International Journal of Signal Processing*, vol. 87, pp. 2283–2302, 2007.

[73] I. Markovsky, D. M. Sima, and S. V. Huffel, "Total least squares methods," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 2, no. 2, pp. 212–217, March/April 2010.

[74] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, "Constrained model predictive control: stability and optimality," *Automatica*, pp. 789–814, 2000.

[75] H. Michalska and D. Q. Mayne, "Moving horizon observers," *presented at the Nonlinear Control Systems Design Symposium (NOLCOS), M. Fliess, Ed., Bordeaux, France,*, 1992.

[76] A. Millner, "Modeling lithium ion battery degradation in electric vehicles," in *Innovative Technologies for an Efficient and Reliable Electricity Supply (CITRES), 2010 IEEE Conference on*. IEEE, 2010, pp. 349–356.

[77] S. Moore and M. Ehsani, "An empirically based electrosource horizon lead-acid battery model," *SAE Technical Paper 960448*, vol. 105, no. 6, pp. 421–424, February 1996.

[78] K. Murphy, "Hidden markov model (hmm) matlab toolbox," Online.

[79] D. N. P. Murthy, M. Xie, and R. Jiang, *Weibull Models*, N. York, Ed. Wiley, 2004.

[80] S. M. Namburu, M. S. Azam, J. Luo, K. Choi, and K. R. Pattipati, "Data-driven modeling, fault diagnosis, and optimal sensor selection for hvac chillers," *IEEE transactions on automation science and engineering*, vol. 4, no. 3, 2007.

[81] D. E. Neumann and S. Lichte, "A multi-dimensional battery discharge model with thermal feedback applied to a lithium-ion battery pack," *NDIA Ground Vehicle Systems Engineering and Technology Symposium-Modeling & Simulation, Testing and Validation (MSTV) Mini-Symposium*, 2011.

[82] G. Ning, B. Haran, and B. N. Popov, "Capacity fade study of lithium-ion batteries cycled at high discharge rates," *Journal of Power Sources*, vol. 117, no. 1, pp. 160–169, 2003.

[83] G. Ning, R. E. White, and B. N. Popov, "A generalized cycle life model of rechargeable li-ion batteries," *Electrochimica Acta*, vol. 51, no. 10, pp. 2012–2022, 2006.

[84] B. Pattipati, B. Balasingam, G. V. Avvari, K. Pattipati, and Y. Bar-Shalom, "Open circuit voltage characterization of lithium-ion batteries," *accepted, Journal of Power Sources*, June 2014.

[85] B. Pattipati, K. Pattipati, Y. A. Ghoneim, M. Howell, and M. Salman, "Electronic returnless fuel system fault diagnosis and isolation: A data-driven approach," *Annual Conference of Prognostics and Health Management Society*, 2013.

[86] B. Pattipati, C. Sankavaram, and K. R. Pattipati, "System identification and estimation framework for pivotal automotive battery management system characteristics," *IEEE transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 41, no. 6, pp. 869–884, November 2011.

[87] S. Piller, M. Perrin, and A. Jossen, "Methods for state-of-charge determination and their applications," *Journal of Power Sources*, vol. 96, pp. 113–120, 2001.

[88] 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.

[89] ——, "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.

[90] PNGV, "Pngv battery test manual," DOE/ID-10597, Rev. 3, February 2001.

[91] V. Pop, H. Bergveld, P. Notten, J. O. het veld, and P. Regtien, "Accuracy analysis of the state-of-charge and remaining run-time determination for lithium-ion batteries," *Measurement Journal*, vol. 42, pp. 1131–1138, 2009.

[92] V. Pop, H. Bergveld, P. Notten, and P. Regtien, "State-of-the-art of battery state-of-charge determination," *Measurement Science and Technology journal*, vol. 16, pp. R93–R110, 2005.

[93] W. H. Press, B. P. Flannery, S. A. Teutolsky, and W. T. Vetterling, *Numerical Recipes: The Art of Scientific Computing*, Cambridge, Ed. Cambridge University Press, 1990.

[94] M. L. Puterman, *Markov Decision Processes*, U. New York, Ed. John Wiley and Sons, 1994.

[95] S. J. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology," *Control Engineering Practice*, pp. 733–764, 2003.

[96] L. Rabiner and B. Juang, "An introduction to hidden markov models," *IEEE ASSP Magazine*, pp. 4–16, 1986.

[97] L. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *proceedings of IEEE*, vol. 77, 1989.

[98] S. K. Rahimian, S. Rayman, and R. E. White, "Comparison of single particle and equivalen circuit analong models for a lithium-ion cell," *Journal of Power Sources*, vol. 196, no. 20, pp. 8450–8462, October 2011.

[99] P. Ramadass, B. Haran, P. M. Gomadam, R. White, and B. N. Popov, "Development of first principles capacity fade model for li-ion cells," *Journal of the Electrochemical Society*, vol. 151, no. 2, pp. A196–A203, 2004.

[100] P. Ramadass, B. Haran, R. White, and B. N. Popov, "Mathematical modeling of the capacity fade of li-ion cells," *Journal of Power Sources*, vol. 123, no. 2, pp. 230–240, 2003.

[101] C. V. Rao, J. B. Rawlings, and D. Q. Mayne, "Constrained state estimation for nonlinear discrete-time systems: stability and moving horizon approximations," *IEEE transactions on Automatic Control*, vol. 48, no. 2, 2003.

[102] J. B. Rawlings, "Tutorial overview of model predictive control," *IEEE Control Systems Magazine*, vol. 20, 2000.

[103] J. W. Rawlings and B. R. Bakshi, "Particle filtering and moving horizon estimation," *Computers and Chemical Engineering*, pp. 1529–1541, 2006.

[104] P. Rong and M. Pedram, "An analytical model for predicting the remaining battery capacity of lithium-ion batteries," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, vol. 14, no. 5, pp. 441–451, 2006.

[105] M. A. Roscher, O. Bohlen, and J. Vetter, "Ocv hysteresis in li-ion batteries including two-phase transition materials," *International Journal of Electrochemistry*, pp. 1369–1384, 2011.

[106] M. A. Roscher and D. U. Sauer, "Dynamic electric behavior and open-circuit-voltage modeling of $lifepo_4$-based lithium ion secondary batteries," *Journal of Power Sources*, vol. 196, pp. 331–336, 2011.

[107] M. Roscher, O. Bohlen, and J. Vetter, "Ocv hysteresis in li-ion batteries including two-phase transition materials," *International Journal of Electrochemistry*, 2011, article ID 984320.

[108] D. G. S. Samar and S. Boyd, "Embedded estimation of fault parameters in an unmanned aerial vehicle," *Proceedings of the IEEE International Conference on Control Applications, Munich, Germany*, 2006.

[109] ——, "Model predictive estimation of evolving faults," *Proceedings of the American Control Conference, Minneapolis, Minnesota*, 2006.

[110] B. Saha, K. Goebel, S. Poll, and J. Christopherson, "Prognostics methods for battery health monitoring using a bayesian framework," *IEEE Transactions on Instrumentation and Measurement*, vol. 58, pp. 291–296, 2009.

[111] C. Sankavaram, A. Kodali, K. Pattipati, B. Wang, M. S. Azam, and S. Singh, "A prognostic framework for health management of coupled systems," *IEEE Conference on Prognostics and Health Management*, 2011.

[112] S. Santhanagopalan, Q. Guo, P. Ramadass, and R. E. White, "Review of models for predicting the cycling performance of lithium ion batteries," *Journal of Power Sources*, vol. 156, no. 2, pp. 620–628, 2006.

[113] S. Santhanagopalan and R. E. White, "Online estimation of the state of charge of a lithium ion cell," *Journal of Power Sources*, vol. 161, no. 2, pp. 1346–1355, 2006.

[114] G. Sarre, P. Blanchard, and M. Broussely, "Aging of lithium-ion batteries," *Journal of Power Sources*, vol. 127, no. 1, pp. 65–71, 2004.

[115] T. Sasaki, Y. Ukyo, and P. Novk, "Memory effect in a lithium-ion battery," *Nature Materials*, vol. 12, pp. 569 – 575, April 2013.

[116] M. A. Schwabacher, "A survey of data-driven prognostics," *American Institute of Aeronautics and Astronautics*, 2005.

[117] W. Shen, C. Chan, E. Lo, and K. Chau, "Estimation of battery available capacity under variable discharge currents," *Journal of Power Sources*, vol. 103, pp. 180–187, 2002.

[118] A. K. Sleigh, J. J. Murray, and W. R. McKinnon, "Memory effects due to phase conversion and hysteresis in li/lixmno2 cells," *Electrochimica Acta*, vol. 36, no. 9, p. 14691474, 1991.

[119] A. J. Smola, P. L. Bartlett, B. Scholkopf, and D. Schuurmans, *Advances in Large Margin Classifiers*, M. Cambridge, Ed. The MIT Press, 2000.

[120] J. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural Processing Letters*, vol. 9, pp. 293–300, 1999.

[121] A. Szumanowski and Y. Chang, "Battery management system based on battery nonlinear dynamics modeling," *IEEE Transactions on Vehicular Technology*, vol. 57, pp. 1425–1432, 2008.

[122] K. P. Ta and J. Newman, "Proton intercalation hysteresis in charging and discharging nickel hydroxide electrodes," *Journal of the Electrochemical Society*, vol. 146, no. 8, pp. 2769–2779, 1999.

[123] P. Technology. (2010, accessed on November 14th) Electronic throttle control (drive by wire or fly by wire). [Online]. Available: http://www.picoauto.com/applications/electronic-throttle-control.html

[124] I. ThermoAnalytics. (2012, September) Battery modeling @ONLINE. [Online]. Available: http://www.thermoanalytics.com/docs/batteries.html

[125] D. A. Tobon-Mejia, K. Medjaher, N. Zerhouni, and G. Tripot, "A data-driven failure prognostics method based on mixture of gaussians hidden markov models," *IEEE transactions on reliability*, vol. 61, no. 2, pp. 491–503, June 2012.

[126] L. E. Unnewehr and S. A. Nasar, *Electric Vehicle Technology*, N. York, Ed. John Wiley & Sons, 1982.

[127] V. Vapnik, *The Nature of Statistical Learning Theory*, U. New York, Ed. Springer, 1995.

[128] M. Verbrugge, D. Frisch, and B. Koch, "Adaptive energy management of electric and hybrid electric vehicles," *Journal of the Electrochemical Society*, vol. 152, pp. A333–A342, 2005.

[129] M. Verbrugge and E. Tate, "Adaptive state of charge algorithm for nickel metal hydride batteries including hysteresis phenomena," *Journal of Power Sources*, vol. 126, pp. 236–249, 2004.

[130] ——, "Adaptive state of charge algorithm for nickel metal hydride batteries including hysteresis phenomena," *Journal of Power Sources*, vol. 126, pp. 236–249, 2004.

[131] J. Vetter, P. Novak, M. Wagner, C. Veit, K.-C. Möller, J. Besenhard, M. Winter, M. Wohlfahrt-Mehrens, C. Vogler, and A. Hammouche, "Ageing mechanisms in lithium-ion batteries," *Journal of power sources*, vol. 147, no. 1, pp. 269–281, 2005.

[132] P. J. Vlok, J. L. Coetzee, D. Banjevic, A. K. S. Jardine, and V. Makis, "Optimal component replacement decisions using vibration monitoring and the phm," *Journal of the Operational Research Society*, vol. 53, pp. 193–202, 2002.

[133] M. O. M. P. W. He, N. Williard, "Prognostics of lithium-ion batteries based on dempstershafer theory and the bayesian monte carlo method," *Journal of Power Sources*, vol. 196, pp. 10 314–10 321, 2011.

[134] Wikipedia. (2013, July) Nameplate capacity @ONLINE. [Online]. Available: http://en.wikipedia.org/wiki/Nameplate_capacity

[135] R. Wright, C. Motloch, J. Belt, J. Christophersen, C. Ho, R. Richardson, I. Bloom, S. Jones, V. Battaglia, G. Henriksen, T. Unkelhaeuser, D. Ingersoll, H. Case, S. Rogers, and R. Sutula, "Calendar-and cycle-life studies of advanced technology development program generation 1 lithium-ion batteries," *Journal of Power Sources*, vol. 110, p. 445470, 2002.

[136] R. Xiong, F. Sun, X. Gong, and C. Gao, "A data-driven based adaptive state of charge estimator of lithium-ion polymer battery used in electric vehicles," *Applied Energy*, vol. 113, pp. 1421–1433, 2014.