

A Study of Computationally Efficient Advanced Battery Management: Modeling, Identification, Estimation and Control

©2020

Ning Tian

Submitted to the graduate degree program in Mechanical Engineering and the Graduate Faculty of the University of Kansas in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

Committee members

Associate Professor Huazhen Fang, Chair

Associate Professor Carl W. Luchies

Associate Professor Sara E. Wilson

Associate Professor Jian Li

Professor Trung Van Nguyen

Date defended: December 1, 2020

The Dissertation Committee for Ning Tian certifies
that this is the approved version of the following dissertation :

A Study of Computationally Efficient Advanced Battery Management: Modeling, Identification,
Estimation and Control

Associate Professor Huazhen Fang, Chair

Date approved: December 1, 2020

Ever Tried. Ever Failed. No Matter. Try Again. Fail Again. Fail Better.

—*Samuel Beckett*

Abstract

Lithium-ion batteries (LiBs) are a revolutionary technology for energy storage. They have become a dominant power source for consumer electronics and are rapidly penetrating into the sectors of electrified transportation and renewable energies, due to the high energy/power density, long cycle life and low memory effect. With continuously falling prices, they will become more popular in foreseeable future. LiBs demonstrate complex dynamic behaviors and are vulnerable to a number of operating problems including overcharging, overdischarging and thermal runaway. Hence, battery management systems (BMSs) are needed in practice to extract full potential from them and ensure their operational safety. Recent years have witnessed a growing amount of research on BMSs, which usually involves topics such as dynamic modeling, parameter identification, state estimation, cell balancing, optimal charging, thermal management, and fault detection. A common challenge for them is computational efficiency since BMSs typically run on embedded systems with limited computing and memory capabilities. Inspired by the challenge, this dissertation aims to address a series of problems towards advancing BMSs with low computational complexity but still high performance. Specifically, the efforts will focus on novel battery modeling and parameter identification (Chapters 2 and 3), highly efficient optimal charging control (Chapter 4) and spatio-temporal temperature estimation of LiB packs (Chapter 5). The developed new LiB models and algorithms can hopefully find use in future LiB systems to improve their performance, while offering insights into some key challenges in the field of BMSs. The research will also entail the development of some fundamental technical approaches concerning parameter identification, model predictive control and state estimation, which have a prospect of being applied to dynamic systems in various other problem domains.

Acknowledgements

This is the best five years of my life. Five years ago I graduated with a master's degree in thermal engineering. Being attracted by the magic of control technology and the growing importance of lithium-ion batteries, I decided to shift from my seven-year thermal engineering study to the field of dynamic systems and control and have since focused on the research of battery management. Looking back upon the past five years, the journey is filled with numerous tough times, but well paid off both physically and mentally at the end of the day.

I would first give my sincere thanks to my supervisor, Dr. Huazhen Fang. As his first PhD student, his devotion and expectation that I enjoy is incomparable. He totally redefined the way of being rigorous and perseverance in my mind and encouraged me to do my best always, which I believe is the key drive for me when working my way through the long march. I also appreciate his continued emphasis on “having a big vision”, making me aspire to the future and gear up to take on challenges.

I would also thank the members of my dissertation committee—Dr. Trung Van Nguyen, Dr. Sara Wilson, Dr. Jian Li and Dr. Carl Luchies—for their time, patience and instructions. I am also thankful for my research collaborators: Mr. Hamidreza Movahedi and Dr. Rajesh Rajamani from the University of Minnesota, Dr. Yebin Wang from Mitsubishi Electric Research Laboratories, and Dr. Jian Chen from Zhejiang University. There are many other people to whom I would like to express my thanks, and they include my lab mates, my colleagues at the ASME DSCD Energy Systems Technical Committee, ANSYS and Romeo Power.

I would also shout out my thanks to my wife, Meihua Zhang. It is my greatest luck to have her company since undergraduate days and a tremendous happiness to have our baby, Hengyi Tian, at this moment. Finally, I would give my most heartfelt thanks to my parents, without whom the dissertation would not have been possible.

Contents

1	Introduction	1
1.1	Background	1
1.2	Lithium-ion Battery	2
1.3	Battery Management System	5
1.4	Statement of Contributions	10
2	Development of the Nonlinear Double-Capacitor Model	11
2.1	Introduction	11
2.2	Model Development	17
2.3	Parameter Identification	21
2.3.1	Approach 1.0: Constant-Current Charging/Discharging	21
2.3.2	Approach 2.0: Variable-Current Charging/Discharging	25
2.4	Experimental Validation	34
2.4.1	Validation Based on Approach 1.0	34
2.4.2	Validation Based on Approach 2.0	38
2.5	Summary	42
3	One-Shot Parameter Identification of the Thevenin Model	44
3.1	Introduction	44
3.2	Model Description	48
3.3	Parameter Identification	51
3.3.1	Identifiability Analysis	51
3.3.2	Identification Methods	54

3.4	Numerical Simulation	59
3.4.1	Identification with Training Data	59
3.4.2	Simulation Results	62
3.5	Experimental Validation	64
3.5.1	Identification with Training Data	64
3.5.2	Validation with Validation Data	67
3.6	Summary	71
4	Real-Time Optimal Charging Design Based on Explicit Model Predictive Control	72
4.1	Introduction	72
4.2	Model Description	76
4.3	Optimal Charging Design	80
4.3.1	Charging Problem Formulation	80
4.3.2	Charging Design Based on eMPC	84
4.4	Numerical Simulation	87
4.4.1	Basic Case Study	88
4.4.2	EKF-Based Output-Feedback Charging Control	93
4.4.3	Effects of Changing Constraints and Horizon Parameters	94
4.4.4	Evaluation of Computational Efficiency	97
4.5	Experimental Validation	98
4.6	Summary	99
5	Real-Time Pack Spatial Thermal Field Estimation via Distributed Kalman Filter	101
5.1	Introduction	101
5.2	Battery Pack Thermal Model	106
5.3	Model Discretization	111
5.3.1	Discretization	111
5.3.2	Discrete-Time State-Space Model	115

5.4	Kalman Filter Estimation	117
5.4.1	Centralized Kalman Filtering	117
5.4.2	Distributed Kalman Filtering	119
5.4.3	Steady-State Distributed Kalman Filtering	121
5.4.4	Computational Complexity Analysis	125
5.5	Numerical Simulation	126
5.5.1	Simulation Setting	127
5.5.2	Simulation Results	129
5.6	Summary	132
6	Conclusions	134
	References	137

List of Figures

1.1	Comparison of commonly used rechargeable batteries [7].	1
1.2	Global battery demand between 2018 and 2030 [4].	2
1.3	Lithium-ion battery: (a) single cell structure and (b) schematic diagram.	3
1.4	An overview of battery management algorithms based on keyword search <i>via</i> Google Scholar. For example, “battery modeling” is based on the searching results of "lithium-ion" AND "battery management" AND "modeling".	6
2.1	(a) The original double-capacitor model and (b) the proposed NDC model.	17
2.2	The single-particle model (top), and a particle (bottom) subdivided into two volumes, core and shell, which correspond to R_b - C_b and R_s - C_s , respectively.	19
2.3	The Wiener-type structure of the nonlinear double-capacitor (NDC) model.	27
2.4	PEC [®] SBT4050 battery tester.	34
2.5	Identification 1.0: parameter identification of $h(\cdot)$ that defines SoC-OCV relation.	35
2.6	Identification 1.0: model fitting with the training data obtained under 3 A constant-current discharging.	36
2.7	Identification 1.0: predictive fitting over validation data obtained by discharging at different constant currents.	36
2.8	Identification 1.0: predictive fitting over validation dataset obtained by discharging at varying currents (0~3 A). (a) Current profile. (b) Voltage fitting.	37
2.9	Identification 1.0: predictive fitting over validation dataset obtained by discharging at varying currents (0 ~ 6 A). (a) Current profile. (b) Voltage fitting.	38
2.10	Identification 2.0: (a) model fitting with training dataset and (b) fitting error in percentage.	40

2.11	Identification 2.0: predictive fitting over validation datasets obtained by discharging at different varying currents.	41
2.12	Identification 2.0: (a) predictive fitting over validation dataset obtained by discharging at varying currents between 0 A and 6 A and (b) predictive fitting error in percentage.	42
2.13	Identification 2.0: identification of the SoC-OCV relation based on different models, compared to the truth.	43
3.1	The Thevenin model.	48
3.2	A model-based computer-generated terminal voltage profile for one Monte Carlo simulation.	60
3.3	Parameter estimation errors for (a) NLS as a benchmark, (b) C-NLS and (c) R-NLS.	62
3.4	Histogram of estimates of θ_6 for (a) NLS as a benchmark, (b) C-NLS and (c) R-NLS.	62
3.5	Theoretical estimation accuracy of the R-NLS method changes with λ	63
3.6	The voltage response of the Samsung INR18650-25R Li-ion battery in a constant-current/constant-voltage discharging experiment.	65
3.7	Comparison of measured and fitted terminal voltage for (a) NLS as a benchmark, (b) C-NLS and (c) R-NLS.	67
3.8	The voltage response of the battery in an intermittent discharging test.	67
3.9	Comparison of measured and estimated SoC-OCV for (a) NLS as a benchmark, (b) C-NLS and (c) R-NLS.	68
3.10	Comparison of measured and estimated $R_0 + R$ for (a) NLS as a benchmark, (b) C-NLS and (c) R-NLS.	68
3.11	Variable currents from normalization of the UDDS profile.	69
3.12	UDDS: (a) comparison of measured and estimated voltage for the benchmark NLS, C-NLS and R-NLS methods and (b) errors.	69
3.13	Variable currents from normalization of the WLTP profile.	70

3.14	WLTP: (a) comparison of measured and estimated voltage for the benchmark NLS, C-NLS and R-NLS methods and (b) errors.	70
4.1	The nonlinear double-capacitor model.	77
4.2	Development of the eMPC-based charging control algorithm.	87
4.3	Multi-segment approximation of $h(V_s)$ and $R_0(V_s)$	89
4.4	Critical region partitioning on the V_b - V_s plane when $I = 2$ A, $\check{r} = 0.9$, and $\check{u} = 0$ A.	89
4.5	A basic case study of the eMPC-based charging control (black dashed lines denote constraints).	91
4.6	Comparison between eMPC and NMPC for charging control.	92
4.7	Output-feedback charging control based on eMPC and EKF.	93
4.8	Charging control under different constraints on $V_s - V_b$	94
4.9	Charging time versus γ_1 under $\gamma_2 = 0.08$	94
4.10	Charging control when the prediction horizon $N = 10, 50, 90$	95
4.11	Charging control when the control horizon $N_u = 2, 5, 9$	96
4.12	Charging control when the constraint horizon $N_c = 2, 5, 9$ for the constraint on $V_s - V_b$	96
4.13	Experimental results based on the proposed charging strategies under different $V_s - V_b$ constraints.	99
5.1	Schematic diagram of a LiB pack.	107
5.2	Schematic diagram of the nodes in a LiB pack comprising three cells, namely, $N_c = 3$	112
5.3	The y-z cross-section of the control volume of a node on the cell-cell interface.	112
5.4	A schematic of the DKF for a three-cell pack.	119
5.5	Schematic diagram of a basic unit in a LiB cell.	128
5.6	Discharging current profile based on UDDS.	128
5.7	First row: real temperature fields through time for the LiB pack discharged at UDDS-based current loads. Second row: CKF-based reconstruction. Third row: DKF-based reconstruction.	130

5.8	Evaluation of the estimation error over time.	131
5.9	Matrix \mathbf{F} plotted in black and white.	131
5.10	Evaluation of the estimation error over time when the simplified heat generation model is used.	132

List of Tables

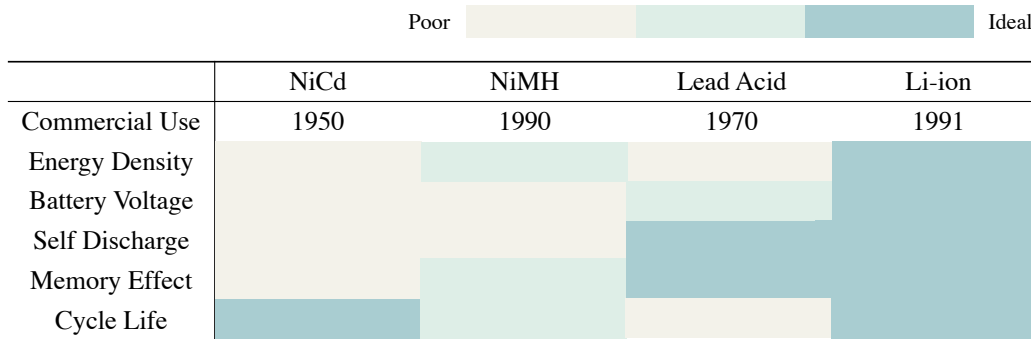
2.1	Quasi-Newton-based implementation for MAP-based Wiener identification.	32
2.2	Identification 1.0: initial guess, bound limits and identification results.	35
2.3	Identification 2.0: initial guess, prior knowledge and identification results.	39
3.1	The simulation setting.	61
3.2	Parameter estimates for the battery used in the experiments.	66
4.1	Battery model parameters.	88
4.2	Linearization setting.	88
4.3	Comparison of computational time.	98
5.1	Arithmetic operation requirements of the CKF, DKF and SS-DKF algorithms. . . .	126
5.2	LiB cell parameters used in the simulation [37].	127

Chapter 1

Introduction

1.1 Background

Lithium-ion batteries (LiBs) are a revolutionary technology for energy storage. They have high energy/power density, low self discharge, low memory effect and long cycle life, notably advantageous over other types of batteries (see Figure 1.1). Since first commercialized in 1991, LiBs have been extensively used in consumer electronics, from wearables and mobile phones to laptops, medical devices and power tools, bringing great convenience to our life.



Note: Lead Acid was first commercialized in 1886 and the above Lead Acid refers to the sealed one.

Figure 1.1: Comparison of commonly used rechargeable batteries [7].

The adoption of LiBs is not confined to the consumer electronics, but has also penetrated into sectors of electrified transportation such as electric vehicles (EVs). This trend is further strengthened with rapidly falling LiB prices. Now LiB-powered EVs have greatly narrowed the gap with internal-combustion-engine vehicles in both price and mileage, stimulating an exponential growth of LiB markets (see Figure 1.2) [2, 6].

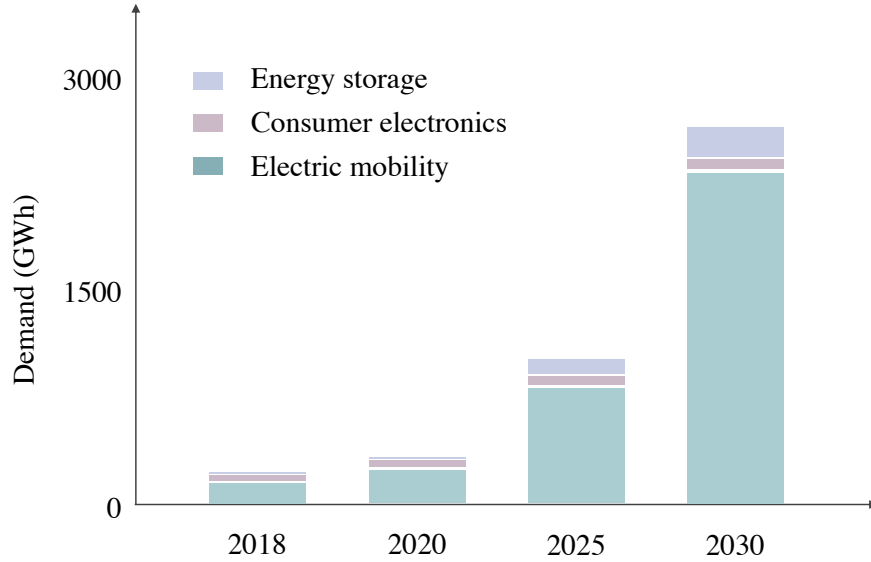


Figure 1.2: Global battery demand between 2018 and 2030 [4].

Another field that foresees a growing application of LiBs especially in the form of large battery packs is grid-scale battery storage [4]. It can enhance grid system flexibility and reliability by peak shaving, load leveling, primary frequency response, regulation and black start [30]. Besides, it can facilitate growing integration of renewable energies such as solar and wind by smoothing out their variability and intermittence [19].

To ensure the safe, reliable and durable use of LiBs in the above applications, the battery management technology is a must, which is also the focus of this dissertation. In the remainder of this chapter, we will first briefly overview the working principle and some common material choices for LiBs and then give an introduction of the battery management system.

1.2 Lithium-ion Battery

Working Principle

Rechargeable LiBs are generally manufactured and available in cylindrical, prismatic and pouch shapes. Despite the different shapes, they share a similar fundamental working principle. The main

components of a LiB include the cathode (positive electrode), anode (negative electrode), electrolyte, separator and current collectors. Figure 1.3 shows a schematic diagram of a cylindrical LiB cell.

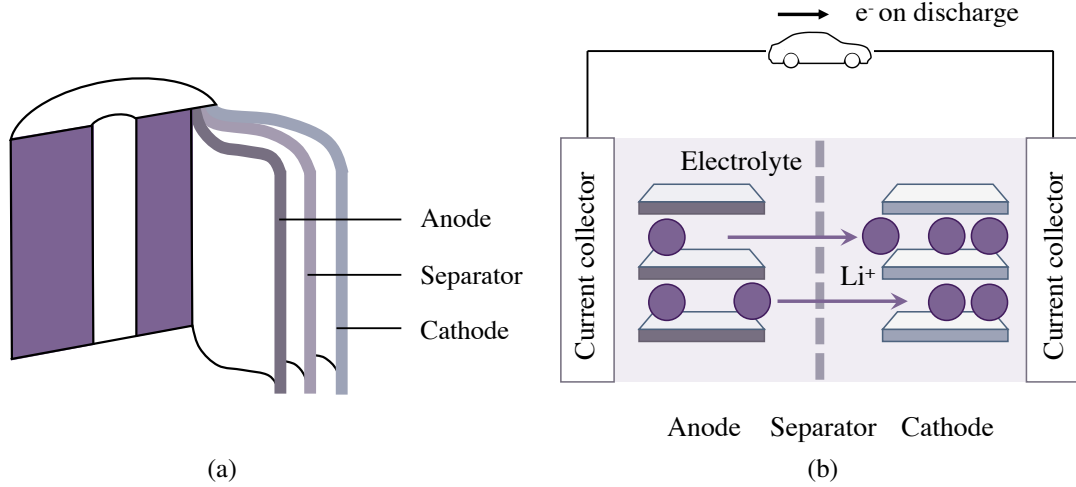
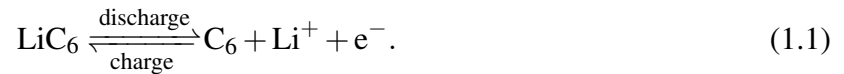
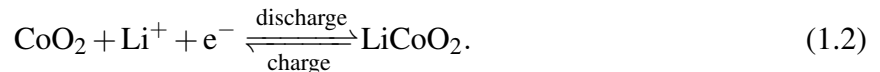


Figure 1.3: Lithium-ion battery: (a) single cell structure and (b) schematic diagram.

LiBs are intercalation-electrode batteries. During discharging, lithium ions de-intercalate from the anode and intercalate into the cathode through oxidation and reduction, respectively [26]. The charging process follows the reverse procedure. For further explanation, we consider lithium cobalt oxide (LiCoO_2) and graphite (C_6) as the active materials for the respective cathode and anode as an example. When the battery is under discharge, lithium atoms in the anode are oxidized and become lithium ions as governed by [26]:



The lithium ions exit the anode and migrate through the electrolyte and separator to the cathode. At the cathode, Co accepts an electron and is reduced from +4 to +3 state. The intercalation of a lithium ion compensates for the loss of the positive charge and keeps the cathode neutral as shown below [26]:



Meanwhile, the separator, ionically conducted but electrically insulated, keeps the electrons from passing through, so they have to make their way through the external circuit and move from the anode to the cathode. During charging, the lithium ions and electrons move in an opposite way. The current collectors are electrical conductors, which are attached to the surfaces of the electrodes and collect the electrons. They are usually made of copper for the anode and aluminum for the cathode.

Electrode Materials

To facilitate the intercalation process of lithium ions, the electrode materials typically have porous open-crystal structures. For the anode, the most common material is graphite, which is structured in layers and provides space for the intercalation of lithium ions. Every six carbon atoms can host up to one lithium atom, and the lithiated graphite is hence denoted as Li_xC_6 , where $0 \leq x \leq 1$. Other than graphite, other materials are being explored for the anode. For instance, silicon can potentially increase the energy density as one silicon atom can bond with as many as four lithium ions, and lithium titanate ($\text{Li}_4\text{Ti}_5\text{O}_{12}$) offers larger specific surface area and can accommodate fast charging more easily.

Compared with the anode, there are more materials available for the cathode. Generally they can be categorized into three groups: *layered*, *spinel*, and *olivine cathodes* [156]. In 1980, John B. Goodenough introduced the first *layered cathode*, LiCoO_2 , also known as LCO. Some of its popular variations include lithium nickel cobalt manganese oxide (LiNiCoMnO_2 or NCM) and lithium nickel cobalt aluminum oxide (LiNiCoAlO_2 or NCA). They are utilized, respectively, for Panasonic NCR18650B and Samsung INR18650-25R batteries, which are used in this dissertation's experimental validation. Nickel is an alternative for cobalt, which can mitigate toxicity and increase energy density and thermal stability of LCO. In 1983, a *spinel cathode* material, lithium manganese oxide (LiMn_2O_4 or LMO), was introduced by Goodenough and Michael M. Thackeray. It is cheaper and safer than LCO but usually has shorter lifetime because the manganese can dissolve into electrolyte under some conditions. In 1997, Goodenough proposed the *olivine cathode*, made of lithium iron phosphate (LiFePO_4 or LFP). It is even cheaper and less toxic, but the

specific energy is low due to low voltage and heavy iron atoms. Besides these active materials, the electrodes also contain other materials that serve as binders of particles of active materials and between particles and current collectors. They are conductive but not involved in electrochemical reactions as in (1.1) and (1.2).

1.3 Battery Management System

Battery management systems (BMSs) are an electronic system that integrates hardware, software and algorithms to manage the use of LiBs, ensuring them to function properly and safely. Their importance stems from at least the following three aspects:

- *Complexity*: LiB charging/discharging involves many electrochemical reactions and physical processes. They are complex, nonlinear and sometimes have elusive dynamics. Just as an example, the voltage response of a LiB cell will depend on a large number of factors ranging from operating conditions to state-dependent resistance and temperatures. Further, a LiB pack that consists of multiple cells will see a much higher complexity in its dynamics.
- *Vulnerability*: LiBs are vulnerable to overcharge, overdischarge, excessive currents and temperatures, and suffer faster aging when such events happen [5]. An example in point is the 2011 Nissan Leaf powered by air-cooled LiBs, which saw an accelerated capacity loss in the areas of extreme heat such as Arizona [1]. In extreme conditions, the vulnerability will lead LiBs to catch fires and explosions.
- *Expenses*: LiBs are still expensive. Their current price is about \$150 per kWh, and a 60 kWh EV battery pack costs about \$9,000 [2]. If LiBs function poorly or even unsafely, heavy economic losses can result from shortened life, operation failures and even replacements.

To handle the above challenges, BMSs generally provide a series of functions, including: 1) monitoring the current, voltage and temperature, 2) detecting and responding to faults to protect a battery, 3) estimating a battery's operating conditions and health, 4) running real-time control of

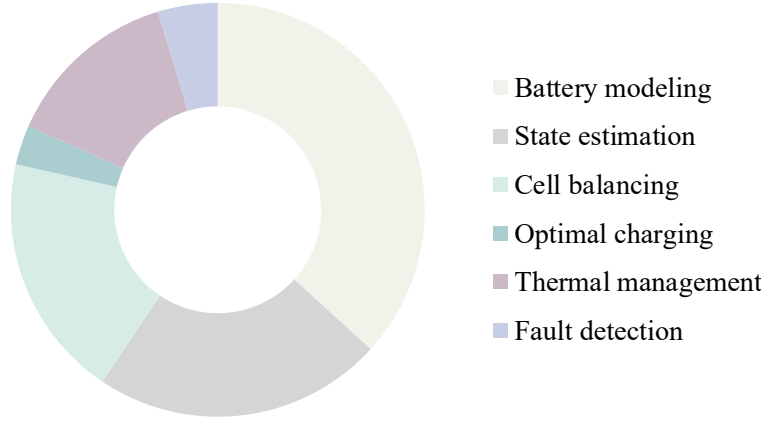


Figure 1.4: An overview of battery management algorithms based on keyword search *via* Google Scholar. For example, “battery modeling” is based on the searching results of "lithium-ion" AND "battery management" AND "modeling".

charging, balancing and temperature conditions, and 5) logging and communicating various types of data and commands with higher-level systems. BMS research has attracted growing attention in recent years due to its crucial importance for LiB systems. A central challenge in this emerging field is development of dynamic models and model-based estimation and control algorithms toward maximizing the functionality and performance of LiBs. Next is an overview of key BMS problems.

Battery Modeling

Battery models are the foundation for many battery management algorithms and have attracted intense interest of researchers as shown in Figure 1.4. There are mainly two kinds of battery models: electrochemical models and equivalent circuit models (ECMs) [35]. The electrochemical models describe a battery’s dynamics due to electrochemical reactions and diffusion processes during charging/discharging, e.g., the diffusion and intercalation of lithium ions and associated effects on ion concentrations and potentials. They can be used to predict not only battery voltage dynamics but also unmeasurable internal states such as lithium-ion concentrations and overpotentials that are important to battery performance, health and aging process [148]. However, they typically manifest a complex structure involving partial differential equations (PDEs) and dozens of states. By contrast, ECMs capture a battery’s dynamics using electrical circuits, which are much simpler in

structure and usually consist of ordinary differential equations (ODEs) with a much less number of states. In general, ECMs provide better computational efficiency and thus are more suitable for embedded control design and applications for real-time BMSs.

State Estimation

Operation of LiBs requires a close monitoring of many important yet unknown states, including state of charge (SoC), state of power (SoP) and state of health (SoH). SoC refers to the amount of available capacity in a LiB, which must be determined to avoid overcharge and overdischarge. SoP indicates the power that a LiB can supply, playing a critical role in power management and safe use of LiB systems. SoH describes a LiB's health condition, which provides a crucial awareness of the aging process and remaining life of a LiB. There are various empirical methods and practices in the industry, which, however, offer limited accuracy in general. An increasing amount of attention in the research community has been turned to model-based state estimation. The driving notion lies in leveraging both knowledge about a LiB's dynamics as captured by a model and real-time current/voltage/temperature measurements to facilitate accurate state estimation. A great number of approaches have been developed in the past years. For instance, nonlinear state observers and estimators, especially the well-known Kalman filter, have been applied and proven as an effective tool for SoC and SoH estimation [154, 155, 18]. Both electrochemical models and ECMs can find good use in LiB state estimation, as shown in the literature [154, 60]. More recently, data-driven methods, which draw upon inspiration from machine learning, have emerged as a useful means, in a context where many of today's LiB systems offer increasingly abundant data [130]. However, challenges continue to remain, because highly complicated dynamics, along with a broad range of factors, underlie the evolution of a LiB's states, thus calling for more research efforts [69].

Cell Balancing

A LiB pack consists of multiple cells connected in series or parallel. In real world, it inevitably faces cell-to-cell variations in capacity and resistance due to manufacturing tolerances, different

aging rates, and uneven work conditions. The variations will increase if left neglected and lead some cells to become weaker in a vicious cycle, thus limiting the entire pack's performance and life [81, 82, 143]. Energy balancing across the cells is thus necessary to reduce the variation, which aims to drive the cells to a common energy or SoC level. There are two main approaches in the literature, namely, passive and active balancing [208]. The passive approaches remove the excess charge of high-energy cells through shunt resistors. They only need simple and cheap circuits but offer slow speed in balancing and cause an undesirable waste of energy. By contrast, active balancing seeks to transfer energy from high-energy cells to low-energy cells through some equalizer circuits. While more complicated in circuit and algorithm design, it can enable faster balancing with little loss of energy.

Charging Control

The charging process is crucial for the performance and longevity of LiBs. Improper charging can cause internal stress accumulation, crystallization, and various other negative effects. The consequences include fast capacity fade, shortened life, and increased risk of use. This fact highlights the significance of health-aware charging. In the meantime, many practical applications, stretching from consumer electronics to EVs, strongly demand fast charging. These two objectives, however, are in conflict with each other. Hence, optimal charging control design has attracted great interest from researchers, which usually aims to strike a balance between them. Extensive studies have been proposed to enable health-conscious fast charging based on model-based optimal control and constrained optimization [120, 184]. There also exist some other methods, e.g., optimized pulse charging [56]. A main challenge for optimal charging control is the computational complexity due to the needed online optimization, which has motivated a broad search for more efficient control designs. This dissertation will present our recent advance about optimal charging based on explicit model predictive control.

Thermal Management

Thermal management is a prominent issue for LiBs. The operation of LiBs entails heat generation and buildup, which is also accompanied by the effects of ambient temperature [63]. For LiBs, the temperature has a strong impact on the performance and safety [211]. They should be operated within a certain temperature range, typically from -20 to 60°C and must avoid too low or too high temperatures. Most types of LiBs are also vulnerable to thermal events due to the highly reactive lithium and flammable electrolyte and may catch fire or explode in extreme cases. Effective thermal management thus carries tremendous significance. An important aspect of this subject is to design systems and methods to adjust the temperature of a LiB system, by cooling (by air, liquid or phase change materials) or heating as dependent on needs [218, 215, 99]. Another important set of efforts, particularly from the control community, aim to monitor the real-time temperature of LiBs. A popular method is model-based estimation, which exploits a thermal model and temperature measurements to infer the unknown temperature at some key locations, e.g., inside a cell [121, 180]. Despite increasing studies, this problem still remains open, because of the spatio-temporal complexity of LiBs' thermal dynamics. This dissertation will provide a study of how to reconstruct the three-dimensional temperature field of a LiB pack at both high accuracy and efficiency.

Fault Detection

Faults can happen during the operation of a battery system like many electrical engineering systems. The faults can be generally classified as battery faults and system faults [191]. The battery faults include overcharge, overdischarge, overmuch current and temperature, internal and external short circuit, and thermal runaway. The system faults include current, voltage and temperature sensor faults, cell connection and cooling system faults. The faults can be catastrophic to the safety and reliability of the battery system, which necessitates the early and accurate fault detection. Recent years have witnessed a growing amount of works on the battery system fault detection [191]. Roughly they can be categorized into three groups: heuristic methods, model-based methods and data-driven methods. Heuristic methods detect faults *via* pre-set thresholds for temperature rise

and voltage change [203, 207]. They are easy to implement, but the thresholds can be hard to determine for different battery chemistries and applications. Model-based methods need much less testing and calibration [12]. They detect faults by analyzing model-based residual signals that are fault-sensitive [138, 39, 132, 49]. By contrast, data-driven methods use machine learning or various data analysis methods to identify faults from data directly [140, 198]. Despite the advances, multiple challenges remain due to the elusiveness of potential faults and BMS hardware limitations, and more research efforts are still needed [191].

1.4 Statement of Contributions

This dissertation focuses on developing computationally efficient methods for several key BMS tasks, with the aim of enhancing potential implementation of advanced BMS algorithms on embedded computing platforms. The following summarizes our contributions.

- We proposed an ECM that is structurally simple and physically sound and accurate, which is the first ECM that can simulate the lithium-ion diffusion in an electrode and capture the nonlinear voltage behavior concurrently. We also developed two efficient and effective identification methods. More details are in Chapter 2.
- We performed a systematic study for one-shot parameter identification of the Thevenin model. It encompasses parameter identifiability analysis, identification method design and validation. More details are provided in Chapter 3.
- We developed an optimal health-aware charging method with high computational efficiency and amenability to practical implementation. More details can be found in Chapter 4.
- We designed a distributed estimation scheme to reconstruct the three-dimensional temperature field of a LiB pack, which is computationally efficient and accurate. More details are discussed in Chapter 5.

Chapter 2

Development of the Nonlinear Double-Capacitor Model

This chapter¹ proposes a new equivalent circuit model for batteries by modifying a double-capacitor model proposed in [101]. It is known that the original model can address the rate capacity effect and energy recovery effect inherent to batteries better than other models. However, it is a purely linear model and includes no representation of a battery's nonlinear phenomena. Hence, this work transforms the original model by introducing a nonlinear-mapping-based voltage source and a serial resistance-capacitance (RC) circuit. The modification is justified by an analogy with the single-particle model. Two offline parameter estimation approaches, termed 1.0 and 2.0, are designed for the new model to deal with the scenarios of constant-current and variable-current charging/discharging, respectively. An extensive experimental evaluation shows that the proposed model offers excellent accuracy and predictive capability. A comparison against the Rint and Thevenin models further points to its superiority. With high fidelity and low mathematical complexity, this model is beneficial for various real-time battery management applications.

2.1 Introduction

Rechargeable batteries have seen an ever-increasing use in today's industry and society as power sources for systems of different scales, ranging from consumer electronic devices to electric vehicles and smart grid. This trend has motivated a growing body of research on advanced battery management algorithms, which are aimed to ensure the performance, safety and life of battery systems. Such algorithms generally require mathematical models that can well characterize a battery's

¹This chapter is based on the dissertation author's first-authored journal paper [182] © 2020 IEEE.

dynamics. This has stimulated significant attention in battery modeling during the past years, with the current literature offering a plethora of results.

There are two main types of battery models: 1) electrochemical models that build on electrochemical principles to describe the electrochemical reactions and physical phenomena inside a battery during charging/discharging, and 2) equivalent circuit models (ECMs) that replicate a battery's current-voltage characteristics using electrical circuits made of resistors, capacitors and voltage sources. With structural simplicity, the latter ones provide great computational efficiency, thus more suitable for real-time battery management. However, as the other side of the coin, the simple circuit-based structures also imply a difficulty to capture a battery's dynamic behavior at a high accuracy. Therefore, this work aims to develop a new ECM based on the one in [101] so that it can offer high fidelity while retaining low mathematical complexity, through systematically investigating the model construction, parameter identification, and experimental validation.

Literature Review

Review of Battery Modeling

As mentioned above, the electrochemical models and ECMs constitute the majority of the battery models available today. The electrochemical modeling approach seeks to characterize the physical and chemical mechanisms underlying the charging/discharging processes. One of the best-known electrochemical models is the Doyle-Fuller-Newman model, which describes the concentrations and transport of lithium ions together with the distribution of separate potential in porous electrodes and electrolyte [51, 66, 35]. While delineating and reproducing a battery's behavior accurately, this model, like many others of similar kind, involves many partial differential equations and causes high computational costs. This has driven the development of some simplified versions, e.g., the single-particle model (SPM) [35, 77], and various model reduction methods, e.g., [142, 225, 92], toward more efficient computation.

By contrast, the ECMs are generally considered as more competitive for real-time battery monitoring and control, having found their way into various battery management systems. The first

ECM to our knowledge is the Randles model proposed in the 1940s [161]. It reveals a lead-acid battery's ohmic and reactive (capacitive and inductive) resistance, demonstrated in the electrochemical reactions and contributing to various phenomena of voltage dynamics, e.g., voltage drop, recovery and associated transients. This model has become a de facto standard for interpreting battery data obtained from electrochemical impedance spectroscopy (EIS) [221]. It also provides a basis for building diverse ECMs to grasp a battery's voltage dynamics during charging/discharging. Adding a voltage source representing the open-circuit voltage (OCV) to the Randles model, one can obtain the popular Thevenin model [67, 83, 156]. The Thevenin model without the RC circuit is called as the Rint model, which includes an ideal voltage source with a series resistor [83]. If more than one RC circuit is added to the Thevenin model, it becomes the dual polarization (DP) model that is capable of capturing multi-time-scale voltage transients during charging/discharging [83].

The literature has also reported a few modifications of the Thevenin model to better characterize a battery's dynamics. Generally, they are based on two approaches. The first one aims to describe a battery's voltage more accurately by incorporating certain phenomena, e.g., hysteresis, into the voltage dynamics, or through different parameterizations of OCV with respect to the state of charge (SoC) [199, 122, 71, 149, 60, 153, 94]. Some literature also models the resistors and capacitors as dependent on SoC, as well as some other factors like the temperature or rate and direction of the current loads in order to improve the accuracy of battery voltage prediction [116, 113]. The second approach sets the focus on improving the runtime prediction for batteries. In [36], a battery's capacity change due to cycle and temperature is considered and parameterized, and the dependence of resistors and capacitors on SoC also characterized. A similar investigation is made in [105] to improve the Thevenin model, which proposes to capture the nonlinear change of a battery's capacity with respect to the current loads.

An ECM that shows emerging importance is a double-capacitor model [101, 100]. It consists of two capacitors configured in parallel, which correspond to an electrode's bulk inner part and surface region, respectively, and can describe the process of charge diffusion and storage in a battery's electrode [59]. Compared to the Thevenin model, this circuit structure allows the rate

capacity effect and charge recovery effect to be captured, making the model an attractive choice for charging control [59, 56]. However, based on a purely linear circuit, this model is unable to grasp nonlinear phenomena innate to a battery—for instance, the nonlinear SoC-OCV relation is beyond its descriptive capability—and thus has its applicability limited. The presented work is motivated to remove this limitation by revamping the model’s structure. The effort will eventually lead to a new ECM that, for the first time, can capture the charge diffusion within a battery’s electrode and its nonlinear voltage behavior simultaneously.

Review of Battery Model Identification

A key problem associated with battery modeling is parameter identification, which pertains to extracting the unknown model parameters from the measurement data. Due to its importance, recent years have seen a growth of research. The existing methods can be divided into two main categories, experiment-based and data-based. The first category conducts experiments of charging, discharging or EIS and utilizes the experimental data to read a model’s parameters. It is pointed out in [166, 10] that the transient voltage responses under constant- or pulse-current charging/discharging can be leveraged to estimate the resistance, capacitance and time constant parameters of the Thevenin model. In addition, the relation between SoC and OCV is a defining characteristic of a battery’s dynamics. It can be experimentally identified by charging or discharging a battery using a very small current [52], or alternatively, using a current of normal magnitude but intermittently (with a sufficiently long rest period applied between two discharging operations) [85, 188]. The EIS experiments have also been widely used to identify a battery’s impedance properties [139, 73, 28]. While involving basic data analysis, the methods of this category generally put emphasis on the design of experiments. In a departure, the second category goes deeper into understanding the model-data relationship and pursues data-driven parameter estimation. It can enable provably correct identification even for complex models, thus often acknowledged as better at extracting the potential of data. It is proposed in [91] to identify the Thevenin model by solving a set of linear and polynomial equations. Another popular means is to formulate model-

data fitting problems and solve them using least squares or other optimization methods to estimate the parameters [86, 210, 186, 62, 157, 172]. When considering more complicated electrochemical models, the identification usually involves large-size nonlinear nonconvex optimization problems. In this case, particle swarm optimization and genetic algorithms are often exploited to search for the best parameter estimates [66, 217, 159, 213]. A recent study presents an adaptive-observer-based parameter estimation scheme for an electrochemical model [118]. While the above works focus on identification of physics-based models, data-driven black-box identification is also examined in [95, 117, 163], which construct linear state-space models *via* subspace identification or nonparametric frequency domain analysis. A topic related with identification is experiment design, which is to find out the best input sequences to excite a battery to maximize the parameter identifiability. In [164, 146], optimal input design is performed by maximizing the Fisher information matrices—an identifiability metric—involved in the identification of the Thevenin model and the SPM, respectively.

The presented work is also related with the literature on Wiener system identification, because the model to be developed has a Wiener-type structure featuring a linear dynamic subsystem in cascade with a static nonlinear subsystem. Wiener systems are an important subject in the field of parameter identification, and a reader is referred to [72] for a collection of recent studies. Wiener system identification based on maximum likelihood (ML) estimation is investigated in [80, 193], which shows significant promises. However, the optimization procedure resulting from the ML formulation can easily converge to local minima due to the presence of the nonlinear subsystem. This hence yields a motivation to enhance the notion of ML-based identification in this work to achieve more effective battery parameter estimation.

Contributions

This work presents the following contributions.

- A new ECM, named the *nonlinear double-capacitor (NDC) model*, is developed. By design, it transforms the linear double-capacitor model in [101] by coupling it with a nonlinear

circuit mimicking a battery's voltage behavior. With this pivotal change, the NDC model introduces two advantages over existing ECMs. First, it can simulate not only the charge diffusion characteristic of a battery's electrochemical dynamics, but also the critical nonlinear electrical phenomena. This unique feature guarantees the model's better accuracy, which comes at only a very slight increase in model complexity. Second, the NDC model can be interpreted as a circuit-based approximation of the SPM. This further justifies its soundness while inspiring a refreshed look at the connections between the SPM and ECMs.

- Parameter identification is investigated for the proposed model. This begins with a study of the constant-current charging/discharging scenario, with an identification approach, termed 1.0, developed by fitting parameters with the measurement data. Then, shifting the focus to the scenario of variable-current charging/discharging, the study introduces a Wiener perspective into the identification of the NDC model due to its Wiener-type structure. A Wiener identification approach is proposed for the NDC model based on maximum *a posteriori* (MAP) estimation, which is termed 2.0. Compared to the ML-based counterparts in the literature, this new approach incorporates into the estimation a prior distribution of the unknown parameters, which represents additional information or prior knowledge and can help drive the parameter search toward physically reasonable values.
- Experimental validation is performed to assess the proposed results. This involves multiple experiments about battery discharging under different kinds of current profiles and a comparison of the NDC model with the Rint and Thevenin models. The validation shows the considerable accuracy and predictive capability of the NDC model, as well as the effectiveness of the 1.0 and 2.0 identification approaches.

Organization

The remainder of the chapter is organized as follows. Section 2.2 presents the construction of the NDC model. Section 2.3.1 studies parameter identification for the NDC model in the constant-

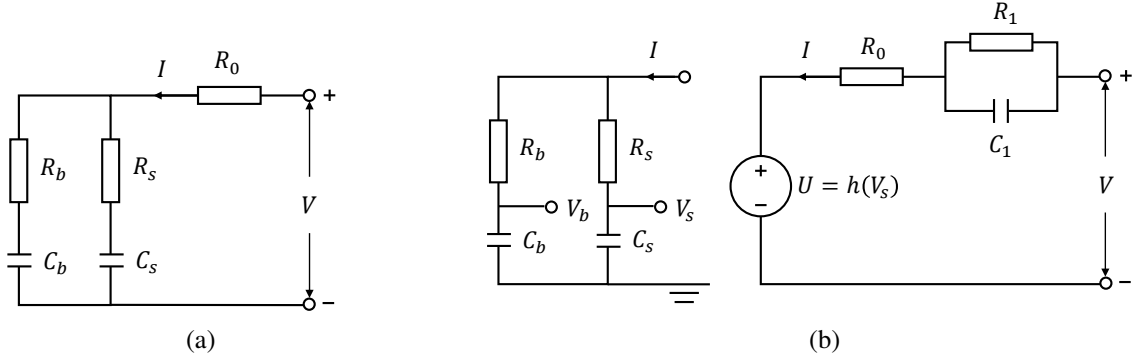


Figure 2.1: (a) The original double-capacitor model and (b) the proposed NDC model.

current charging/discharging scenario. Inspired by Wiener system identification, Section 2.3.2 proceeds to develop an MAP-based parameter estimation approach to identify the NDC model. Section 2.4 offers the experimental validation. Finally, Section 2.5 gathers concluding remarks.

2.2 Model Development

This section develops the NDC model and presents the mathematical equations governing its dynamic behavior.

To begin with, let us review the original linear double-capacitor model proposed in [101]. As shown in Figure 2.1a, this model includes two capacitors in parallel, C_b and C_s , each connected with a serial resistor, R_b and R_s , respectively. The double-capacitor structure simulates a battery's electrode, providing storage for electric charge, and the parallel connection between them allows the transport of charge within the electrode to be described. Specifically, one can consider the R_s - C_s circuit as corresponding to the electrode surface region exposed to the electrolyte; the R_b - C_b circuit represents an analogy of the bulk inner part of the electrode. As such, this model has the following features:

- $C_b \gg C_s$ and $R_b \gg R_s$;
- C_b is where the majority of the charge is stored, and R_b - C_b accounts for low-frequency responses during charging/discharging;

- C_s is much smaller, and its voltage changes at much faster rates than that of C_b during charging/discharging, making R_s - C_s responsible for high-frequency responses.

In addition, R_0 is included to embody the electrolyte resistance. This model was designed in [101] for high-power lithium-ion batteries, and its application can naturally extend to double-layer capacitors that are widely used in hybrid energy storage systems, e.g., [48].

As pointed out in [59], the linear double-capacitor model can grasp the rate capacity effect, i.e., the total charge absorbed (or released) by a battery goes down with the increase in charging (or discharging) current. To see this, just notice that the terminal voltage V mainly depends on V_s (the voltage across C_s), which changes faster than V_b (the voltage across C_b). Thus, when the current I is large, the fast rise (or decline) of V_s will make V hit the cut-off threshold earlier than when C_b has yet to be fully charged (or discharged). Another phenomenon that can be seized is the capacity and voltage recovery effect. That is, the usable capacity and terminal voltage would increase upon the termination of discharging due to the migration of charge from C_b to C_s . However, this model by nature is a linear system, unable to describe a defining characteristic of batteries—the nonlinear dependence of OCV on the SoC. It hence is effective only when a battery is restricted to operate conservatively within some truncated SoC range that permits a linear approximation of the SoC-OCV curve.

To overcome the above issue, the NDC model is proposed, which is shown in Figure 2.1b. It includes two changes. The primary one is to introduce a voltage source U , which is a nonlinear mapping of V_s , i.e., $U = h(V_s)$. Second, an RC circuit, R_1 - C_1 , is added in series to U . Next, let us justify the above modifications from a perspective of the SPM, a simplified electrochemical model that has recently attracted wide interest.

Figure 2.2 gives a schematic diagram of the SPM. The SPM represents an electrode as a single spherical particle. It describes the mass balance and diffusion of lithium ions in a particle during charging/discharging by Fick's second law of diffusion in a spherical coordinate system [77]. If subdividing a spherical particle into two finite volumes, the bulk inner domain (core) and the near-surface domain (shell), one can simplify the diffusion of lithium ions between them as the

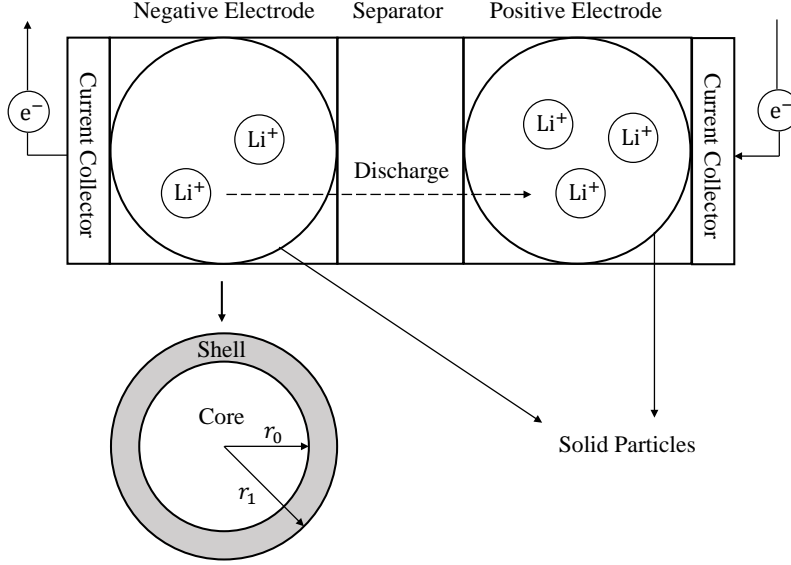


Figure 2.2: The single-particle model (top), and a particle (bottom) subdivided into two volumes, core and shell, which correspond to R_b-C_b and R_s-C_s , respectively.

charge transport between the capacitors of the double-capacitor model, as proven in [59]. For SPM, the terminal voltage consists of three elements: the difference in the open-circuit potential of the positive and negative electrodes, the difference in the reaction overpotential, and the voltage across the film resistance [35]. The open-circuit potential depends on the lithium-ion concentration in the surface region of the sphere, which is akin to the role of V_s here. Therefore, it is appropriate as well as necessary to introduce a nonlinear function of V_s , i.e., $h(V_s)$, as an analogy to the open-circuit potential. With $U = h(V_s)$, the NDC model can correctly show the influence of the charge state on the terminal voltage, while inheriting all the capabilities of the original model.

Furthermore, the NDC model also contains an RC circuit, R_1-C_1 , which, together with R_0 , simulates the impedance-based part of the voltage dynamics. Here, R_0 characterizes the linear kinetic aspect of the impedance, which relates to the ohmic resistance and solid electrolyte interface (SEI) resistance [137]; R_1-C_1 accounts for the voltage transients related with the charge transfer on the electrode/electrolyte interface and the ion mass diffusion in the battery [14]. This work finds that one RC circuit can offer sufficient fidelity, though it is possible to connect more RC circuits serially with R_1-C_1 to gain better accuracy.

The dynamics of the NDC model can be expressed in the state-space form as follows:

$$\begin{cases} \begin{bmatrix} \dot{V}_b(t) \\ \dot{V}_s(t) \\ \dot{V}_1(t) \end{bmatrix} = \begin{bmatrix} \frac{-1}{C_b(R_b+R_s)} & \frac{1}{C_b(R_b+R_s)} & 0 \\ \frac{1}{C_s(R_b+R_s)} & \frac{-1}{C_s(R_b+R_s)} & 0 \\ 0 & 0 & \frac{-1}{R_1C_1} \end{bmatrix} \begin{bmatrix} V_b(t) \\ V_s(t) \\ V_1(t) \end{bmatrix} + \begin{bmatrix} \frac{R_s}{C_b(R_b+R_s)} \\ \frac{R_b}{C_s(R_b+R_s)} \\ \frac{-1}{C_1} \end{bmatrix} I(t), \\ V(t) = h(V_s(t)) - V_1(t) + R_0I(t), \end{cases} \quad (2.1a)$$

$$(2.1b)$$

In above, $I > 0$ for charging, $I < 0$ for discharging, and V_1 refers to the voltage across the R_1 - C_1 circuit. One can parameterize $h(V_s)$ as a polynomial. A fifth-order polynomial is empirically selected here:

$$h(V_s) = \alpha_0 + \alpha_1 V_s + \alpha_2 V_s^2 + \alpha_3 V_s^3 + \alpha_4 V_s^4 + \alpha_5 V_s^5,$$

where α_i for $i = 0, 1, \dots, 5$ are coefficients. Note that $h(V_s)$ should be lower and upper bounded, depending on a battery's operating voltage range. This implies that V_b and V_s must also be bounded. For any bounds selected for them, it is always possible to find out a set of coefficients α_i 's to satisfy $h(\cdot)$. Hence, one can straightforwardly normalize V_b and V_s to let them lie between 0 V and 1 V, without loss of generality. In other words, $V_b = V_s = 1$ V at full charge (SoC = 1) and that $V_b = V_s = 0$ V for full depletion (SoC = 0). Following this setting, SoC is given by

$$\text{SoC} = \frac{Q_a}{Q_t} = \frac{C_b V_b + C_s V_s}{C_b + C_s}, \quad (2.2)$$

where $Q_t = C_b + C_s$ denotes the total capacity, and $Q_a = C_b V_b + C_s V_s$ the available capacity, respectively. It is easy to verify that the SoC's dynamics is governed by

$$\dot{\text{SoC}} = \begin{bmatrix} \frac{C_b}{C_b+C_s} & \frac{C_s}{C_b+C_s} & 0 \end{bmatrix} \begin{bmatrix} \dot{V}_b \\ \dot{V}_s \\ \dot{V}_1 \end{bmatrix} = \frac{1}{Q_t} I. \quad (2.3)$$

Meanwhile, it is worth noting that the SoC-OCV function would share the same form with $h(\cdot)$.

To see this point, recall that OCV refers to the terminal voltage when the battery is at equilibrium without current load. For the NDC model, the equilibrium happens when $V_b = V_s$, $V_1 = 0$ V and $I = 0$ A, and in this case, $V_s = \text{SoC}$ according to (2.2), and $\text{OCV} = h(V_s)$. This suggests that $\text{OCV} = h(\text{SoC})$. In addition, the internal resistance R_0 is also assumed to be SoC-dependent following the recommendation in [53], taking the form of

$$R_0 = \gamma_1 + \gamma_2 e^{-\gamma_3 \text{SoC}} + \gamma_4 e^{-\gamma_5 (1 - \text{SoC})}. \quad (2.4)$$

The rest of this chapter will center on developing parameter identification approaches to determine the model parameters using measurement data and apply identified models to experimental datasets to evaluate their predictive accuracy.

2.3 Parameter Identification

2.3.1 Approach 1.0: Constant-Current Charging/Discharging

This section studies parameter identification for the NDC model when a constant current is applied to a battery. The discharging case is considered here without loss of generality. In a two-step procedure, the $h(\cdot)$ function is identified first, and the impedance and capacitance parameters estimated next.

Identification of $h(\cdot)$

The SOC-OCV relation of the NDC model is given by $\text{OCV} = h(\text{SoC})$, as aforementioned in Section 2.2. Hence, one can identify $h(\cdot)$ by fitting it with a battery's SoC-OCV data. To obtain the SoC-OCV curve, one can discharge a battery using a small current (e.g., 1/25 C-rate as suggested in [52]) from full to empty. In this process, the terminal voltage V can be taken as OCV. Immediately one can see that $\alpha_0 = \underline{V}$ and $\sum_{i=0}^5 \alpha_i = \overline{V}$, where \underline{V} and \overline{V} are the minimum and maximum value of V in the process. Therefore, $\text{OCV} = h(\text{SoC})$ can be written as a function of α_i for

$i = 1, 2, \dots, 4$ as follows:

$$\text{OCV} = \underline{V} + \sum_{i=1}^4 \alpha_i \text{SoC}^i + \left(\bar{V} - \underline{V} - \sum_{i=1}^4 \alpha_i \right) \text{SoC}^5,$$

where OCV can be read directly from the terminal voltage measurements. By (2.3), SoC can be calculated using the coulomb counting method as follows:

$$\text{SoC} = 1 + \frac{1}{Q_t} It.$$

From above, one can observe that α_i for $i = 1, 2, \dots, 4$ can be identified by solving a data fitting problem, which can be addressed as a linear least squares problem. The identification results are unique and can be easily obtained. Then with $\alpha_0 = \underline{V}$ and $\alpha_5 = \bar{V} - \underline{V} - \sum_{i=1}^4 \alpha_i$, the function $h(\cdot)$ becomes explicit and ready for use.

Identification of Impedance and Capacitance

Now consider discharging the battery by a constant current of normal magnitude to determine the impedance and capacitance parameters. The identification can be attained by expressing the terminal voltage in terms of the parameters and then fitting it to the measurement data.

Given a battery left idling for a long period of time and then discharged under a constant current, the dynamics of V_s , according to (2.1a), can be derived as

$$V_s(t) = V_s(0) + \frac{It}{C_b + C_s} + \frac{C_b(R_b C_b - R_s C_s)I}{(C_b + C_s)^2} \left[1 - \exp \left(-\frac{C_b + C_s}{C_b C_s (R_b + R_s)} t \right) \right], \quad (2.5)$$

where $V_s(0)$ is known to us as it can be accessed from $\text{SoC}(0)$ when the battery is initially relaxed. However, it is impossible to identify C_b , R_b , C_s and R_s altogether. This issue can be seen from (2.5), where V_s depends on three parameters, i.e., $1/(C_b + C_s)$, $C_b(R_b C_b - R_s C_s)/(C_b + C_s)^2$ and $(C_b + C_s)/[C_b C_s (R_b + R_s)]$. Even if the three parameters are known, it is still not possible to extract all the four individual impedance and capacitance parameters from them due to the parameter redundancy.

Therefore, one can sensibly assume $R_s = 0$, as recommended in [172]. This is a tenable assumption for the NDC model since $R_s \ll R_b$ as aforementioned. As a result, (2.5) reduces to

$$V_s(t) = V_s(0) + \beta_1 It + \beta_2 I \left(1 - e^{-\beta_3 t}\right), \quad (2.6)$$

where

$$\beta_1 = \frac{1}{C_b + C_s}, \quad \beta_2 = \frac{R_b C_b^2}{(C_b + C_s)^2}, \quad \beta_3 = \frac{C_b + C_s}{C_b C_s R_b}.$$

Here, β_1 is known because Q_t can be calibrated by coulomb counting. When β_2 and β_3 are also available, C_b , C_s and R_b can be reconstructed as follows:

$$C_b = \frac{\beta_2 \beta_3}{\beta_1 (\beta_1 + \beta_2 \beta_3)}, \quad C_s = \frac{1}{\beta_1 + \beta_2 \beta_3}, \quad R_b = \frac{1}{\beta_1 \beta_3 C_b C_s}.$$

Further, in the above constant-current discharging scenario, the evolution of V_1 follows

$$V_1(t) = e^{-\beta_5 t} V_1(0) - I \beta_4 \left(1 - e^{-\beta_5 t}\right), \quad (2.7)$$

where

$$\beta_4 = R_1, \quad \beta_5 = \frac{1}{R_1 C_1}.$$

Since the battery has idled for a long period prior to discharging, $V_1(0)$ relaxes at zero and can be removed from (2.7).

Then, combining (2.1b), (2.4), (2.6) and (2.7), the terminal voltage response is given by

$$V(\boldsymbol{\theta}; t) = \sum_{i=0}^5 \alpha_i V_s^i(\boldsymbol{\theta}; t) + I \theta_3 \left(1 - e^{-\theta_4 t}\right) + I \theta_5 + I \theta_6 e^{-\theta_7 \text{SoC}(t)} + I \theta_8 e^{-\theta_9 (1 - \text{SoC}(t))}, \quad (2.8)$$

with

$$\boldsymbol{\theta} = \left[\beta_2 \quad \beta_3 \quad \beta_4 \quad \beta_5 \quad \gamma_1 \quad \gamma_2 \quad \gamma_3 \quad \gamma_4 \quad \gamma_5 \right]^\top,$$

and

$$V_s(\boldsymbol{\theta}; t) = V_s(0) + It/Q_t + \theta_1 I \left(1 - e^{-\theta_2 t}\right),$$

$$\text{SoC}(t) = \text{SoC}(0) + It/Q_t.$$

In above, the terminal voltage V is expressed in terms of $\boldsymbol{\theta}$, allowing one to identify $\boldsymbol{\theta}$ by minimizing the difference between the measured voltage and the voltage predicted by (2.8). Hence, a data fitting problem can be formulated. It should be noted that the resultant optimization will be nonlinear and nonconvex due to the presence of $h(\cdot)$. As a consequence, a numerical algorithm may get stuck in local minima and eventually give unreasonable estimates. A promising way of mitigating this challenge is to constrain the numerical optimization search within a parameter space that is believably correct. Specifically, one can roughly determine the lower and upper bounds of part or all of the parameters, set up a limited search space, and run numerical optimization within this space. With this notion, the identification problem can be formulated as a constrained optimization problem:

$$\hat{\boldsymbol{\theta}} = \arg \min_{\boldsymbol{\theta}} \frac{1}{2} [\mathbf{y} - \mathbf{V}(\boldsymbol{\theta})]^\top \mathbf{Q}^{-1} [\mathbf{y} - \mathbf{V}(\boldsymbol{\theta})], \quad (2.9a)$$

$$\text{s.t. } \underline{\boldsymbol{\theta}} \leq \boldsymbol{\theta} \leq \overline{\boldsymbol{\theta}}, \quad (2.9b)$$

where $\hat{\boldsymbol{\theta}}$ is the estimate of $\boldsymbol{\theta}$, $\underline{\boldsymbol{\theta}}$ and $\overline{\boldsymbol{\theta}}$ are the pre-set lower and upper bounds of $\boldsymbol{\theta}$, respectively, \mathbf{y} the terminal voltage measurement vector, \mathbf{Q} an $M \times M$ symmetric positive definite matrix representing the covariance of the measurement noise, with M being the number of the data points. Besides,

$$\mathbf{y} = \begin{bmatrix} y(t_1) & y(t_2) & \cdots & y(t_M) \end{bmatrix}^\top,$$

$$\mathbf{V}(\boldsymbol{\theta}) = \begin{bmatrix} V(\boldsymbol{\theta}; t_1) & V(\boldsymbol{\theta}; t_2) & \cdots & V(\boldsymbol{\theta}; t_M) \end{bmatrix}^\top.$$

Multiple numerical algorithms are available in the literature to solve (2.9), a choice among which is the interior-point-based trust-region method [186].

2.3.2 Approach 2.0: Variable-Current Charging/Discharging

While it is not unusual to charge or discharge a battery at a constant current, real-world battery systems such as those in electric vehicles generally operate at variable currents. Motivated by practical utility, an interesting and challenging question is: *Will it be possible to estimate all the parameters of the NDC model in one shot when an almost arbitrary current profile is applied to a battery?* Having this question addressed will greatly improve the availability of the model, even to an on-demand level, for battery management tasks. This section offers a study in this regard from a Wiener identification perspective. It first unveils the NDC model's inherent Wiener-type structure and then develops an MAP-based identification approach. Here, the study assumes R_0 to be constant for convenience.

Wiener-Type Structure of the NDC Model

The NDC model is structurally similar to a Wiener system—the double RC circuits constitute a linear dynamic subsystem, and cascaded with it is a nonlinear mapping. The following outlines the discrete-time Wiener-type formulation of (2.1).

Suppose that (2.1a) is sampled with a time period ΔT and then discretized by the zero-order-hold (ZOH) method. The discrete-time model is expressed as

$$x(t_{k+1}) = A_d x(t_k) + B_d I(t_k), \quad (2.10)$$

where k is the discrete-time index with $t_k = k\Delta T$, and

$$A_d = e^{A\Delta T}, \quad B_d = \left(\int_0^{\Delta T} e^{A\tau} d\tau \right) B.$$

Let us use t instead of t_k to represent the discrete time instant in sequel for notational simplicity.

Then, (2.10) can be written as

$$x(t) = (qI_{3 \times 3} - A_d)^{-1} B_d I(t) + (qI_{3 \times 3} - A_d)^{-1} q x(0),$$

where q is the forward shift operator, and $I_{3 \times 3} \in \mathbb{R}^{3 \times 3}$ is an identity matrix, respectively. Since $V_s(t) = [0 \ 1 \ 0]x(t)$ and $V_1(t) = [0 \ 0 \ 1]x(t)$, one can obtain the following after some lengthy derivation:

$$V_s(t) = G_1(q)I(t) + G_2(q)V_s(0), \quad (2.11)$$

$$V_1(t) = G_3(q)I(t) + G_4(q)V_1(0), \quad (2.12)$$

where

$$G_1(q) = \frac{(\beta_1 + \beta_2)q^{-1} - (\beta_1\beta_3 + \beta_2)q^{-2}}{1 - (1 + \beta_3)q^{-1} + \beta_3q^{-2}}, \quad G_2(q) = \frac{1}{1 - q^{-1}},$$

$$G_3(q) = \frac{\beta_4 q^{-1}}{1 + \beta_5 q^{-1}}, \quad G_4(q) = \frac{1}{1 + \beta_5 q^{-1}},$$

with

$$\beta_1 = \frac{A_{21}B_{11} + A_{12}B_{21}}{A_{12} + A_{21}}\Delta T, \quad \beta_2 = \frac{A_{21}(B_{21} - B_{11})}{(A_{12} + A_{21})^2}(1 - \beta_3),$$

$$\beta_3 = e^{-(A_{12} + A_{21})\Delta T}, \quad \beta_4 = -(\beta_5 + 1)B_{31}/A_{33}, \quad \beta_5 = -e^{A_{33}\Delta T}.$$

Note that the notation β is slightly abused above without causing confusion. Assume that the battery has been at rest for a sufficiently long time to achieve an equilibrium state before a test. In this setting, $V_s(0) = \text{SoC}(0)$, $V_1(0) = 0$ V, and $G_4(q)V_1(0) = 0$. Besides, one can also see that the same parameter redundancy issue as in Section 2.3.1 occurs again—only three parameters, β_1 through β_3 , appear in (2.11), but four physical parameters, C_b , C_s , R_b and R_s , need to be identified. To fix this, let $R_s = 0$ as was done before. Then β_1 through β_3 reduce to be

$$\beta_1 = \frac{\Delta T}{C_b + C_s}, \quad \beta_2 = \frac{R_b C_b^2 (1 - \beta_3)}{(C_b + C_s)^2}, \quad \beta_3 = e^{-\frac{C_b + C_s}{C_b C_s R_b} \Delta T}.$$

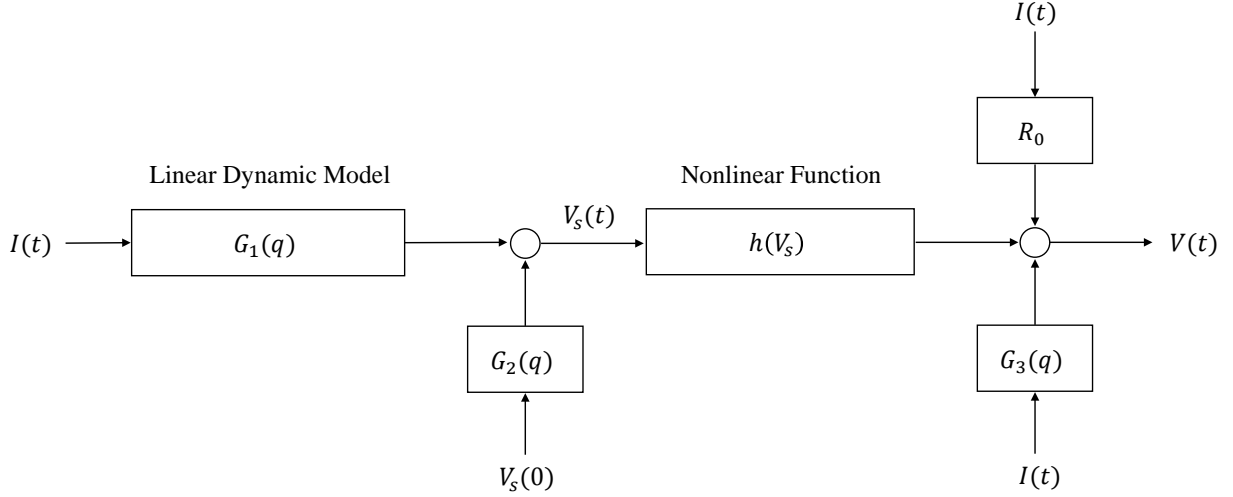


Figure 2.3: The Wiener-type structure of the nonlinear double-capacitor (NDC) model.

If β_1 through β_5 become available, the physical parameters can be reconstructed as follows:

$$C_b = \frac{\Delta T}{\beta_1} - C_s, \quad C_s = \frac{(1 - \beta_3) \Delta T}{\beta_1 - \beta_1 \beta_3 - \beta_2 \log \beta_3},$$

$$R_b = -\frac{(\Delta T)^2}{C_b C_s \beta_1 \log \beta_3}, \quad R_1 = \frac{-\beta_4}{\beta_5 + 1}, \quad C_1 = \frac{-\Delta T}{\log(-\beta_5) R_1}.$$

Finally, it is obvious that

$$V(t) = h[G_1(q)I(t) + G_2(q)V_s(0)] - G_3(q)I(t) + R_0 I(t). \quad (2.13)$$

The above equation reveals the block-oriented Wiener-type structure of the NDC model, as depicted in Figure 2.3, in which the linear dynamic model $G_1(q)$ and the nonlinear function $h(V_s)$ are interconnected sequentially. Given (2.13), the next pursuit is to estimate all of the parameters simultaneously, which include α_i for $i = 1, 2, \dots, 4$, β_i for $i = 1, 2, \dots, 5$, and R_0 . Here, α_0 and α_5 are free of identification as they can be expressed by α_i for $i = 1, 2, \dots, 4$ (see Section 2.3.1).

MAP-Based Wiener Identification

Consider the following model based on (2.13) for notational convenience:

$$z(t) = V(\boldsymbol{\theta}; u(t)) + v(t), \quad (2.14)$$

where u is the input current I , z the measured voltage, v the measurement noise added to V and assumed to follow a Gaussian distribution $\mathcal{N}(0, \sigma^2)$, and

$$V(\boldsymbol{\theta}; u(t)) = h[G_1(q, \boldsymbol{\theta})u(t) + G_2(q)V_s(0), \boldsymbol{\theta}] - G_3(q, \boldsymbol{\theta})u(t) + \theta_{10}u(t),$$

with

$$\boldsymbol{\theta} = \begin{bmatrix} \alpha_1 & \alpha_2 & \alpha_3 & \alpha_4 & \beta_1 & \beta_2 & \beta_3 & \beta_4 & \beta_5 & R_0 \end{bmatrix}^\top.$$

The input and output datasets are denoted as

$$\begin{aligned} \mathbf{u} &= \begin{bmatrix} u(t_1) & u(t_2) & \cdots & u(t_N) \end{bmatrix}^\top \in \mathbb{R}^{N \times 1}, \\ \mathbf{z} &= \begin{bmatrix} z(t_1) & z(t_2) & \cdots & z(t_N) \end{bmatrix}^\top \in \mathbb{R}^{N \times 1}, \end{aligned}$$

where N is the total number of data samples. A combination of them is expressed as

$$\mathbf{Z} = \begin{bmatrix} \mathbf{u} & \mathbf{z} \end{bmatrix}.$$

An ML-based approach is developed in [80] to deal with Wiener system identification. If applied to (2.14), it leads to consideration of the following problem:

$$\hat{\boldsymbol{\theta}} = \arg \max_{\boldsymbol{\theta}} p(\mathbf{Z}|\boldsymbol{\theta}).$$

Following this line, one can derive a likelihood cost function and perform minimization to find out $\hat{\boldsymbol{\theta}}$. However, this method can be vulnerable to the risk of local minima because of the nonconvexity

issue resulting from the static nonlinear function $h(\cdot)$. This can cause unphysical estimates. While carefully selecting an initial guess is suggested to alleviate this problem [79], it is often found inadequate for many practical systems. In particular, our study showed that it could hardly deliver reliable parameter estimation when used to handle the NDC model identification.

MAP-based Wiener identification thus is proposed here to overcome this problem. The MAP estimation can incorporate some prior knowledge about parameters to help drive the parameter search toward a reasonable minimum point. Specifically, consider maximizing the *a posteriori* probability distribution of $\boldsymbol{\theta}$ conditioned on \mathbf{Z} :

$$\hat{\boldsymbol{\theta}} = \arg \max_{\boldsymbol{\theta}} p(\boldsymbol{\theta}|\mathbf{Z}). \quad (2.15)$$

By the Bayes' theorem, it follows that

$$p(\boldsymbol{\theta}|\mathbf{Z}) = \frac{p(\mathbf{Z}|\boldsymbol{\theta}) \cdot p(\boldsymbol{\theta})}{p(\mathbf{Z})} \propto p(\mathbf{Z}|\boldsymbol{\theta}) \cdot p(\boldsymbol{\theta}).$$

In above, $p(\boldsymbol{\theta})$ quantifies the prior information available about $\boldsymbol{\theta}$. A general way is to characterize it as a Gaussian random vector following the distribution $p(\boldsymbol{\theta}) \sim \mathcal{N}(\mathbf{m}, \mathbf{P})$. Based on (2.14), $p(\mathbf{z}|\boldsymbol{\theta}) \sim \mathcal{N}(\mathbf{V}(\boldsymbol{\theta}; \mathbf{u}), \mathbf{R})$, where $\mathbf{R} = \sigma^2 \mathbf{I}$ and

$$\mathbf{V}(\boldsymbol{\theta}; \mathbf{u}) = \begin{bmatrix} V(\boldsymbol{\theta}; u(t_1)) & \cdots & V(\boldsymbol{\theta}; u(t_N)) \end{bmatrix}^\top.$$

Then,

$$p(\mathbf{Z}|\boldsymbol{\theta}) \cdot p(\boldsymbol{\theta}) \propto \exp\left(-\frac{1}{2} [\mathbf{z} - \mathbf{V}(\boldsymbol{\theta}; \mathbf{u})]^\top \mathbf{R}^{-1} [\mathbf{z} - \mathbf{V}(\boldsymbol{\theta}; \mathbf{u})]\right) \cdot \exp\left(-\frac{1}{2} (\boldsymbol{\theta} - \mathbf{m})^\top \mathbf{P}^{-1} (\boldsymbol{\theta} - \mathbf{m})\right).$$

If using the log-likelihood, the problem in (2.15) is equivalent to

$$\hat{\boldsymbol{\theta}} = \arg \min_{\boldsymbol{\theta}} J(\boldsymbol{\theta}), \quad (2.16)$$

where

$$J(\boldsymbol{\theta}) = \frac{1}{2} [\mathbf{z} - \mathbf{V}(\boldsymbol{\theta}; \mathbf{u})]^\top \mathbf{R}^{-1} [\mathbf{z} - \mathbf{V}(\boldsymbol{\theta}; \mathbf{u})] + \frac{1}{2} (\boldsymbol{\theta} - \mathbf{m})^\top \mathbf{P}^{-1} (\boldsymbol{\theta} - \mathbf{m}).$$

For the nonlinear optimization problem in (2.16), one can exploit the quasi-Newton method to numerically solve it [80]. This method iteratively updates the parameter estimate through

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \lambda_k \mathbf{s}_k. \quad (2.17)$$

Here, λ_k denotes the step size at iteration step k , and \mathbf{s}_k is the gradient-based search direction given by

$$\mathbf{s}_k = -\mathbf{B}_k \mathbf{g}_k, \quad (2.18)$$

where $\mathbf{B}_k \in \mathbb{R}^{10 \times 10}$ is a positive definite matrix that approximates the inverse of the Hessian matrix $\nabla^2 J(\boldsymbol{\theta}_k)$, and $\mathbf{g}_k = \nabla J(\boldsymbol{\theta}_k) \in \mathbb{R}^{10 \times 1}$. Based on the well-known BFGS update strategy [201], \mathbf{B}_k can be updated by

$$\mathbf{B}_k = \left(\mathbf{I} - \frac{\boldsymbol{\delta}_k \boldsymbol{\gamma}_k^\top}{\boldsymbol{\delta}_k^\top \boldsymbol{\gamma}_k} \right) \mathbf{B}_{k-1} \left(\mathbf{I} - \frac{\boldsymbol{\gamma}_k \boldsymbol{\delta}_k^\top}{\boldsymbol{\delta}_k^\top \boldsymbol{\gamma}_k} \right) + \frac{\boldsymbol{\delta}_k \boldsymbol{\delta}_k^\top}{\boldsymbol{\delta}_k^\top \boldsymbol{\gamma}_k}, \quad (2.19)$$

with $\boldsymbol{\delta}_k = \boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-1}$ and $\boldsymbol{\gamma}_k = \mathbf{g}_k - \mathbf{g}_{k-1}$. In addition,

$$\begin{aligned} \mathbf{g}_k = & - \left(\frac{\partial \mathbf{V}(\boldsymbol{\theta}_k; \mathbf{u})}{\partial \boldsymbol{\theta}_k} \right)^\top \mathbf{R}^{-1} [\mathbf{z} - \mathbf{V}(\boldsymbol{\theta}_k; \mathbf{u})] \\ & + \mathbf{P}^{-1} (\boldsymbol{\theta}_k - \mathbf{m}), \end{aligned} \quad (2.20)$$

where each column of $\frac{\partial \mathbf{V}(\boldsymbol{\theta}; \mathbf{u})}{\partial \boldsymbol{\theta}} \in \mathbb{R}^{N \times 10}$ is given by

$$\begin{aligned}\frac{\partial \mathbf{V}(\boldsymbol{\theta}; \mathbf{u})}{\partial \theta_i} &= \mathbf{x}^{\circ i} - \mathbf{x}^{\circ 5} \text{ for } i = 1, 2, \dots, 4, \\ \frac{\partial \mathbf{V}(\boldsymbol{\theta}; \mathbf{u})}{\partial \theta_5} &= \boldsymbol{\Sigma} \circ \frac{q^{-1} - \theta_7 q^{-2}}{1 - (1 + \theta_7)q^{-1} + \theta_7 q^{-2}} \mathbf{u}, \\ \frac{\partial \mathbf{V}(\boldsymbol{\theta}; \mathbf{u})}{\partial \theta_6} &= \boldsymbol{\Sigma} \circ \frac{q^{-1} - q^{-2}}{1 - (1 + \theta_7)q^{-1} + \theta_7 q^{-2}} \mathbf{u}, \\ \frac{\partial \mathbf{V}(\boldsymbol{\theta}; \mathbf{u})}{\partial \theta_7} &= \boldsymbol{\Sigma} \circ \frac{\theta_6 q^{-2} - 2\theta_6 q^{-3} + \theta_6 q^{-4}}{(1 - (1 + \theta_7)q^{-1} + \theta_7 q^{-2})^2} \mathbf{u}, \\ \frac{\partial \mathbf{V}(\boldsymbol{\theta}; \mathbf{u})}{\partial \theta_8} &= \frac{-q^{-1}}{1 + \theta_9 q^{-1}} \mathbf{u}, \\ \frac{\partial \mathbf{V}(\boldsymbol{\theta}; \mathbf{u})}{\partial \theta_9} &= \frac{\theta_8 q^{-2}}{1 + 2\theta_9 q^{-1} + \theta_9^2 q^{-2}} \mathbf{u}, \\ \frac{\partial \mathbf{V}(\boldsymbol{\theta}; \mathbf{u})}{\partial \theta_{10}} &= \mathbf{u},\end{aligned}$$

with

$$\begin{aligned}\mathbf{x} &= G_1(q, \boldsymbol{\theta}) \mathbf{u} + G_2(q) V_s(0) \mathbf{1}, \\ \boldsymbol{\Sigma} &= \sum_{i=1}^4 i \theta_i \mathbf{x}^{\circ(i-1)} + 5 \left(\bar{V} - \underline{V} - \sum_{i=1}^4 \theta_i \right) \mathbf{x}^{\circ 4}.\end{aligned}$$

Here, $\mathbf{x} \circ \mathbf{u}$ denotes the Hadamard product of \mathbf{x} and \mathbf{u} , $\mathbf{x}^{\circ 2}$ denotes the Hadamard power with $\mathbf{x}^{\circ 2} = \mathbf{x} \circ \mathbf{x}$, and $\mathbf{1} \in \mathbb{R}^{N \times 1}$ denotes a column vector with all elements equal to one.

Finally, note that λ_k needs to be chosen carefully to make $J(\boldsymbol{\theta})$ decrease monotonically. One can use the Wolfe conditions and let λ_k be selected such that

$$J(\boldsymbol{\theta}_k + \lambda_k \mathbf{s}_k) \leq J(\boldsymbol{\theta}_k) + c_1 \lambda_k \mathbf{g}_k^\top \mathbf{s}_k, \quad (2.21a)$$

$$\nabla J(\boldsymbol{\theta}_k + \lambda_k \mathbf{s}_k)^\top \mathbf{s}_k \geq c_2 \nabla J(\boldsymbol{\theta}_k)^\top \mathbf{s}_k, \quad (2.21b)$$

with $0 < c_1 < c_2 < 1$. For the quasi-Newton method, c_1 is usually set to be quite small, e.g., $c_1 = 10^{-6}$, and c_2 is typically set to be 0.9. The selection of λ_k can be based on trial and error in implementation. One can start with picking a number and check the Wolfe conditions. If the conditions are not satisfied, reduce the number and check again. An interested reader is referred

Table 2.1: Quasi-Newton-based implementation for MAP-based Wiener identification.

```

Initialize  $\boldsymbol{\theta}_0$  and set the convergence tolerance
repeat
  Compute  $\mathbf{g}_k$  via (2.20)
  if  $k = 0$  then
    Initialize  $\mathbf{B}_0 = 0.001 \frac{1}{\|\mathbf{g}_0\|} \mathbf{I}$ 
  else
    Compute  $\mathbf{B}_k$  via (2.19)
  end if
  Compute  $\mathbf{s}_k$  via (2.18)
  Find  $\lambda_k$  that satisfies the Wolfe conditions (2.21)
  Perform the update via (2.17)
until  $J(\boldsymbol{\theta}_k)$  converges
return  $\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}_k$ 

```

to [201] for detailed discussion about the λ_k selection. Summarizing the above, Table 2.1 outlines the implementation procedure for the MAP-based Wiener identification.

Remark 2.1. While the MAP estimation has enjoyed a long history of addressing a variety of estimation problems, no study has been reported about its application to Wiener system identification to our knowledge. Here, it is found to be a very useful approach for providing physically reasonable parameter estimation for practical systems, as it takes into account some prior knowledge about the unknown parameters. In a Gaussian setting as adopted here, the prior $p(\boldsymbol{\theta})$ translates into a regularization term in $J(\boldsymbol{\theta})$, which prevents incorrect fitting and enhances the robustness of the numerical optimization against nonconvexity.

Remark 2.2. The proposed 2.0 identification approach requires some prior knowledge of the parameters to be available, which can be developed in several ways in practice. First, R_0 can be

roughly estimated using the voltage drop at the beginning of the discharge, to which it is a main contributor. Second, the polynomial coefficients of $h(\cdot)$ can be approximately obtained from an experimentally calibrated SoC-OCV curve if there is any. Third, one can derive a rough range for $C_b + C_s$ if a battery's capacity is approximately known. Finally, as the parameters of batteries of the same kind and brand are usually close, one can take the parameter estimates acquired from one battery as prior knowledge for another.

Remark 2.3. In general, a prerequisite for successful identification is that the parameters must be identifiable in a certain sense. Following along similar lines as in [60, 192], one can rigorously define the parameters' local identifiability for the considered Wiener identification problem and find out that a sufficient condition for it to hold is the full rankness of the sensitivity matrix $\partial \mathbf{V}(\boldsymbol{\theta}; \mathbf{u}) / \partial \boldsymbol{\theta}$, which can be used for identifiability testing. Using this idea, our simulations consistently showed the full rankness of the sensitivity matrix under variable current profiles such as those in Figure 2.8, indicating that the NDC model can be locally identifiable. Related with identification is optimal input design, which concerns designing the best current profile to maximize the parameter identifiability [164, 146]. It will be part of our future research to explore this interesting problem for the NDC model.

Remark 2.4. It is worth mentioning that the 2.0 identification approach can be readily extended to identify some other ECMs that have a Wiener-like structure like the Rint and Thevenin models. One can follow similar lines to develop the computational procedures for each, and hence the details are skipped here.

Remark 2.5. The 1.0 and 2.0 identification approaches are designed to perform offline identification for the NDC model, each with its own advantages. The 1.0 approach is designed for in-lab battery modeling and analysis, using simple two-step (trickle- and constant-current discharging) battery testing protocols. While requiring a long time for experiments, it can offer high accuracy in parameter estimation. More sophisticated by design, the 2.0 approach can extract the parameters all at once from data based on variable current profiles. It can be conveniently exploited to determine the NDC model for batteries operating in real-world applications.



Figure 2.4: PEC® SBT4050 battery tester.

2.4 Experimental Validation

This section presents experimental validation of the proposed NDC model and parameter identification 1.0 and 2.0 approaches. All the experiments in this section were conducted on a PEC® SBT4050 battery tester (see Figure 2.4). The tester can support charging/discharging with arbitrary current-, voltage- and power-based loads (up to 40 V and 50 A). A specialized server is used to prepare and configure a test offline and collect experimental data online *via* the associated software LifeTest™. Using this facility, charging/discharging tests were performed to generate data on a Panasonic NCR18650B lithium-ion battery cell, which was set to operate between 3.2 V (fully discharged) and 4.2 V (fully charged).

2.4.1 Validation Based on Approach 1.0

This validation first extracts the NDC model from training dataset using the 1.0 identification approach in Section 2.3.1 and then applies the identified model to validation datasets to assess its predictive capability.

As a first step, the cell was fully charged and relaxed for a long time period. Then, a full discharge test was applied to the cell using a trickle constant current of 0.1 A (about 1/30 C-rate). With this test, the total capacity is determined to be $Q_t = 3.06$ Ah by coulomb counting, implying

Table 2.2: Identification 1.0: initial guess, bound limits and identification results.

Name	β_2	β_3	β_4	β_5	γ_1	γ_2	γ_3	γ_4	γ_5
Initial guess	0.02	0.05	0.005	1/100	0.05	0.2	8	0.07	12
$\underline{\theta}$	0.005	0.005	0.001	1/800	0.01	0.05	1	0.01	1
$\overline{\theta}$	0.2	0.2	0.03	1/10	0.09	0.35	15	0.12	15
$\hat{\theta}$	0.0163	0.0575	0.02	1/65	0.0531	0.1077	3.807	0.0533	7.613

Note: quantities are given in SI standard units in Tables 2.2 and 2.3.

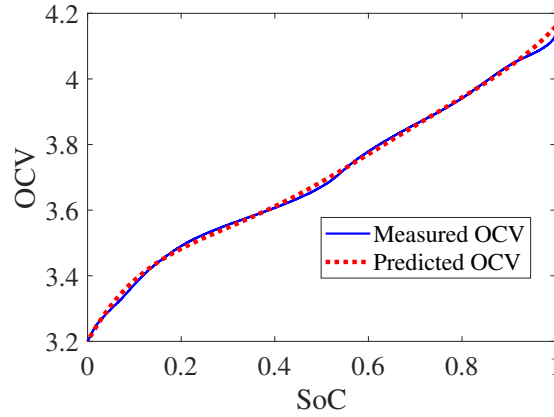


Figure 2.5: Identification 1.0: parameter identification of $h(\cdot)$ that defines SoC-OCV relation.

$C_b + C_s = 11,011$ F. Further, from the SoC-OCV curve fitting, we obtain

$$\text{OCV} = 3.2 + 2.59 \cdot \text{SoC} - 9.003 \cdot \text{SoC}^2 + 18.87 \cdot \text{SoC}^3 - 17.82 \cdot \text{SoC}^4 + 6.325 \cdot \text{SoC}^5,$$

which establishes $h(\cdot)$ immediately. The measured and identified SoC-OCV curves are compared in Figure 2.5. Next, the cell was fully charged again and left idling for a long time. This was then followed by a full discharge using a constant current of 3 A to produce data for estimation of the impedance and capacitance parameters. The identification was achieved by solving the constrained optimization problem in (2.9). The computation took around 1 sec, performed on a Dell Precision Tower 3620 equipped with 3 GHz Inter Xeon CPU, 16 Gb RAM and MATLAB R2018b. Table 2.2 summarizes the initial guess, lower and upper bounds, and obtained estimates of the parameters.

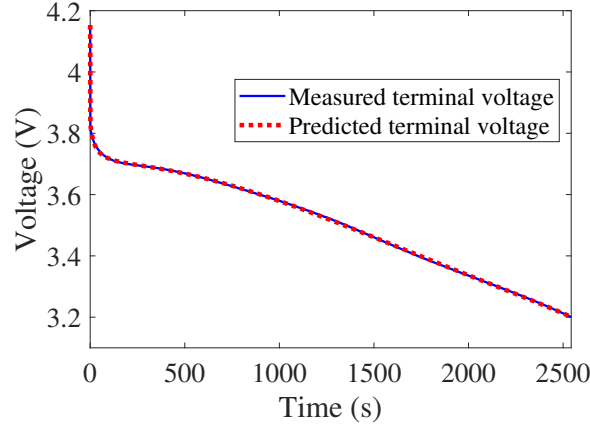


Figure 2.6: Identification 1.0: model fitting with the training data obtained under 3 A constant-current discharging.

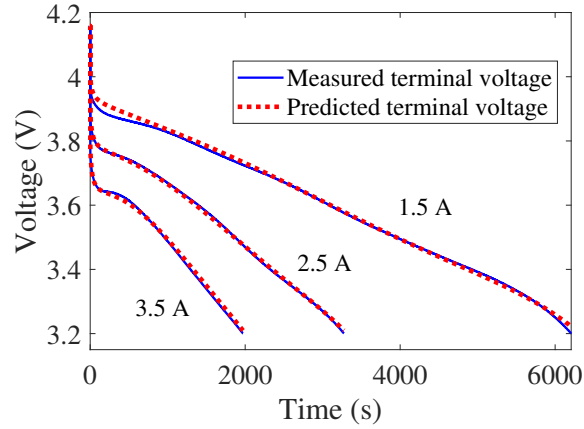


Figure 2.7: Identification 1.0: predictive fitting over validation data obtained by discharging at different constant currents.

The physical parameter estimates are extracted as: $C_b = 10,037$ F, $C_s = 973$ F, $R_b = 0.019$ Ω , $R_s = 0$, $R_1 = 0.02$ Ω , $C_1 = 3,250$ F, and

$$R_0 = 0.0531 + 0.1077e^{-3.807 \cdot \text{SoC}} + 0.0533e^{-7.613 \cdot (1 - \text{SoC})}.$$

The model is now fully available from the two steps. Figure 2.6 shows that it accurately fits with the measurement data.

While an identified model generally can well fit a training dataset, it is more meaningful and revealing to examine its predictive performance on some different datasets. Hence, five more tests were conducted by discharging the cell using constant currents of 1.5 A, 2.5 A and 3.5 A and

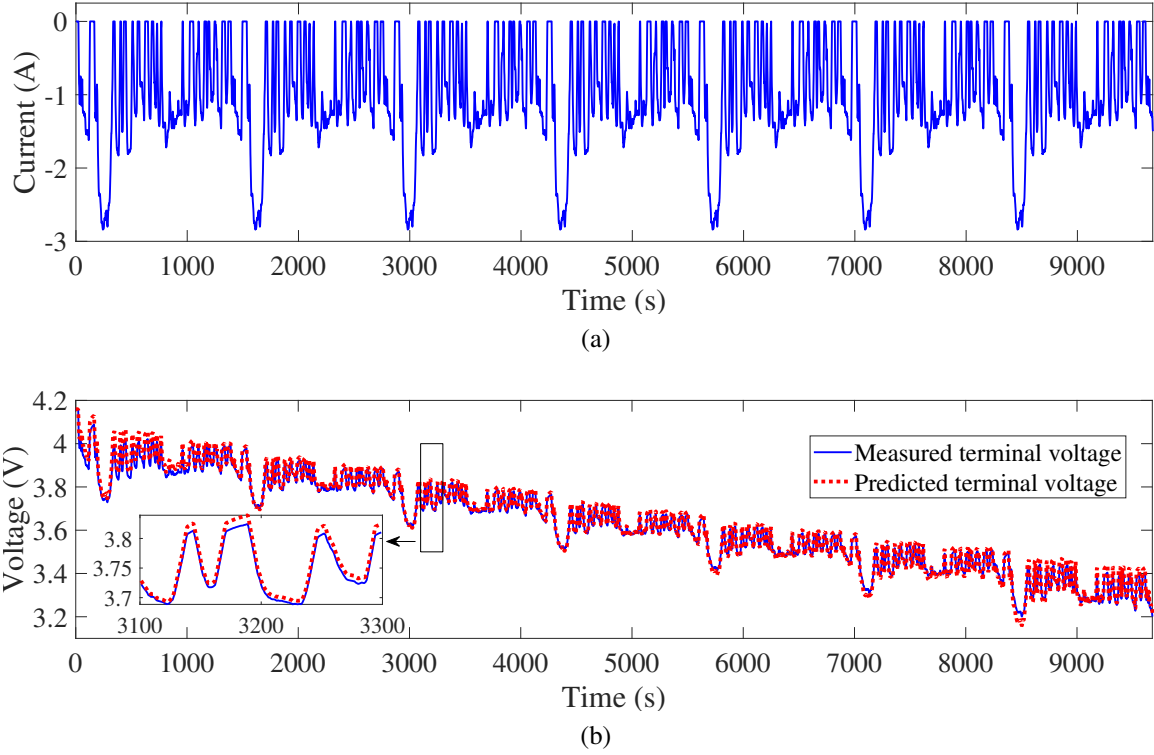


Figure 2.8: Identification 1.0: predictive fitting over validation dataset obtained by discharging at varying currents (0~3 A). (a) Current profile. (b) Voltage fitting.

two variable current profiles, respectively. Figure 2.7 shows what the identified model predicts for discharging at constant currents. An overall high accuracy is observed, even though the prediction is slightly less accurate when the current is 1.5 A, probably because the parameters are current-dependent to a certain extent. The variable current profiles are portrayed in Figures 2.8a and 2.9a, which were created by scaling the Urban Dynamometer Driving Schedule (UDDS) profile in [3] to span the ranges of 0~3 A and 0~6 A, respectively. Figures 2.8b and 2.9b present the predictive fitting results. Both of them illustrate that the model-based voltage prediction is quite close to the actual measurements. These results demonstrate the excellent predictive capability of the NDC model.

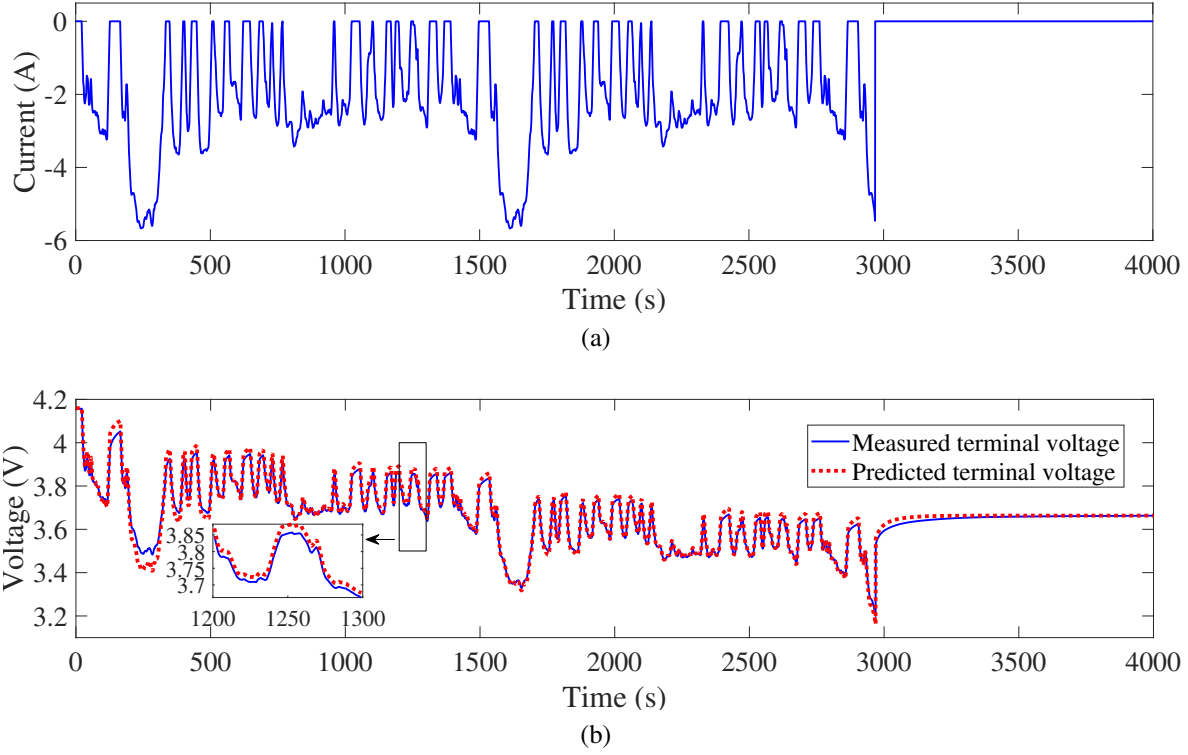


Figure 2.9: Identification 1.0: predictive fitting over validation dataset obtained by discharging at varying currents (0 ~ 6 A). (a) Current profile. (b) Voltage fitting.

2.4.2 Validation Based on Approach 2.0

Let us now consider the 2.0 approach developed in Section 2.3.1, which treats the NDC model as a Wiener-type system and performs MAP-based parameter estimation. This approach advantageously allows all the parameters to be estimated in a convenient one-shot procedure.

Following the manner in Section 2.4.1, one can apply the 2.0 approach to a training dataset to extract an NDC model and then use it to predict the responses over several other different datasets. The validation here is also set to evaluate the NDC model against the Rint model [83] and the Thevenin model with one serial RC circuit [83], which are commonly used in the literature. The comparison also extends to a basic version of the NDC model (referred to as “basic NDC” in sequel), one with a constant R_0 and without R_1 - C_1 circuit, with the purpose of examining the utility of the NDC model when it is reduced to a simpler form. Note that, even though the NDC model is the most sophisticated among them, all of the four models offer high computational efficiency by

Table 2.3: Identification 2.0: initial guess, prior knowledge and identification results.

Name	α_1	α_2	α_3	α_4	$\check{\beta}_1/10^{-5}$	$\check{\beta}_2/10^{-4}$	$\check{\beta}_3$	$\beta_4/10^{-4}$	β_5	R_0
Initial Guess	2.59	-9.003	18.87	-17.82	9.078	8.914	0.964	-4.938	-0.9753	0.08
\mathbf{m}	-	-	-	-	9.078	8.914	0.964	-4.938	-0.9753	0.08
$\sqrt{\text{diag}(\mathbf{P})}/m_i$	-	-	-	-	0.001	0.15	0.15	0.15	0.15	0.15
$\hat{\boldsymbol{\theta}}$	2.32	-8.15	19.345	-20.78	9.082	9.227	0.982	-4.859	-0.8153	0.069

requiring only a small number of arithmetic operations.

These four models are all Wiener-type, so the 2.0 identification approach can be used to identify them on the same training dataset, i.e., the one shown in Figure 2.8, thus ensuring a fair comparison. The parameter setting for the NDC model identification and the estimation result are summarized in Table 2.3. The computation took around 4 sec. The resultant physical parameter estimates are given by: $C_b = 10,031$ F, $C_s = 979$ F, $R_b = 0.063$ Ω , $R_s = 0$, $R_1 = 0.003$ Ω , $C_1 = 2,449$ F and $R_0 = 0.069$ Ω . The identification results for the Rint, Thevenin model and basic NDC models are omitted here for the sake of space.

Figure 2.10a depicts how the identified models fit with the training dataset. One can observe that the NDC model and its basic version show excellent fitting accuracy, overall better than the Rint and Thevenin models. A more detailed comparison is given in Figure 2.10b, which displays the fitting error in percentage. It is seen that the Rint model shows the least accuracy, followed by the Thevenin model. The NDC model and its basic version well outperform them, with the NDC model performing slightly better.

Proceeding forward, let us investigate the predictive performance of the four models over several validation datasets. First, consider the datasets obtained by constant-current discharging at 1.5 A, 2.5 A and 3.5 A, as illustrated in Figure 2.7. Figure 2.11 demonstrates that the NDC model and its basic version can predict the voltage responses under different currents much more accurately than the Rint and Thevenin models. Next, consider the dataset in Figure 2.9 based on variable-current discharging. Figure 2.12 shows that the prediction accuracy of all the models is lower than

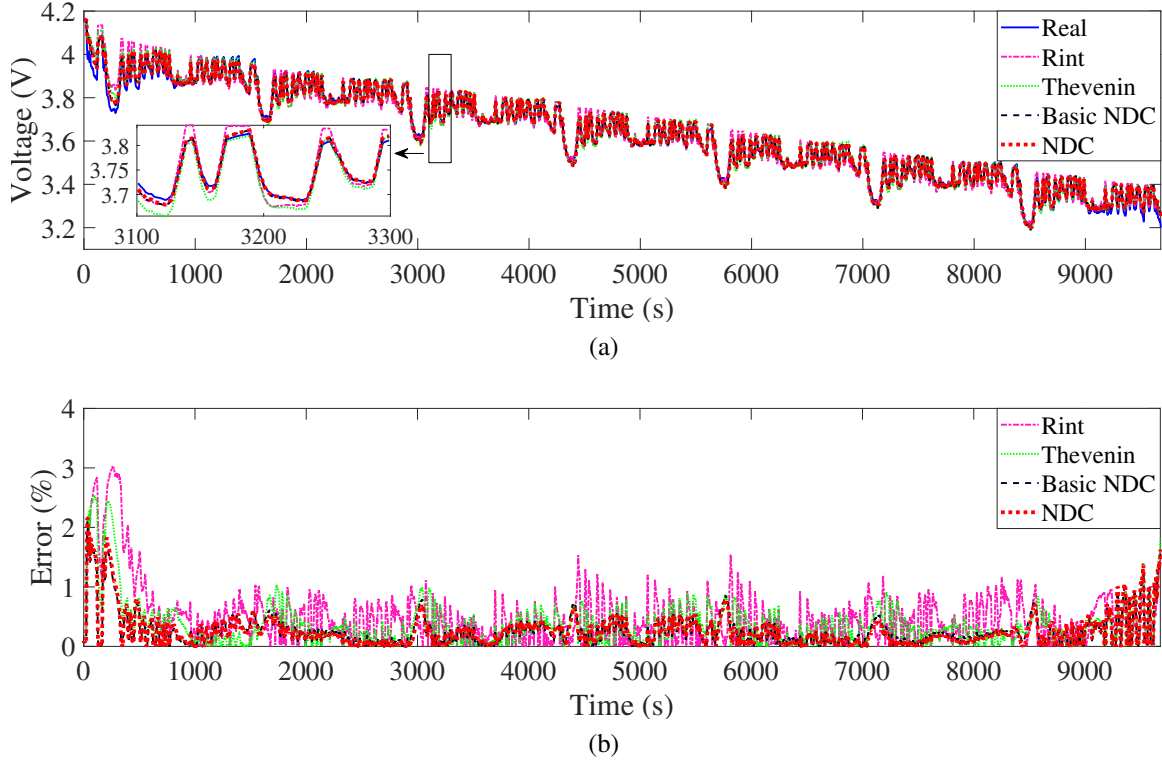


Figure 2.10: Identification 2.0: (a) model fitting with training dataset and (b) fitting error in percentage.

the fitting accuracy, which is understandable. However, the NDC model and its basic version are still again the most capable of predicting, with the error mostly lying below 1%. As a contrast, while the Thevenin model can offer a decent fit with the training dataset as shown in Figure 2.10, its prediction accuracy over the validation dataset is not as satisfactory. This implies that it is less predictive than the NDC model.

Another evaluation of interest is about the SoC-OCV relation. As mentioned earlier, the 2.0 approach can estimate all the parameters, including the function $h(\cdot)$. This allows one to write the SoC-OCV function directly based on the identified $h(\cdot)$ as it also characterizes the SoC-OCV relation. That is,

$$\text{OCV} = 3.2 + 2.32 \cdot \text{SoC} - 8.15 \cdot \text{SoC}^2 + 19.345 \cdot \text{SoC}^3 - 20.78 \cdot \text{SoC}^4 + 8.222 \cdot \text{SoC}^5.$$

Identification of the other three models can also lead to estimation of this function. Figure 2.13

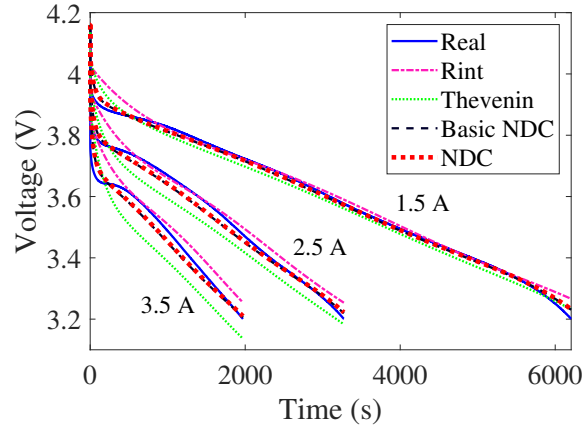


Figure 2.11: Identification 2.0: predictive fitting over validation datasets obtained by discharging at different varying currents.

compares them with the benchmark shown in Figure 2.5, which is obtained experimentally by discharging the cell using a small current of 0.1 A. It is obvious that the SoC-OCV curves obtained in the identification of the NDC model and its basic version are closer to the benchmark overall. This further shows the benefit of the NDC model as well as the efficacy of the 2.0 approach.

Summing up the above validation results, one can draw the following observations:

- The NDC model is the most competent among the four considered models for grasping and predicting a battery's dynamic behavior, justifying its validity and soundness.
- The basic NDC model can offer fitting and prediction accuracy almost comparable to that of the full model. It thus can be well qualified if a practitioner wants to use a simpler NDC model yet without much loss of accuracy.
- The 2.0 identification approach is effective in estimating all the parameters of the NDC model as well as the Rint and Thevenin models in one shot from variable-current-based data profiles. It can not only ease the cost of identification considerably but also provide on-demand model availability potentially in practice.

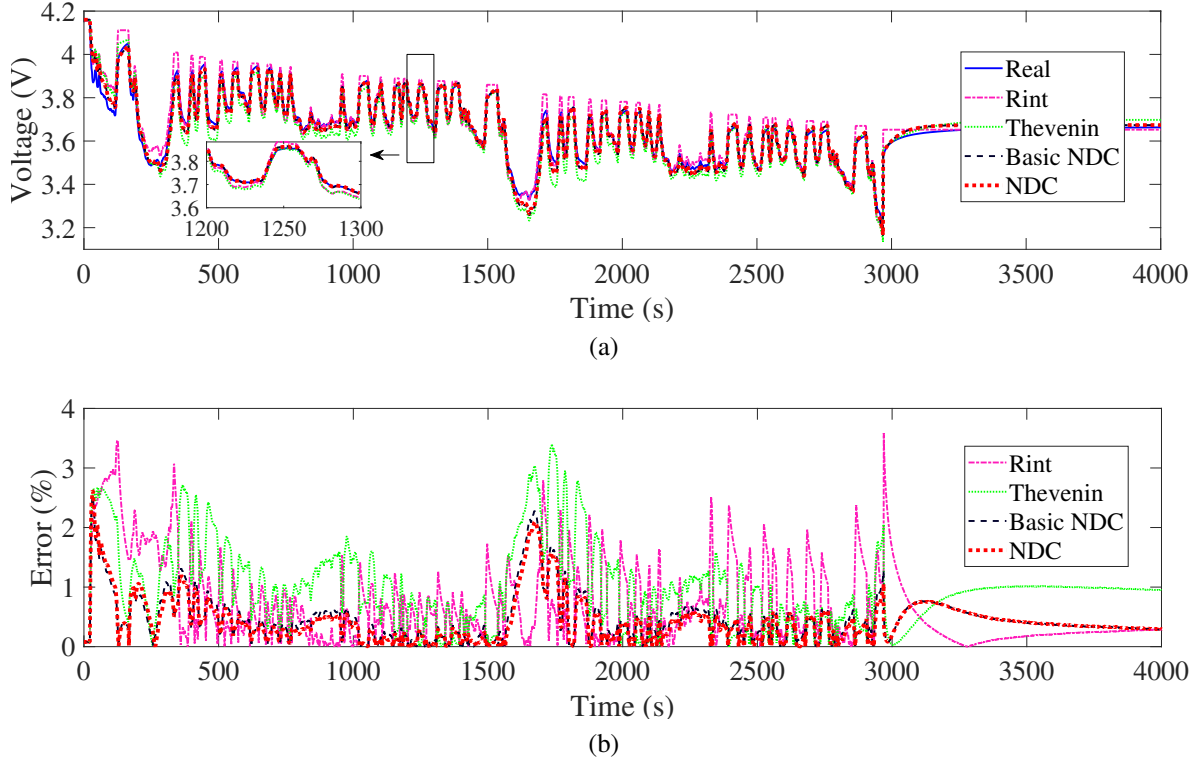


Figure 2.12: Identification 2.0: (a) predictive fitting over validation dataset obtained by discharging at varying currents between 0 A and 6 A and (b) predictive fitting error in percentage.

2.5 Summary

This chapter improved on the original double-capacitor model by adding a nonlinear-mapping-based voltage source and a serial RC circuit and proposed the NDC model. This development was justified through an analogous comparison with the SPM. Furthermore, two offline parameter estimation approaches, which were named 1.0 and 2.0, respectively, were designed to identify the model from current/voltage data. The 1.0 approach considered the constant-current charging/discharging scenarios, determining the SoC-OCV relationship first and then estimating the impedance and capacitance parameters. With the observation that the NDC model has a Wiener-type structure, the 2.0 approach was derived from the Wiener perspective. As the first of its kind, it leverages the notion of MAP to address the issue of local minima that may reduce or damage the performance of the nonlinear Wiener system identification. It well lends itself to the variable-current charging/discharging scenarios and can desirably estimate all the parameters in one shot. The experimental

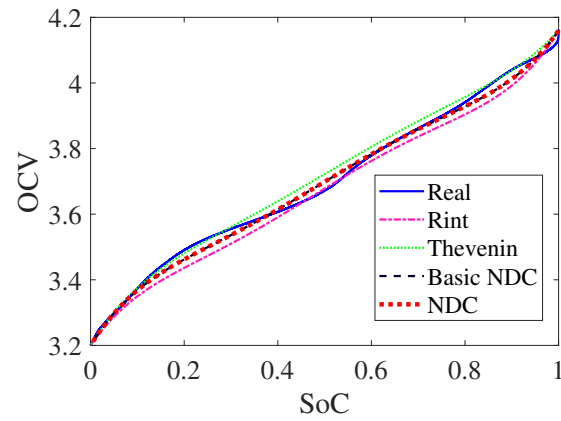


Figure 2.13: Identification 2.0: identification of the SoC-OCV relation based on different models, compared to the truth.

evaluation demonstrated that the NDC model outperformed the popularly used Rint and Thevenin models in predicting a battery's behavior, in addition to showing the effectiveness of the identification approaches for extracting parameters.

Chapter 3

One-Shot Parameter Identification of the Thevenin Model

This chapter¹ studies one-shot identification of all parameters in the Thevenin model, including the resistance and capacitance parameters and the parameters in the parameterized function mapping from state of charge to open-circuit voltage. It begins with the parameter identifiability analysis, showing that all the parameters are locally identifiable. Then, it formulates the parameter identification problem in a prediction-error-minimization framework. As the non-convexity intrinsic to the problem may lead to physically meaningless estimates, two methods are developed to overcome this issue. The first one is to constrain the parameter search within a reasonable space by setting parameter bounds, and the other adopts regularization of the cost function using prior parameter guess. The proposed identifiability analysis and identification methods are extensively validated through simulations and experiments.

3.1 Introduction

Rechargeable batteries have found wide use nowadays in the consumer electronics, transportation and grid sectors by the billions. Their ever-widening use has excited an intense interest in advanced battery management system research. The main subjects of inquiry in this area include state of charge (SoC) and state of health (SoH) estimation, optimal charging control, cell balancing, thermal management, e.g., [41, 114, 150, 197, 119, 59, 56, 149, 127, 143, 122, 183], and the references therein. Playing a foundational role in many of the existing studies are the equivalent circuit models (ECMs), which replicate a battery's electrical dynamics using a circuit composed of resistors,

¹This chapter is based on the dissertation author's first-authored journal paper [187].

capacitors and voltage sources. However, the parameters of an ECM are often unknown in many real-world circumstances or drift with cycled charging/discharging. This presents the question of how to accurately and efficiently identify them from the current/voltage measurements made on a battery cell.

Literature Review

The current literature on ECM identification can be roughly divided into three categories:

- **Experiment-based analysis.** This category develops and conducts charging/discharging experiments such that a battery's dynamics can be exposed and used to estimate parameters. For instance, the transient voltage responses are leveraged in [52, 166] to identify the internal resistance and resistance-capacitance-based (RC-based) time constants by charging or discharging a battery using constant or pulse currents. Another example is the experimental determination of the relationship between SoC and open-circuit voltage (OCV). Conventionally, it can be accomplished by charging or discharging the battery using a very small current [154, 199, 206], or alternatively, applying a current of normal magnitude intermittently (a sufficiently long rest period is applied between two discharging operations) [85, 188, 151]. While easy to implement, these approaches introduce significant time costs—an SoC-OCV calibration experiment can take more than one day [154, 52], unaffordable especially in massive battery testing. One can also find studies about the design of specialized charging/discharging protocols to expedite parameter identification, e.g. [10], which, however, would still take more than ten hours.
- **Electrochemical impedance spectroscopy (EIS).** EIS is an important means of observing electrochemical processes within batteries. The EIS data reveal the impedance properties of a battery, and the literature includes a few methods that fit an ECM to collected EIS data to extract the resistance and RC parameters [32, 141, 73, 14]. These methods focus on impedance identification as needed in many applications and meanwhile, leave other parts

of a battery's dynamics such as the SoC-OCV function beyond consideration.

- **Analytical data-based parameter estimation.** This category seeks to determine an ECM's unknown parameters using the current/voltage measurement data from a system identification perspective. The studies in [36, 210, 20, 111, 87, 68, 219, 105] consider pulse current charging/discharging experiments and use the data to identify an ECM's RC parameters by fitting the model to the pulse phase voltage [36, 210], the relaxation phase voltage [20, 111, 87, 68], or both of them [219, 105]. Some other studies perform similar data-fitting-based RC parameter estimation using data from variable current charging/discharging tests, while exploiting generic optimization [54], grey-box optimization [28], sequential quadratic programming [115], or particle swarm optimization [94, 213] as the solution tools. The studies in [75, 91, 46, 93] propose to estimate the RC parameters by converting the identification problem into a problem of solving a set of linear and polynomial equations. In [95, 117], a linear state-space model is formulated for batteries, and subspace identification is then performed to infer the system matrices. An approach presented in [204] directly identifies the parameters of a continuous-time ECM using sampled discrete-time measurements. Associated with parameter identification, there is a growing amount of work on combined estimation of SoC and part of the model parameters [40, 172, 179, 60, 84, 158, 64, 62, 212, 214]. It is noted that these studies usually require an accurate SoC-OCV relationship to be available prior to identification, which requires long-time testing as aforementioned.

Despite the importance, the methods surveyed above, however, share one limitation: they are designed to identify only a subset of an ECM's parameters, on the premise that the other parameters are known. This brings about an intriguing question: *Is it possible to extract the parameters of an ECM all at once?* Here, by "all", it means the RC parameters as well as the parameters of the nonlinear SoC-OCV function. At least two benefits can result if this can be achieved. First, it will enhance the efficiency of battery model identification considerably by avoiding the tedious SoC-OCV calibration. Second, it can help ensure the availability of an accurate model for battery management during a battery's service life. However, one-shot identification for batteries has been

long known as a challenge, because of the increase in the number of parameters to estimate and the serious nonlinearity and non-convexity in the identification procedure. The only studies on this topic, to our knowledge, are given in [96, 97, 31, 136], which use the genetic algorithms to search for the best parameter estimates for a battery model. However, their application still face some limitations. The primary one among them is the weak parameter identifiability due to the many parameters (more than 30) to determine, which can potentially compromise the estimation accuracy. In addition, the genetic algorithms generally converge slowly and impose high computational expenses.

Contributions

This work is motivated to develop new and efficient approaches for one-shot battery parameter identification. The Thevenin model, which has been a popular choice for battery management [162, 102, 156, 168], is considered here. This model is nonlinear by nature, and consequently, an identification effort can suffer pitfalls caused by the nonlinearity and non-convexity, which may eventually produce inaccurate or unphysical parameter estimation. This work hence presents a systematic study to overcome such an issue, yielding the following contributions.

- The study for the first time reveals the feasibility of the one-shot identification for the Thevenin model through an in-depth parameter identifiability analysis. Specifically, it shows that all the parameters are locally identifiable, indicating that they can be uniquely determined in a local domain.
- With the model's local identifiability guaranteed, this work synthesizes novel one-shot parameter identification methods. The methods are developed to minimize the model prediction error through numerical optimization. Different from the literature, we introduce two critical mechanisms to address the non-convexity issue: 1) constrained optimization, which constrains the search space by applying upper and lower bounds to some parameters, and 2) generalized Tikhonov regularization, which adds a regularization term to the considered cost

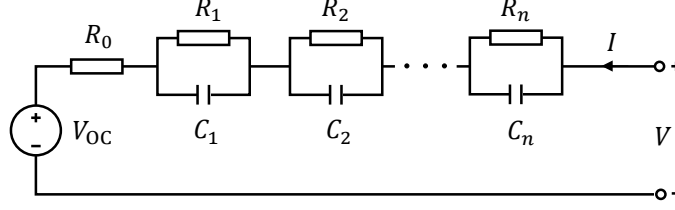


Figure 3.1: The Thevenin model.

function to drive the optimization toward a reasonable minimum point.

- The theoretical estimation accuracy of the proposed identification methods is rigorously characterized. The approaches and the associated analysis are validated using extensive simulations and experiments.

With the above contributions, our study can enable easier availability of accurate Thevenin models for advanced battery management ranging from charging control to SoC estimation and aging prognostics.

Organization

The rest of the chapter is organized as follows. Section 3.2 reviews the Thevenin model and sets up the parameter identification problem. Section 3.3 investigates the parameter identifiability and develops the identification approaches. Section 3.4 evaluates the efficacy of the proposed results through Monte Carlo simulations. Section 3.5 further presents validation based on experiments. Finally, some concluding remarks are gathered in Section 3.6.

3.2 Model Description

This section first introduces the Thevenin model and then derives the voltage response under constant-current discharging. The formulation of the parameter identification problem then follows.

A general form of the Thevenin model is shown in Figure 3.1. The first main component of the model is a voltage source, which emulates a battery's OCV. The OCV is SoC-dependent, and the

SoC's dynamics is given by

$$\dot{\text{SoC}} = \frac{1}{3600Q_f}I, \quad (3.1)$$

where Q_f is the battery's maximum capacity in ampere-hour (Ah) and I is the current in ampere ($I < 0$ for discharging and $I > 0$ for charging). The second component of this model, R_0 , represents the battery's internal resistance, which grasps the voltage drop in discharging and jump in charging. As the third component, a set of serially connected RC pairs can characterize the transient behavior in the battery's voltage response. This work considers only one RC pair, which is often sufficient practically and brings simplicity of analysis and computation. It is noteworthy that this will cause no loss of generality as the ensuing results can be extended to the case of multiple RC pairs.

By the Kirchhoff's circuit laws, the voltage dynamics based on the one-RC Thevenin model can be expressed as

$$\begin{cases} \dot{V}_{\text{RC}} = -\frac{1}{RC}V_{\text{RC}} - \frac{1}{C}I, \\ V = V_{\text{OC}} - V_{\text{RC}} + R_0I, \end{cases} \quad (3.2a)$$

$$(3.2b)$$

where V_{RC} is the voltage across the RC pair, V_{OC} the OCV, and V the terminal voltage. As aforementioned, V_{OC} is dependent on SoC. Following [178, 199], it can be characterized as a fifth-order polynomial with respect to SoC:

$$V_{\text{OC}}(\text{SoC}) = \sum_{i=0}^5 \alpha_i \text{SoC}^i, \quad (3.3)$$

where α_i for $i = 0, 1, \dots, 5$ are coefficients, and V_{OC} is lower and upper bounded, respectively, by $\underline{V}_{\text{OC}} = V_{\text{OC}}(\text{SoC} = 0)$ and $\bar{V}_{\text{OC}} = V_{\text{OC}}(\text{SoC} = 1)$. In addition, the literature often suggests the internal resistance R_0 as a function of SoC, which is in general constant but increases exponentially when SoC nears 0 [36, 134]. Hence, R_0 is parameterized as

$$R_0(\text{SoC}) = \beta_0 + \beta_1 e^{-\beta_2 \text{SoC}}, \quad (3.4)$$

where the coefficients $\beta_i > 0$ for $i = 0, 1, 2$.

From above, the Thevenin model with one RC pair is represented by (3.1)-(3.4). Now, suppose that the battery is fully charged, left idling for a long period and then discharged by a constant current to cut-off voltage. Here, the focus on a constant-current discharging scenario for the sake of formulating a mathematically tractable identification problem. As another advantage, such a discharging protocol can be easily implemented in practice. Solving (3.2a) with $V_{RC}(t = 0) = 0$, one can obtain the evolution of V_{RC} under constant current through time as follows:

$$V_{RC}(t) = -IR \left(1 - e^{-\frac{t}{RC}} \right). \quad (3.5)$$

As a result, the terminal voltage V through time is

$$\begin{aligned} V(t) = & \alpha_0 + \alpha_1 \text{SoC}(t) + \alpha_2 \text{SoC}^2(t) + \alpha_3 \text{SoC}^3(t) + \alpha_4 \text{SoC}^4(t) + \alpha_5 \text{SoC}^5(t) \\ & + I\beta_0 + I\beta_1 e^{-\beta_2 \text{SoC}(t)} + IR \left(1 - e^{-\frac{t}{RC}} \right), \end{aligned} \quad (3.6)$$

where $\text{SoC}(t) = 1 + It/(3600Q_f)$ as indicated by (3.1), with $\text{SoC}(0) = 1$.

It is worth noting that \underline{V}_{OC} and \bar{V}_{OC} can be preset and used in the discharging experiments. It is also assumed that the total capacity Q_c is known or can be experimentally determined for given \underline{V}_{OC} and \bar{V}_{OC} . Because $\alpha_0 = \underline{V}_{OC}$ and $\sum_{i=0}^5 \alpha_i = \bar{V}_{OC}$, only α_i for $i = 1, 2, \dots, 4$ need to be determined. It is seen that (3.6) defines an explicit relationship between V , I and the unknown parameters, i.e., α_i for $i = 1, \dots, 4$, β_i for $i = 0, 1, 2$, R and C . The next problem to tackle is two-fold: 1) deciding if the parameters are identifiable, and 2) designing approaches to achieve effective parameter identification, which will be the focus of the next section. Note that the parameters are assumed to be constant throughout this chapter in order to consider a representative and tractable problem.

3.3 Parameter Identification

Based on Section 3.2, this section aims to identify the unknown model parameters. It starts with investigating the parameter identifiability analysis, which shows that the parameters are locally identifiable and offers a method of determining the theoretical identification accuracy. Then, it synthesizes two identification methods, which are based on the notion of nonlinear least squares (NLS) but use constrained optimization and regularization, respectively, to address the non-convexity issue that arises in parameter estimation.

3.3.1 Identifiability Analysis

Consider (3.6), for which the unknown parameter vector is

$$\boldsymbol{\theta} = \left[\alpha_1 \quad \alpha_2 \quad \alpha_3 \quad \alpha_4 \quad \beta_0 \quad \beta_1 \quad \beta_2 \quad R \quad (RC)^{-1} \right]^\top.$$

Note that θ_i for $i = 1, 2, \dots, 9$ and its corresponding parameter will be used interchangeably in sequel. The terminal voltage V is measured at a sequence of sampling instants $\{t_k\}$ for $k = 1, 2, \dots, N$ with $t_1 = 0$. By (3.6), $V(t_k)$ can be compactly written as

$$V(t_k) = \phi(\boldsymbol{\theta}; t_k), \tag{3.7}$$

where

$$\begin{aligned} \phi(\boldsymbol{\theta}; t_k) = & \underline{V}_{OC} + \theta_1 \text{SoC}(t_k) + \theta_2 \text{SoC}^2(t_k) + \theta_3 \text{SoC}^3(t_k) + \theta_4 \text{SoC}^4(t_k) \\ & + \left(\bar{V}_{OC} - \underline{V}_{OC} - \sum_{i=1}^4 \theta_i \right) \text{SoC}^5(t_k) + \theta_5 I + \theta_6 I e^{-\theta_7 \text{SoC}(t_k)} + \theta_8 I \left(1 - e^{-\theta_9 t_k} \right). \end{aligned}$$

Given (3.7), a fundamental question is whether $\boldsymbol{\theta}$ can be uniquely identified from the data sequence $\{V(t_k)\}$, which boils down to the identifiability issue. To proceed, a definition of the local parameter identifiability is introduced for (3.7) as follows [60, 192].

Definition 3.1. A model structure $\phi(\boldsymbol{\theta}; t_k)$ is said to be locally identifiable at some point $\boldsymbol{\theta}^0$ in the parameter space for a given data sequence $\{V(t_1), V(t_2), \dots, V(t_N)\}$, if for any $\boldsymbol{\theta}^1$ and $\boldsymbol{\theta}^2$ within the neighborhood of $\boldsymbol{\theta}^0$, $\phi(\boldsymbol{\theta}^1; t_k) = \phi(\boldsymbol{\theta}^2; t_k)$ holds if and only if $\boldsymbol{\theta}^1 = \boldsymbol{\theta}^2$.

Definition 3.1 indicates that there should exist no two different parameter sets in the neighborhood of a point that can produce the same data sequence if the parameters are locally identifiable.

Before moving on to discuss the identifiability testing condition, let us consider the parameter identification problem for (3.7) in a prediction-error framework, seeking to find out the parameter estimates to minimize the difference between the measurements and the model-based predictions. To this end, an NLS cost function can be formulated as

$$J(\boldsymbol{\theta}) = \frac{1}{2} [\mathbf{y} - \boldsymbol{\phi}(\boldsymbol{\theta})]^\top \mathbf{Q}^{-1} [\mathbf{y} - \boldsymbol{\phi}(\boldsymbol{\theta})], \quad (3.8)$$

where \mathbf{Q} is a symmetric positive definite matrix interpretable as the covariance matrix of an additive measurement noise, and

$$\begin{aligned} \mathbf{y} &= \begin{bmatrix} V(t_1) & V(t_2) & \cdots & V(t_N) \end{bmatrix}^\top, \\ \boldsymbol{\phi}(\boldsymbol{\theta}) &= \begin{bmatrix} \phi(\boldsymbol{\theta}; t_1) & \phi(\boldsymbol{\theta}; t_2) & \cdots & \phi(\boldsymbol{\theta}; t_N) \end{bmatrix}^\top. \end{aligned}$$

The parameter estimation then lies in finding out $\hat{\boldsymbol{\theta}}$ to minimize $J(\boldsymbol{\theta})$, i.e.,

$$\hat{\boldsymbol{\theta}} = \arg \min_{\boldsymbol{\theta}} J(\boldsymbol{\theta}). \quad (3.9)$$

It is pointed out in [192] that evaluating the local identifiability of $\boldsymbol{\theta}$ around $\hat{\boldsymbol{\theta}}$ (see Definition 3.1) can be achieved by testing whether the minimization problem in (3.8) has a unique solution in the local parameter space. Further, a sufficient condition to guarantee a unique $\hat{\boldsymbol{\theta}}$ (or in other words, the local identifiability of $\boldsymbol{\theta}$) is that the Hessian matrix $\partial^2 J(\boldsymbol{\theta}) / \partial \boldsymbol{\theta}^2 > \mathbf{0}$ at $\hat{\boldsymbol{\theta}}$. To satisfy this condition, $\partial \boldsymbol{\phi}(\boldsymbol{\theta}) / \partial \boldsymbol{\theta}$ at $\hat{\boldsymbol{\theta}}$ must be of full rank. This is because the local first-order approximation

of $\partial^2 J(\boldsymbol{\theta})/\partial \boldsymbol{\theta}^2$ is

$$\frac{\partial^2 J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}^2} \approx \left(\frac{\partial \boldsymbol{\phi}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right)^\top \mathbf{Q}^{-1} \frac{\partial \boldsymbol{\phi}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}.$$

Here, define $\mathbf{S}(\boldsymbol{\theta}) = \partial \boldsymbol{\phi}(\boldsymbol{\theta})/\partial \boldsymbol{\theta}$. It is referred to as the sensitivity matrix and measures how the parameter perturbations can change the terminal voltage. The terminal voltage can be said to be sensitive to a parameter if it sees a relatively large change when the parameter varies slightly. A parameter with higher sensitivity generally allows for more accurate estimation. Meanwhile, $\mathbf{S}(\boldsymbol{\theta})$ should be of full rank as discussed above, so that when the parameters change, their effects on the terminal voltage are differentiable. Specifically, $\mathbf{S}(\boldsymbol{\theta})$ is given by

$$\mathbf{S}(\boldsymbol{\theta}) = \begin{bmatrix} \vdots & \vdots & \vdots & \vdots \\ \frac{\partial \phi(\boldsymbol{\theta}; t_k)}{\partial \theta_1} & \frac{\partial \phi(\boldsymbol{\theta}; t_k)}{\partial \theta_2} & \dots & \frac{\partial \phi(\boldsymbol{\theta}; t_k)}{\partial \theta_9} \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix}_{N \times 9}, \quad (3.10)$$

where

$$\frac{\partial \phi(\boldsymbol{\theta}; t_k)}{\partial \theta_i} = \text{SoC}^i(t_k) - \text{SoC}^5(t_k), \text{ for } i = 1, \dots, 4, \quad (3.11a)$$

$$\frac{\partial \phi(\boldsymbol{\theta}; t_k)}{\partial \theta_5} = I, \quad (3.11b)$$

$$\frac{\partial \phi(\boldsymbol{\theta}; t_k)}{\partial \theta_6} = I e^{-\theta_7 \text{SoC}(t_k)}, \quad (3.11c)$$

$$\frac{\partial \phi(\boldsymbol{\theta}; t_k)}{\partial \theta_7} = -I \text{SoC}(t_k) \theta_6 e^{-\theta_7 \text{SoC}(t_k)}, \quad (3.11d)$$

$$\frac{\partial \phi(\boldsymbol{\theta}; t_k)}{\partial \theta_8} = I \left(1 - e^{-\theta_9 t_k} \right), \quad (3.11e)$$

$$\frac{\partial \phi(\boldsymbol{\theta}; t_k)}{\partial \theta_9} = I \theta_8 t_k e^{-\theta_9 t_k}, \quad (3.11f)$$

for $k = 1, 2, \dots, N$. From above one can observe that the columns of $\mathbf{S}(\boldsymbol{\theta})$ are linearly independent, making $\mathbf{S}(\boldsymbol{\theta})$ a full-rank matrix. This conclusion can be further verified by computationally evaluating $\mathbf{S}(\boldsymbol{\theta})$ around the nominal parameters of a battery model, with further discussion offered in Section 3.4. Hence, one can claim that the parameter vector $\boldsymbol{\theta}$ is locally identifiable.

Further, suppose that $\hat{\boldsymbol{\theta}}$ is successfully estimated and minimizes $J(\boldsymbol{\theta})$. The covariance of $\hat{\boldsymbol{\theta}}$ in

the Gaussian case then is given by

$$\text{Cov}(\hat{\boldsymbol{\theta}}) = \text{E} \left(\left[\frac{\partial^2 J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}^2} \bigg|_{\boldsymbol{\theta}} \right]^{-1} \right) \approx [\mathbf{s}^\top(\boldsymbol{\theta}) \mathbf{Q}^{-1} \mathbf{s}(\boldsymbol{\theta})]^{-1}, \quad (3.12)$$

from which the variance of the estimate $\hat{\theta}_i$ for $i = 1, 2, \dots, 9$, is the i -th diagonal element of $\text{Cov}(\hat{\boldsymbol{\theta}})$, i.e., $[\text{Cov}(\hat{\boldsymbol{\theta}})]_{ii}$. Here, it should be noted that $[\text{Cov}(\hat{\boldsymbol{\theta}})]_{ii}$ measures the estimation error if the estimation is unbiased and one can use (3.12) to calculate the theoretically possible parameter estimation accuracy.

3.3.2 Identification Methods

The foregoing discussion poses a basic parameter estimation problem as shown in (3.8)-(3.9). Although the parameters are locally identifiable, solving the problem is still a challenge due to two difficulties. The first one stems from the nonlinearity of the model, as is seen from the voltage equation (3.6). As the second and more challenging difficulty, the minimization problem in (3.9) is non-convex. Although a numerical optimization procedure, e.g., the Gauss-Newton method, can be deployed to tackle the nonlinearity issue, the non-convexity may still lead the parameter search to local minimum points that are unphysical or unreasonable. To ensure correct parameter estimation, this work introduces constrained optimization and regularization to improve the problem formulation by using prior knowledge about parameters.

Constrained Optimization

To prevent parameter search approaching physically meaningless local minima, one can constrain the search within a parameter space believably correct. Specifically, one can roughly determine the lower and upper bounds of some parameters, use them to set up a limited search space and run numerical optimization within this space. In practice, it is not difficult to determine the bounds of some parameters for the Thevenin model, because some coarse-grained knowledge of a battery, e.g., internal impedance, can be obtained from both experience and some simple observation or

analysis of the measurement data. A further discussion of extracting prior knowledge of parameters is provided in Section 3.5. With this idea, the identification problem in (3.9) is modified as a constrained optimization problem:

$$\hat{\boldsymbol{\theta}} = \min_{\boldsymbol{\theta}} J(\boldsymbol{\theta}), \quad (3.13a)$$

$$\text{s.t. } \underline{\boldsymbol{\theta}} \leq \boldsymbol{\theta} \leq \overline{\boldsymbol{\theta}}, \quad (3.13b)$$

where $\underline{\boldsymbol{\theta}}$ and $\overline{\boldsymbol{\theta}}$ are the lower and upper bounds for $\boldsymbol{\theta}$, respectively. It is noted that (3.13) represents a constrained NLS problem, which can be addressed by the trust region or line search algorithms. Here, a trust region method is considered, and a brief overview about it is taken from [42] and offered below for completeness.

Let us begin with the unconstrained case and consider (3.13) without the constraint (3.13b). Suppose there is a current guess of the solution $\hat{\boldsymbol{\theta}}_k$, where k is the iteration step number. For the trust region method, it first constructs a function ψ_k , referred to as “model function”, to approximate the actual objective function J around the current guess $\hat{\boldsymbol{\theta}}_k$. The model function ψ_k is based on the Taylor-series expansion of J around $\hat{\boldsymbol{\theta}}_k$ and defined as follows:

$$\psi_k(\mathbf{s}_k) = J(\hat{\boldsymbol{\theta}}_k) + \mathbf{g}_k^\top \mathbf{s}_k + \frac{1}{2} \mathbf{s}_k^\top \mathbf{B}_k \mathbf{s}_k, \quad (3.14)$$

where \mathbf{s}_k is a step to be determined, $\mathbf{g}_k = \nabla J(\hat{\boldsymbol{\theta}}_k)$, and $\mathbf{B}_k = \nabla^2 J(\hat{\boldsymbol{\theta}}_k)$. Then, minimizing the objective function J reduces to finding a minimizer of the model function $\psi_k(\mathbf{s}_k)$. Since $\psi_k(\mathbf{s}_k)$ can be a poor approximation of J when the step \mathbf{s}_k is too large, the trust region method bounds the search for a minimizer of $\psi_k(\mathbf{s}_k)$ within some region around $\hat{\boldsymbol{\theta}}_k$. Hence, one needs to solve the following subproblem in order to find out the best candidate step \mathbf{s}_k :

$$\min_{\mathbf{s}_k} \psi_k(\mathbf{s}_k) \quad \text{s.t. } \|\mathbf{D}_k \mathbf{s}_k\| \leq \Delta_k, \quad (3.15)$$

where \mathbf{D}_k is a scaling matrix, $\|\cdot\|$ is the Euclidean norm, and Δ_k is the so-called trust region radius. It is then interesting to see whether \mathbf{s}_k can bring a large enough drop in $J(\boldsymbol{\theta})$ if added to $\hat{\boldsymbol{\theta}}_k$. A

worthy metric for this is

$$\rho_k = \frac{J(\hat{\boldsymbol{\theta}}_k) - J(\hat{\boldsymbol{\theta}}_k + \mathbf{s}_k)}{\psi_k(0) - \psi_k(\mathbf{s}_k)}, \quad (3.16)$$

where $J(\hat{\boldsymbol{\theta}}_k) - J(\hat{\boldsymbol{\theta}}_k + \mathbf{s}_k)$ is the actual reduction in $J(\boldsymbol{\theta})$, and $\psi_k(0) - \psi_k(\mathbf{s}_k)$ the anticipated reduction. If ρ_k is greater than a threshold, it is safe to increase the trust-region radius by letting $\hat{\boldsymbol{\theta}}_{k+1} = \hat{\boldsymbol{\theta}}_k + \mathbf{s}_k$ and then continuing the search. Otherwise, the trust region should be reduced, and rerun (3.15) at the same iterate point $\hat{\boldsymbol{\theta}}_k$.

In the case of (3.13b) applied as a constraint, there are different ways to cope with this. An approach offered in [42] defines a new subproblem, in which $\psi_k(\mathbf{s}_k)$ is replaced by

$$\psi'_k(\mathbf{s}_k) = J(\hat{\boldsymbol{\theta}}_k) + \mathbf{g}_k^\top \mathbf{s}_k + \frac{1}{2} \mathbf{s}_k^\top (\mathbf{B}_k + \mathbf{C}_k) \mathbf{s}_k,$$

where \mathbf{C}_k is a matrix that results from the constraint. Similar to (3.14), this subproblem is then iteratively implemented to search for the optimal parameter estimates. Summarizing the above, one can obtain the constrained NLS method for the Thevenin model identification, which is named as C-NLS.

Regularization

Tikhonov regularization offers another way to help overcome the non-convexity issue. It involves a pre-estimation of the parameters and uses the prior guess to regularize the original cost function. Here, the pre-estimation is expected to be close to the truth so that its presence will then drive the optimization to run in the vicinity of the true parameter values. The regularization-based parameter estimation problem is expressed as follows:

$$\min_{\boldsymbol{\theta}} \bar{J}(\boldsymbol{\theta}) = J(\boldsymbol{\theta}) + \frac{1}{2} (\boldsymbol{\theta} - \boldsymbol{\theta}_0)^\top \mathbf{P}_0^{-1} (\boldsymbol{\theta} - \boldsymbol{\theta}_0), \quad (3.17)$$

where $\boldsymbol{\theta}_0$ represents the prior guess of $\boldsymbol{\theta}$, and \mathbf{P}_0 is a diagonal positive matrix acting as a quantification of the confidence in the quality of $\boldsymbol{\theta}_0$.

The formulation in (3.17) can be interpreted from a Bayesian perspective [133]. Bayesian probabilistic estimation is concerned with determining the *a posteriori* distribution of $\boldsymbol{\theta}$ given the measurements \mathbf{y} or $p(\boldsymbol{\theta}|\mathbf{y})$, where $p(\cdot|\cdot)$ denotes conditional probability density function (pdf). According to the Bayes' rule, one has

$$p(\boldsymbol{\theta}|\mathbf{y}) = \frac{p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{y})} \propto p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta}).$$

Consider that the measurement \mathbf{y} is corrupted by noise \mathbf{w} , as often happens in practice. That is,

$$\mathbf{y} = \boldsymbol{\phi}(\boldsymbol{\theta}) + \mathbf{w}.$$

If \mathbf{w} is independent of $\boldsymbol{\theta}$ and follows a Gaussian distribution $\mathcal{N}(\mathbf{0}, \mathbf{Q})$, then $p(\mathbf{y}|\boldsymbol{\theta}) \sim \mathcal{N}(\boldsymbol{\phi}(\boldsymbol{\theta}), \mathbf{Q})$. Meanwhile, $\boldsymbol{\theta}$ is a Gaussian random vector. Its *a priori* pdf before the observation of \mathbf{y} is $p(\boldsymbol{\theta}) \sim \mathcal{N}(\boldsymbol{\theta}_0, \mathbf{P}_0)$. Then, it follows that

$$p(\boldsymbol{\theta}|\mathbf{y}) \propto \exp\left(-\frac{1}{2}[\mathbf{y} - \boldsymbol{\phi}(\boldsymbol{\theta})]^\top \mathbf{Q}^{-1}[\mathbf{y} - \boldsymbol{\phi}(\boldsymbol{\theta})]\right) \exp\left(-\frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\theta}_0)^\top \mathbf{P}_0^{-1}(\boldsymbol{\theta} - \boldsymbol{\theta}_0)\right).$$

From the perspective of estimation, it is desired to find out $\boldsymbol{\theta}$ to maximize $p(\boldsymbol{\theta}|\mathbf{y})$. This is known as maximum *a posteriori* (MAP) estimation and can reduce to the problem shown in (3.17) by considering the log likelihood. Here, the selection of $\boldsymbol{\theta}_0$ and \mathbf{P}_0 depends on some prior knowledge of a battery under consideration, and further discussion is given in Section 3.5. To address the problem in (3.17), one can also consider the aforementioned trust region method. Thus far the regularization-based NLS method is obtained, which is named as R-NLS.

Comparison of C-NLS and R-NLS

Both the C-NLS and R-NLS methods are developed to accomplish the parameter estimation for the Thevenin model. Sharing the objective of overcoming the non-convexity issue, they both require some advance knowledge of the parameters (i.e., the parameter bounds $\underline{\boldsymbol{\theta}}$ and $\bar{\boldsymbol{\theta}}$ for C-NLS and

prior guess $\boldsymbol{\theta}_0$ and \mathbf{P}_0 for R-NLS). Our experience suggests that the advance knowledge does not have to be accurate—even a coarse inference can lead to satisfactory estimation accuracy. It also turns out that one can gather it in some convenient ways, with a further discussion to be offered in Section 3.5 along with the elaboration of experiments.

Meanwhile, there are several interesting differences between the two approaches. The first one lies in the computational efficiency. It is noteworthy that the R-NLS is usually computationally faster as it is based on unconstrained optimization. A second difference is concerned with the parameter estimation accuracy. Based on [88], the theoretical estimation error for both approaches can be quantified as $\text{tr}(\boldsymbol{\Sigma})$, where tr denotes trace and

$$\boldsymbol{\Sigma} = \text{E} \left[(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})^\top \right] = \text{Cov}(\hat{\boldsymbol{\theta}}) + \left(\text{E}[\hat{\boldsymbol{\theta}}] - \boldsymbol{\theta} \right) \left(\text{E}[\hat{\boldsymbol{\theta}}] - \boldsymbol{\theta} \right)^\top. \quad (3.18)$$

Here, one can approximately view $[\boldsymbol{\Sigma}]_{ii}$ as a squared error between $\hat{\theta}_i$ and θ_i , and the second term of the rightmost-hand side is due to the bias of $\text{E}[\hat{\boldsymbol{\theta}}]$ from $\boldsymbol{\theta}$. For the C-NLS method, the bias term in (3.18) will reduce to zero if the bounds are ideally selected. In view of (3.12), its theoretical estimation accuracy then is given by

$$\text{tr}(\boldsymbol{\Sigma}) = \text{tr} \left(\left[\mathbf{S}^\top(\boldsymbol{\theta}) \mathbf{Q}^{-1} \mathbf{S}(\boldsymbol{\theta}) \right]^{-1} \right). \quad (3.19)$$

By contrast, the R-NLS method produces biased estimates as it pushes the estimates toward $\boldsymbol{\theta}_0$. It follows from (3.17) that

$$\text{E}[\hat{\boldsymbol{\theta}}] = \boldsymbol{\theta} + \left(\mathbf{I} + \mathbf{P}_0 \mathbf{S}^\top(\boldsymbol{\theta}) \mathbf{Q}^{-1} \mathbf{S}(\boldsymbol{\theta}) \right)^{-1} (\boldsymbol{\theta} - \boldsymbol{\theta}_0).$$

This hence suggests that the theoretical estimation error of the R-NLS is

$$\begin{aligned} \text{tr}(\mathbf{\Sigma}) = \text{tr} \left(\left[\mathbf{S}^\top(\boldsymbol{\theta}) \mathbf{Q}^{-1} \mathbf{S}(\boldsymbol{\theta}) + \mathbf{P}_0^{-1} \right]^{-1} \right) + (\boldsymbol{\theta} - \boldsymbol{\theta}_0)^\top \left(\mathbf{I} + \mathbf{P}_0 \mathbf{S}^\top(\boldsymbol{\theta}) \mathbf{Q}^{-1} \mathbf{S}(\boldsymbol{\theta}) \right)^{-\top} \\ \cdot \left(\mathbf{I} + \mathbf{P}_0 \mathbf{S}^\top(\boldsymbol{\theta}) \mathbf{Q}^{-1} \mathbf{S}(\boldsymbol{\theta}) \right)^{-1} (\boldsymbol{\theta} - \boldsymbol{\theta}_0). \end{aligned} \quad (3.20)$$

Comparing (3.19) and (3.20), it is easy to notice the theoretical estimation accuracy attained by the two methods can be different, with the former being fixed and the latter dependent on $\boldsymbol{\theta}_0$ and \mathbf{P}_0 . With distinct choices of $\boldsymbol{\theta}_0$ and \mathbf{P}_0 , the R-NLS can be either more accurate than the C-NLS (e.g., $\boldsymbol{\theta}_0 = \boldsymbol{\theta}$ and $\mathbf{P}_0 = \mathbf{0}$ as an extreme case) or less accurate (e.g., $\boldsymbol{\theta}_0$ is far away from $\boldsymbol{\theta}$ and \mathbf{P}_0 is close to $\mathbf{0}$). This indicates that one can leverage (3.20) as a guidance for the selection of $\boldsymbol{\theta}_0$ and \mathbf{P}_0 , and a detailed discussion of this will be given in Section 3.4.

3.4 Numerical Simulation

This section presents Monte Carlo simulation to verify the effectiveness of the C-NLS and R-NLS methods in extracting the Thevenin model parameters. The simulation will further compare them in terms of estimation accuracy and computation time in accordance with the analysis in Section 3.3.2.

3.4.1 Identification with Training Data

The training data used in the Monte Carlo simulation are generated using the Thevenin model with the following parameters: the nominal capacity $Q_f = 2.17$ Ah, $R = 0.0313 \, \Omega$, $C = 1,858$ F, and

$$R_0(\text{SoC}) = 0.0313 + 0.0678 \cdot e^{-13.2 \cdot \text{SoC}},$$

$$V_{\text{OC}}(\text{SoC}) = 3.3 + 2.61 \cdot \text{SoC} - 9.36 \cdot \text{SoC}^2 + 19.7 \cdot \text{SoC}^3 - 19.0 \cdot \text{SoC}^4 + 6.9 \cdot \text{SoC}^5,$$

with $\underline{V}_{\text{OC}} = 3.3$ V and $\overline{V}_{\text{OC}} = 4.15$ V. Here, the battery parameters are drawn from the identification results on a Samsung INR18650-25R Li-ion battery used in Section 3.5.

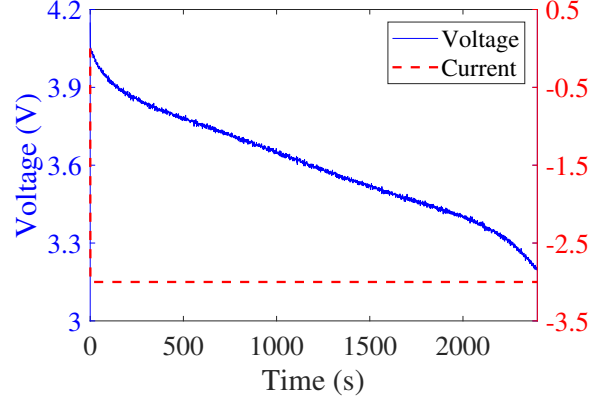


Figure 3.2: A model-based computer-generated terminal voltage profile for one Monte Carlo simulation.

Suppose the battery is discharged from full charge under a constant current I of -3 A by 2,400 s until when the terminal voltage reaches the cut-off voltage of 3.2 V. By (3.1), the SoC will decrease from 1 following

$$\text{SoC}(t) = 1 - \frac{1}{2604}t. \quad (3.21)$$

According to (3.6), the terminal voltage through time is given by

$$V(t) = 3.3 + \theta_1 \cdot \text{SoC}(t) + \theta_2 \cdot \text{SoC}^2(t) + \theta_3 \cdot \text{SoC}^3(t) + \theta_4 \cdot \text{SoC}^4(t) + \left(4.15 - 3.3 - \sum_{i=1}^4 \theta_i\right) \cdot \text{SoC}^5(t) - \theta_5 \cdot 3 - \theta_6 \cdot 3e^{-\theta_7 \cdot \text{SoC}(t)} - \theta_8 \cdot 3 \left(1 - e^{-\theta_9 t}\right), \quad (3.22)$$

where θ_i for $i = 1, 2, \dots, 9$ make up the parameter vector $\boldsymbol{\theta}$ with

$$\boldsymbol{\theta} = \begin{bmatrix} 2.61 & -9.36 & 19.7 & -19.0 & 0.0313 & 0.0678 & 13.2 & 0.0313 & 0.0172 \end{bmatrix}^\top.$$

Let the terminal voltage $V(t)$ be sampled every second and corrupted by an additive white Gaussian noise with variance of 2.5×10^{-5} (i.e., $\mathbf{Q} = 2.5 \times 10^{-5} \mathbf{I}$ in (3.13) and (3.17)). Running (3.21)-(3.22), one can obtain an I - V dataset, with an example plotted in Figure 3.2. Further, repeat the same procedure for $M = 500$ times to create 500 datasets. These datasets, accounting for the effect of random noise sampling, can help develop statistically convincing conclusions about the

Table 3.1: The simulation setting.

	Par	$\alpha_1/1$	$\alpha_2/1$	$\alpha_3/1$	$\alpha_4/1$	β_0/Ω	β_1/Ω	$\beta_2/1$	R/Ω	$(RC)^{-1}/s^{-1}$
	Initial Guess	1	1	1	1	0.029	0.4	40	0.2	1/40
C-NLS	$\underline{\theta}$	-	-	-	-	0.01	0	0	0	1/200
	$\bar{\theta}$	-	-	-	-	0.04	0.8	80	0.4	1
R-NLS	θ_0	1	1	1	1	0.029	0.4	40	0.2	1/40
	$\sqrt{\text{diag}(\mathbf{P}_0)}$	50	50	50	50	0.001	0.1	10	0.06	0.005

performance of the proposed methods.

Before proceeding to parameter identification, one can first calculate the parameter sensitivity matrix $\mathbf{S}(\theta)$ based on the datasets to computationally verify the full rankness of $\mathbf{S}(\theta)$. It can be easily and consistently found out that $\mathbf{S}(\theta)$ is of full rank, showing that θ is locally identifiable as previously analyzed.

With the datasets generated and the parameters' local identifiability verified, let us now apply the C-NLS and R-NLS methods to each dataset for parameter estimation. Meanwhile, the original NLS problem (3.8) is also solved to provide a benchmark for comparison. Table 3.1 summarizes the initial guess for all the methods, $\underline{\theta}$ and $\bar{\theta}$ for the C-NLS, and θ_0 and \mathbf{P}_0 for the R-NLS. Note that the setting in Table 3.1 is coarse-grained relative to the truth. Here, consider the normalized root-mean-square error (NRMSE) to quantify the parameter estimation accuracy, which is defined as

$$\text{NRMSE}(\theta_i) = \sqrt{\frac{\frac{1}{M} \sum_{k=1}^M (\hat{\theta}_i[k] - \theta_i)^2}{\theta_i^2}}, \quad (3.23)$$

for $i = 1, 2, \dots, 9$. Recalling (3.19)-(3.20), one can notice that $\text{NRMSE}(\theta_i)$ is a sample-based computational estimation of $\sqrt{[\Sigma]_{ii}/\theta_i^2}$. They are expected to be close if M is large enough.

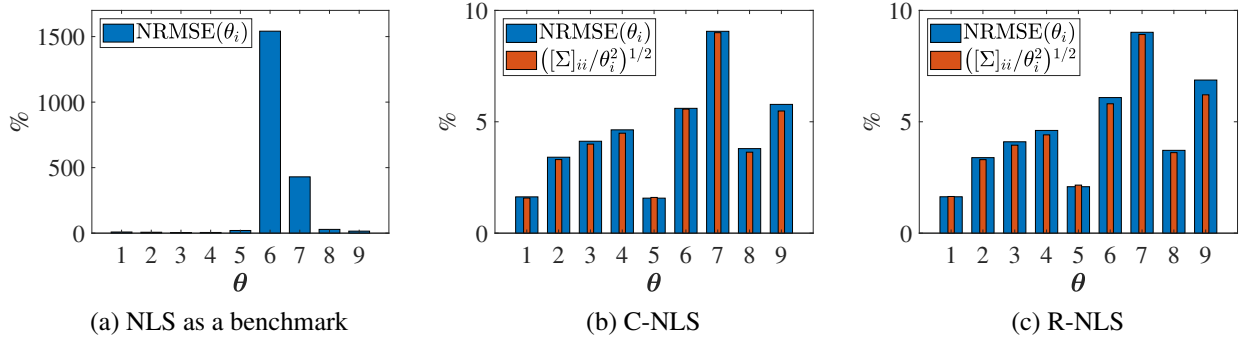


Figure 3.3: Parameter estimation errors for (a) NLS as a benchmark, (b) C-NLS and (c) R-NLS.

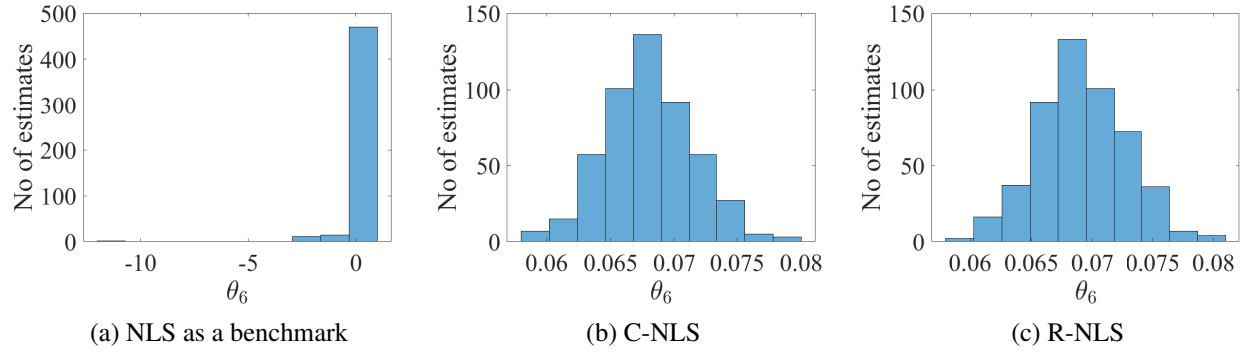


Figure 3.4: Histogram of estimates of θ_6 for (a) NLS as a benchmark, (b) C-NLS and (c) R-NLS.

3.4.2 Simulation Results

Figure 3.3 shows the NRMSE obtained by solving the original NLS problem (3.8) and applying the C-NLS and R-NLS methods, respectively. It is first seen that the NLS leads to extremely poor estimation accuracy, due to the intrinsic non-convexity as aforementioned. By comparison, both the C-NLS and R-NLS methods achieve substantial success, with the NRMSE lying below 10% for every parameter. Going deeper, let us look in more detail at θ_6 , i.e., the resistance β_1 . Figure 3.4 shows the histogram of the estimates of θ_6 produced by the three methods. As is seen, the NLS estimation constantly strays far away from the truth, but the C-NLS and R-NLS give estimates near the true value 0.0678 with considerable consistency. These observations suggest that the proposed methods are highly competent in identifying the Thevenin model parameters, even though the prior knowledge used is rough.

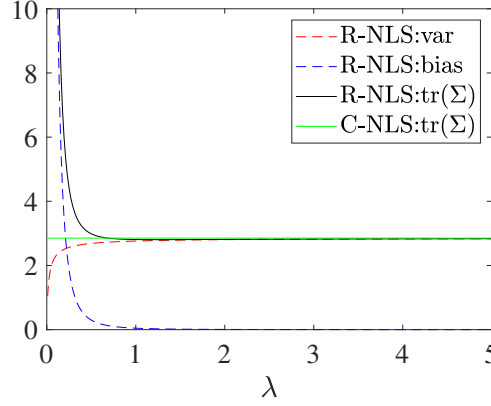


Figure 3.5: Theoretical estimation accuracy of the R-NLS method changes with λ .

Taking a closer look at Figure 3.3, one can find out that the estimation errors calculated based on the Monte Carlo simulation results match well with the theoretical values. This validates the correctness of the theoretical analysis. In addition, an implication specific for the R-NLS is that one can make use of (3.20) to select $\boldsymbol{\theta}_0$ and \boldsymbol{P}_0 . To explore this implication, let us consider applying $\{\boldsymbol{\theta}_0, \lambda \boldsymbol{P}_0\}$ with λ varying between 0 and 5 and $\{\boldsymbol{\theta}_0, \boldsymbol{P}_0\}$ shown in Table 3.1. Figure 3.5 depicts the theoretical accuracy of the R-NLS with respect to λ , where var and bias correspond to the two terms in the rightmost-hand side of (3.20), respectively. Several observations can be drawn. First, λ plays a role in the estimation performance, or in other words, the selection of $\boldsymbol{\theta}_0$ and \boldsymbol{P}_0 will affect the eventual estimation accuracy. Second, the estimation accuracy will improve when λ increases and then keep at a same level after λ becomes large enough, and the nominal bias will approach zero. This indicates that the prior knowledge should be imposed with appropriate confidence and that it is wise to adopted a relatively large \boldsymbol{P}_0 if the prior knowledge is not precise. Finally, based on such a plot, a practitioner can choose $\boldsymbol{\theta}_0$ and \boldsymbol{P}_0 deemed as the best.

Concluding this section is a comparison of the running time. All the simulations in this section are conducted on a Dell Precision Tower 3620 with 3 GHz Inter Xeon CPU and 16 Gb RAM running MATLAB R2018b. The computation time is averaged on the results of the 500 Monte Carlo runs. The original NLS, C-NLS and R-NLS on average take 88 ms, 142 ms, and 84 ms, respectively. From this result, one can find that both the C-NLS and R-NLS methods are computationally

well affordable, especially given that they are designed for offline identification. Furthermore, the R-NLS computes faster, which comes as an additional advantage, because the regularization expedites the parameter search.

3.5 Experimental Validation

This section validates the C-NLS and R-NLS methods on experimental data. The experiments were conducted using a PEC[®] SBT4050 battery tester (see Figure 2.4) and a Samsung INR18650-25R Li-ion battery. One test was conducted first to generate the training data, from which the parameters are extracted. Then, three more experiments were performed to generate validation datasets, to which the identified model is applied to assess its predictive performance.

3.5.1 Identification with Training Data

Considering $\underline{V}_{OC} = 3.3$ V and $\overline{V}_{OC} = 4.15$ V, the first experiment discharged the battery under room temperature using a constant current of -3 A from full charge until the terminal voltage hit the cut-off threshold of 3.2 V and then switched to constant-voltage discharging until the battery was fully depleted (when the current reduced to 0.125 A). The voltage asymptotically recovered to 3.3 V after the battery rested for a long time. Figure 3.6 shows the obtained current and voltage profiles. It should be noticed that a constant-voltage discharging phase was included here in order to determine the battery's total capacity. One can safely remove it if the capacity is known prior. Based on Figure 3.6, the battery's nominal capacity was calculated as $Q_f = 2.17$ Ah using coulomb counting [111]. Provided the I - V data in the constant-current discharging stage (encompassed by the black dotted-line box in Figure 3.6), the C-NLS and R-NLS methods were applied to extract parameter estimates, using the information setting in Table 3.1. Meanwhile, the original NLS problem (3.8) was also solved as a benchmark.

Before examining the parameter identification results, let us explain how the information in Table 3.1 is determined through the summary below. Here, the initial guess is used to initialize the

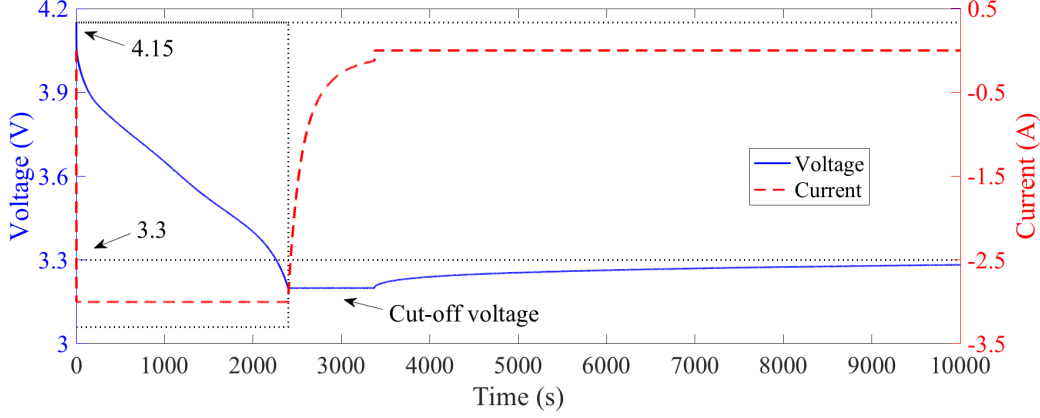


Figure 3.6: The voltage response of the Samsung INR18650-25R Li-ion battery in a constant-current/constant-voltage discharging experiment.

identification run for all the methods; $\underline{\theta}$ and $\bar{\theta}$ are the parameter bounds for the C-NLS, and θ_0 and P_0 represent the advance knowledge used by the R-NLS.

- The parameters α_1 through α_4 are hard to guess in advance as they have no physical meaning, so they are simply initialized to be one. There is also no need to assign bounds for them in the C-NLS case. In the R-NLS case, it is safe to leave θ_0 and P_0 that correspond to them as blank, but if one wants to calculate the theoretical accuracy using (3.20), it is acceptable to set θ_0 without much care about accuracy and meanwhile set a large P_0 as in Table 3.1.
- The parameter β_0 is roughly equal to the immediate voltage drop upon the start of discharging divided by the applied current. From Figure 3.6, it is estimated as 0.029. Since this guess can be close to the truth, it is reasonable to stipulate a prior 99% confidence interval of 0.029 ± 0.003 for β_0 , where the standard deviation is 0.001, to run the R-NLS. For the execution of the C-NLS, the bounds are loosely set as 0.01 and 0.04.
- The voltage recovery after the end of the entire discharging process is largely due to the loss of voltage across $R_0(\text{SoC} = 0) = \beta_0 + \beta_1$. Hence, β_1 is upper bounded by $0.1/0.125 = 0.8$ based on Figure 3.6. Its lower bound is assigned to be zero for simplicity. The average of the bounds, 0.4, hence is chosen as the initial guess of β_1 , along with a standard deviation of $0.4/3$.

Table 3.2: Parameter estimates for the battery used in the experiments.

Name	$\alpha_1/1$	$\alpha_2/1$	$\alpha_3/1$	$\alpha_4/1$	β_0/Ω	β_1/Ω	$\beta_2/1$	R/Ω	$(RC)^{-1}/s^{-1}$
$\hat{\theta}_{\text{Benchmark}}$	3.48	-11.3	20.5	-17.0	0.0360	0.0465	155	0.0668	0.0070
$\hat{\theta}_{\text{C-NLS}}$	2.61	-9.36	19.7	-19.0	0.0313	0.0678	13.2	0.0313	0.0172
$\hat{\theta}_{\text{R-NLS}}$	2.60	-9.34	19.7	-19.0	0.0308	0.0689	13.2	0.0313	0.0179

- The parameter β_2 mostly determines the curve shape of $R_0(\text{SoC})$. Its lower and upper bounds are very loosely set as 0 and 80. It is hence initially guessed as 40, which is the average of the bounds, with a standard deviation of $40/3$.
- The total voltage decline, following the immediate voltage drop when the discharging starts and lasting until when the constant-current discharging terminates, is from 4.15 V to 3.2 V. This is contributed by the combined change in the OCV and the voltage across R_0 and R . Note that the voltage across R will also approach a constant (i.e., IR) after sufficiently long time. The information can be used to infer an upper bound for R at 0.4 and a simple lower bound at 0. The average of the bounds, 0.2, is set as the initial guess, and the standard deviation set as $0.2/3$.
- Because RC is the time constant for the RC circuit, its value can be roughly seen from the voltage curve during the constant-current discharging. The voltage starts to decline almost linearly after the first 200 s, implying that RC is vaguely around $200/5 = 40$. The initial guess of $(RC)^{-1}$ is thus taken to be $1/40$, with a standard deviation of 0.005. Loose lower and upper bounds are chosen, which are $1/200$ and 1, respectively.

From above, one can develop some advance knowledge of the parameters through straightforward observation and analysis of the data and build an information setting as shown in Table 3.1. This allows the proposed C-NLS and R-NLS methods to easily lend themselves to use in practice.

With the above settings and letting $\mathbf{Q} = 2.5 \times 10^{-5} \mathbf{I}$, the original NLS, C-NLS and R-NLS are applied to estimate the parameters. The parameter estimates are summarized in Table 3.2, and the

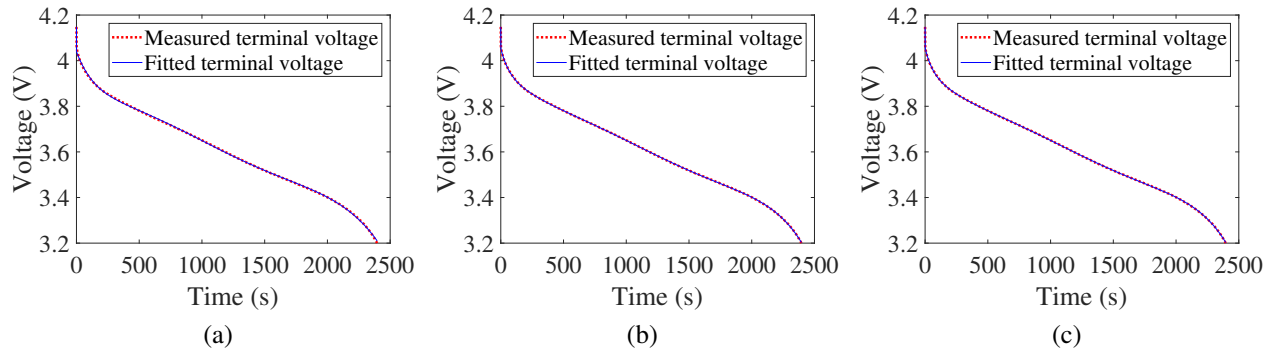


Figure 3.7: Comparison of measured and fitted terminal voltage for (a) NLS as a benchmark, (b) C-NLS and (c) R-NLS.

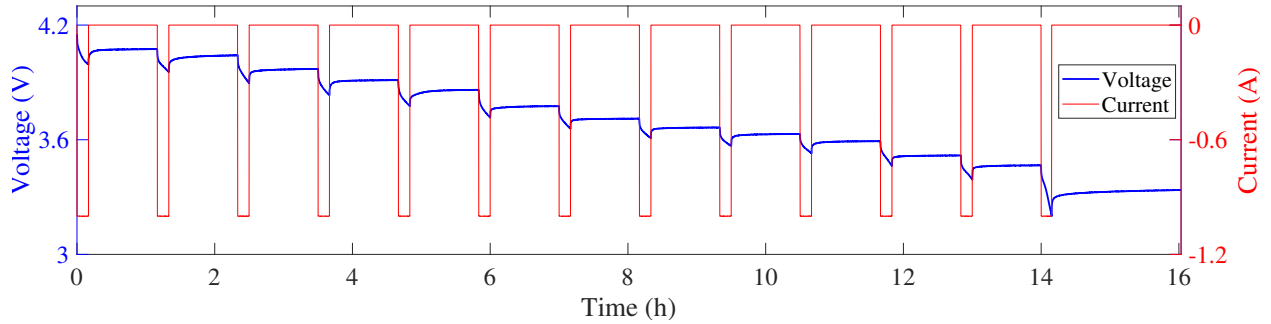


Figure 3.8: The voltage response of the battery in an intermittent discharging test.

voltage data fitting shown in Figure 3.7. Figure 3.7 displays that the predicted terminal voltage well overlaps the measurements in all three cases. However, with a look at Table 3.2, it is interesting to find out that the C-NLS and R-NLS produce very close estimation, whereas the estimates of the original NLS are quite different. Next, validation data are utilized to further evaluate the quality of the three identified models.

3.5.2 Validation with Validation Data

Further experiments were conducted to generate datasets used to assess the performance of the above identified model. The first experiment, based on intermittent discharging, was designed to validate the accuracy of the estimated SoC-OCV relationships and impedance parameters. In this experiment, the battery was discharged by a constant current of -1 A for ten minutes and then left at rest for two hours such that the battery voltage would recover to a steady state. Such a

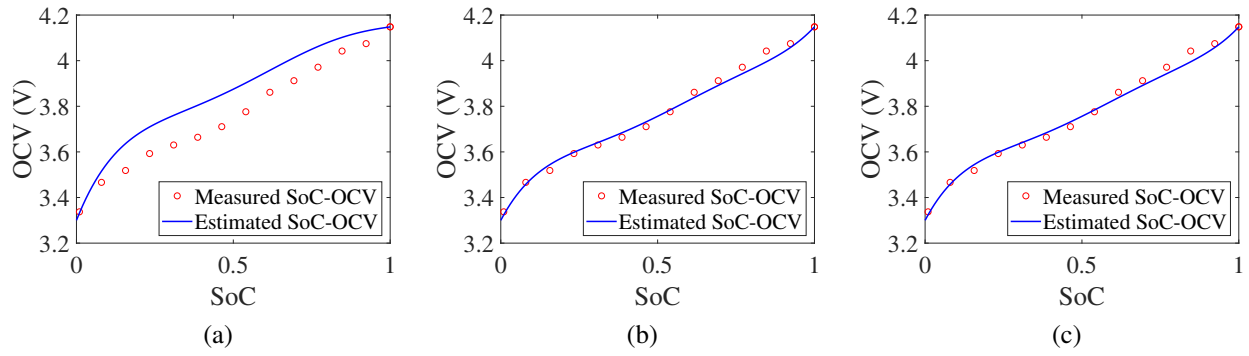


Figure 3.9: Comparison of measured and estimated SoC-OCV for (a) NLS as a benchmark, (b) C-NLS and (c) R-NLS.

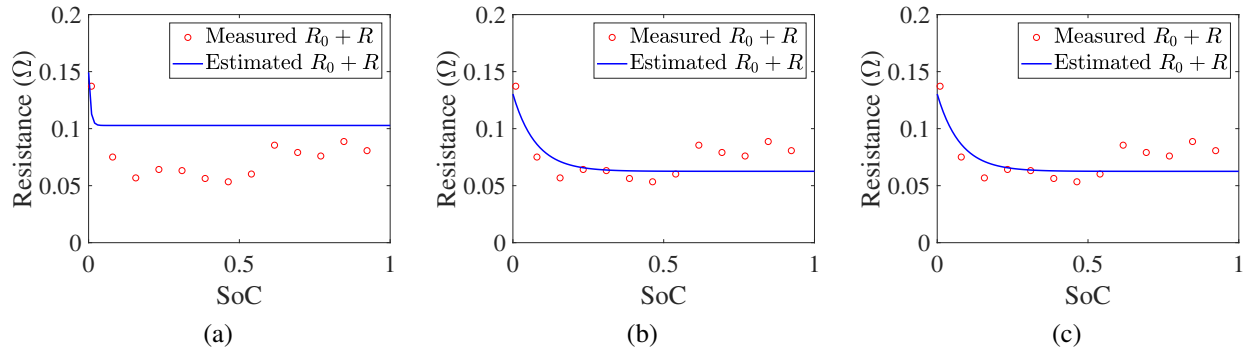


Figure 3.10: Comparison of measured and estimated $R_0 + R$ for (a) NLS as a benchmark, (b) C-NLS and (c) R-NLS.

procedure repeated itself until the voltage declined to the cut-off voltage of 3.2 V. The current and voltage profiles of this experiment are shown in Figure 3.8. This dataset allowed us to perform the following validation to verify the effectiveness of the proposed C-NLS and R-NLS methods.

- First, since the terminal voltage after a two-hour recovery can be regarded as the OCV, one can plot the SoC-OCV and use it as the ground truth to evaluate the estimated SoC-OCV relationship. Figure 3.9 offers such a comparison. Evidently, the SoC-OCV relationships identified by the C-NLS and R-NLS methods match well with the measured one. Contrasting this is a serious discrepancy in the benchmark case based on the original NLS.
- Second, the voltage recovery upon a pause of discharging can be largely attributed to the change in the voltage across R_0 and R , which is equal to $I(R_0 + R)$. Hence, $R_0 + R$ cor-

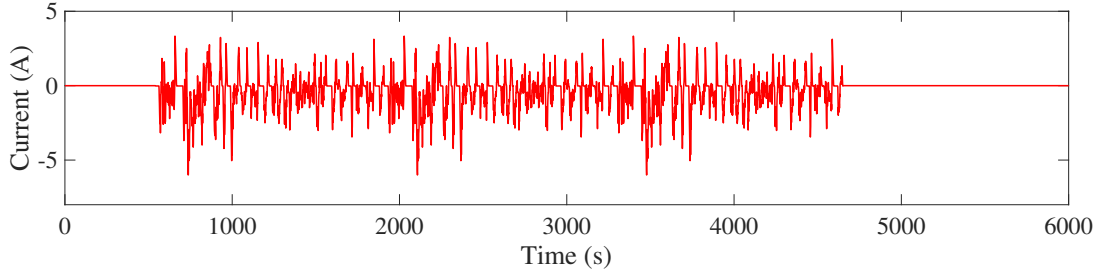


Figure 3.11: Variable currents from normalization of the UDDS profile.

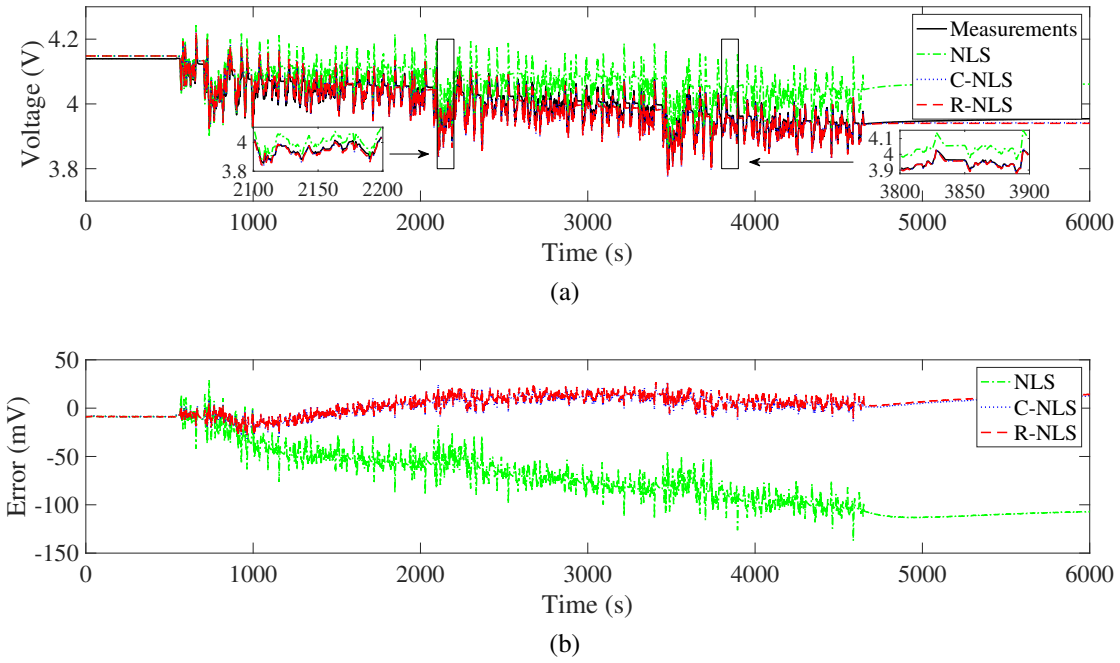


Figure 3.12: UDDS: (a) comparison of measured and estimated voltage for the benchmark NLS, C-NLS and R-NLS methods and (b) errors.

responding to different SoC levels can be determined and used to appraise the estimated values, as shown in Figure 3.10. The C-NLS and R-NLS are also observed to achieve more precise estimation than the original NLS in this case.

Another experiment applied a variable current profile to the battery, which was created based on the Urban Dynamometer Driving Schedule (UDDS) [3]. Figure 3.11 displays the UDDS-based current profile, which involves both charging and discharging in the run. Then, the identified models by the NLS, C-NLS and R-NLS methods are used to predict the terminal voltage under the current profile. Figure 3.12a compares the predicted voltage against the actual measurements. An

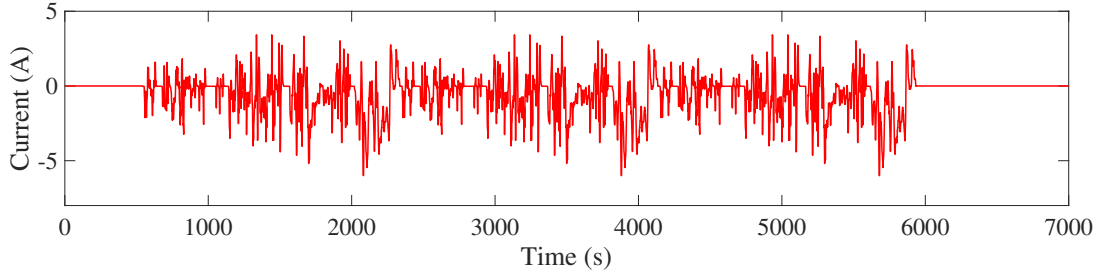


Figure 3.13: Variable currents from normalization of the WLTP profile.

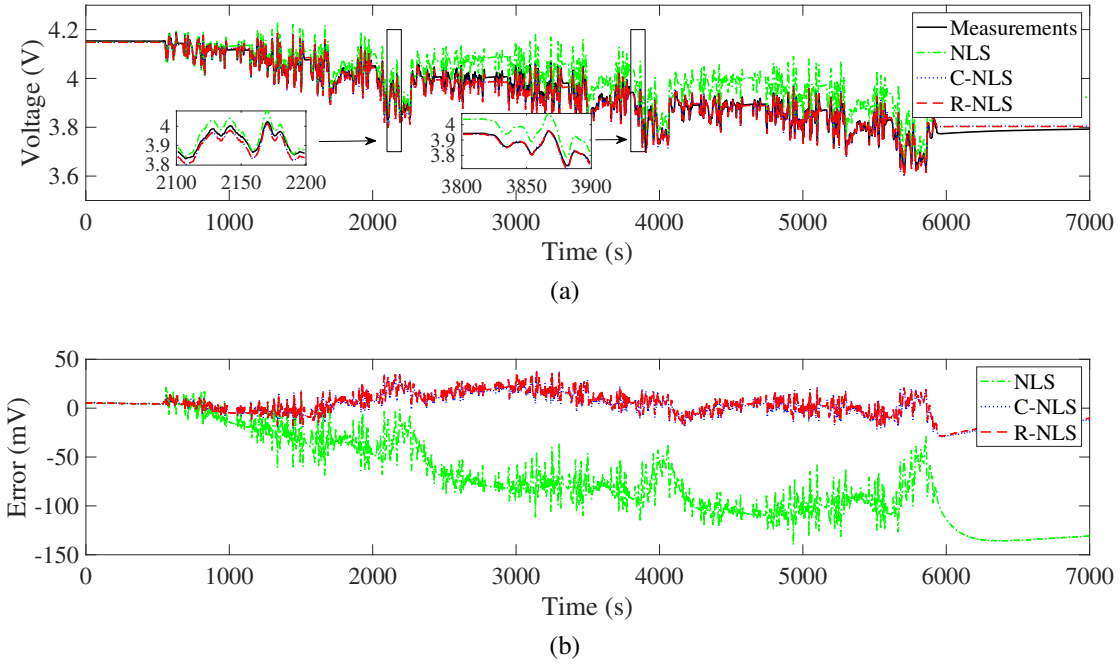


Figure 3.14: WLTP: (a) comparison of measured and estimated voltage for the benchmark NLS, C-NLS and R-NLS methods and (b) errors.

overall excellent fitting accuracy can be observed for the models identified by the C-NLS and R-NLS, whereas for the benchmark NLS the accuracy is far from satisfactory. Figure 3.12b further plots the prediction errors, which are found to be generally less than 20 mV for the C-NLS and R-NLS and larger by a great margin for the original NLS. This validation provides clear-cut evidence that the identified models by the C-NLS and R-NLS are accurate, justifying the effectiveness and competence of the two methods for parameter identification. Another experiment (see Figure 3.13) based on the Worldwide harmonized Light-duty vehicles Test Procedure (WLTP) [8] also shows similar results, as depicted in Figure 3.14.

3.6 Summary

This chapter studied the open problem of one-shot parameter identification for the popular Thevenin battery model. The objective was to build an approach capable of estimating all model parameters offline from the current/voltage data in a single run. The study formulated the identification problem in a prediction-error-based framework and conducted the identifiability analysis, unveiling that the parameters of the Thevenin model are locally identifiable. Given the nonlinearity of the identification problem, numerical optimization was considered, but the non-convexity of the cost function inflicted local minimum trouble on the optimization procedure. This work then took two approaches to deal with this challenge, one imposing parameter bounds to constrain the search space, and the other regularizing the cost function with prior knowledge of the parameters. The resultant two identification methods, along with the identifiability analysis, were extensively validated through simulation and experiments. The proposed methods can provide high-quality parameter estimation and are relatively easy to implement with a one-shot run, having a potential for many battery management applications relying on accurate models, e.g., optimal charging, SoC estimation and monitoring of aging conditions.

Chapter 4

Real-Time Optimal Charging Design Based on Explicit Model Predictive Control

This chapter¹ presents a framework for real-time charging control based on explicit model predictive control (MPC), exploiting its advantage in characterizing an explicit solution to an MPC problem to enable real-time charging control. The study begins with the formulation of MPC charging based on the nonlinear double-capacitor model. Then, multi-segment linearization is conducted to the original model, and applying the explicit MPC (eMPC) design to the obtained linear models leads to a charging control algorithm. The proposed algorithm shifts the constrained optimization offline by precomputing explicit solutions to the charging problem and expressing the charging law as piecewise affine functions. This drastically reduces not only the online computational costs in the control run but also the difficulty of coding. Extensive numerical simulation and experimental results verify the effectiveness of the proposed eMPC charging control framework and algorithm. The research results can potentially meet the needs for real-time battery management running on embedded hardware.

4.1 Introduction

Lithium-ion batteries (LiBs) have seen ever-increasing application across various sectors, including consumer electronics, electrified transportation and renewable energy, due to their appealing features like high voltage, high energy and power density, no memory effect, low self-discharge rates and long service life [197]. This trend has been driving a surge of research on advanced bat-

¹This chapter is based on the dissertation author's first-authored journal paper [185] © 2020 IEEE.

tery management to ensure the performance, safety and longevity of LiBs. Among the problems of interest, a critical one is optimal charging design in pursuit of two main objectives: reducing side reactions and effects to prolong life, and increasing the charging speed to meet efficiency needs. This work proposes a novel optimal charging control approach based on the notion of MPC. While accommodating the above objectives, it is particularly designed *via* exploiting the recent advances of eMPC to attain real-time implementation. The proposed approach may find important prospective use in future real-time battery management systems.

Literature Review

Finding the best ways to charge LiBs has attracted sustained attention in the past two decades. Currently, the most popular industrial practice is the so-called constant-current/constant-voltage (CC/CV) charging [98]. It applies a constant current to charge a LiB cell until it reaches a threshold voltage and then enforces a constant voltage to charge the cell at a gradually diminishing current. Another often endorsed practice is pulse charging that feeds energy into a battery using current pulses [43]. These methods, however, usually involve some heuristic determination of charging parameters, giving only empirical or conservative guarantee for charging safety and speed. This hence has motivated researchers to develop optimal charging protocols by combining physics-based LiB models and optimization to meet certain objectives concerning LiB health and/or charging time. A study is offered in [177] to build current profiles that can maximize the charge stored in a given time while suppressing the internal stress buildup, using a single particle model (SPM) supplemented with an intercalation-induced stress generation model. To enhance the conventional pulse charging, the study in [56] optimizes the magnitudes and duty cycles of current pulses to reconcile health effects with rates of charging. The investigations in [91, 128, 149, 131] lead to the design of health-aware, fast and thermal-safe charging protocols *via* multi-objective optimization based on coupled electro-thermal-aging models.

It is known that charging protocols are first generated offline and then run online, thus subjecting LiB charging to de facto open-loop control. Nonetheless, closed-loop control is arguably

more capable of improving the charging performance, since it incorporates dynamically the feedback about a LiB's present state to regulate the charging process. The past years have witnessed a growing body of work on this subject. Linear quadratic control is leveraged in [59] to enable health-aware LiB charging, with cost functions therein by design restricting the use of aggressive currents. Meanwhile, MPC, a constrained optimal control strategy, holds considerable promise here for two reasons. 1) It can handle hard state and input constraints. This gives a leverage to guarantee satisfaction of health- or safety-related constraints necessary for LiB operation. 2) It can optimize different kinds of objective functions to meet different charging needs or considerations. As another benefit, its formulation well admits nonlinear systems, thus bearing applicability to different types of nonlinear LiB models.

A lead is taken in [109] with the development of minimum-time charging control by applying nonlinear MPC to a one-dimensional (1-D) electrochemical model of LiBs. However, a barrier in the way of MPC-based charging is the high computational complexity that results from the numerical constrained optimization procedure at the core of an MPC algorithm. This can be more serious in the context of complex models, e.g., nonlinear electrochemical models involving various partial differential equations. Significant research hence has been devoted to computationally efficient MPC charging control design. The study in [126] considers nonlinear MPC for SPM and exploits the differential flatness of Fick's law of diffusion to reduce computational load. As another important way, model reduction is often used in the literature to simplify an electrochemical model and make it amenable for the design of efficient MPC. For example, the approach in [223] linearizes a nonlinear electrochemical model successively along a reference SoC trajectory. Other examples, e.g., [190], develop input-output approximations of a pseudo 2-D (P2D) model such as step response models and autoregressive exogenous models, so that application of MPC to them causes less computation. Particularly, the fast quadratic dynamic matrix control is used in [190] to further improve the computational efficiency.

Equivalent circuit models (ECMs) represent another appealing choice for MPC-based charging control due to their much less computation than electrochemical models. An early study is in [209],

which yet adopts a genetic algorithm as the optimizer despite its costly computation. There is a consensus today that it is still a critical need to design fast MPC for ECMs for the sake of practical implementation. To this end, the literature derives either simpler models or computationally frugal control frameworks. The method in [129] proposes to identify a time-series model recursively as an input-output approximation of the Thevenin model and takes advantage of its simplicity to achieve efficient generalized predictive control. Optimal charging based on the Thevenin model is formulated as a standard linear MPC problem in [202] that eases computation. Similarly, a linear-time-varying MPC method is proposed in [222]. A hierarchical MPC design in [144] features the generation of reference current profiles at a slow time scale and the reference tracking at a faster time scale, which lowers the cost of computation.

The above survey highlights the main advances in the development of computationally efficient MPC charging control. However, the existing methods still demand a relatively large amount of online computation, arising from the need to solve a constrained optimization problem at each sampling time. In addition, such online optimizers require strong computing capability, which is rarely available for the hardware on which a battery management system runs. A central challenge then lies in how to offload the primary part of the computational effort offline and run a lean controller online. The eMPC strategy, pioneered in [23, 22], is set to address this challenge. It pre-computes the control law offline by deriving explicit solutions to an MPC problem. The control law is composed of piecewise affine (PWA) functions of the system's current state, which can be run online through only straightforward arithmetic operations. The advantages of eMPC are remarkable. First, it can achieve MPC functionality with microsecond-millisecond online computational efficiency. Second, it is easy to code and executable on cheap embedded control hardware. Therefore, eMPC can hopefully provide a solution to bridge the gaps in computation and execution facing the current breed of MPC charging methods.

Contributions

The contribution of this work is threefold.

- This work presents the first framework for eMPC-based optimal charging control. Distinguished from the literature, this framework exploits eMPC to considerably reduce the online computational time and complexity of coding, paving the way for real-time execution of charging control.
- Based on the framework, an eMPC-based charging control algorithm is developed. The study considers the nonlinear double-capacitor (NDC) model, an ECM proposed in [181], and formulates a general nonlinear MPC charging problem. To deal with the nonlinearity inherent in the dynamics of batteries, it then simplifies the nonlinear MPC problem into a combination of approximate linear MPC problems. On such a basis, synthesis of eMPC-based control is performed to build an optimal charging control algorithm, which constructs on a set of PWA functions over a parameter space.
- The proposed charging control framework and algorithm are evaluated through extensive simulations and experiments, with their performance well validated.

Organization

The rest of the chapter is organized as follows. Section 4.2 introduces the NDC model along with the charging-related constraints. Section 4.3 contains: 1) the statement of an MPC-based health-aware charging control problem, 2) the piecewise linearization of the model, and 3) the formulation of eMPC-based charging control. The proposed charging control law is evaluated by simulation in Section 4.4 and experiments in Section 4.5. Finally, Section 4.6 gathers concluding remarks.

4.2 Model Description

Laying the groundwork for the charging control design, this section describes the NDC model (see Figure 4.1) and its governing dynamics. Further, the constraints to be enforced during charging are outlined.

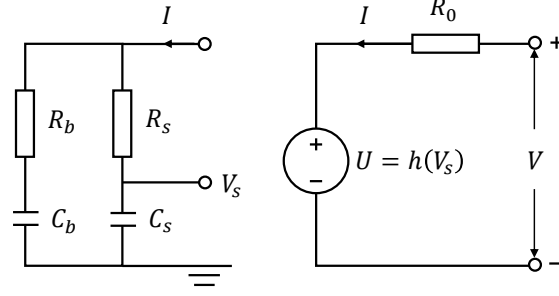


Figure 4.1: The nonlinear double-capacitor model.

As Section 2.2 in Chapter 2 has provided a detailed description of the NDC model, this section only gives a brief overview of the model dynamics for completeness. For the NDC model, its dynamics is governed by the following state-space equations:

$$\begin{cases} \begin{bmatrix} \dot{V}_b(t) \\ \dot{V}_s(t) \end{bmatrix} = A \begin{bmatrix} V_b(t) \\ V_s(t) \end{bmatrix} + BI(t), \\ V(t) = h(V_s(t)) + R_0(V_s(t))I(t), \end{cases} \quad (4.1a)$$

$$(4.1b)$$

where V_b is the voltage across C_b , V_s the voltage across C_s , I the applied current with $I > 0$ for charging and $I < 0$ for discharging, and

$$A = \begin{bmatrix} -\frac{1}{C_b(R_b+R_s)} & \frac{1}{C_b(R_b+R_s)} \\ \frac{1}{C_s(R_b+R_s)} & -\frac{1}{C_s(R_b+R_s)} \end{bmatrix}, \quad B = \begin{bmatrix} \frac{R_s}{C_b(R_b+R_s)} \\ \frac{R_b}{C_s(R_b+R_s)} \end{bmatrix}.$$

Here, V_b and V_s are limited to a range. For both, the lower bound of the range is set to be $V_{s,\min} = 0$ V, and the upper bound $V_{s,\max} = 1$ V for simplicity and without loss of generality. In other words, $V_b = V_s = 0$ V when the LiB is fully depleted (SoC = 0%), and $V_b = V_s = 1$ V when it is fully charged (SoC = 100%). Furthermore, the following shows the relation between SoC and V_b and V_s :

$$\text{SoC} = \frac{C_b V_b + C_s V_s}{(C_b + C_s) V_{s,\max}} \times 100\%,$$

where $(C_b + C_s)V_{s,\max}$ is a battery's total capacity, and $(C_b V_b + C_s V_s)$ the available capacity. Note

that the function $h(\cdot)$ also characterizes the SoC-OCV relationship, because $V_b = V_s = \text{SoC}$ when the battery is at equilibrium.

In (4.1b), $h(V_s)$ is parameterized as a polynomial:

$$h(V_s) = \sum_{i=0}^5 \alpha_i V_s^i,$$

where α_i for $i = 0, 1, \dots, 5$ are coefficients. Besides, R_0 is considered to monotonically increase with V_s . Such a dependence is described as

$$R_0(V_s) = \beta_1 + \beta_2 e^{-\beta_3(1-V_s)},$$

where $\beta_i > 0$ for $i = 1, 2, 3$.

To ensure health-conscious and safe charging, some constraints must be imposed during a charging process, a summary of which is as below. To begin with, the SoC must be constrained to avoid overcharging. That is,

$$\text{SoC}_{\min} \leq \text{SoC} \leq \text{SoC}_{\max}. \quad (4.2)$$

The charging current and terminal voltage must also be subject to limitations to circumvent safety issues, implying

$$I_{\min} \leq I \leq I_{\max}, \quad (4.3)$$

$$V_{\min} \leq V \leq V_{\max}. \quad (4.4)$$

In addition, V_b and V_s should be kept within the pre-set range. Given the dynamics shown in (4.1a), $V_s \geq V_b$ always holds during charging if the LiB is at equilibrium initially. One hence only needs to limit V_s by

$$V_{s,\min} \leq V_s \leq V_{s,\max}. \quad (4.5)$$

As explained in Section 2.2, V_s strikes an analogy to the Li-ion concentration at the surface region of an electrode, which needs to be constrained as suggested in some studies about MPC charging

based on electrochemical models, e.g., [223]. This makes (4.5) correspond to those constraints though it is an ECM considered here.

The final constraint to add concerns $V_s - V_b$. It is seen that $V_s - V_b$ drives the migration of charge from C_s to C_b during charging. The study in [181] points out that this variable is comparable to the Li-ion concentration gradients within an electrode when proving the approximate equivalence between the NDC model and the SPM. The Li-ion concentration gradients are a cause for internal stress buildup, and it can also lead to heating and formation of solid electrolyte interphase (SEI) film indirectly. These phenomena eventually will degrade the capacity, cycle life and thermal stability of LiBs [200, 17, 152]. As such, too steep gradients should be circumvented during charging. This implies a necessity for restricting $V_s - V_b$. Besides, the restriction should be increasingly stricter as SoC grows, because a LiB becomes more vulnerable to a large Li-ion concentration gradient. The constraint about $V_s - V_b$ is then designed as an affine decreasing function of SoC:

$$V_s - V_b \leq \gamma_1 \text{SoC} + \gamma_2, \quad (4.6)$$

where $\gamma_1 \leq 0$ and $\gamma_2 \geq 0$ are two coefficients. It can be rewritten as

$$\eta \leq \gamma_2, \quad (4.7)$$

where

$$\eta = -\frac{C_b + \gamma_1 C_b + C_s}{C_b + C_s} V_b + \frac{C_b + C_s - \gamma_1 C_s}{C_b + C_s} V_s.$$

Remark 4.1. The constraints in (4.2)-(4.5) are either standard or can find equivalents in the literature. But (4.6) or (4.7) is unique as no similar ones have been considered in previous studies about MPC charging, despite their implications for enhancing the health consciousness in charging. Here, it is the NDC model that allows such a constraint to be applied. Furthermore, this model, as shown in [181, 182], offers higher predictive accuracy than other popular ECMs, e.g., the Rint and Thevenin models, thus in a better position to ensure accurate charging control. These factors

point to the advantage and appeal of using the NDC model to perform charging control design.

4.3 Optimal Charging Design

This section states an MPC-based charging control problem and then develops an eMPC-based charging control law. The latter effort will involve model linearization and computation of explicit solutions to the stated MPC problem.

4.3.1 Charging Problem Formulation

To begin with, let us convert the original state-space model (4.1) into a form that admits standard MPC formulation, which includes no direct input-output feedthrough in the measurement equation.

To this end, define

$$\begin{aligned} x &= \begin{bmatrix} V_b & V_s & I \end{bmatrix}^\top, \\ y &= \begin{bmatrix} \text{SoC} & V_s & I & V & \eta \end{bmatrix}^\top, \end{aligned}$$

where x is the state vector, and y the output vector. Then, one can transform (4.1) by some manipulation and discretization into the following form:

$$\begin{cases} x_{k+1} = \mathcal{A}x_k + \mathcal{B}u_k, & (4.8a) \\ y_k = g(x_k), & (4.8b) \end{cases}$$

where $u_k = I_{k+1} - I_k$,

$$\mathcal{A} = \begin{bmatrix} \tilde{A} & \tilde{B} \\ 0 & 1 \end{bmatrix}, \mathcal{B} = \begin{bmatrix} 0 \\ 1 \end{bmatrix},$$

and $g(\cdot)$ represents the nonlinear mapping from x to y . In above, \tilde{A} and \tilde{B} are the discretization-based counterparts of A and B under sampling interval Δt . Furthermore, the constraints in (4.2)-

(4.7) can be put together in a compact form:

$$y_{\min} \leq y \leq y_{\max}.$$

In this setting, the health-aware charging control problem can be achieved by solving the following nonlinear MPC (NMPC) problem at time step k :

$$\begin{aligned} \min_z \quad & \sum_{k=0}^{N-1} \frac{1}{2} (\text{SoC}_k - \check{r})^\top Q (\text{SoC}_k - \check{r}) + \frac{1}{2} \Delta u_k^\top R \Delta u_k, \\ \text{s.t.} \quad & (4.8), \quad x_0 = \check{x}, \quad u_{-1} = \check{u}, \\ & u_k = u_{k-1} + \Delta u_k, \quad k = 0, \dots, N-1, \\ & \Delta u_k = 0, \quad k = N_u, \dots, N-1, \\ & y_{\min} \leq y_k \leq y_{\max}, \quad k = 0, \dots, N_c - 1, \end{aligned} \tag{4.9}$$

where $z = \begin{bmatrix} \Delta u_0 & \dots & \Delta u_{N_u-1} \end{bmatrix}^\top \in \mathbb{R}^{N_u}$ is the future input sequence to be optimized, \check{r} the target SoC, \check{x} the model state at current time instant, and \check{u} the control input applied in the previous sampling interval, respectively. Besides, N represents the prediction horizon, N_u the input horizon, N_c the constraint horizon, $Q = Q^\top \succeq 0$ and $R = R^\top \succ 0$. The problem (4.9) can be solved using nonlinear programming at each time instant. When the optimal solution z^* is found, i.e.,

$$z^* = \begin{bmatrix} \Delta u_0^* & \dots & \Delta u_{N_u-1}^* \end{bmatrix}^\top, \tag{4.10}$$

its first element Δu_0^* is used to compute

$$u_0 = \check{u} + \Delta u_0^*.$$

The current to be applied for charging then is given by

$$I_1 = u_0 + I_0 = \check{u} + \Delta u_0^* + \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \check{x}. \quad (4.11)$$

After this, the entire optimization problem is resolved at the next time instant with a new starting point.

The problem in (4.9) gives a complete description of MPC-based charging control based on the NDC model. However, the online computation for the nonlinear programming is relatively formidable, which limits its applicability to embedded charging control. Hence, it is our aim to address (4.9) *via* eMPC for easy online computation. As eMPC is designed for linear systems, one must first linearize (4.8), and the linearization is concerned with $h(V_s)$ and $R_0(V_s)$. Note that a single linear function is not accurate enough to approximate it as V_s changes in charging. This motivates us to adopt multi-segment linear approximation to enhance the approximation accuracy.

Proceeding to show this idea, consider linearizing $h(V_s)$ and $R_0(V_s)$ around a general fixed operating point, V_s^{op} , as a first step. For $h(V_s)$, a linear approximation is considered, i.e.,

$$h(V_s) \approx \lambda_1 V_s + \lambda_2, \quad (4.12)$$

where

$$\lambda_1 = \left. \frac{\partial h(V_s)}{\partial V_s} \right|_{V_s = V_s^{\text{op}}}, \quad \lambda_2 = h(V_s^{\text{op}}) - \lambda_1 V_s^{\text{op}}.$$

For $R_0(V_s)$, it is approximated as a constant, i.e.,

$$R_0(V_s) \approx R_0(V_s^{\text{op}}). \quad (4.13)$$

By (4.12)-(4.13), one can modify (4.8b) into a linear form as:

$$y_k = \mathcal{C}x_k + \mathcal{D}, \quad (4.14)$$

where

$$\mathcal{C} = \begin{bmatrix} \frac{C_b}{C_b+C_s} & \frac{C_s}{C_b+C_s} & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & \lambda_1 & R_0(V_s^{\text{op}}) \\ -\frac{C_b+\gamma_1 C_b+C_s}{C_b+C_s} & \frac{C_b+C_s-\gamma_1 C_s}{C_b+C_s} & 0 \end{bmatrix}, \quad \mathcal{D} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \lambda_2 \\ 0 \end{bmatrix}.$$

Accordingly, the original nonlinear MPC problem (4.9) would reduce to a linear one, which can be expressed as

$$\min_z \sum_{k=0}^{N-1} \frac{1}{2} (\text{SoC}_k - \check{r})^\top Q (\text{SoC}_k - \check{r}) + \frac{1}{2} \Delta u_k^\top R \Delta u_k, \quad (4.15)$$

$$\text{s.t. (4.8a), (4.14), } x_0 = \check{x}, u_{-1} = \check{u},$$

$$u_k = u_{k-1} + \Delta u_k, k = 0, \dots, N-1,$$

$$\Delta u_k = 0, k = N_u, \dots, N-1,$$

$$y_{\min} \leq y_k \leq y_{\max}, k = 0, \dots, N_c - 1.$$

Next is to extend this procedure to multi-segment approximation. Specifically, one can select multiple linearization points, denoting them as $V_{s,i}^{\text{op}}$ for $i = 1, 2, \dots, N_{\text{op}}$. The range of V_s then is subdivided into N_{op} partitions. The same procedure as in (4.12)-(4.14) can be repeated for each $V_{s,i}^{\text{op}}$. Finally, a set of linear MPC subproblems akin to (4.15) will be obtained.

Remark 4.2. The above procedure decomposes the original nonlinear MPC problem into a set of linear MPC problems with each based on a locally linearized model. In general, constraint violation may happen upon model switching during the execution of multiple linear MPCs. But this issue does not cause much concern here. First, the linearization can be very precise. For LiB cells, the function $h(\cdot)$, which characterizes the SOC-OCV curve, is roughly composed of several almost flat regions, thus well lending itself to multi-segment linearization. The accuracy can be also easily improved further by using more operating points. Second, one can use the upper end of the segment as V_s^{op} to evaluate and approximate $R_0(V_s)$. Then $R_0(V_s)$ is replaced by a number larger than it should be. Such a conservatism will effectively reduce the chance of constraint violation.

4.3.2 Charging Design Based on eMPC

Consider the linear MPC charging control problem (4.15). Based on (4.8a) and (4.14), one can obtain

$$\begin{aligned} y_k &= \mathcal{CA}^k \check{x} + \sum_{j=0}^{k-1} \mathcal{CA}^j \mathcal{B} u_{k-1-j} + \mathcal{D}, \\ u_k &= \check{u} + \sum_{j=0}^{k-1} \Delta u_j. \end{aligned}$$

Further, define the following vector of parameters:

$$\theta = \begin{bmatrix} \check{x}^\top & \check{r}^\top & \check{u}^\top \end{bmatrix}^\top \in \mathbb{R}^m.$$

The optimization problem (4.15) then can be recast as a convex quadratic program (QP) taking a standard form [22]:

$$\min_z \frac{1}{2} z^\top \Sigma z + (F\theta)^\top z, \quad (4.16a)$$

$$\text{s.t. } Gz \leq S\theta + W, \quad (4.16b)$$

where $z \in \mathbb{R}^{N_u}$, $\Sigma \succ 0 \in \mathbb{R}^{N_u \times N_u}$ and $F \in \mathbb{R}^{N_u \times m}$.

The QP problem (4.16) is also a multiparametric QP (mpQP) problem, as the characterization of its solution fundamentally involves the parameter vector θ . The solution can be described as a set-valued function $Z^*(\theta) : \Theta \rightarrow 2^{\mathbb{R}^n}$, where $Z^*(\theta)$ is a set of optimizer functions $z^*(\theta)$, Θ the set of feasible parameters (parameters that allow a non-empty set of z to satisfy (4.16b)), and $2^{\mathbb{R}^n}$ the set of subsets of \mathbb{R}^n . It has been proven in [23] that, Θ , which is provably a polyhedral set, can be partitioned into convex polyhedral regions, also referred to as critical regions and denoted as CR_i for $i = 1, 2, \dots, N_{CR}$. For each critical region,

$$z^*(\theta) = K_i \theta + g_i, \quad \forall \theta \in CR_i.$$

In other words, $Z^*(\theta)$ is PWA and continuous over Θ . An immediate implication is that, given the formulated problem, the charging control law will be PWA functions of the present charging state,

the target SoC and the input in last time instant.

A few mpQP algorithms have been developed in the literature. Such an algorithm usually has a two-fold functionality: determining the partition of Θ into critical regions CR_i , and finding out the control law $z^*(\theta)$ associated with each CR_i . To design them, an important approach is the so-called geometric approach [29]. It 1) decides a critical region around a specified parameter by using the sufficient and necessary conditions for optimality, 2) solves the mpQP for this region, and 3) partitions the rest of the feasible parameter space and continues the optimization until the space is fully explored. The literature also contains some other approaches, and an interested reader is referred to [29] for a review.

Putting together the above developments, the eMPC-based charging control algorithm is summarized as follows:

-
- *Offline mpQP computation*
 - Consider the first linear model
 - * Select a parameter θ_0
 - * Determine the critical region in the neighborhood of θ_0 , and denote it as CR_0
 - * Solve the mpQP problem (4.16) to obtain $z^*(\theta) = K_0\theta + g_0$ for $\theta \in CR_0$
 - * Partition the parameter space outside CR_0 , and determine $z^*(\theta)$ for new critical regions
 - * Repeat the procedure until when the entire parameter space has been explored
 - * Store in a table all (K_i, g_i) for $i = 1, 2, \dots, N_{CR}$
 - Repeat the procedure for all the other linear models
 - *Online eMPC-controlled charging*
 - Determine the governing linear model at every time step k
 - Given θ_k , search the stored (K, g) table to find CR_j such that $\theta_k \in CR_j$

- Determine $z^*(\theta) = K_j \theta_k + g_j$ as well as the charging current I_{k+1} as in (4.11)
- Repeat the charging control procedure until the condition for charging completion is satisfied

The following remarks summarize further discussion of the above eMPC-based charging control algorithm.

Remark 4.3. The eMPC-based charging control algorithm moves the constrained optimization, which is computationally expensive, from online to offline by deriving explicit solutions to the considered optimization problem. The online control run at every time step involves only searching through a look-up table comprising critical regions to find out the correct PWA function and then evaluating it. Our analysis, together with instructions in [24], shows that the computational and storage costs in the control run are generally affordable for embedded systems. Further, the literature also contains studies about efficient eMPC implementation by minimizing the time to evaluate the PWA functions and reducing the memory needed to store numbers, e.g., [189, 70]. They can be integrated with our work to enable highly efficient charging in practice. Therefore, the eMPC-based charging control algorithm would yield substantial online computational economy and implementability on relatively low-end computing hardware, filling a gap that exists in previous research on MPC-based charging.

Remark 4.4. The above design assumes state-feedback design for convenience of discussion. It can be easily extended to output-feedback control, which is necessary in practice as the internal states of the NDC model, V_b and V_s , cannot be measured. To attain this end, one can just use a nonlinear state observer to perform real-time state estimation. Then, the charging control setup is a closed loop between the LiB, observer and eMPC controller, which is outlined in Figure 4.2. To further illustrate this, Section 4.4.2 offers a case study that uses the extended Kalman filter (EKF) for state estimation and combines it with the proposed eMPC control law to perform output-feedback charging control.

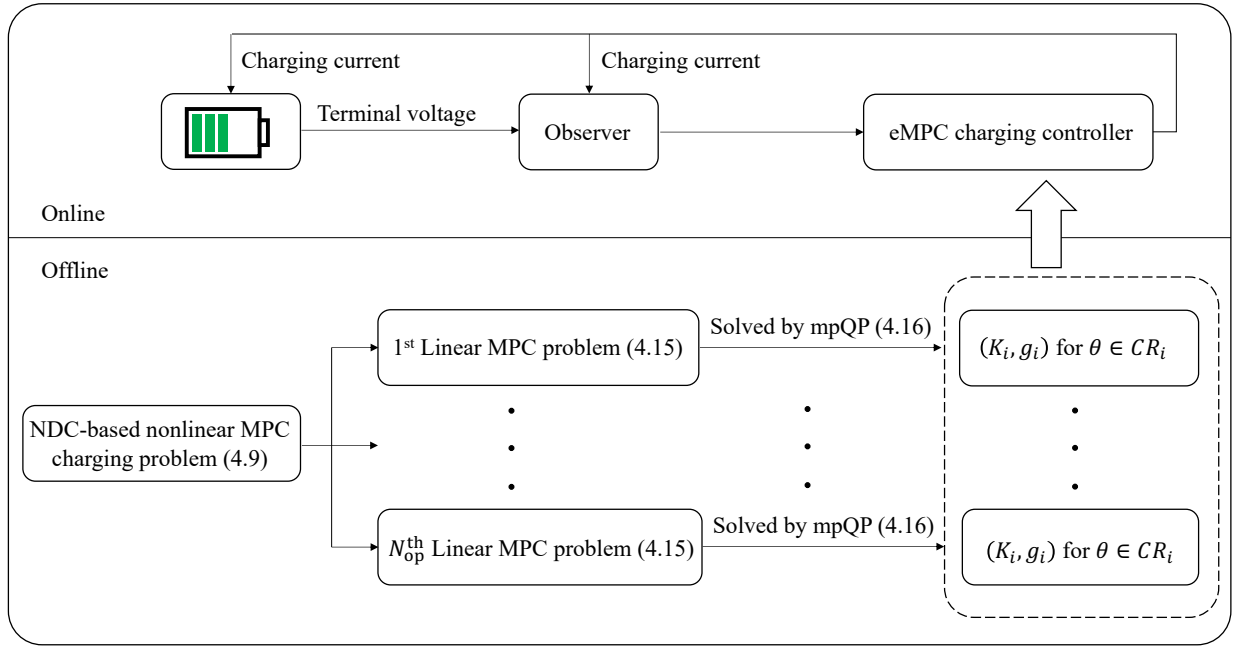


Figure 4.2: Development of the eMPC-based charging control algorithm.

Remark 4.5. The proposed design allows for extension to a more sophisticated model. For example, if a thermal model is coupled with the NDC model, one can follow the design approach to do linearization and then conduct eMPC design to enable temperature-conscious charging control. Such a treatment can also be modified to deal with the case when the model parameters are temperature-dependent. It is also noteworthy that the above can be applied to other ECMs, such as the Thevenin model, with custom-built eMPC charging control algorithms.

4.4 Numerical Simulation

This section presents simulation results to validate the proposed eMPC-based charging control algorithm. It offers an overview of the algorithm first through a basic case study and then investigates its performance in different settings.

Table 4.1: Battery model parameters.

Name	C_b/F	C_s/F	R_b/Ω	R_s/Ω	α_0	α_1	α_2	α_3	α_4	α_5	β_1	β_2	β_3
Value	9,913	887	0.025	0	3.2	3.041	-11.475	24.457	-23.536	8.513	0.09	0.35	10

Table 4.2: Linearization setting.

No.	V_s	V_s^{op}	λ_1	λ_2	R_0	N_{CR}
I	[0.20,0.50]	0.39	0.6505	3.3701	0.091	15
II	[0.50,0.60]	0.60	0.8659	3.2685	0.096	14
III	[0.60,0.70]	0.70	0.8562	3.2752	0.107	14
IV	[0.70,0.74]	0.74	0.8503	3.2794	0.116	14
V	[0.74,0.78]	0.78	0.8581	3.2734	0.129	14
VI	[0.78,0.81]	0.81	0.8810	3.2551	0.142	15
VII	[0.81,0.84]	0.84	0.9259	3.2181	0.161	15
VIII	[0.84,0.87]	0.87	1.0002	3.1544	0.185	15
IX	[0.87,0.90]	0.90	1.1123	3.0551	0.219	15

4.4.1 Basic Case Study

Given a 3 Ah LiB cell governed by the NDC model with the parameters shown in Table 4.1, consider the optimal charging problem based on the formulation in (4.9). Here, the charging objective is to raise the SoC from 20% to 90% under the following constraints:

$$V_s \leq 0.95 \text{ V}, 0 \text{ A} \leq I \leq 3 \text{ A}, V \leq 4.2 \text{ V},$$

$$V_s - V_b \leq -0.04 \cdot \text{SoC} + 0.08.$$

The sampling interval Δt for model discretization is 1 min. For the eMPC run, $Q = 1$, $R = 0.1$, the prediction horizon $N = 10$, the control horizon $N_u = 2$, and the constraint horizon $N_c = 2$ for (4.6) and $N_c = 1$ for the other constraints.

Next, the nonlinear MPC problem (4.9) is broken down into nine linear MPC problems through

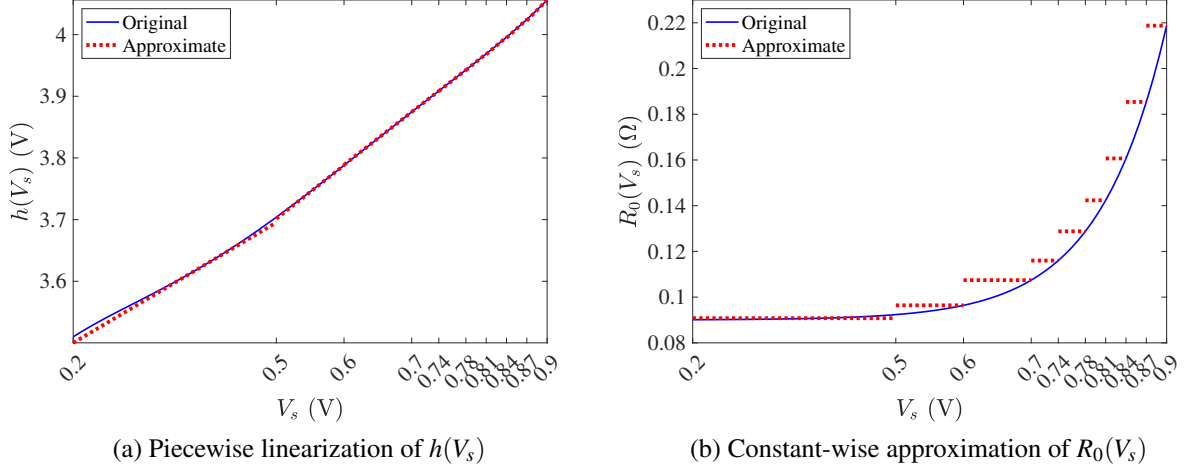


Figure 4.3: Multi-segment approximation of $h(V_s)$ and $R_0(V_s)$.

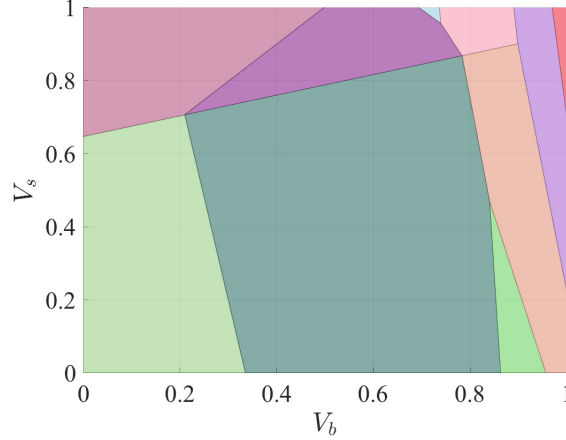


Figure 4.4: Critical region partitioning on the V_b - V_s plane when $I = 2$ A, $\check{r} = 0.9$, and $\check{u} = 0$ A.

multi-segment approximation. Table 4.2 summarizes the linearization setting, which includes the range of V_s for each segment, the linearization point of V_s , and the obtained linearization results for $h(V_s)$ and $R_0(V_s)$. The approximation of $h(V_s)$ and $R_0(V_s)$ is depicted in Figure 4.3. Thanks to the linearization, each linear MPC problem can be characterized in the form of (4.15). One can then compute the explicit solution to every problem by conveniently resorting to the MATLAB[®] MPC Toolbox [24], which leads to nine eMPCs that combine to make up the charging control algorithm. Note that the eMPCs have different numbers of critical region partitions, as they are based on different models and operating ranges. For each eMPC, the N_{CR} is shown in Table 4.2. To give the

reader a flavor of the control law, let us look at the second eMPC. For this one, the five-dimensional parameter space Θ is divided into 14 convex polyhedral critical regions, with each one associated with a PWA function of θ . For example, the tenth region is given by

$$\begin{bmatrix} 0.25 & -0.90 & -0.35 & 0 & 0 \\ 0.40 & 0.42 & 0.02 & -0.81 & -0.06 \\ 0.89 & 0.44 & 0.15 & 0 & 0 \\ -0.69 & 0.67 & 0.28 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} V_{b,k} \\ V_{s,k} \\ I_k \\ \check{r} \\ u_{k-1} \end{bmatrix} \leq \begin{bmatrix} -1.29 \\ 0 \\ 1.37 \\ 0.89 \\ 2 \\ 2 \\ 10 \\ 1 \\ 0 \\ 10 \end{bmatrix},$$

and the corresponding charging control law is

$$I_{k+1} = \begin{bmatrix} -2.263 & 2.263 & 0.938 & 0 & 0 \end{bmatrix} \begin{bmatrix} V_{b,k} \\ V_{s,k} \\ I_k \\ \check{r} \\ u_{k-1} \end{bmatrix},$$

which is affine and easy to code and compute. While it is impossible to visualize the critical region partitioning in the five-dimensional space, one can make a cross-sectional view by fixing part of the parameters. With this idea, Figure 4.4 displays the partitioning of the critical regions on the two-dimensional V_b - V_s plane when $I = 2$ A, $\check{r} = 0.9$, and $\check{u} = 0$ A. It is seen that there are ten critical regions from this point of observation, with each one being a convex polygon.

Running the simulation, the eMPC charging control algorithm yields an optimal current profile as shown in Figure 4.5a, which distinguishes itself significantly from existing results. It is observed

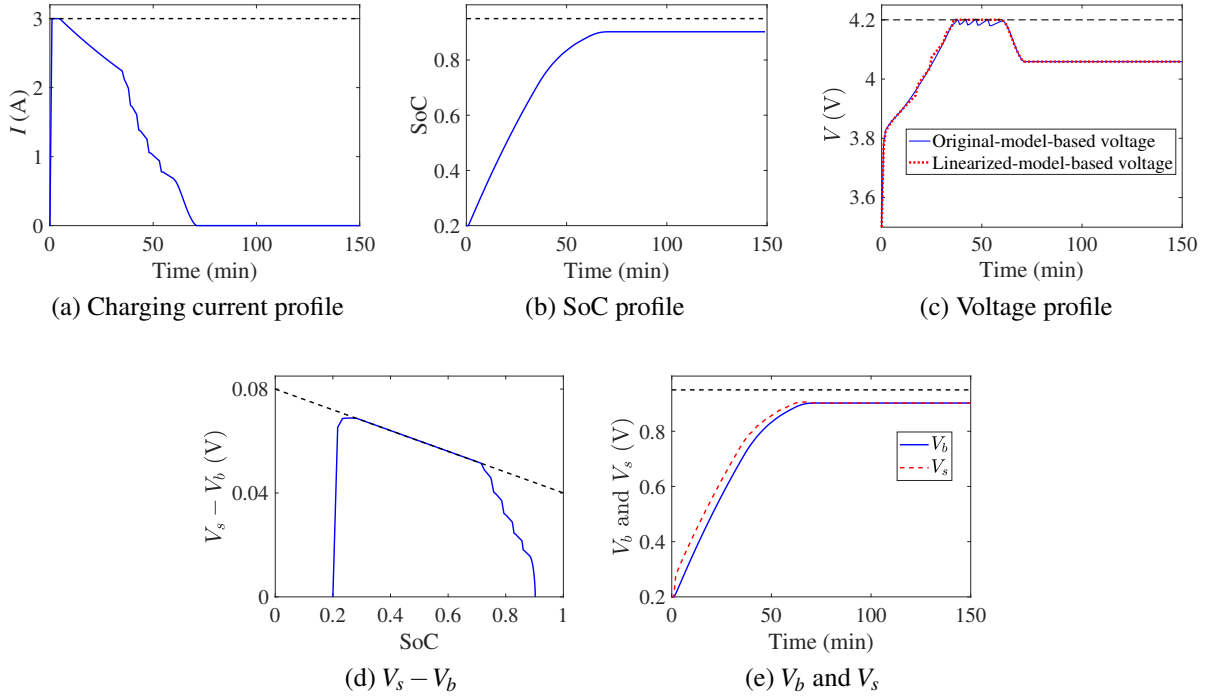


Figure 4.5: A basic case study of the eMPC-based charging control (black dashed lines denote constraints).

that this profile roughly includes three stages. Stage 1 features constant-current charging, which lasts for a relatively short period. Following it, Stage 2 sees the current decreasing at an approximately linear rate. The SoC increases fast during the two stages, and so does the terminal voltage V , as shown in Figures 4.5b-4.5c, respectively. When the charging continues in Stage 3, the magnitude of the current decreases at a faster rate overall, and the increase of SoC becomes slower. Meanwhile, it is seen that rate of decrease is not uniform and alternates between fast and slow rhythms. This is because the control law is seeking to achieve charging efficiency and constraint satisfaction simultaneously. If compared with the CC/CV charging, the optimal charging profile demonstrates more active regulation of the charging process, which believably can mitigate the effects of charging on the LiB cell's health more. Figure 4.5c illustrates the actual terminal voltage based on the original nonlinear model and the one predicted by the linear models. One can observe a discrepancy between them, which results from the model approximation. However, the actual voltage always lies below the pre-set upper limit due to the reasons elaborated in Remark 4.2. Fig-

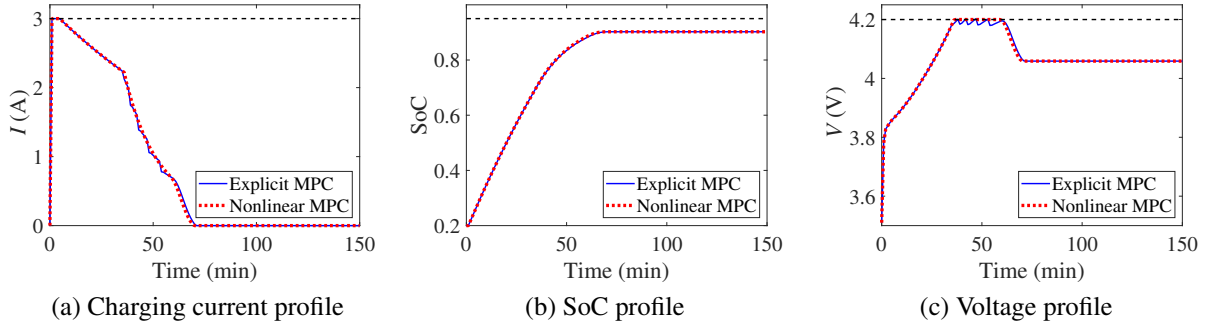


Figure 4.6: Comparison between eMPC and NMPC for charging control.

ures 4.5d and 4.5e show $V_s - V_b$ and the profiles of V_b and V_s , respectively. From all the figures, it is seen that every constraint is well satisfied throughout the charging process. These results indicate the efficacy of the proposed eMPC charging control algorithm to practical execution and its promise for health-aware charging.

It is understood that the eMPC algorithm approximates an NMPC formulated in (4.9), which involves no linearization and conducts online optimization. Figure 4.6 compares the profiles of the charging current, SoC and terminal voltage generated by the two methods. It is seen that both lead to very close results. This indicates that the eMPC can almost reproduce the NMPC while offering much higher computational efficiency.

Recalling Remark 4.3, the online implementation of an eMPC mainly concerns search and evaluation of the PWA functions in the look-up table. One can use the sequential search method to retrieve the correct PWA function from the look-up table [189]. For the above simulation, if considering that the search at every time step checks all critical regions in the worst case, the computation would involve around 840 multiply-accumulate operations in total. Section 4.4.4 discusses further about the running time. As for memory cost, the nine eMPCs here require a storage space for roughly 9,756 real numbers that encode all PWA functions and critical regions in the worst case. Our extensive simulations also illustrate an arithmetic precision of three digits after the decimal point can assure sufficient control accuracy and performance.

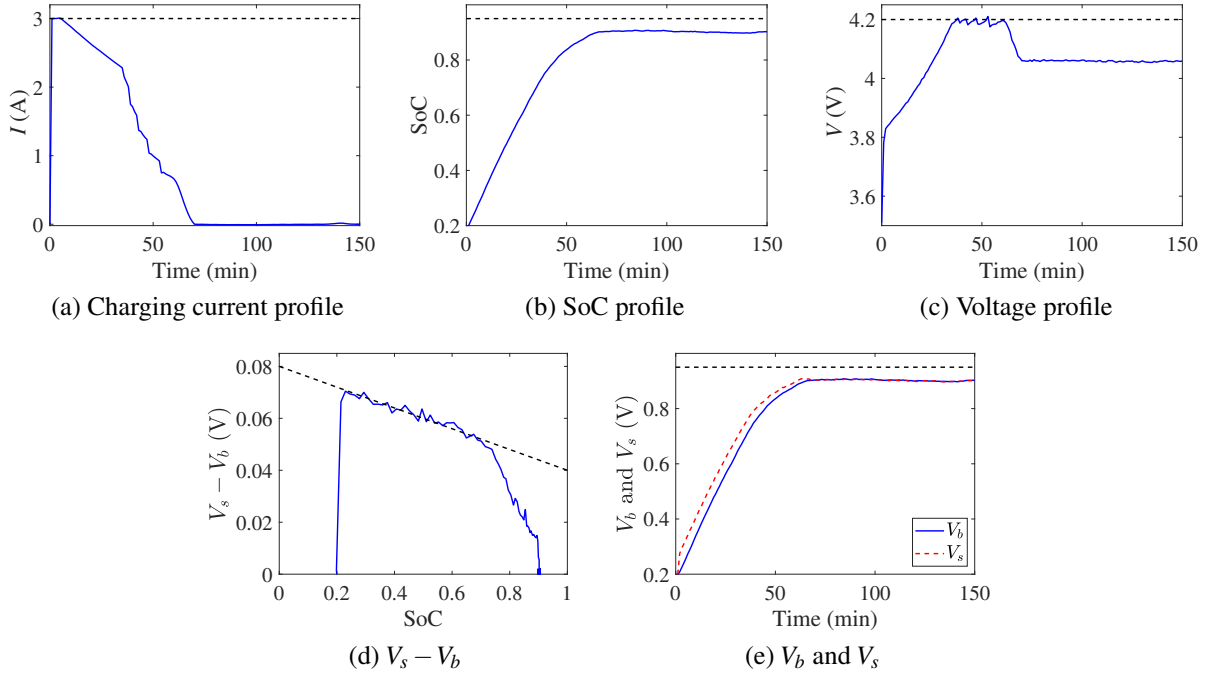


Figure 4.7: Output-feedback charging control based on eMPC and EKF.

4.4.2 EKF-Based Output-Feedback Charging Control

This works focuses on the eMPC charging control design on the state-feedback assumption. The result can be easily modified to enable the more practical output-feedback charging control by adding an observer to carry out state estimation (see Figure 4.2 and Remark 4.5). Here, let us investigate this extension by choosing the well-known EKF [58] as the observer and feeding the estimated states to the eMPC charging control algorithm. For the application of EKF, the model in (4.8) is considered and expanded to include process and measurement noises, which are both zero-mean white Gaussian and have covariances of $10^{-6}I$ and 9×10^{-6} , respectively. The other setting for simulation is the same as in Section 4.4.1. Figures 4.7a-4.7c show the resultant profiles of the charging current, SoC and terminal voltage. Comparing them one-on-one with Figures 4.5a-4.5c, one can find out some difference between them due to the noises and state estimation errors, which nonetheless is very small. Figure 4.7d displays $V_s - V_b$ slightly fluctuating around the pre-set constraint, still because of the estimation errors. While the constraint is not fully satisfied here, the violation is at a quite minor level without making a concern. Figure 4.7e depicts the profiles of

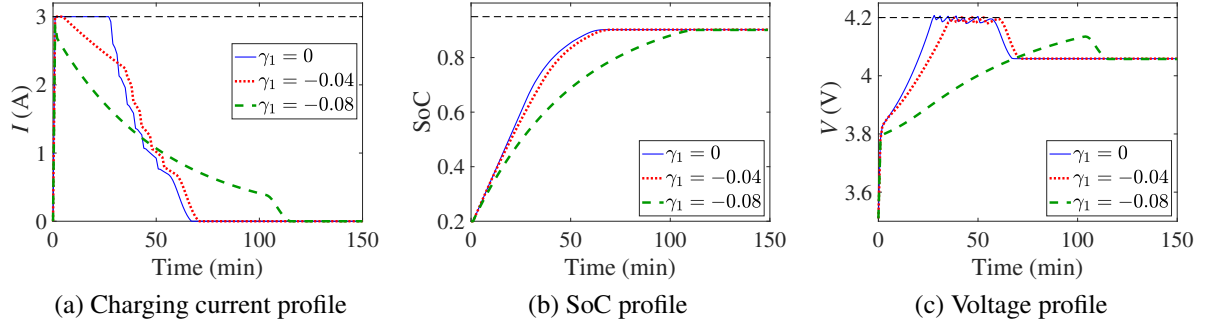


Figure 4.8: Charging control under different constraints on $V_s - V_b$.

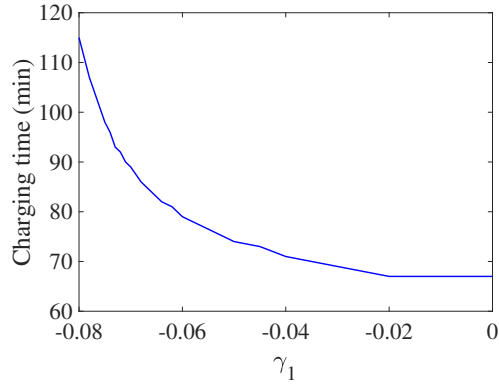


Figure 4.9: Charging time versus γ_1 under $\gamma_2 = 0.08$.

the actual V_b and V_s , which are only marginally different from their counterparts in Figure 4.5e. The results highlight that the proposed eMPC charging control algorithm can be implemented efficiently and effectively in the output-feedback manner in practice.

4.4.3 Effects of Changing Constraints and Horizon Parameters

Constraints and horizon parameters play a vital role in MPC-based optimal charging design. This section examines how the constraint (4.6) and the prediction, control and constraint horizons affect the proposed algorithm.

Recall that the constraint (4.6) is determined by γ_1 and γ_2 . Different choices of them will lead to different levels of health consciousness. Here, fix γ_2 at 0.08. Then let γ_1 take 0, -0.04 and -0.08, respectively, yielding increasingly stricter restrictions on $V_s - V_b$. The other parameters follow the

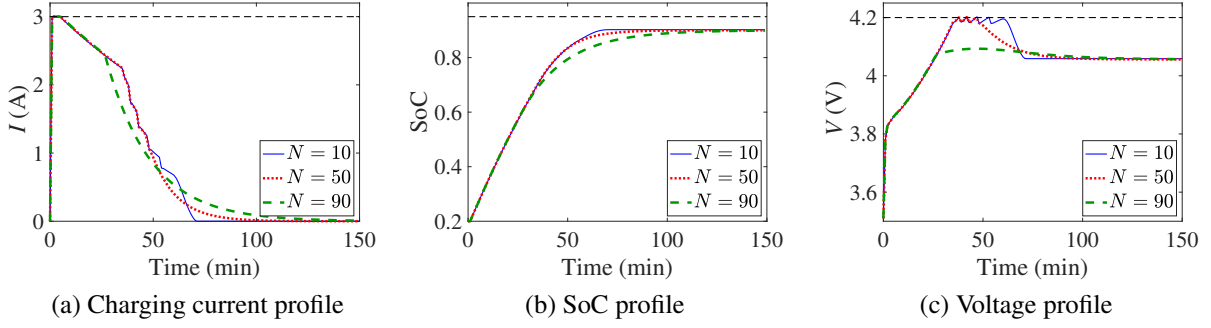


Figure 4.10: Charging control when the prediction horizon $N = 10, 50, 90$.

ones in Section 4.4.1. Figure 4.8 summarizes the simulation results in this setting. Figure 4.8a illustrates that, when γ_1 decreases, the “head” of the charging current profile lowers, and the “tail” raises and lengthens, leading to a longer charging time. This results from the constraint $V_s - V_b$ becoming more restrictive to enforce stronger health protection in charging. The corresponding SoC and terminal voltage profiles are plotted in Figures 4.8b and 4.8c. Both of them rise more slowly when γ_1 switches from 0 to -0.04 and then -0.08 . From these plots, one can observe that the stronger health protection in charging can be compromised by the longer charging time. This is verified by Figure 4.9, which gives an illustration of the charging time versus γ_1 when γ_2 is fixed at 0.08. A major implication is that a practitioner will need to select γ_1 and γ_2 to strike a balance between charging time and battery health, depending on the considered application. A simulation-based trial-and-error procedure can be used for the search.

Now, let us vary the horizon parameters and assess their influence on the charging results. To this end, it is beneficial to focus on only one horizon parameter at one time and have all other parameters remain the same as in Section 4.4.1. First, consider the prediction horizon N , and let it be 10, 50 and 90, respectively. Figure 4.10 depicts the charging current, SoC and terminal voltage profiles with different choices of N . Here, it is seen that an increasing N would make the charging process smoother and slower. This is because the optimization now is about a cost function defined and evaluated over a longer-term future, the solution of which will hence lead to less aggressive short-term control. It is also noteworthy that the charging current profiles differ only moderately

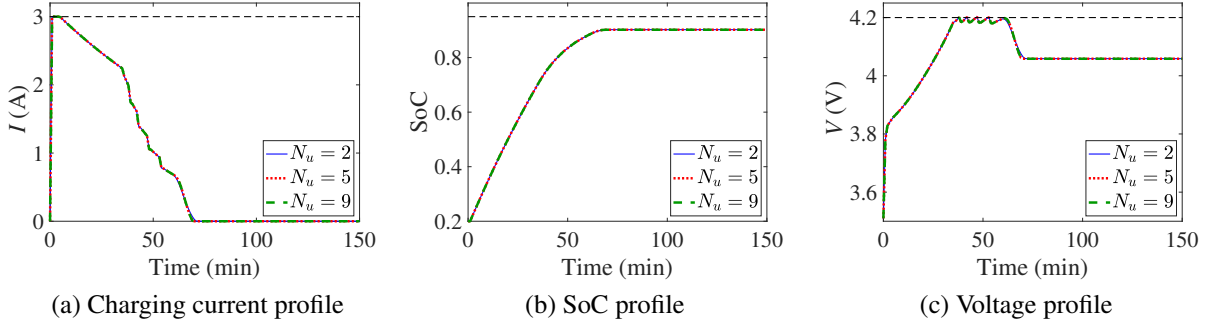


Figure 4.11: Charging control when the control horizon $N_u = 2, 5, 9$.

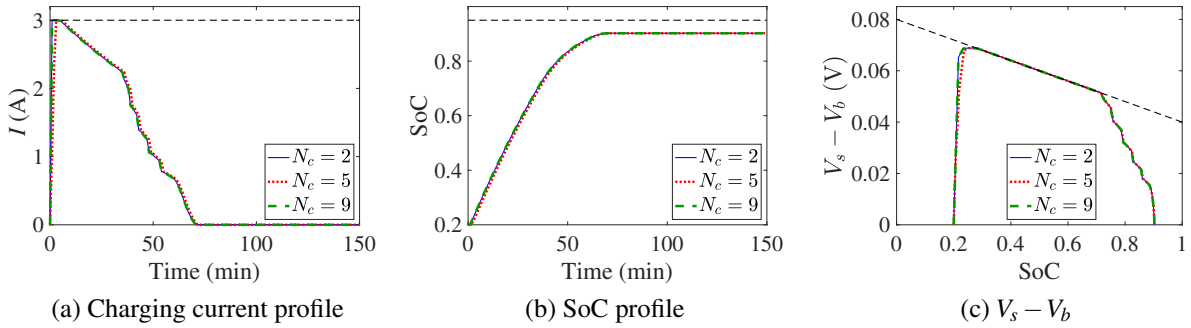


Figure 4.12: Charging control when the constraint horizon $N_c = 2, 5, 9$ for the constraint on $V_s - V_b$.

despite different N . The reason lies in the slow dynamics of LiBs, which traces its origin to the fact that the two eigenvalues of A are either at or close to the origin. Because of this, if comparing the eMPC with the original NMPC running with a very large prediction horizon, one can also find out that they lead to close performance. This implies that a relatively small N can be a safe choice.

Next, look at the control horizon N_u , which represents the number of charging moves to be optimized. Usually, $N_u \ll N$, and here, use $N_u = 2, 5$ and 9 , respectively. Figure 4.11 shows that almost the same results are acquired for different choices of N_u , even though the charging becomes a little faster when N_u increases. This is also largely due to the slow dynamics of LiBs. As a result, changing N_u by a scale of several minutes will produce little change to Δu_k . An additional reason is that the enforced constraints further suppress the variation of Δu_k . This observation suggests that a small N_u would be sufficient in practice, which will also reduce the computation advantageously as it leads to fewer critical regions. As an extra benefit, a smaller N_u implies fewer control variables

in the mpQP computation and brings a parameter space divided by fewer critical regions, thus yielding greater computational efficiency for both offline and online computation. Finally, let the constraint horizon N_c for the constraint in (4.6) vary among 2, 5 and 9, respectively. Figure 4.12 shows almost identical results for the different choices of N_c . To see why, one can examine the dynamics of $V_s - V_b$, which, by (4.1), is given by

$$\dot{V}_s - \dot{V}_b = -\frac{C_b + C_s}{C_b C_s (R_b + R_s)} (V_s - V_b) + \frac{R_b C_b - R_s C_s}{C_b C_s (R_b + R_s)} I.$$

Since the first term of the right-hand side is larger than the second one by at least two orders of magnitude, the current I has only marginal influence on the dynamics of $V_s - V_b$. The current profile hence will not change much even if N_c changes. As is with the case for N_u , this phenomenon allows us to use a small N_c for the constraint on $V_s - V_b$, which can assure control performance while promoting faster computation.

Although the horizon parameter selection for the proposed eMPC charging control algorithm would require some empirical optimization in practice, the following suggestions are offered summarizing the above:

1. One can select an N such that the prediction horizon is about ten minutes.
2. It is advisable to let $N_u, N_c \ll N$. Small N_u and N_c will lead to charging current profiles similar to those resulting from large N_u and N_c but can reduce online computational costs. The above results show that it is sufficient to set $N_u, N_c = 2$ if $N = 10$.
3. Trial-and-error tuning based on simulations can help a practitioner determine the best horizon parameters for specific applications.

4.4.4 Evaluation of Computational Efficiency

Finally, let us conclude the simulation by comparing the running time of the proposed eMPC and the NMPC charging algorithms based on (4.9). To this end, a series of simulations were run on a

Table 4.3: Comparison of computational time.

Method	(N, N_u, N_c)	Time (s)	Case
eMPC	(10,2,2)	0.42	Figures 4.5-4.6, 4.10-4.12
	(50,2,2)	0.42	Figure 4.10
	(90,2,2)	0.43	Figure 4.10
	(10,5,2)	0.42	Figure 4.11
	(10,9,2)	0.42	Figure 4.11
	(10,2,5)	0.42	Figure 4.12
	(10,2,9)	0.42	Figure 4.12
NMPC	(10,2,2)	6.45	Figure 4.6

Macbook Pro equipped with a 2.3 GHz Inter Core i5, 8 Gb RAM and MATLAB R2018b. Here, the evaluation of eMPC used different horizon parameter settings as considered in Section 4.4.3. For each simulation, the entire control run comprised 150 time steps representing a charging session of 2.5 hours. To make a fair assessment, 20 simulation runs were conducted in each case, and the average running time was calculated. Table 4.3 summarizes the computational time for all cases. It shows that eMPC on average takes no more than $\frac{1}{15}$ time of NMPC and its computation for each time step is only about 3 milliseconds. The results highlight the computational superiority of eMPC over NMPC and show its promise for real-time charging management.

4.5 Experimental Validation

The experimental validation was conducted using a PEC[®] SBT4050 battery tester (see Figure 2.4) and a Panasonic NCR18650B LiB cell with a rated capacity of 3 Ah.

The objective and setting of the experiment follow those for the simulation in Section 4.4.1. The cell's parameters are also the ones used in the simulation and given in Table 4.1. Hence, the experiment directly used the charging current profiles plotted in Figure 4.8a to charge the cell,

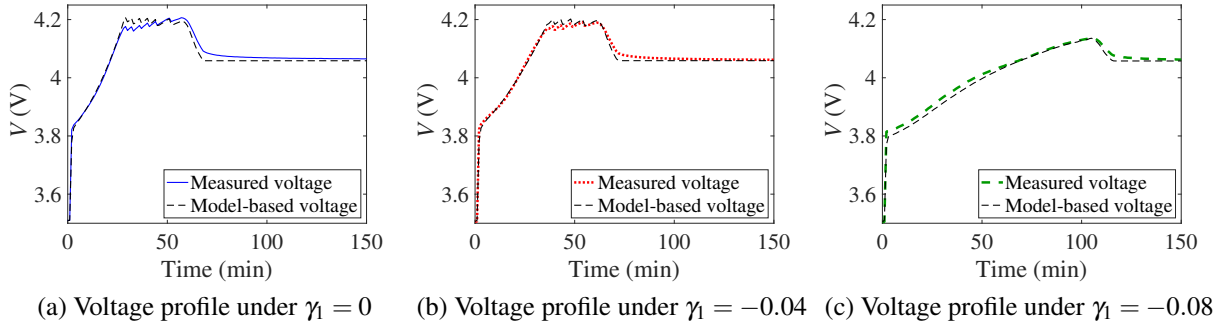


Figure 4.13: Experimental results based on the proposed charging strategies under different $V_s - V_b$ constraints.

which represent optimal health-conscious charging designs when the constraint (4.6) takes different parameters. Figure 4.13 illustrates the measured terminal voltage for each charging current profile and makes a comparison with the model-based prediction, where a close match between them is observed. The experiment verifies that the proposed algorithm can be well used for practical charging control. It is noted that battery life cycle testing is desired to further evaluate its role in mitigating health degradation, which will be part of our future work.

Here, due to the limitation of the tester, a charging profile was computed offline and then was uploaded to the tester to charge the cell in an open-loop control manner. Even though this would lead to certain performance loss, the results shown above still illustrate the effectiveness of the charging control. It is believed that better control performance will be achieved if applying the algorithm to closed-loop control, which will be pursued in our future work.

4.6 Summary

This chapter proposed to exploit eMPC to enhance LiB charging control. It inherits all the merits of MPC but enables highly efficient computation. Our design started with formulating an MPC charging control problem based on the NDC model. As the model is nonlinear, multi-segment linearization was developed to approximate the original MPC problem by a combination of multiple linear MPC problems. The solutions to the linear MPCs were computed offline and explicitly ex-

pressed as PWA functions, which made up an eMPC charging control algorithm. Contrasting the previous counterparts, this new algorithm is tremendously easy to code and fast to run online, potentially applicable to embedded computing hardware. The effectiveness of the proposed design was verified through extensive simulation and experiments.

Chapter 5

Real-Time Pack Spatial Thermal Field Estimation via Distributed Kalman Filter

This chapter¹ studies real-time reconstruction of the three-dimensional temperature field of a lithium-ion battery (LiB) pack in charging or discharging. The major challenge lies in how to acquire a high-fidelity reconstruction with constrained computation time. In this study, a three-dimensional thermal model is established first for a LiB pack configured in series. Although spatially resolved, this model captures spatial thermal behavior with a combination of high integrity and low complexity. Given the model, the standard Kalman filter is then distributed to attain temperature field estimation at substantially reduced computational complexity. The arithmetic operation analysis and numerical simulation illustrate that the proposed distributed estimation achieves a comparable accuracy as the centralized approach but with much less computation. This work can potentially contribute to the safer operation of the LiB packs in various systems dependent on LiB-based energy storage, potentially widening the access of this technology to a broader range of engineering areas.

5.1 Introduction

It is known that LiB packs are prone to heat buildup in operation, which can cause many side reactions, degrade the powering performance and accelerate aging. Due to the heat transfer mechanisms, the heat accumulation can spread from one cell to another. Fires and explosions may occur then and devastate the entire pack in a few seconds, as evidenced by a few incidents raising public

¹This chapter is based on the dissertation author's first-authored journal paper [183] © 2019 IEEE.

worry. This requires an effective monitoring of the thermal status, which, however, is a need unmet by the present literature. Connecting spatially resolved thermal modeling with estimation based on the Kalman filtering (KF), this work derives computationally efficient algorithms to reconstruct a LiB pack's three-dimensional temperature field, which opens up a new opportunity for LiB pack thermal management. The results, offering a promising means of reducing thermal risks facing LiB packs, can potentially find significant use in a broad range of LiB-based systems in electrified transportation, renewable energy farms and grid energy storage.

Literature Review

Since the commercialization in the early 1990s, LiBs have gained widespread use in various applications due to their high energy/power density, long cycle life and low self-discharge rate. They have established a dominant role in the consumer electronics sector and, owing to a continually declining manufacturing cost, are rapidly penetrating into the emerging sectors of electric vehicles and smart grid, where high-energy high-capacity energy storage is needed [17]. This trend has stimulated an intense interest in the research of high-performing battery management algorithms, with most of the existing works on state of charge (SoC) estimation to infer the amount of energy available in LiBs [50, 60, 196, 61, 160, 174, 21, 47], state of health (SoH) estimation to track the aging status [226, 167, 55, 106, 224], and optimal charging protocol design to optimally charge LiBs [109, 177, 209, 147, 59]. These efforts have provided strong support for the advancement of LiB systems.

Though widely considered as a promising technology, LiBs are susceptible to the thermal effects. Heat generation in LiBs during charging and discharging is always associated with irreversible overpotential heating, reversible entropic heating from electrochemical reactions, mixing heating and phase change heating, etc. [27]. The heating process can be intensified by high charging/discharging currents. The heat, without timely removal, will gradually build up, leading to many parasitic and side reactions involving by a complex mix of multiple physical and electrochemical process. Performance degradation and aging acceleration will result consequently [17].

Due to the high reactivity of the lithium metal and flammability of the electrolyte, overheating of LiBs may cause fires and explosions in extreme cases. This is known as the thermal runaway, in which heat generation occurs at a much faster rate than dissipation and eventually ignites the LiBs. Testaments to this catastrophic threat are given by a few well-publicized LiB fire incidents that happened to Tesla Model S, Toyota Prius, Boeing 787 Dreamliner, NASA’s Mars Surveyor and a navy’s submarine (for an overview, see, e.g., [9] and the references therein), which have raised serious concerns over the safe use of LiBs. This situation thus has motivated a growing interest in real-time temperature monitoring, which represents a crucial way toward taming the thermal threats.

Currently, a significant amount of work has been devoted to temperature estimation using a thermal model and the surface temperature measurements [65, 124, 122]. These studies consider lumped thermal models that concentrate the spatial dimensions into singular points and thus reduces thermal partial differential equations (PDEs) into very low-order ordinary differential equations (ODEs). Though advantageous for computation, this introduces a significant simplification because the temperature distribution is nonuniform spatially within a cell. For improvement, some recent works [176, 45, 205] study the temperature estimation with some awareness of the spatial nonuniformity, which uses thermal models accounting for the LiB cell’s spatial dimensions to a certain extent. Yet the models are still simplified at the sacrifice of their physical fidelity. It is noteworthy that the foregoing studies are focused on thermal management for a single LiB cell. The issue can become much more challenging when LiB packs are considered. With tens of cells stacked in a compact space, a LiB pack has larger dimensions and significantly complicated thermal behavior that will render the cell-level approaches unproductive. Meanwhile, LiB packs are more susceptible to heating issues—a “hot spot” can quickly spread from one cell to the others in a domino effect—and thus have a stronger need for thermal monitoring. The challenge can become more daunting as large-format high-capacity LiB cells are preferred increasingly for forming packs, since heat will be generated in larger amounts and more complex manners when the cell increases in size. In [145, 135, 123, 125, 170], the idea of lumped models is extended to characterize

the thermal dynamics of LiB packs composed of small cylindrical cells. However, studies of the spatial thermal behavior, which are crucial for effective thermal management for battery packs, are scant due to the research and development difficulty. The status quo, hence, is not even close to eliminating thermal threats that stand in the way of wider and safer LiB use.

Research Overview

Among the first of its kind, this work proposes approaches to reconstruct the three-dimensional temperature field of a LiB pack in real time. The fundamental notion is to acquire a spatially resolved thermal model for a LiB pack and then apply the KF technique to estimate the spatially distributed temperature. However, the task is nontrivial given the complexity of a LiB pack's thermal behavior. To fulfill the goal, a twofold effort is made, which lies in modeling and KF-based estimation.

Thermal modeling for LiBs has attracted a wealth of research, with the existing methodologies falling in three categories: 1) thermal models, which are concerned only with the heat phenomena and based on the thermal energy conservation principle, often given in the form of PDEs in three-dimensional space [37, 38, 76], 2) coupled thermal-electrochemical models, which associate the equations for thermal behavior with those for electrochemical reactions [74, 195, 175, 173, 103], and 3) lumped parameter models, which reduce the spatially distributed heat transfer into a heat flow passing through several discrete points (e.g., two points representing the cell's core and surface and connected by a thermal resistance) [65, 124, 122, 176, 45, 205, 170]. Among them, coupled thermal-electrochemical models can offer a sophisticated view of the LiB behavior with electrochemical reactions characterized at multiple scales. This, however, comes at the expense of computing burden formidable enough to defy any real-time estimation. For lumped models, the simplicity is conducive to estimation design for thermal management, but the spatial information loss weakens their capability for more effective spatial temperature monitoring. While the two types of models represent two extremes in terms of model fidelity or computational efficiency, the thermal models strike a valuable balance, thereby offering great promises for thermal management

with spatial awareness. Currently, despite prolific results on cell-level thermal modeling, the search for pack-level model is still at an early stage. This work, hence, will present the development of a three-dimensional LiB pack thermal model.

When the thermal model is available, the temperature field reconstruction will depend on the estimation technique, which is meant to estimate the temperature at any spatial point using the model and the temperature measurement data. Here, the KF, which is arguably the most celebrated estimation method, is one of the most promising candidate tools due to its ability to deal with the stochastic dynamic systems affected by noise—the thermal dynamics of a LiB pack can be subjected to the process noise in its evolution and the measurement noise when the temperature is measured by sensors. A direct application of the standard centralized KF (CKF) here is possible but will cause hefty computational costs that will not allow for an execution on real-time computing platforms. This is because of the KF’s computational complexity being cubic with the size of the state space [90], and a spatially resolved LiB pack model will have a substantial number of states, especially when the pack comprises many cells. To address this problem, a distributed KF (DKF) approach will be undertaken to achieve computational efficiency, which reduces a global KF into multiple local KFs running in parallel but with information exchange. The overall computational complexity of this approach will only increase linearly with the number of LiB cells in the pack, in contrast with the cubic increase for the CKF. This improvement thus places the proposed work in a more advantageous position for practical application.

Contributions

The primary contribution of this work lies in the three-dimensional temperature field reconstruction to enable accurate and computationally efficient LiB thermal monitoring, which is the first study that we are aware of in this direction. LiB packs are prone to heat buildup, and monitoring the spatially distributed temperature field over time will be valuable for ensuring the thermal safety. To this end, spatially resolved thermal modeling for LiB packs is developed in this work, which characterizes the dynamic thermal behavior with high integrity but still manageable computation

time. Based on the model, a DKF approach is then leveraged to reconstruct the temperature field, which features notably lower computational complexity than the centralized estimation. These efforts thus can guarantee reconstruction at both high accuracy and affordable computational costs, which is verified through extensive simulations. Providing a useful way for tracking the temperature in and on the surface of a LiB pack, this work can generate potential benefits for safety-critical application of LiBs in many engineering systems.

Organization

The rest of the chapter is organized as follows. Section 5.2 introduces a three-dimensional thermal model for a LiB pack, which is built on the principles of heat transfer and energy balance. In Section 5.3, the PDE-based model obtained in Section 5.2 is reduced to an ODE-based state-space system through discretization over time and space for convenience of estimation. Section 5.4 investigates the design of the estimation approaches based on the KF. Beginning with an introduction of the standard KF, the study will then focus on the distributed versions in order to cut down the computation, which is accompanied by a detailed analysis of the reduction in computational complexity. The simulation is offered in Section 5.5 to demonstrate the efficacy of the proposed approaches. Finally, some concluding remarks are gathered in Section 5.6.

5.2 Battery Pack Thermal Model

This section is devoted to spatially resolved modeling for a LiB pack with cells in serial configuration in order to characterize its spatial thermal behavior. This effort is based on the fundamental principles of heat transfer and energy balance and extends the cell-level modeling in [37] to a pack.

Consider the LiB pack shown in Figure 5.1. This pack consists of multiple identical prismatic cells configured in series (the cell is the same as in [37]). Each cell can be divided into two areas: the core region and the cell case that seals the core region inside. The core region is the main body of a prismatic cell. It consists of many smaller cell units connected in parallel to provide

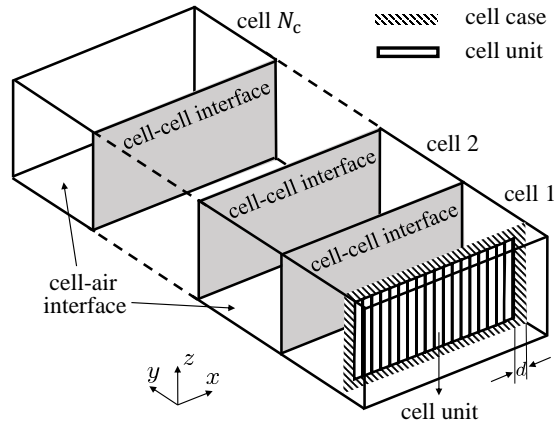


Figure 5.1: Schematic diagram of a LiB pack.

high capacity, with each unit composed of electrodes, separators and current collectors. While this makes the prismatic cell similar to a module, we still refer it as a cell as it is the basic building block of a battery pack. The cell case is a metal container and also includes a contact layer filled with liquid electrolyte and in touch with the core region. No heat is produced within the case. In this setting, modeling will be performed next for the thermal dynamics in the core region and the case and on the boundaries, e.g., the core-case interface boundary, cell-cell interface boundary and cell-air interface boundary.

To proceed further, a brief review of the heat transfer mechanisms is offered. It is known that there are three ways for heat transfer from one place to another: conduction, convection and radiation. Conduction happens within a substance or between substances in direct contact, caused by collision between atoms. Convection generally refers to the heat transfer with a fluid that moves between areas with temperature difference. Radiation is the energy emission by objects at nonzero temperature in the form of electromagnetic waves. An interested reader is referred to [25] for more details.

Consider the thermal dynamics in the core region first. Here, convection and radiation can be ignored since electromagnetic waves can hardly transmit through the cell and the liquid electrolyte is of limited mobility. Consequently, heat transfer within the core region is dominated by conduc-

tion, which can be expressed by the three-dimensional heat diffusion equation

$$\rho_{\text{co}} c_{\text{co}} \frac{\partial T}{\partial t} = \lambda_{\text{co}} \nabla^2 T + q, \quad (5.1)$$

where ρ_{co} , c_{co} and λ_{co} are the mass density, specific heat and thermal conductivity of the core region, respectively. In addition, T and q , respectively, denote the core region's temperature in kelvins and heat generation density. A general characterization of q , as discussed in [11, 38, 17], is offered by

$$q = \frac{I}{V_{\text{co}}} \left[(U_{\text{ocv}} - U_{\text{t}}) - T \frac{dU_{\text{ocv}}}{dT} \right], \quad (5.2)$$

where V_{co} , I , U_{ocv} , U_{t} and dU_{ocv}/dT denote the total volume of the core region, the current through pack ($I > 0$ for discharge and $I < 0$ for charge), the open-circuit voltage, the terminal voltage and the entropic heat coefficient, respectively. Here, the heat is assumed to be generated uniformly across the core region. The first term on the right-hand side of (5.2) is the irreversible heating and the second term is the reversible entropic heating from electrochemical reactions.

Similarly, conduction makes up the dominant part of the heat transfer in the case region, that is,

$$\rho_{\text{ca}} c_{\text{ca}} \frac{\partial T}{\partial t} = \lambda_{\text{ca}} \nabla^2 T, \quad (5.3)$$

where ρ_{ca} , c_{ca} and λ_{ca} are the mass density, specific heat and thermal conductivity of the case, respectively.

Next, consider the boundaries. To begin with, heat transfer on the core-case interface is mainly due to conduction. Assuming that core region and cell case are in perfect contact, the temperature and heat flux can be considered continuous on the interface [171]. Here, the boundary conditions on the core-case interface are given by

$$T|_{\text{core}} = T|_{\text{case}}, \quad \lambda_{\text{co}} \frac{\partial T}{\partial n} \Big|_{\text{core}} = \lambda_{\text{ca}} \frac{\partial T}{\partial n} \Big|_{\text{case}}, \quad (5.4)$$

where n is the normal direction of the core-case interface. The continuity of temperature and heat flux at a boundary also holds for the cell-cell interface with assumption that LiB cells are in close contact. Take cells 1 and 2 in Figure 5.1 as an example. On their cell-cell interface,

$$T|_{\text{cell1}} = T|_{\text{cell2}}, \lambda_{\text{ca}} \frac{\partial T}{\partial n} \Big|_{\text{cell1}} = \lambda_{\text{ca}} \frac{\partial T}{\partial n} \Big|_{\text{cell2}}, \quad (5.5)$$

where n is the normal direction of the interface of cells 1 and 2. Different from the regions and interfaces above, the cell-air interface will see heat transfer due to all of conduction, convection and radiation. Therefore, the energy balance on this boundary is

$$\lambda_{\text{ca}} \frac{\partial T}{\partial n} \Big|_{\text{cell}} = q_{\text{conv}} + q_{\text{r}}, \quad (5.6)$$

where q_{conv} and q_{r} represent the convective heat flux and the radiative heat flux on the cell-air interface, respectively. Here, q_{conv} and q_{r} are given by

$$q_{\text{conv}} = h_{\text{conv}} (T - T_{\text{air}}), \quad (5.7)$$

$$q_{\text{r}} = \varepsilon \sigma (T^4 - T_{\text{air}}^4), \quad (5.8)$$

where n , h_{conv} , ε , σ and T_{air} are the normal direction of the cell-air interface, the convective heat transfer coefficient, the emissivity, the Stefan-Boltzmann constant and the ambient air temperature, respectively. Note that if $|T - T_{\text{air}}|/T_{\text{air}} \ll 1$ as is often the case of LiB operation (both T and T_{air} in kelvins), (5.8) can be linearized around T_{air} as

$$q_{\text{r}} = 4\varepsilon \sigma T_{\text{air}}^3 (T - T_{\text{air}}) = h_{\text{r}} (T - T_{\text{air}}), \quad (5.9)$$

where h_{r} is the radiative heat transfer coefficient. Combining (5.7) and (5.9), the energy balance (5.6) can be rewritten as

$$\lambda_{\text{ca}} \frac{\partial T}{\partial n} \Big|_{\text{cell}} = h (T - T_{\text{air}}), \quad (5.10)$$

where h is the combined convective-radiative heat transfer coefficient.

Summarizing (5.1) through (5.10), one will obtain a complete thermal model for the considered LiB pack. This model builds on an awareness of the spatial temperature distribution. According to (5.1) and (5.2), heat will be produced within the core region when a current flows through the pack and transferred across the region by conduction. Conduction will also enable the propagation of heat within the case region, which is shown in (5.3). The boundary conditions at the core-case and cell-cell interfaces can be determined as in (5.4) and (5.5) on the reasonable assumption of continuous temperature and heat flux. Meanwhile, heat will travel from the cell surface to the air driven by a mix of conduction, convection and radiation, as shown in (5.6). The radiation effect at the cell-air interface is further linearized to simplify the model, as allowed by some mild conditions that can be easily met in a LiB pack's operation. This would lead to (5.10). Finally, the initial condition is also specified. In spite of the PDE-based representation, the obtained model still has constrained overall complexity advantageous for computation. In the next section, it will be further converted to ODE-based state-space form for the purpose of estimation design.

Remark 5.1. (Extensions of the thermal model). The thermal model above is developed in a basic battery pack setting and able to capture the most critical heat transfer phenomena underlying a pack's thermal behavior. Meanwhile, it can be readily extended to deal with more sophisticated settings. 1) Extension to a battery pack with a cooling system. The cooling effects can be accounted for in two ways. First, as suggested in [108], one can regard the cooling system as the boundaries of the battery pack's thermal model and thus modify the boundary conditions accordingly. Second, one can develop a separate heat transfer model for the cooling system and determine its interaction with the pack's model. The two models can be combined to offer a complete description of the battery pack under cooling conditions. This idea is exploited in [135, 123, 125]. 2) Extension to non-uniform heat generation. As is shown in (5.2), the heat generation is assumed to be even across a cell. The model built on this assumption can be modified to cope with spatially nonuniform heating if the gradient distribution of the potential and current density is captured. The literature includes some studies in this regard, see [17] and the references therein. In addition, the heat pro-

duced by electrical connections can also be included into the thermal model, which supposedly results from the passage of an electrical current through a resistance [107]. 3) Extension to heterogeneous cells. While the above considers identical cells, cells of the same type but of different state or aging level can be dealt with by changing the model parameters. Further, if cells of different electrochemistries are used in an extreme case, one can first build separate models for each cell type and then couple them using the same heat transfer principles to obtain a pack-level model. It is noteworthy that, though based on the basic model, the methodology to be proposed next is still applicable to these extended models for thermal field reconstruction.

5.3 Model Discretization

In this section, the PDE-based model in Section 5.2 is discretized in space and time to derive the state-space model using the finite difference formulation based on energy balance.

5.3.1 Discretization

The finite difference formulation is applied to fulfill the discretization. Consider the LiB pack comprising N_c cells shown in Figure 5.1. It can be subdivided into a large number of volume elements, giving rise to a three-dimensional grid with many nodes, see Figure 5.2. Specifically, a LiB cell, with dimensions of $L_x \times L_y \times L_z$, is subdivided by a grid with $(m \times n \times p)$ nodes. Here, two adjacent cells share the mp nodes on the contact interface between them. Then, each node can be labeled by its xyz -coordinates, i.e., (i, j, k) , which ranges from $(1, 1, 1)$ to $(m, nN_c - N_c + 1, p)$. A node is linked with a control volume. Within this volume, the temperature is considered uniform and assigned as the temperature of the node. The control volume of a node in the core region is uniform with a size of $(a \times b \times c)$, where a , b and c are the volume's edge length in x , y and z directions, respectively; yet the control volume of a node in the case region varies subject to its specific location. Since the case thickness is quite slim relative to the cell's dimensions, the temperature change in the thickness direction then will be minor. Thus for the node in the case, we

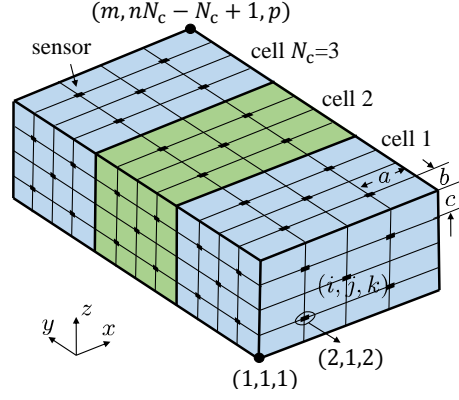


Figure 5.2: Schematic diagram of the nodes in a LiB pack comprising three cells, namely, $N_c = 3$.

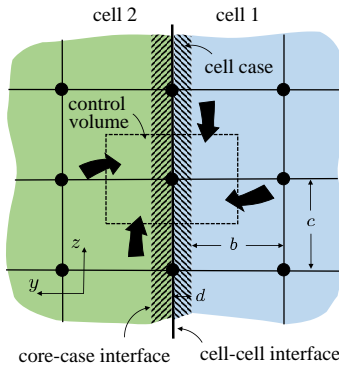


Figure 5.3: The y-z cross-section of the control volume of a node on the cell-cell interface.

let its edge parallel to the case's thickness direction be larger than the case's thickness length d , e.g., the control volume of (2,1,2) has a size of $(a \times (b/2 + d) \times c)$. As such, $L_x = (m - 1)a + 2d$, $L_y = (n - 1)b + 2d$, and $L_z = (p - 1)c + 2d$.

In this work, the finite-difference equation at each node is developed by the *energy balance approach* [25]. That is, the energy conservation law is applied to each volume. Here the direction of heat transfer on each volume's surfaces is assumed to be toward the node, see Figure 5.3. For a node in the core region, e.g., (2,2,2) of the grid in Figure 5.2, it receives the conductive heat flow

from its surrounding six nodes. The finite-difference equation can then be expressed as

$$\begin{aligned}
\rho c_p abc \frac{T_{i,j,k}^{t+\Delta t} - T_{i,j,k}^t}{\Delta t} = & \lambda_{co,x} \frac{T_{i+1,j,k}^t - T_{i,j,k}^t}{a} bc + \lambda_{co,x} \frac{T_{i-1,j,k}^t - T_{i,j,k}^t}{a} bc \\
& + \lambda_{co,y} \frac{T_{i,j+1,k}^t - T_{i,j,k}^t}{b} ac + \lambda_{co,y} \frac{T_{i,j-1,k}^t - T_{i,j,k}^t}{b} ac + \lambda_{co,z} \frac{T_{i,j,k+1}^t - T_{i,j,k}^t}{c} ab \\
& + \lambda_{co,z} \frac{T_{i,j,k-1}^t - T_{i,j,k}^t}{c} ab + \frac{abc(U_{ocv} - U_t - \frac{dU_{ocv}}{dT} T_{i,j,k}^t)}{V_{co}} I,
\end{aligned} \tag{5.11}$$

where ρ , c_p and Δt are the mass density, specific heat capacity of the cell and time step, respectively. Here, the thermophysical parameters, (e.g., ρ , c_p and λ_{co}), are time invariant, and dU_{ocv}/dT is also assumed to be independent of time within the operating range of the LiB pack [37]. Therefore, (5.11) can be regarded as a linear equation. The product of ρ and c_p is averaged based on the volume of the core region V_{co} and the case V_{ca} for simplification, namely,

$$\rho c_p = \frac{\rho_{co} c_{co} V_{co} + \rho_{ca} c_{ca} V_{ca}}{V_{co} + V_{ca}}. \tag{5.12}$$

The strategy in the calculation of $\rho_{co} c_{co}$, $\rho_{ca} c_{ca}$ and the three-dimensional thermal conductivity λ_{co} is the same with that used in [37] and thus omitted here.

Next, consider a node on the cell-cell boundary interface as shown in Figure 5.3. Its control volume in the y direction is $2d + b$. With properly selected edge length in x and z directions, the control volume of this node can be regarded as isothermal. Since the core region and the case region are made up of different materials, the thermal conduction length between the node and its adjacent node in the y direction is b instead of $b + d$. The finite-difference equation [34] for this

boundary is thus given by

$$\begin{aligned}
\rho c_p a c (b + 2d) \frac{T_{i,j,k}^{t+\Delta t} - T_{i,j,k}^t}{\Delta t} = & \lambda_{co,x} \frac{T_{i+1,j,k}^t - T_{i,j,k}^t}{a} b c + \lambda_{ca,x} \frac{T_{i+1,j,k}^t - T_{i,j,k}^t}{a} 2cd \\
& + \lambda_{co,x} \frac{T_{i-1,j,k}^t - T_{i,j,k}^t}{a} b c + \lambda_{ca,x} \frac{T_{i-1,j,k}^t - T_{i,j,k}^t}{a} 2cd + \lambda_{co,y} \frac{T_{i,j+1,k}^t - T_{i,j,k}^t}{b} a c \\
& + \lambda_{co,y} \frac{T_{i,j-1,k}^t - T_{i,j,k}^t}{b} a c + \lambda_{co,z} \frac{T_{i,j,k+1}^t - T_{i,j,k}^t}{c} a b + \lambda_{ca,z} \frac{T_{i,j,k+1}^t - T_{i,j,k}^t}{c} 2ad \\
& + \lambda_{co,z} \frac{T_{i,j,k-1}^t - T_{i,j,k}^t}{c} a b + \lambda_{ca,z} \frac{T_{i,j,k-1}^t - T_{i,j,k}^t}{c} 2ad + \frac{abc(U_{ocv} - U_t - \frac{dU_{ocv}}{dT} T_{i,j,k}^t)}{V_{co}} I. \quad (5.13)
\end{aligned}$$

Then, consider a node on the cell-air boundary interface of the pack. Take the node (2,1,2) in Figure 5.2 as an example. It receives both conductive heat transfer from its surrounding nodes and combined convective-radiative heat transfer from the ambient air. Similar to the nodes on the cell-cell interface, the edge length of the control volume of node (2,1,2) should be larger than d in the y direction, and the effective thermal conduction length between the node and its adjacent node in the y direction would also be equal to b . Hence, the finite-difference equation will be

$$\begin{aligned}
\rho c_p a c \left(\frac{b}{2} + d \right) \frac{T_{i,j,k}^{t+\Delta t} - T_{i,j,k}^t}{\Delta t} = & \lambda_{co,x} \frac{T_{i+1,j,k}^t - T_{i,j,k}^t}{a} \frac{b}{2} c + \lambda_{ca,x} \frac{T_{i+1,j,k}^t - T_{i,j,k}^t}{a} c d \\
& + \lambda_{co,x} \frac{T_{i-1,j,k}^t - T_{i,j,k}^t}{a} \frac{b}{2} c + \lambda_{ca,x} \frac{T_{i-1,j,k}^t - T_{i,j,k}^t}{a} c d + \lambda_{co,y} \frac{T_{i,j+1,k}^t - T_{i,j,k}^t}{b} a c \\
& + \lambda_{co,z} \frac{T_{i,j,k+1}^t - T_{i,j,k}^t}{c} a \frac{b}{2} + \lambda_{ca,z} \frac{T_{i,j,k+1}^t - T_{i,j,k}^t}{c} a d + \lambda_{co,z} \frac{T_{i,j,k-1}^t - T_{i,j,k}^t}{c} a \frac{b}{2} \\
& + \lambda_{ca,z} \frac{T_{i,j,k-1}^t - T_{i,j,k}^t}{c} a d + a c h (T_{air} - T_{i,j,k}^t) + \frac{abc(U_{ocv} - U_t - \frac{dU_{ocv}}{dT} T_{i,j,k}^t)}{2V_{co}} I. \quad (5.14)
\end{aligned}$$

Thus far, the PDE-based thermal model has been systematically discretized in space and time. The obtained equations present a linear ODE system that is high-dimensional but more amenable to estimation design and implementation.

5.3.2 Discrete-Time State-Space Model

Reorganizing the ODEs (5.11) through (5.14), one can derive a high-dimensional state-space representation of the following general form:

$$\mathbf{x}_{k+1} = \mathbf{F}_k \mathbf{x}_k + \mathbf{G}_k \mathbf{u}_k, \quad (5.15)$$

where \mathbf{x} is the state vector summarizing the temperature at all the nodes, \mathbf{u} the input vector based on the ambient temperature T_{air} and I the current. Specifically,

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}^{(1)} \\ \vdots \\ \mathbf{x}^{(N_c)} \end{bmatrix} \in \mathbb{R}^{N \times 1},$$

where

$$\mathbf{x}^{(l)} = \begin{bmatrix} T_{1,1,1}^{(l)} & \cdots & T_{m,n,p}^{(l)} \end{bmatrix}^\top \in \mathbb{R}^{N_l \times 1},$$

Here, for ease of the decomposition strategy design toward distributed estimation, the global state vector \mathbf{x} is established as an aggregation of local state vector $\mathbf{x}^{(l)}$ of each cell. To accommodate this representation, the pack-based numbering method introduced in Section 5.3.1 is modified accordingly to be cell-based—for each cell, the nodes are numbered from $(1, 1, 1)$ to (m, n, p) . Although the length of the vector \mathbf{x} is increased from $N - mp(N_c - 1)$ to N as a result of the nodes in the cell-cell interface accounted for separately, this will allow each cell to be treated as a subsystem and pave the way for distributed estimation algorithm design, as will be shown later. Besides, for notational convenience, the discrete time index will be denoted by k in the remainder of the chapter in replacement of t . The input \mathbf{u} is given by

$$\mathbf{u} = \begin{bmatrix} T_{\text{air}} & I \end{bmatrix}^\top \in \mathbb{R}^{2 \times 1},$$

which sums up the ambient temperature and current in the role of driving the system. Furthermore, the matrices \mathbf{F} and \mathbf{G} can be expressed in the block form:

$$\mathbf{F} = \begin{bmatrix} \mathbf{F}_{11} & \mathbf{F}_{12} & & \mathbf{0} \\ \mathbf{F}_{21} & \mathbf{F}_{22} & \ddots & \\ & \ddots & \ddots & \ddots \\ \mathbf{0} & & \ddots & \mathbf{F}_{N_c, N_c} \end{bmatrix} \in \mathbb{R}^{N \times N}, \quad \mathbf{G} = \begin{bmatrix} \mathbf{G}_1 \\ \vdots \\ \mathbf{G}_{N_c} \end{bmatrix} \in \mathbb{R}^{N \times 2},$$

where the block entries can be readily determined from (5.11)-(5.14). It is interesting to note that the matrix \mathbf{F} is not only tridiagonal but also diagonally dominant and sparse. This is mainly because each cell will only exchange heat with its adjacent cells and adjacent cells only share a limited number of nodes.

Let cell l in the pack be equipped with M_l thermocouples to measure the temperature. The measurement equation can then be expressed as

$$\mathbf{y}_k^{(l)} = \mathbf{H}_l \mathbf{x}_k^{(l)}, \quad (5.16)$$

where $\mathbf{y}^{(l)} \in \mathbb{R}^{M_l}$ and $\mathbf{H}_l \in \mathbb{R}^{M_l \times N_l}$ are the temperature measurement and measurement matrix, respectively. For \mathbf{H} , the entries corresponding to the nodes directly measured will be set equal to 1, and all the other entries zero. Aggregating all the measurements together, the measurement equation for the entire pack is then given by

$$\mathbf{y}_k = \mathbf{H} \mathbf{x}_k, \quad (5.17)$$

where

$$\mathbf{y} = \begin{bmatrix} \mathbf{y}^{(1)} \\ \vdots \\ \mathbf{y}^{(N_c)} \end{bmatrix} \in \mathbb{R}^{M \times 1}, \quad \mathbf{H} = \begin{bmatrix} \mathbf{H}_1 & & \\ & \ddots & \\ & & \mathbf{H}_{N_c} \end{bmatrix} \in \mathbb{R}^{M \times N},$$

where $M = N_c M_l$.

From above, (5.15) and (5.17) form the state-space model, which characterizes the propagation and measurement of a LiB pack's thermal dynamics based on a group of ODEs. With this model, one will be in a good position to conduct state estimation toward reconstructing the temperature field.

5.4 Kalman Filter Estimation

The state-space thermal model developed in Section 5.3 describes the dynamic thermal behavior of a LiB pack in charging/discharging. With this model, optimal estimation can be employed to enable reconstruction of the temperature field from the temperature measurement data. In this work, the KF will be exploited as the solution tool, which has achieved proven success in a wide range of engineering fields. However, the standard centralized KF will not fit with this application, because of a fine characterization of the LiB pack's thermal dynamics will imply a large size of the state space and cause heavy computation. This thus motivates the use of a distributed approach to build computational efficiency. In the following, the centralized KF will be introduced first, and a distributed version presented and analyzed in detail.

5.4.1 Centralized Kalman Filtering

Consider the state-space equations (5.15) and (5.17) and for convenience, replicate them with noise terms added as follows:

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{F}_k \mathbf{x}_k + \mathbf{G}_k \mathbf{u}_k + \mathbf{w}_k, \\ \mathbf{y}_k = \mathbf{H} \mathbf{x}_k + \mathbf{v}_k. \end{cases} \quad (5.18)$$

Here, the vectors with compatible dimensions, \mathbf{w}_k and \mathbf{v}_k , are added to account for the process noise and measurement noise that exist in the thermal dynamic processes of a LiB pack. They are assumed to be zero-mean Gaussian white noises with covariances of $\mathbf{Q} \geq 0$ and $\mathbf{R} > 0$, respectively. Suppose the initial state \mathbf{x}_0 is a Gaussian random vector with mean $\hat{\mathbf{x}}_{0|0}$ and covariance $\mathbf{P}_{0|0}$.

Note that $\hat{\mathbf{x}}_{0|0}$ indeed makes up the initial guess of \mathbf{x}_0 , and that $\mathbf{P}_{0|0}$ represents the estimation error covariance. Then application of the standard CKF to (5.18) can be performed at each future time instant. This procedure consists of two steps, *prediction* and *update*. The one-step-forward prediction yields the estimate of \mathbf{x}_k , denoted as $\hat{\mathbf{x}}_{k|k-1}$, using the measurements collected up to time $k-1$. Then upon the arrival of \mathbf{y}_k , $\hat{\mathbf{x}}_{k|k-1}$ will be updated to $\hat{\mathbf{x}}_{k|k}$ leveraging the information conveyed by \mathbf{y}_k about \mathbf{x}_k . In the meantime, the estimation error covariances associated with both estimates are computed accordingly to quantify the uncertainties of the obtained estimates.

When the state estimate $\hat{\mathbf{x}}_{k-1|k-1}$ is generated, the one-step-forward prediction can be made through

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{F}_{k-1}\hat{\mathbf{x}}_{k-1|k-1} + \mathbf{G}_{k-1}\mathbf{u}_{k-1}, \quad (5.19)$$

$$\mathbf{P}_{k|k-1} = \mathbf{F}_{k-1}\mathbf{P}_{k-1|k-1}\mathbf{F}_{k-1}^\top + \mathbf{Q}. \quad (5.20)$$

After $\hat{\mathbf{x}}_{k|k-1}$ is produced, it will be of next interest to investigate the updated state estimate. When the new measurement \mathbf{y}_k becomes available, the update step can be performed as follows to correct the prediction:

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k(\mathbf{y}_k - \mathbf{H}\hat{\mathbf{x}}_{k|k-1}), \quad (5.21)$$

$$\mathbf{K}_k = \mathbf{P}_{k|k-1}\mathbf{H}^\top(\mathbf{H}\mathbf{P}_{k|k-1}\mathbf{H}^\top + \mathbf{R})^{-1}, \quad (5.22)$$

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{K}_k\mathbf{H}\mathbf{P}_{k|k-1}. \quad (5.23)$$

The CKF will execute the above steps recursively over time to generate the state estimate at each time instant. The CKF is the optimal among all filters if the aforementioned Gaussian assumptions are satisfied and optimal among all linear filters in the non-Gaussian case. Such an optimality establishes the foundation for the practical utility of the CKF approach. The CKF, however, is not well suited to estimate the temperature for a LiB pack, because the high-dimensional thermal model of the LiB pack will imply considerable computation (detailed computational complexity analysis will be given in Section 5.4.4). Hence, the CKF will be distributed next to bring down the computational cost.

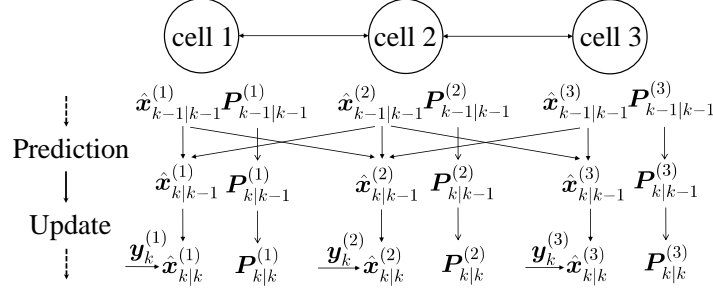


Figure 5.4: A schematic of the DKF for a three-cell pack.

5.4.2 Distributed Kalman Filtering

Rather than estimate the global state in a centralized manner, the DKF will consider the pack system as a combination of multiple cell-based subsystems and run a series of local KFs in parallel, each one corresponding to a cell. Because of the mutual influence between the cells and their thermal behavior, the local KFs will exchange information according to the existing communication topology to accomplish the estimation. The local estimates, when collected and put together, will comprise a complete picture of the entire pack's temperature field.

Consider a LiB pack composed of N_c cells wired in series, which are numbered in order from 1 to N_c . A three-cell pack is shown in Figure 5.4 as an example. For cell l and i , they are said to be neighbors if they are adjacent. The neighborhood of l , \mathcal{N}_l , is defined as the set of its neighbor cells, and in this setting, $\mathcal{N}_l = \{l-1, l+1\}$. Here, it is assumed that cell l stores the structural information of its neighbors, specifically, \mathbf{F}_{li} for $i \in \mathcal{N}_l$. To minimize the communication cost, information exchange protocol is only enforced between neighbors. That is, cell l only communicates with \mathcal{N}_l .

As such, the state-space equation for subsystem S_l can be written as

$$\begin{cases} \mathbf{x}_{k+1}^{(l)} = \mathbf{F}_{ll,k} \mathbf{x}_k^{(l)} + \sum_{i \in \mathcal{N}_l} \mathbf{F}_{li,k} \mathbf{x}_k^{(i)} + \mathbf{G}_{l,k} \mathbf{u}_k + \mathbf{w}_k^{(l)}, \\ \mathbf{y}_k^{(l)} = \mathbf{H}_l \mathbf{x}_k^{(l)} + \mathbf{v}_k^{(l)}, \end{cases} \quad (5.24)$$

where cell-wise decomposition is also applied to \mathbf{w}_k and \mathbf{v}_k . It is noted that the evolution of cell l 's state is not only self-driven but also affected by the neighboring cells, as a result of the cell-to-cell

coupling of the thermal dynamics. All cells yet share the same \mathbf{u}_k because the serial connection implies the same charging/discharging current across the circuit. In addition, each cell is only aware of its own temperature measurements.

For the above cell l -based subsystem, the KF approach can be adjusted for local state estimation. This can be attained by applying the prediction-update procedure in analogy to (5.19)-(5.23). Specifically, the prediction can be given by

$$\hat{\mathbf{x}}_{k|k-1}^{(l)} = \mathbf{F}_{ll,k-1}\hat{\mathbf{x}}_{k-1|k-1}^{(l)} + \sum_{i \in \mathcal{N}_l} \mathbf{F}_{li,k-1}\hat{\mathbf{x}}_{k-1|k-1}^{(i)} + \mathbf{G}_{l,k-1}\mathbf{u}_{k-1}, \quad (5.25)$$

$$\mathbf{P}_{k|k-1}^{(l)} = \mathbf{F}_{ll,k-1}\mathbf{P}_{k-1|k-1}^{(l)}\mathbf{F}_{ll,k-1}^\top + \mathbf{Q}_l. \quad (5.26)$$

Here, cell l 's state prediction, $\hat{\mathbf{x}}_{k|k-1}^{(l)}$, depends on not only its own but also its neighbors' state estimates from the previous time instant. All cells are still driven by the same input \mathbf{u}_k because cells in serial connection are subjected to the same current. On its arrival, $\mathbf{y}_k^{(l)}$ can be used to update $\hat{\mathbf{x}}_{k|k-1}^{(l)}$ as follows:

$$\hat{\mathbf{x}}_{k|k}^{(l)} = \hat{\mathbf{x}}_{k|k-1}^{(l)} + \mathbf{K}_k^{(l)}(\mathbf{y}_k^{(l)} - \mathbf{H}_l\hat{\mathbf{x}}_{k|k-1}^{(l)}), \quad (5.27)$$

$$\mathbf{K}_k^{(l)} = \mathbf{P}_{k|k-1}^{(l)}\mathbf{H}_l^\top (\mathbf{H}_l\mathbf{P}_{k|k-1}^{(l)}\mathbf{H}_l^\top + \mathbf{R}_l)^{-1}, \quad (5.28)$$

$$\mathbf{P}_{k|k}^{(l)} = \mathbf{P}_{k|k-1}^{(l)} - \mathbf{K}_k^{(l)}\mathbf{H}_l\mathbf{P}_{k|k-1}^{(l)}. \quad (5.29)$$

Note that no information exchange with neighbors is required in the update step. The two steps shown in (5.25) through (5.29) then constitute the DKF algorithm. Running the DKF in parallel for each cell, the local temperature field will be estimated through time, and all the local estimation results when combined will provide the full view of the pack's temperature field.

It is noteworthy that the above DKF algorithm, compared with the CKF, involves approximation for two reasons. First, each cell only has a local rather than global knowledge of the system's dynamic behavior, making certain information loss inherent in each local DKF. To see this, the nominal prediction error covariance $\mathbf{P}_{k|k-1}^{(l)}$, differing from the CKF, evolves only from its prede-

cessor without fusion of the counterparts of the other cells, as shown in (5.26). This shows $\mathbf{P}_{k|k-1}^{(l)}$ is only approximate to the true prediction error covariance. Second, part of the approximation is made to reduce the communication and computation costs. Looking at (5.26) again, one can see that a local cell does not consider its neighbors in its forward propagation of its prediction error covariance. This will obviate the need for the exchange of the estimation error covariance between neighboring cells and further, the local computational effort. However, such an approximation will not seriously compromise the estimation accuracy. Since \mathbf{F} is diagonally dominant and \mathbf{H} block-diagonal due to the pack's serial connection architecture, \mathbf{P} will also be diagonally dominant. The self-propagation of the local estimation error covariance, as a result, will not bring much loss of estimation accuracy. The above DKF algorithm, to our knowledge, is the most computationally fast among its kind and highly suitable for the considered LiB pack thermal monitoring problem.

5.4.3 Steady-State Distributed Kalman Filtering

An opportunity can be identified that, if some mild reduction is introduced for the considered thermal model, we can obtain another DKF approach with much higher computational efficiency. To be specific, consider the heat generation equation (5.2). Many studies in the literature suggest that its second term often has a negligible magnitude in comparison with the first term and thus can be ignored [122, 108]. With this simplification, (5.2) can be reduced as $q = I(U_{ocv} - U_t)/V_{co}$. It is then found that \mathbf{F} becomes time-invariant in this case, which will allow us to develop a more computationally efficient DKF for temperature field reconstruction. The development is as follows.

Before proceeding further, we make an assumption at first.

Assumption 5.1. The pair $(\mathbf{F}_l, \mathbf{H}_l)$ is detectable and the pair $(\mathbf{F}_l, \mathbf{Q}_l^{\frac{1}{2}})$ stabilizable for $l \in \{1, \dots, N_c\}$.

If Assumption 5.1 holds, the DKF algorithm in (5.25) through (5.29) will achieve steady state. Specifically, $\mathbf{P}_{k|k-1}^{(l)}$ will converge to a unique stabilizing solution, $\bar{\mathbf{P}}^{(l)}$, of the discrete algebraic

Riccati equation

$$\mathbf{X} = \mathbf{F}_{ll} \mathbf{X} \mathbf{F}_{ll}^\top - \mathbf{F}_{ll} \mathbf{X} \mathbf{H}_l^\top \left(\mathbf{H}_l \mathbf{X} \mathbf{H}_l^\top + \mathbf{R}_l \right)^{-1} \mathbf{H}_l \mathbf{X} \mathbf{F}_{ll}^\top + \mathbf{Q}_l,$$

where \mathbf{X} is an unknown symmetric positive-definite matrix with compatible dimensions. The gain matrix $\mathbf{K}_k^{(l)}$ in (5.28) will consequently reach a fixed point

$$\bar{\mathbf{K}}^{(l)} = \bar{\mathbf{P}}^{(l)} \mathbf{H}_l^\top \left(\mathbf{H}_l \bar{\mathbf{P}}^{(l)} \mathbf{H}_l^\top + \mathbf{R}_l \right)^{-1}, \quad (5.30)$$

which can ensure $\mathbf{F}_{ll}(\mathbf{I}_l - \bar{\mathbf{K}}^{(l)} \mathbf{H}_l)$ to be stable [13]. With fixed $\bar{\mathbf{P}}^{(l)}$ and $\bar{\mathbf{K}}^{(l)}$, the state prediction and update can be accomplished more efficiently:

$$\hat{\mathbf{x}}_{k|k-1}^{(l)} = \mathbf{F}_{ll} \hat{\mathbf{x}}_{k-1|k-1}^{(l)} + \sum_{i \in \mathcal{N}_l} \mathbf{F}_{li} \hat{\mathbf{x}}_{k-1|k-1}^{(i)} + \mathbf{G}_l \mathbf{u}_{k-1}, \quad (5.31)$$

$$\hat{\mathbf{x}}_{k|k}^{(l)} = \hat{\mathbf{x}}_{k|k-1}^{(l)} + \bar{\mathbf{K}}^{(l)} \left(\mathbf{y}_k^{(l)} - \mathbf{H}_l \hat{\mathbf{x}}_{k|k-1}^{(l)} \right), \quad (5.32)$$

which together form the SS-DKF algorithm. It is seen that the SS-DKF does not maintain the estimation error covariance and that its gain matrix can be computed offline prior to the estimation run. Although this comes at certain sacrifice of estimation accuracy, it still maintains stability under some assumptions and presents much appeal from a computational perspective. Before moving on to the computational complexity analysis in Section 5.4.4, the stability of the SS-DKF algorithm is examined in the remainder of this section.

We define the real state error $\mathbf{e}_k = \hat{\mathbf{x}}_{k|k} - \mathbf{x}_k$. For the real state error $\mathbf{e}_k^{(l)}$ at subsystem l , combining (5.31) and (5.32), we have

$$\mathbf{e}_k^{(l)} = (\mathbf{I} - \bar{\mathbf{K}}^{(l)} \mathbf{H}_l) \mathbf{F}_{ll} \mathbf{e}_{k-1}^{(l)} + \sum_{i \in \mathcal{N}_l} (\mathbf{I} - \bar{\mathbf{K}}^{(l)} \mathbf{H}_l) \mathbf{F}_{li} \mathbf{e}_{k-1}^{(i)} \quad (5.33)$$

$$- (\mathbf{I} - \bar{\mathbf{K}}^{(l)} \mathbf{H}_l) \mathbf{w}_{k-1}^{(l)} + \bar{\mathbf{K}}^{(l)} \mathbf{v}_k^{(l)}, \quad (5.34)$$

where $\mathbf{I} \in \mathbb{R}^{N_l \times N_l}$. Aggregating $\mathbf{e}_k^{(l)}$ for $l \in \{1, \dots, N_c\}$ together yields

$$\mathbf{e}_k = (\mathbf{I} - \bar{\mathbf{K}}\mathbf{H})\mathbf{F}\mathbf{e}_{k-1} - (\mathbf{I} - \bar{\mathbf{K}}\mathbf{H})\mathbf{w}_{k-1} + \bar{\mathbf{K}}\mathbf{v}_k, \quad (5.35)$$

where $\bar{\mathbf{K}} = \text{blkdiag} \left\{ \bar{\mathbf{K}}^{(1)}, \dots, \bar{\mathbf{K}}^{(N_c)} \right\}$ and identity matrix $\mathbf{I} \in \mathbb{R}^{N \times N}$.

The stability of the SS-DKF algorithm is summarized as follows.

Theorem 5.1. Let $\mathbf{A} = (\mathbf{I} - \bar{\mathbf{K}}\mathbf{H})\mathbf{F}$ and $\mathbf{B} = (\mathbf{I} - \bar{\mathbf{K}}\mathbf{H})\mathbf{Q}(\mathbf{I} - \bar{\mathbf{K}}\mathbf{H})^\top + \bar{\mathbf{K}}\mathbf{R}\bar{\mathbf{K}}^\top$. If \mathbf{A} is stable, the true error covariance $\boldsymbol{\Sigma}_k = \mathbf{E} [\mathbf{e}_k \mathbf{e}_k^\top]$ will converge to the unique solution of the discrete-time Lyapunov equation

$$\boldsymbol{\Sigma} = \mathbf{A}\boldsymbol{\Sigma}\mathbf{A}^\top + \mathbf{B}. \quad (5.36)$$

Proof. It is seen that the propagation of $\boldsymbol{\Sigma}_k$ is governed by

$$\boldsymbol{\Sigma}_k = \mathbf{A}\boldsymbol{\Sigma}_{k-1}\mathbf{A}^\top + \mathbf{B}. \quad (5.37)$$

Since \mathbf{A} is stable, $\lim_{k \rightarrow \infty} \boldsymbol{\Sigma}_k = \boldsymbol{\Sigma}$ (see Chapter 3.3 in [194]). □

From above, Assumption 5.1 lays the foundation for the derivation of the SS-DKF, and Theorem 5.1 indicates that a stable $(\mathbf{I} - \bar{\mathbf{K}}\mathbf{H})\mathbf{F}$ can guarantee the stability of the SS-DKF algorithm. However, a question then arises: *will Assumption 5.1 and the stability of $(\mathbf{I} - \bar{\mathbf{K}}\mathbf{H})\mathbf{F}$ hold for the LiB pack model in (5.18)?*

An examination is given as follows. First, consider Assumption 5.1. Note that the thermal physics imply that the model established in Section 5.2 is stable if the LiB pack operates normally and the numerical stability criteria is satisfied in discretization. Hence, the model matrix \mathbf{F} will be stable. Next, we partition the model matrix \mathbf{F} into the following form

$$\mathbf{F} = \mathbf{F}_d + \mathbf{F}_{od}, \quad (5.38)$$

where the subscripts d and od, respectively, denote diagonal blocks and off-diagonal blocks and

$\mathbf{F}_d = \text{blkdiag}\{\mathbf{F}_{11}, \dots, \mathbf{F}_{N_c, N_c}\}$. According to Corollary 5.6.14 in [89], one will have

$$\rho(\mathbf{F}_{\text{od}}) = \lim_{k \rightarrow \infty} \|\mathbf{F}_{\text{od}}^k\|^{1/k}. \quad (5.39)$$

It is interesting to note that $\mathbf{F}_{\text{od}}^2 = \mathbf{0}$ in this application due to the serial structure of the LiB pack, which will be further shown in Section 5.5. Therefore, $\rho(\mathbf{F}_{\text{od}}) = 0$ in this case. Invoking Lemma 5.6.10 in [89], there exists a matrix norm $\|\cdot\|_*$ for any given $\varepsilon > 0$ such that

$$\rho(\mathbf{F}_d) \leq \|\mathbf{F}_d\|_* \leq \rho(\mathbf{F}_d) + \varepsilon. \quad (5.40)$$

Since it satisfies the triangle inequality, then

$$\begin{aligned} \|\mathbf{F}_d\|_* &\leq \|\mathbf{F}\|_* + \|\mathbf{F}_{\text{od}}\|_* \\ &\leq (\rho(\mathbf{F}) + \varepsilon_1) + (\rho(\mathbf{F}_{\text{od}}) + \varepsilon_2), \end{aligned} \quad (5.41)$$

where $\varepsilon_1 > 0$ and $\varepsilon_2 > 0$. Since $\rho(\mathbf{F}_{\text{od}}) = 0$, (5.41) can be rewritten as

$$\|\mathbf{F}_d\|_* \leq \rho(\mathbf{F}) + \varepsilon_1 + \varepsilon_2. \quad (5.42)$$

Thus, one can always find ε_1 and ε_2 to prove that \mathbf{F}_d is stable, thus validating Assumption 5.1.

Now, consider the stability of $(\mathbf{I} - \bar{\mathbf{K}}\mathbf{H})\mathbf{F}$. For notational simplicity, we denote $\mathbf{I} - \bar{\mathbf{K}}\mathbf{H}$ with $\tilde{\mathbf{I}}$. Then the objective is to show that $\tilde{\mathbf{I}}\mathbf{F}$ is stable. Recalling the matrix norm $\|\cdot\|_*$ in (5.40), it is also submultiplicative and implies

$$\begin{aligned} \|\tilde{\mathbf{I}}\mathbf{F}\|_* &\leq \|\tilde{\mathbf{I}}\mathbf{F}_d\|_* + \|\tilde{\mathbf{I}}\mathbf{F}_{\text{od}}\|_* \leq \|\tilde{\mathbf{I}}\mathbf{F}_d\|_* + \|\tilde{\mathbf{I}}\|_* \|\mathbf{F}_{\text{od}}\|_* \\ &\leq (\rho(\tilde{\mathbf{I}}\mathbf{F}_d) + \varepsilon_3) + (\rho(\tilde{\mathbf{I}}) + \varepsilon_4)(\rho(\mathbf{F}_{\text{od}}) + \varepsilon_5) \\ &\leq \rho(\tilde{\mathbf{I}}\mathbf{F}_d) + \varepsilon_3 + \rho(\tilde{\mathbf{I}})\varepsilon_5 + \varepsilon_4\varepsilon_5, \end{aligned} \quad (5.43)$$

where $\varepsilon_3 > 0$, $\varepsilon_4 > 0$, and $\varepsilon_5 > 0$. Then, because $\mathbf{F}_{ll}(\mathbf{I}_l - \bar{\mathbf{K}}^{(l)}\mathbf{H}_l)$ is stable, the matrix $\mathbf{F}_d\tilde{\mathbf{I}}$ is stable. Following that $\lim_{k \rightarrow \infty}(\mathbf{F}_d\tilde{\mathbf{I}})^k = \mathbf{0}$, $(\tilde{\mathbf{I}}\mathbf{F}_d)^k$ can be constructed as $\tilde{\mathbf{I}}(\mathbf{F}_d\tilde{\mathbf{I}})^{k-1}\mathbf{F}_d$, such that $\lim_{i \rightarrow \infty}(\tilde{\mathbf{I}}\mathbf{F}_d)^k = \mathbf{0}$. Therefore, it is always possible to find ε_3 , ε_4 and ε_5 to make the right-hand side of (5.43) smaller than 1. Subsequently, $\rho(\tilde{\mathbf{I}}\mathbf{F}) < 1$ and $(\mathbf{I} - \bar{\mathbf{K}}\mathbf{H})\mathbf{F}$ is stable.

Remark 5.2. (Extension to thermal runaway detection). In above, the DKF and SS-DKF are developed to reconstruct the temperature field of a battery pack. They can be used as a tool to monitor the spatially distributed thermal behavior critical for a battery pack's safety. An extension of them to detect thermal runaway can be hopefully made. An idea is to consider the thermal runaway as an unknown disturbance that abruptly appears and applies to the model in (5.18). Then, the thermal runaway detection can be formulated as the problem of disturbance detection. KF-based approaches have been studied extensively for disturbance detection in the literature, e.g., [78], and can be potentially exploited here. Combining this idea and the design in this work, we can promisingly build distributed KF-based approaches for thermal runaway detection. This will be an important part of our future work.

5.4.4 Computational Complexity Analysis

As aforementioned, the objective of distributing the CKF across the cells is to improve the computational efficiency toward enabling real-time reconstruction of a LiB pack's temperature field. In this subsection, the CKF, DKF and SS-DKF algorithms are analyzed and compared in terms of the computational complexity. The analysis is based on the number of arithmetic operations needed by each algorithm. Before proceeding further, let us consider the basic matrix operations. For two $n \times n$ matrices, their addition involves n^2 elementary additions, and their multiplication involves n^3 elementary multiplications and $(n-1)n^2$ elementary additions. The inverse of an $n \times n$ matrix requires n^3 elementary multiplications and n^3 elementary additions. These are the basic algebraic arithmetics involved in the considered algorithms. The complexity of each algorithm can be assessed by summing up all the arithmetic operations required at each time instant, with the results shown in Table 5.1.

Table 5.1: Arithmetic operation requirements of the CKF, DKF and SS-DKF algorithms.

Algorithm	Number of multiplications	Number of additions	Complexity
CKF	$3N^3 + 2N^2M + 2NM^2 + M^3 + N^2 + 2NM + 2N$	$3N^3 + 2N^2M + 2NM^2 + M^3 - N^2 + N$	$O(N^3)$
DKF	$N_c (3N_l^3 + 2N_l^2M_l + 2N_lM_l^2 + M_l^3 + 3N_l^2 + 2N_lM_l + 2N_l)$	$N_c (3N_l^3 + 2N_l^2M_l + 2N_lM_l^2 + M_l^3 + N_l^2 + N_l)$	$O(N_cN_l^3)$
SS-DKF	$N_c (2N_l^2 + 2N_lM_l + 2N_l)$	$N_c (3N_l^2 + 2N_lM_l + N_l)$	$O(N_cN_l^2)$

It is demonstrated in Table 5.1 that the CKF has the heaviest computation at $O(N^3)$, which increases cubically with the size of the state space of the entire pack. This also implies that, when a pack has more cells, the computation would rise cubically with the cell number. The computational complexity at such a level can be unaffordable by a real-world onboard computing platforms. By comparison, the DKF is much more efficient than the CKF. Given that $N = N_cN_l$ with $N_c \ll N_l$, the DKF's arithmetic operations at $O(N_cN_l^3)$ are only about one N_c^2 -th of the CKF's. In addition, with the computation increasing only linearly with the cell number, the DKF well lends itself to parallel processing, where the estimation for each cell is performed on a separate micro-processor at a complexity $O(N_l^3)$. In this scenario, an increase in the cell number will not add cost to the existing micro-processors. The SS-DKF unsurprisingly is the most computationally competitive. Its complexity at $O(N_cN_l^2)$ is even one order less than that of the DKF. Just like the DKF, it is also well suited for execution based on parallel processing.

5.5 Numerical Simulation

To reconstruct a LiB pack's temperature field, the previous sections presented a spatially resolved thermal model and distributed the KF for computationally fast estimation. In this section, numerical simulation with a practical LiB pack is offered to illustrate the effectiveness of the proposed work. The simulation is performed using MATLAB R2016a.

Table 5.2: LiB cell parameters used in the simulation [37].

Layer/ Material	Thickness [cm]	Density [kg/m ³]	Heat capacity [J/(kg·K)]	Conductivity [W/(m·K)]
Anode*	0.0116	1347.33	1437.4	1.04
Cathode*	0.014	2328.5	1269.21	1.58
Separator*	0.0035	1008.98	1978.16	0.3344
Cu foil*	0.0014	8933	385	398
Al foil*	0.002	2702	903	238
Metal case*	0.07	2770	875	170
Contact layer*	0.05	1129.95	2055.1	0.60

Note: * denotes a component of cell unit and * a component of cell case.

5.5.1 Simulation Setting

Consider a LiB pack that consists of three large-format high-capacity prismatic LiB cells connected and stacked in series. Here, the cells are the same ones as in [37]. Each cell has a capacity of 185.3 Ah and is 19.32 cm long, 10.24 cm wide, and 10.24 cm high. As mentioned in Section 5.2, the cell has two portions: the core region and the metallic case. The core region is 19.08 cm long, 10 cm wide, and 10 cm high, housing three hundred smaller cell units connected in parallel. The structure of a cell unit is schematically shown in Figure 5.5. The key parameters of the LiB cell are summarized in Table 5.2. In the simulation, it is assumed that the LiB pack operates in an environment with temperature maintained at 300 K. The convective heat transfer coefficient, emissivity on pack surface, and entropic heat transfer coefficient dU_{ocv}/dT are set to be 30 W/(m²·K), 0.25, and 0.00022 VK⁻¹ [37], respectively. The battery pack is discharged using a time-varying current profile, which is shown in Figure 5.6 and derived from the Urban Dynamometer Driving Schedule (UDDS) [3]. Sensors are mounted on the battery pack as shown in Figure 5.2. That is, five sensors are placed on each cell-air interface of a cell. Such a placement is straightforward and easy to implement. Associated with this, an intriguing question is how to optimally deploy the sensors

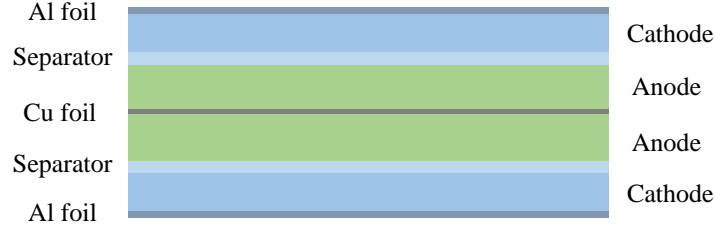


Figure 5.5: Schematic diagram of a basic unit in a LiB cell.

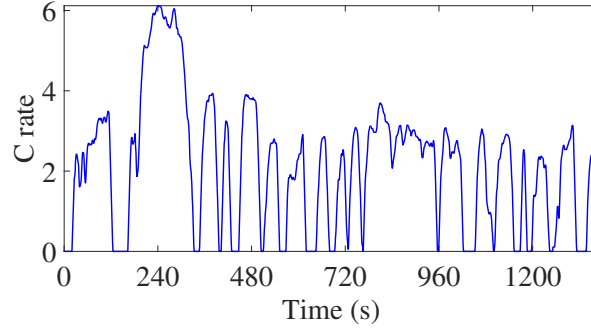


Figure 5.6: Discharging current profile based on UDDS.

toward achieving satisfactory estimation performance with a minimum number of sensors. While some results are reported in the literature, e.g., [125, 57, 216], further research is still required to fully address this question.

The thermal dynamics of the considered pack can be characterized by the PDE-based model in Section 5.2. Each cell is gridded in space with $m = 9$, $n = 5$ and $p = 5$ and in time with $\Delta t = 1$ s. In general, one can increase m , n and p and reduce Δt to increase the accuracy of simulation. This, however, will come at the sacrifice of computational efficiency. Another risk lies in numerical instability, which can be caused if the selected m , n , p and Δt fail to satisfy certain conditions [25]. To find a satisfactory set, one can consider a few candidates. S/he can first check the numerical stability for each set using the conditions in [25]. Then, run the simulation for the sets that pass the check, and choose the set that leads to acceptable accuracy with minimum computational cost. This process understandably may require repeated trial effort.

By discretization, the PDEs are converted to a state-space model as shown in Section 5.3. On each cell-air interface of the cell, five sensors are deployed as shown in Figure 5.2. It follows that

the model matrices \mathbf{F} and \mathbf{H} for this pack take the following form:

$$\mathbf{F} = \begin{bmatrix} \mathbf{F}_{11} & \mathbf{F}_{12} & & \\ \mathbf{F}_{21} & \mathbf{F}_{22} & \mathbf{F}_{23} & \\ & \mathbf{F}_{32} & \mathbf{F}_{33} & \end{bmatrix} \in \mathbb{R}^{675 \times 675}, \mathbf{H} = \begin{bmatrix} \mathbf{H}_1 & & \\ & \mathbf{H}_2 & \\ & & \mathbf{H}_3 \end{bmatrix} \in \mathbb{R}^{70 \times 675}.$$

In the simulation, the pack's initial temperature is 300 K, the same with the ambient temperature. Yet, for the purpose of illustrating the estimation, the initial guess is 295 K in the simulation. The noise covariance matrices \mathbf{Q} and \mathbf{R} are chosen as

$$\mathbf{Q} = 0.05^2 \cdot \mathbf{I}^{675 \times 675}, \mathbf{R} = 0.3^2 \cdot \mathbf{I}^{70 \times 70},$$

where \mathbf{I} is the identity matrix.

5.5.2 Simulation Results

Consider the stochastic discrete-time state-space model (5.18). The real temperature field is obtained by running the state equation of (5.18) using MATLAB R2016a with the effects of process noise included. Sensor-based measurements are obtained according to the measurement equation of (5.18), which are subjected to sensor noise. Using the measurements and based on the model, the CKF and DKF are applied to reconstruct the temperature field. The simulation results are summarized in Figure 5.7. The first row shows the true temperature field that evolves over time. Here, trilinear spatial interpolation is used to generate spatially continuous temperature fields. As is shown, the pack sees an obvious temperature rise, despite the convection cooling and only three cells. In addition, it can be easily found that the temperature differs spatially across the pack, with a high gradient buildup at the end of the simulation. The second row shows the reconstructed temperature fields using the CKF, and the third row shows the ones based on the DKF. It is seen that, although the initial guess differs from the truth, both the CKF and DKF can generate temperature field estimation that gradually catches up with the truth. For both, satisfactory reconstruction starts

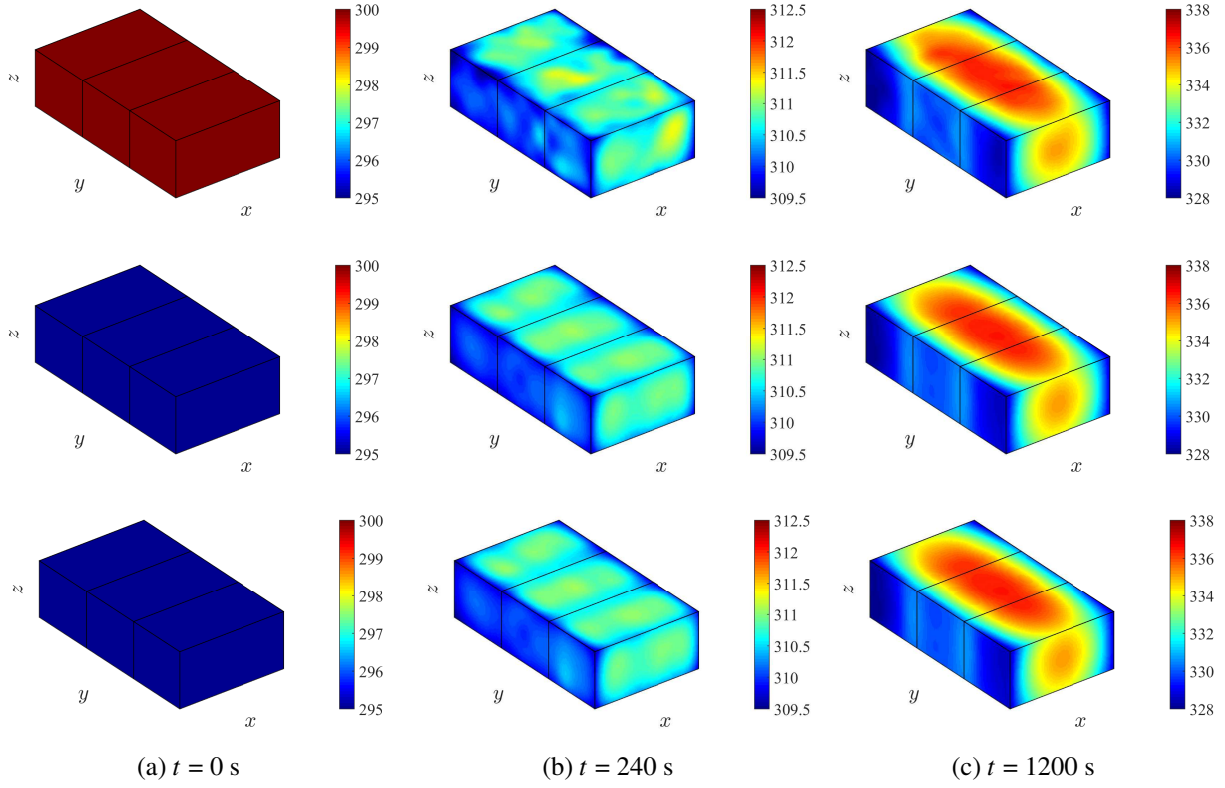


Figure 5.7: First row: real temperature fields through time for the LiB pack discharged at UDDS-based current loads. Second row: CKF-based reconstruction. Third row: DKF-based reconstruction.

from around 240 s and is maintained afterwards. Improved estimation accuracy can be expected if the initial temperature guess is made closer to the truth, which is not difficult on many practical occasions, since a LiB pack at rest for a long period will have almost the same temperature with the ambient environment.

In Figure 5.7, the CKF and DKF present approximately the same estimation results. To further evaluate their accuracy, we consider the following metric:

$$\frac{1}{N} \text{trace} \left(\mathbf{E} \left[(\hat{\mathbf{x}}_{k|k} - \mathbf{x}_k)(\hat{\mathbf{x}}_{k|k} - \mathbf{x}_k)^\top \right] \right),$$

which is the trace of the estimation error covariance averaged over the state space and represents a statistical quantification of the state estimation error. Evaluation of the two algorithms based on this metric is illustrated in Figure 5.8. One can see that they both exhibit a decreasing trend over

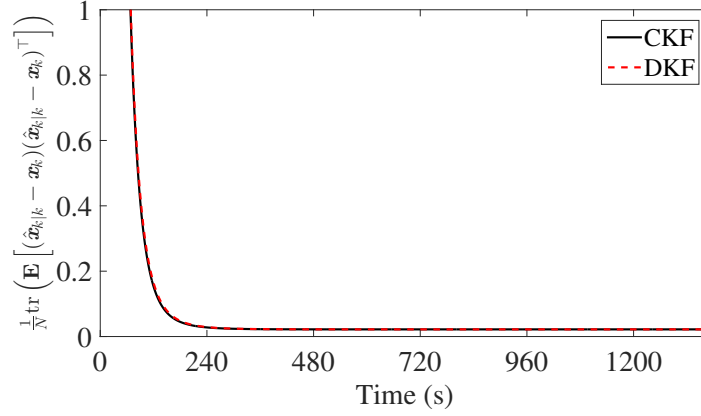


Figure 5.8: Evaluation of the estimation error over time.

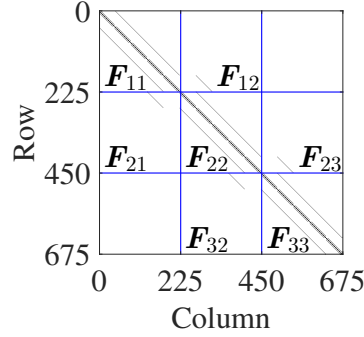


Figure 5.9: Matrix \mathbf{F} plotted in black and white.

time and remain quite close to each other. This implies that the DKF is quite comparable to the CKF in estimation performance. In addition, as analyzed in Section 5.4.4, the DKF considerably outperforms the CKF in terms of computational complexity, which thus makes it a more desirable solution for the considered problem of temperature field reconstruction.

Next, let us further examine the application of the SS-DKF approach proposed in Section 5.4.3. In this case, the heat generation model (5.2) is simplified by removing the negligible reversible entropic heating, and consequently, \mathbf{F} is time-invariant. Figure 5.9 offers a visual black-white display of \mathbf{F} , which shows non-zero elements in black and zero elements in white. Figure 5.9 shows that \mathbf{F} is diagonally dominant, fundamentally ascribed to the pack's serial architecture. It is

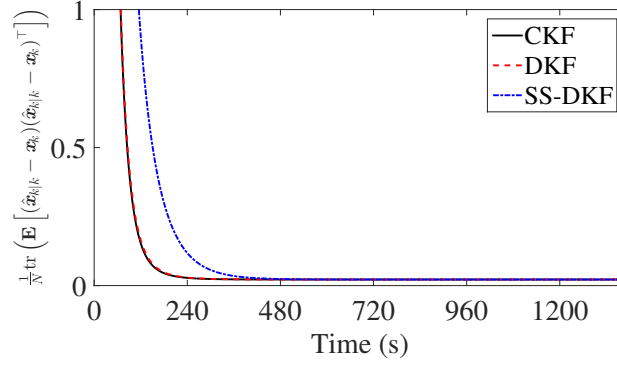


Figure 5.10: Evaluation of the estimation error over time when the simplified heat generation model is used.

also found that \mathbf{F}_{od}^2 equals $\mathbf{0}$, where

$$\mathbf{F}_{\text{od}} = \begin{bmatrix} & \mathbf{F}_{12} & \\ \mathbf{F}_{21} & & \mathbf{F}_{23} \\ & \mathbf{F}_{32} & \end{bmatrix} \in \mathbb{R}^{675 \times 675}.$$

Specifically, it is verifiable that each row vector of \mathbf{F}_{od} multiplied by any column vector $\in \mathbb{R}^{675 \times 1}$ in \mathbf{F}_{od} gives rise to zero. It can be further verified that this finding is applicable to any LiB pack with the considered configuration. Having established the stability of \mathbf{F} and zero spectral radius of \mathbf{F}_{od} , the SS-DKF algorithm will be asymptotically stable, as discussed in Section 5.4.3. Now running the CKF, DKF and SS-DKF, one can reconstruct the temperature field. The visual demonstration of the temperature field estimation is omitted to save space. However, a comparison of accuracy is provided in Figure 5.10. It is observed that the SS-DKF is less accurate in the initial stage. However, it can achieve approximately the same accuracy after about 400 s. Given this result and its superior computational efficiency, the SS-DKF can be a worthy tool in practice.

5.6 Summary

This chapter studied the real-time reconstruction of the three-dimensional temperature field across a LiB pack and addressed this challenge through model-based temperature estimation. A ther-

mal model was presented first to capture the thermal dynamic process of a LiB pack, which is based on heat transfer and energy balance analysis. Compared to the lumped models and thermo-electrochemical models, such a model achieves a balance between computational efficiency and physical integrity. Based on the model, the well-known KF approach was then distributed to achieve global temperature field reconstruction through localized estimation, reducing the computational complexity remarkably. A DKF algorithm, which is straightforward but well fits with the considered problem, was offered, and its steady-state version, SS-DKF, would require even less computation time. A detailed computational complexity analysis highlighted the advantages of the distributed estimation approaches. Simulation with a LiB pack based on genuine cells demonstrated the effectiveness of the DKF and SS-DKF algorithms. The results point to the promises of the proposed methodology for creating more informative thermal monitoring toward improving thermal safety of LiB packs.

Chapter 6

Conclusions

Lithium-ion batteries (LiBs) are an enabling energy storage technology for numerous applications for today's and future world. They have not only achieved dominance in the consumer electronics sector, but also are rapidly penetrating into the transportation, grid, building, and aerospace/space/marine sectors where large-scale energy storage is needed. Looking into the upcoming decades, they represent an important part of the promise for a clean energy era for the human society. Battery management systems (BMSs) are crucial in making LiBs live up to their potential and ensuring safety and reliability in operation, and the research on advanced BMS algorithms has hence gained significant momentum in the past years. As the overarching motivation and focus, our study aims to develop LiB models and management algorithms that can provide both high accuracy and computational efficiency to facilitate the potential use of BMS methods in practical systems. Specifically, this dissertation contributes to the study of four major problems:

- *LiB modeling*: We proposed the nonlinear double-capacitor (NDC) model in search of a more accurate equivalent circuit model. This new model, for the first time, can simultaneously capture the lithium-ion diffusion in electrodes and nonlinear voltage dynamics, while retaining a simple structure.
- *Parameter identification*: We developed identification methods for our proposed NDC model and the popular Thevenin model. For the NDC model, our methods can extract parameters using current/voltage data efficiently. For the Thevenin model, our methods can identify all model parameters in one shot.
- *Optimal charging*: Based on the proposed NDC model, we proposed a computationally effi-

cient health-aware charging strategy based on model predictive control. Our method moves online optimization as needed by traditional methods offline and thus eventually allows fast implementation by table lookup and search, which makes it promising for embedded-system-based charging control.

- *Thermal monitoring*: We, for the first time, proposed an efficient estimation method for real-time monitoring of a pack's three-dimensional temperature field.

This dissertation advances the state-of-the-art of BMS modeling and algorithms toward both high accuracy and computational efficiency. Meanwhile, there exist a few opportunities to further improve the works in this dissertation.

- 1) *LiB modeling*: The NDC model in its current form can emulate lithium-ion diffusion in the battery electrode. It is appealing to further enhance this model by accounting for other electrochemical processes or physical phenomena such as lithium-ion diffusion in the electrolyte, effects of temperature, aging and voltage hysteresis. The improvements can be achieved in different ways, by including additional electrical components, or introducing data-driven modules, or coupling with other types of models. They will allow the NDC model to provide higher accuracy and fidelity for LiBs operating in different conditions.
- 2) *Parameter identification*: The proposed identification approach 2.0 for the NDC model in Chapter 2 is based on an assumption that the training data are sampled periodically with constant time intervals. Non-uniform sampling, however, may occur in practice, making it interesting to extend the approach to this case. A potential solution is to investigate and perform continuous-time system identification for the NDC model. The lifting technique may offer another promising way to enable the use of non-uniformly sampled data for parameter estimation here.
- 3) *Optimal charging*: The experimental validation of the proposed charging control algorithms in Chapter 4 was carried out in an open-loop manner, due to the test facility's limitation. A

closed-loop experimental validation will be more desirable. Besides, the use of the constraint on $V_s - V_b$ is intended to mitigate lithium-ion concentration gradient growth during charging and alleviate a LiB's aging process. It will be useful to conduct further experimental validation to verify its effectiveness.

- 4) *Thermal monitoring*: The battery pack thermal model in Chapter 5 is developed in a basic setting. It would be of great interest to extend it to more complex cases such as packs with heterogeneous cells, non-uniform heat generation and active cooling systems. Prospective research in this direction will likely involve an integration of heat transfer modeling, finite-element analysis, data-driven learning, and experimental validation.

Besides the above extensions, more explorations can be made along the overarching line of research in this dissertation—advanced battery management with both high computational efficiency and accuracy—in the future. Two important example problems among them are outlined as follows.

First, pack-level BMS algorithms will be worthy of more research, which would particularly demand computationally efficient designs due to the more complex and higher-dimensional dynamics [33, 44, 112, 183, 220]. It will be interesting to investigate fast and accurate pack estimation and control by expanding the research in this dissertation. Associated with this, pack-level BMS algorithms amenable to frugal sensing and tolerant to sensor biases or failures will be useful from a practical perspective [110, 165].

Second, as future LiB systems may generate abundant data throughout the operation, an important subject is to develop new BMS methods based upon machine learning [15, 140, 169] and cloud computing [16, 104]. The computational efficiency, however, will still remain a pressing need. The notions provided in this dissertation, including parsimonious modeling and distributed computation, may offer inspirations for future pursuits in this regard.

References

- [1] Are air-cooled batteries hurting Nissan Leaf range? Available: <https://mittr-frontend-prod.herokuapp.com/s/429282/are-air-cooled-batteries-hurting-nissan-leaf-range/amp/>.
- [2] A behind the scenes take on lithium-ion battery prices. Available: <https://about.bnef.com/blog/behind-scenes-take-lithium-ion-battery-prices/>.
- [3] The EPA Urban Dynamometer Driving Schedule (UDDS). Available: <https://www.epa.gov/sites/production/files/2015-10/uddscol.txt>.
- [4] Global battery demand between 2018 and 2030. Available: <https://www.statista.com/statistics/1103218/global-battery-demand-forecast/>.
- [5] Lithium-ion battery management. Available: <http://ieeecss.org/sites/ieeecss/files/2019-06/IoCT2-RC-Trnka-1.pdf>.
- [6] Tesla Model 3. Available: <https://www.tesla.com/model3>.
- [7] What's the best battery? Available: https://batteryuniversity.com/learn/archive/whats_the_best_battery.
- [8] The Worldwide harmonized Light-duty vehicles Test procedure (WLTP). Available: <http://www.unece.org/fileadmin/DAM/trans/doc/2012/wp29grpe/WLTP-DHC-12-07e.xls>.
- [9] Abada, S., Marlair, G., Lecocq, A., Petit, M., Sauvante-Moynot, V., & Huet, F. (2016). Safety focused modeling of lithium-ion batteries: A review. *Journal of Power Sources*, 306, 178–192.
- [10] Abu-Sharkh, S. & Doerffel, D. (2004). Rapid test and non-linear model characterisation of solid-state lithium-ion batteries. *Journal of Power Sources*, 130(1), 266–274.
- [11] Al Hallaj, S., Maleki, H., Hong, J.-S., & Selman, J. R. (1999). Thermal modeling and design considerations of lithium-ion batteries. *Journal of Power Sources*, 83(1), 1–8.

- [12] Alavi, S. M., Samadi, M. F., & Saif, M. (2013). Diagnostics in lithium-ion batteries: Challenging issues and recent achievements. In *Integration of Practice-Oriented Knowledge Technology: Trends and Prospectives* (pp. 277–291). Springer.
- [13] Anderson, B. D. & Moore, J. B. (2015). *Optimal Filtering*. Englewood Cliffs, N.J.: Prentice-Hall.
- [14] Andre, D., Meiler, M., Steiner, K., Wimmer, C., Soczka-Guth, T., & Sauer, D. (2011). Characterization of high-power lithium-ion batteries by electrochemical impedance spectroscopy. I. Experimental investigation. *Journal of Power Sources*, 196(12), 5334–5341.
- [15] Attia, P. M., Grover, A., Jin, N., Severson, K. A., Markov, T. M., Liao, Y.-H., Chen, M. H., Cheong, B., Perkins, N., Yang, Z., et al. (2020). Closed-loop optimization of fast-charging protocols for batteries with machine learning. *Nature*, 578(7795), 397–402.
- [16] Balasingam, B. & Pattipati, K. (2018). Elements of a robust battery-management system: From fast characterization to universality and more. *IEEE Electrification Magazine*, 6(3), 34–37.
- [17] Bandhauer, T. M., Garimella, S., & Fuller, T. F. (2011). A critical review of thermal issues in lithium-ion batteries. *Journal of the Electrochemical Society*, 158(3), R1–R25.
- [18] Barillas, J. K., Li, J., Günther, C., & Danzer, M. A. (2015). A comparative study and validation of state estimation algorithms for li-ion batteries in battery management systems. *Applied Energy*, 155, 455–462.
- [19] Barnes, F. S. & Levine, J. G. (2018). *Handbook on Battery Energy Storage System*. Asian Development Bank, Manila, Philippines.
- [20] Baronti, F., Fantechi, G., Leonardi, E., Roncella, R., & Saletti, R. (2010). Enhanced model for lithium-polymer cells including temperature effects. In *IECON 2010-36th Annual Conference on IEEE Industrial Electronics Society* (pp. 2329–2333).: IEEE.
- [21] Bartlett, A., Marcicki, J., Onori, S., Rizzoni, G., Yang, X. G., & Miller, T. (2016). Electrochemical model-based state of charge and capacity estimation for a composite electrode lithium-ion battery. *IEEE Transactions on Control Systems Technology*, 24(2), 384–399.
- [22] Bemporad, A. et al. (2015). A multiparametric quadratic programming algorithm with polyhedral computations based on nonnegative least squares. *IEEE Trans. Automat. Contr.*, 60(11), 2892–2903.
- [23] Bemporad, A., Morari, M., Dua, V., & Pistikopoulos, E. N. (2002). The explicit linear quadratic regulator for constrained systems. *Automatica*, 38(1), 3–20.

- [24] Bemporad, A., Ricker, N. L., & Morari, M. (2018). *Model Predictive Control Toolbox User's Guide R2018b*. The MathWorks.
- [25] Bergman, T. L., Incropera, F. P., DeWitt, D. P., & Lavine, A. S. (2006). *Fundamentals of Heat and Mass Transfer*. John Wiley & Sons, 6 edition.
- [26] Bergveld, H. (2001). *Battery Management Systems Design by Modelling*. PhD thesis, University Twente.
- [27] Bernardi, D., Pawlikowski, E., & Newman, J. (1985). A general energy balance for battery systems. *Journal of the Electrochemical Society*, 132(1), 5–12.
- [28] Birkel, C. & Howey, D. (2013). Model identification and parameter estimation for LiFePO₄ batteries. In *Proceedings of IET Hybrid and Electric Vehicles Conference* (pp. 1–6).
- [29] Borrelli, F., Bemporad, A., & Morari, M. (2017). *Predictive Control for Linear and Hybrid Systems*. Cambridge University Press.
- [30] Bowen, T., Chernyakhovskiy, I., & Denholm, P. L. (2019). *Grid-Scale Battery Storage: Frequently Asked Questions*. Technical report, National Renewable Energy Lab (NREL), Golden, CO (United States).
- [31] Brand, J., Zhang, Z., & Agarwal, R. K. (2014). Extraction of battery parameters of the equivalent circuit model using a multi-objective genetic algorithm. *Journal of Power Sources*, 247, 729–737.
- [32] Buller, S., Thele, M., Karden, E., & De Doncker, R. W. (2003). Impedance-based non-linear dynamic battery modeling for automotive applications. *Journal of Power Sources*, 113(2), 422–430.
- [33] Castano, S., Gauchia, L., Voncila, E., & Sanz, J. (2015). Dynamical modeling procedure of a Li-ion battery pack suitable for real-time applications. *Energy Conversion and Management*, 92, 396–405.
- [34] Cengel, Y. A., Ghajar, A. J., & Ma, H. (2015). *Heat and Mass Transfer: Fundamentals & Applications*. McGraw-Hill, 5 edition.
- [35] Chaturvedi, N., Klein, R., Christensen, J., Ahmed, J., & Kojic, A. (2010). Algorithms for advanced battery-management systems. *IEEE Control Systems Magazine*, 30(3), 49–68.
- [36] Chen, M. & Rincon-Mora, G. A. (2006). Accurate electrical battery model capable of predicting runtime and I-V performance. *IEEE Transactions on Energy Conversion*, 21(2), 504–511.
- [37] Chen, S., Wan, C., & Wang, Y. (2005). Thermal analysis of lithium-ion batteries. *Journal of Power Sources*, 140(1), 111–124.

- [38] Chen, S.-C., Wang, Y.-Y., & Wan, C.-C. (2006). Thermal analysis of spirally wound lithium batteries. *Journal of the Electrochemical Society*, 153(4), A637–A648.
- [39] Chen, W., Chen, W.-T., Saif, M., Li, M.-F., & Wu, H. (2013). Simultaneous fault isolation and estimation of lithium-ion batteries via synthesized design of Luenberger and learning observers. *IEEE Transactions on Control Systems Technology*, 22(1), 290–298.
- [40] Chiang, Y.-H., Sean, W.-Y., & Ke, J.-C. (2011). Online estimation of internal resistance and open-circuit voltage of lithium-ion batteries in electric vehicles. *Journal of Power Sources*, 196(8), 3921–3932.
- [41] Coleman, M., Lee, C. K., Zhu, C., & Hurley, W. G. (2007). State-of-charge determination from EMF voltage estimation: Using impedance, terminal voltage, and current for lead-acid and lithium-ion batteries. *IEEE Transactions on Industrial Electronics*, 54(5), 2550–2557.
- [42] Coleman, T. F. & Li, Y. (1996). An interior trust region approach for nonlinear minimization subject to bounds. *SIAM Journal on Optimization*, 6(2), 418–445.
- [43] Cope, R. C. & Podrazhansky, Y. (1999). The art of battery charging. In *Fourteenth Annual Battery Conference on Applications and Advances. Proceedings of the Conference* (pp. 233–235).: IEEE.
- [44] Cordoba-Arenas, A., Onori, S., & Rizzoni, G. (2015). A control-oriented lithium-ion battery pack model for plug-in hybrid electric vehicle cycle-life studies and system design with consideration of health management. *Journal of Power Sources*, 279, 791–808.
- [45] Debert, M., Colin, G., Bloch, G., & Chamaillard, Y. (2013). An observer looks at the cell temperature in automotive battery packs. *Control Engineering Practice*, 21(8), 1035–1042.
- [46] Devarakonda, L. & Hu, T. (2014). Algebraic method for parameter identification of circuit models for batteries under non-zero initial condition. *Journal of Power Sources*, 268, 928–940.
- [47] Dey, S., Ayalew, B., & Pisu, P. (2015). Nonlinear robust observers for state-of-charge estimation of lithium-ion cells based on a reduced electrochemical model. *IEEE Transactions on Control Systems Technology*, 23(5), 1935–1942.
- [48] Dey, S., Mohon, S., Ayalew, B., Arunachalam, H., & Onori, S. (2019). A novel model-based estimation scheme for battery-double-layer capacitor hybrid energy storage systems. *IEEE Transactions on Control Systems Technology*, 27(2), 689–702.

- [49] Dey, S., Mohon, S., Pisu, P., & Ayalew, B. (2016). Sensor fault detection, isolation, and estimation in lithium-ion batteries. *IEEE Transactions on Control Systems Technology*, 24(6), 2141–2149.
- [50] Domenico, D., Stefanopoulou, A., & Fiengo, G. (2010). Lithium-ion battery state of charge and critical surface charge estimation using an electrochemical model-based extended Kalman filter. *ASME Journal of Dynamic Systems, Measurement, and Control*, 132(6), 061302–061302–11.
- [51] Doyle, M., Fuller, T. F., & Newman, J. (1993). Modeling of galvanostatic charge and discharge of the lithium/polymer/insertion cell. *Journal of The Electrochemical Society*, 140(6), 1526–1533.
- [52] Dubarry, M. & Liaw, B. Y. (2007). Development of a universal modeling tool for rechargeable lithium batteries. *Journal of Power Sources*, 174(2), 856–860.
- [53] Dubarry, M., Vuillaume, N., & Liaw, B. Y. (2009). From single cell model to battery pack simulation for Li-ion batteries. *Journal of Power Sources*, 186(2), 500–507.
- [54] Einhorn, M., Conte, F. V., Kral, C., & Fleig, J. (2012). Comparison, selection, and parameterization of electrical battery models for automotive applications. *IEEE Transactions on Power Electronics*, 28(3), 1429–1437.
- [55] El Lakkis, M., Sename, O., Corno, M., & Bresch Pietri, D. (2015). Combined battery SoC/SoH estimation using a nonlinear adaptive observer. In *Proceedings of European Control Conference* (pp. 1522–1527).
- [56] Fang, H., Depcik, C., & Lvovich, V. (2018a). Optimal pulse-modulated lithium-ion battery charging: Algorithms and simulation. *Journal of Energy Storage*, 15, 359–367.
- [57] Fang, H., Sharma, R., & Patil, R. (2014a). Optimal sensor and actuator deployment for HVAC control system design. In *American Control Conference* (pp. 2240–2246).: IEEE.
- [58] Fang, H., Tian, N., Wang, Y., Zhou, M., & Haile, M. A. (2018b). Nonlinear Bayesian estimation: From Kalman filtering to a broader horizon. *IEEE/CAA Journal of Automatica Sinica*, 5(2), 401–417.
- [59] Fang, H., Wang, Y., & Chen, J. (2017). Health-aware and user-involved battery charging management for electric vehicles: Linear quadratic strategies. *IEEE Transactions on Control Systems Technology*, 25(3), 911–923.
- [60] Fang, H., Wang, Y., Sahinoglu, Z., Wada, T., & Hara, S. (2014b). State of charge estimation for lithium-ion batteries: An adaptive approach. *Control Engineering Practice*, 25, 45–54.
- [61] Fang, H., Zhao, X., Wang, Y., Sahinoglu, Z., Wada, T., Hara, S., & de Callafon, R. A. (2014c). Improved adaptive state-of-charge estimation for batteries using a multi-model approach. *Journal of Power Sources*, 254, 258 – 267.

- [62] Feng, T., Yang, L., Zhao, X., Zhang, H., & Qiang, J. (2015). Online identification of lithium-ion battery parameters based on an improved equivalent-circuit model and its implementation on battery state-of-power prediction. *Journal of Power Sources*, 281, 192–203.
- [63] Feng, X., Ouyang, M., Liu, X., Lu, L., Xia, Y., & He, X. (2018). Thermal runaway mechanism of lithium ion battery for electric vehicles: A review. *Energy Storage Materials*, 10, 246–267.
- [64] Fleischer, C., Waag, W., Heyn, H.-M., & Sauer, D. U. (2014). On-line adaptive battery impedance parameter and state estimation considering physical principles in reduced order equivalent circuit battery models: Part 2. Parameter and state estimation. *Journal of Power Sources*, 262, 457–482.
- [65] Forgez, C., Do, D. V., Friedrich, G., Morcrette, M., & Delacourt, C. (2010). Thermal modeling of a cylindrical LiFePO₄/graphite lithium-ion battery. *Journal of Power Sources*, 195(9), 2961–2968.
- [66] Forman, J. C., Moura, S. J., Stein, J. L., & Fathy, H. K. (2012). Genetic identification and fisher identifiability analysis of the Doyle–Fuller–Newman model from experimental cycling of a LiFePO₄ cell. *Journal of Power Sources*, 210, 263 – 275.
- [67] G., S. M. & Nikdel, M. (2014). Various battery models for various simulation studies and applications. *Renewable and Sustainable Energy Reviews*, 32, 477–485.
- [68] Gandolfo, D., Brandão, A., Patiño, D., & Molina, M. (2015). Dynamic model of lithium polymer battery–load resistor method for electric parameters identification. *Journal of the Energy Institute*, 88(4), 470–479.
- [69] Gao, Y., Jiang, J., Zhang, C., Zhang, W., Ma, Z., & Jiang, Y. (2017). Lithium-ion battery aging mechanisms and life model under different charging stresses. *Journal of Power Sources*, 356, 103–114.
- [70] Gersnoviez, A., Brox, M., & Baturone, I. (2017). High-speed and low-cost implementation of explicit model predictive controllers. *IEEE Trans. Control Syst. Technol.*, 27(2), 647–662.
- [71] Gholizadeh, M. & Salmasi, F. R. (2014). Estimation of state of charge, unknown nonlinearities, and state of health of a lithium-ion battery based on a comprehensive unobservable model. *IEEE Transactions on Industrial Electronics*, 61(3), 1335–1344.
- [72] Giri, F. & Bai, E.-W. (2010). *Block-Oriented Nonlinear System Identification*, volume 1. London, UK: Springer.
- [73] Goebel, K., Saha, B., Saxena, A., Celaya, J. R., & Christophersen, J. P. (2008). Prognostics in battery health management. *IEEE Instrumentation & Measurement Magazine*, 11(4).

- [74] Gomadam, P. M., Weidner, J. W., Dougal, R. A., & White, R. E. (2002). Mathematical modeling of lithium-ion and nickel battery systems. *Journal of Power Sources*, 110(2), 267–284.
- [75] Gu, H. (1983). Mathematical analysis of a Zn/NiOOH cell. *Journal of The Electrochemical Society*, 130(7), 1459–1464.
- [76] Guo, G., Long, B., Cheng, B., Zhou, S., Xu, P., & Cao, B. (2010). Three-dimensional thermal finite element modeling of lithium-ion battery in thermal abuse application. *Journal of Power Sources*, 195(8), 2393–2398.
- [77] Guo, M., Sikha, G., & White, R. E. (2011). Single-particle model for a lithium-ion cell: Thermal behavior. *Journal of The Electrochemical Society*, 158(2), A122–A132.
- [78] Gustafsson, F. (2000). *Adaptive Filtering and Change Detection*, volume 1. Wiley New York.
- [79] Hagenblad, A. (1999). *Aspects of the Identification of Wiener Models*. PhD thesis, Department of Electrical Engineering, Linköping University, Sweden.
- [80] Hagenblad, A., Ljung, L., & Wills, A. (2008). Maximum likelihood identification of Wiener models. *Automatica*, 44(11), 2697 – 2705.
- [81] Han, W., Zhang, L., & Han, Y. (2015). Computationally efficient methods for state of charge approximation and performance measure calculation in series-connected battery equalization systems. *Journal of Power Sources*, 286, 145–158.
- [82] Han, W., Zou, C., Zhou, C., & Zhang, L. (2018). Estimation of cell SOC evolution and system performance in module-based battery charge equalization systems. *IEEE Transactions on Smart Grid*, 10(5), 4717–4728.
- [83] He, H., Xiong, R., & Fan, J. (2011a). Evaluation of lithium-ion battery equivalent circuit models for state of charge estimation by an experimental approach. *Energies*, 4, 582–598.
- [84] He, H., Xiong, R., & Guo, H. (2012). Online estimation of model parameters and state-of-charge of LiFePO₄ batteries in electric vehicles. *Applied Energy*, 89(1), 413–420.
- [85] He, H., Xiong, R., Zhang, X., Sun, F., & Fan, J. (2011b). State-of-charge estimation of the lithium-ion battery using an adaptive extended Kalman filter based on an improved Thevenin model. *IEEE Transactions on Vehicular Technology*, 60(4), 1461–1469.
- [86] He, Z., Yang, G., & Lu, L. (2016). A parameter identification method for dynamics of lithium iron phosphate batteries based on step-change current curves and constant current curves. *Energies*, 9(6), 444.

- [87] Hentunen, A., Lehmuspelto, T., & Suomela, J. (2014). Time-domain parameter extraction method for Thévenin-equivalent circuit battery models. *IEEE Transactions on Energy Conversion*, 29(3), 558–566.
- [88] Hoerl, A. E. & Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1), 55–67.
- [89] Horn, R. A. & Johnson, C. R. (1985). *Matrix Analysis*. Cambridge University Press, 1 edition.
- [90] Hsieh, C.-S. & Chen, F.-C. (1999). Optimal solution of the two-stage Kalman estimator. *IEEE Transactions on Automatic Control*, 44(1), 194–199.
- [91] Hu, T. & Jung, H. (2013). Simple algorithms for determining parameters of circuit models for charging/discharging batteries. *Journal of Power Sources*, 233, 14–22.
- [92] Hu, X., Cao, D., & Egardt, B. (2018). Condition monitoring in advanced battery management systems: Moving horizon estimation using a reduced electrochemical model. *IEEE/ASME Transactions on Mechatronics*, 23(1), 167–178.
- [93] Hu, X., Collins, L., Stanton, S., & Jiang, S. (2013). A model parameter identification method for battery applications. *SAE Technical Paper*.
- [94] Hu, X., Li, S., & Peng, H. (2012). A comparative study of equivalent circuit models for Li-ion batteries. *Journal of Power Sources*, 198, 359–367.
- [95] Hu, Y. & Yurkovich, S. (2011). Linear parameter varying battery model identification using subspace methods. *Journal of Power Sources*, 196(5), 2913–2923.
- [96] Hu, Y., Yurkovich, S., Guezennec, Y., & Yurkovich, B. (2009). A technique for dynamic battery model identification in automotive applications using linear parameter varying structures. *Control Engineering Practice*, 17(10), 1190–1201.
- [97] Hu, Y., Yurkovich, S., Guezennec, Y., & Yurkovich, B. (2011). Electro-thermal battery model identification for automotive applications. *Journal of Power Sources*, 196(1), 449–457.
- [98] Hussein, A. A.-H. & Batarseh, I. (2011). A review of charging algorithms for nickel and lithium battery chargers. *IEEE Transactions on Vehicular Technology*, 60(3), 830–838.
- [99] Jaguemont, J., Boulon, L., & Dubé, Y. (2016). A comprehensive review of lithium-ion batteries used in hybrid and electric vehicles at cold temperatures. *Applied Energy*, 164, 99–114.

- [100] Johnson, V. (2002). Battery performance models in ADVISOR. *Journal of Power Sources*, 110(2), 321–329.
- [101] Johnson, V. H. & Pesaran, A. A. (2000). *Temperature-Dependent Battery Models for High-Power Lithium-Ion Batteries*. Technical Report NREL/CP-540-28716, National Renewable Energy Laboratory.
- [102] Jongerden, M. & Haverkort, B. (2008). Battery modeling. *Enschede, January*.
- [103] Kim, G.-H., Pesaran, A., & Spotnitz, R. (2007). A three-dimensional thermal abuse model for lithium-ion cells. *Journal of Power Sources*, 170(2), 476–489.
- [104] Kim, T., Makwana, D., Adhikaree, A., Vagdoda, J. S., & Lee, Y. (2018). Cloud-based battery condition monitoring and fault diagnosis platform for large-scale lithium-ion battery energy storage systems. *Energies*, 11(1), 125.
- [105] Kim, T. & Qiao, W. (2011). A hybrid battery model capable of capturing dynamic circuit characteristics and nonlinear capacity effects. *IEEE Transactions on Energy Conversion*, 26(4), 1172–1180.
- [106] Kim, T., Wang, Y., Fang, H., Sahinoglu, Z., Wada, T., Hara, S., & Qiao, W. (2015). Model-based condition monitoring for lithium-ion batteries. *Journal of Power Sources*, 295, 16–27.
- [107] Kim, U. S., Shin, C. B., & Kim, C.-S. (2008). Effect of electrode configuration on the thermal behavior of a lithium-polymer battery. *Journal of Power Sources*, 180(2), 909–916.
- [108] Kim, Y., Mohan, S., Siegel, J. B., Stefanopoulou, A. G., & Ding, Y. (2014). The estimation of temperature distribution in cylindrical battery cells under unknown cooling conditions. *IEEE Transactions on Control Systems Technology*, 22(6), 2277–2286.
- [109] Klein, R., Chaturvedi, N. A., Christensen, J., Ahmed, J., Findeisen, R., & Kojic, A. (2011). Optimal charging strategies in lithium-ion battery. In *Proc. of American Control Conference* (pp. 382–387).: IEEE.
- [110] Kong, X., Zheng, Y., Ouyang, M., Li, X., Lu, L., Li, J., & Zhang, Z. (2017). Signal synchronization for massive data storage in modular battery management system with controller area network. *Applied Energy*, 197, 52–62.
- [111] Lam, L., Bauer, P., & Kelder, E. (2011). A practical circuit-based model for Li-ion battery cells in electric vehicle applications. In *2011 IEEE 33rd International Telecommunications Energy Conference (INTELEC)* (pp. 1–9).: IEEE.
- [112] Lee, J., Ahn, J.-H., & Lee, B. K. (2017). A novel Li-ion battery pack modeling considering single cell information and capacity variation. In *2017 IEEE Energy Conversion Congress and Exposition (ECCE)* (pp. 5242–5247).: IEEE.

- [113] Lee, K.-T., Dai, M.-J., & Chuang, C.-C. (2018). Temperature-compensated model for lithium-ion polymer batteries with extended Kalman filter state-of-charge estimation for an implantable charger. *IEEE Transactions on Industrial Electronics*, 65(1), 589–596.
- [114] Lee, S., Kim, J., Lee, J., & Cho, B. (2008). State-of-charge and capacity estimation of lithium-ion battery using a new open-circuit voltage versus state-of-charge. *Journal of Power Sources*, 185(2), 1367–1373.
- [115] Li, J., Mazzola, M., Gafford, J., & Younan, N. (2012). A new parameter estimation algorithm for an electrical analogue battery model. In *2012 Twenty-Seventh Annual IEEE Applied Power Electronics Conference and Exposition (APEC)* (pp. 427–433).: IEEE.
- [116] Li, K., Wei, F., Tseng, K. J., & Soong, B.-H. (2018). A practical lithium-ion battery model for state of energy and voltage responses prediction incorporating temperature and ageing effects. *IEEE Transactions on Industrial Electronics*, 65(8), 6696–6708.
- [117] Li, Y., Liao, C., Wang, L., Wang, L., & Xu, D. (2014). Subspace-based modeling and parameter identification of lithium-ion batteries. *International Journal of Energy Research*, 38(8), 1024–1038.
- [118] Limoge, D. W. & Annaswamy, A. M. (2018). An adaptive observer design for real-time parameter estimation in lithium-ion batteries. *IEEE Transactions on Control Systems Technology*. in press.
- [119] Lin, C.-C., Peng, H., Grizzle, J. W., & Kang, J.-M. (2003). Power management strategy for a parallel hybrid electric truck. *IEEE Transactions on Control Systems Technology*, 11(6), 839–849.
- [120] Lin, Q., Wang, J., Xiong, R., Shen, W., & He, H. (2019a). Towards a smarter battery management system: A critical review on optimal charging methods of lithium ion batteries. *Energy*, 183, 220–234.
- [121] Lin, X., Kim, Y., Mohan, S., Siegel, J. B., & Stefanopoulou, A. G. (2019b). Modeling and estimation for advanced battery management. *Annual Review of Control, Robotics, and Autonomous Systems*, 2, 393–426.
- [122] Lin, X., Perez, H. E., Mohan, S., Siegel, J. B., Stefanopoulou, A. G., Ding, Y., & Castanier, M. P. (2014a). A lumped-parameter electro-thermal model for cylindrical batteries. *Journal of Power Sources*, 257, 1–11.
- [123] Lin, X., Perez, H. E., Siegel, J. B., Stefanopoulou, A. G., Ding, Y., & Castanier, M. P. (France, 2011). Parameterization and observability analysis of scalable battery clusters for onboard thermal management. In *Oil & Gas Science and Technology — Rev. IFP Energies nouvelles*.
- [124] Lin, X., Perez, H. E., Siegel, J. B., Stefanopoulou, A. G., Li, Y., Anderson, R. D., Ding, Y., & Castanier, M. P. (2013). Online parameterization of lumped thermal dynamics in cylindrical lithium ion batteries for core

- temperature estimation and health monitoring. *IEEE Transactions on Control Systems Technology*, 21(5), 1745–1755.
- [125] Lin, X., Stefanopoulou, A. G., Siegel, J. B., & Mohan, S. (2014b). Temperature estimation in a battery string under frugal sensor allocation. In *ASME Dynamic Systems and Control Conference, American Society of Mechanical Engineers* (pp. V001T19A006–V001T19A006).
- [126] Liu, J., Li, G., & Fathy, H. K. (2016a). A computationally efficient approach for optimizing lithium-ion battery charging. *Journal of Dynamic Systems, Measurement, and Control*, 138(2), 021009.
- [127] Liu, J., Li, G., & Fathy, H. K. (2017a). An extended differential flatness approach for the health-conscious nonlinear model predictive control of lithium-ion batteries. *IEEE Transactions on Control Systems Technology*, 25(5), 1882–1889.
- [128] Liu, K., Li, K., Yang, Z., Zhang, C., & Deng, J. (2016b). Battery optimal charging strategy based on a coupled thermoelectric model. In *2016 IEEE Congress on Evolutionary Computation* (pp. 5084–5091).
- [129] Liu, K., Li, K., & Zhang, C. (2017b). Constrained generalized predictive control of battery charging process based on a coupled thermoelectric model. *Journal of Power Sources*, 347, 145–158.
- [130] Liu, K., Li, Y., Hu, X., Lucu, M., & Widanage, W. D. (2019). Gaussian process regression with automatic relevance determination kernel for calendar aging prediction of lithium-ion batteries. *IEEE Transactions on Industrial Informatics*, 16(6), 3767–3777.
- [131] Liu, K., Zou, C., Li, K., & Wik, T. (2018). Charging pattern optimization for lithium-ion batteries with an electrothermal-aging model. *IEEE Trans. Ind. Informat.*, 14(12), 5463–5474.
- [132] Liu, Z., He, H., Ahmed, Q., & Rizzoni, G. (2015). Structural analysis based fault detection and isolation applied for a lithium-ion battery pack. *IFAC-PapersOnLine*, 48(21), 1465–1470.
- [133] Ljung, L. & Chen, T. (2013). What can regularization offer for estimation of dynamical systems? *IFAC Proceedings Volumes*, 46(11), 1–8. 11th IFAC Workshop on Adaptation and Learning in Control and Signal Processing.
- [134] Lu, L., Han, X., Li, J., Hua, J., & Ouyang, M. (2013). A review on the key issues for lithium-ion battery management in electric vehicles. *Journal of Power Sources*, 226, 272–288.
- [135] Mahamud, R. & Park, C. (2011). Reciprocating air flow for Li-ion battery thermal management to improve temperature uniformity. *Journal of Power Sources*, 196(13), 5685–5696.

- [136] Malik, A., Zhang, Z., & Agarwal, R. K. (2014). Extraction of battery parameters using a multi-objective genetic algorithm with a non-linear circuit model. *Journal of Power Sources*, 259, 76–86.
- [137] Mamun, A., Sivasubramaniam, A., & Fathy, H. (2018). Collective learning of lithium-ion aging model parameters for battery health-conscious demand response in datacenters. *Energy*, 154, 80–95.
- [138] Marcicki, J., Onori, S., & Rizzoni, G. (2010). Nonlinear fault detection and isolation for a lithium-ion battery management system. In *Dynamic Systems and Control Conference*, volume 44175 (pp. 607–614).
- [139] Mauracher, P. & Karden, E. (1997). Dynamic modelling of lead/acid batteries using impedance spectroscopy for parameter identification. *Journal of Power Sources*, 67(1-2), 69–84.
- [140] Naha, A., Khandelwal, A., Agarwal, S., Tagade, P., Hariharan, K. S., Kaushik, A., Yadu, A., Kolake, S. M., Han, S., & Oh, B. (2020). Internal short circuit detection in Li-ion batteries using supervised machine learning. *Scientific Reports*, 10(1), 1–10.
- [141] Nelatury, S. R. & Singh, P. (2004). Equivalent circuit parameters of nickel/metal hydride batteries from sparse impedance measurements. *Journal of Power Sources*, 132(1), 309–314.
- [142] Northrop, P. W. C., Suthar, B., Ramadesigan, V., Santhanagopalan, S., Braatz, R. D., & Subramanian, V. R. (2014). Efficient simulation and reformulation of lithium-ion battery models for enabling electric transportation. *Journal of The Electrochemical Society*, 161(8), E3149–E3157.
- [143] Ouyang, Q., Chen, J., Zheng, J., & Fang, H. (2018a). Optimal cell-to-cell balancing topology design for serially connected lithium-ion battery packs. *IEEE Transactions on Sustainable Energy*, 9(1), 350–360.
- [144] Ouyang, Q., Chen, J., Zheng, J., & Fang, H. (2018b). Optimal multi-objective charging for lithium-ion battery packs: A hierarchical control approach. *IEEE Transactions on Industrial Informatics*.
- [145] Park, C. & Jaura, A. K. (2003). Dynamic thermal model of Li-ion battery for predictive behavior in hybrid and fuel cell vehicles. In *Future Transportation Technology Conference*.
- [146] Park, S., Kato, D., Gima, Z., Klein, R., & Moura, S. (2018). Optimal input design for parameter identification in an electrochemical Li-ion battery model. In *Proceedings of American Control Conference* (pp. 2300–2305).
- [147] Perez, H., Shahmohammadhamedani, N., & Moura, S. (2015). Enhanced performance of Li-ion batteries via modified reference governors and electrochemical models. *IEEE/ASME Transactions on Mechatronics*.

- [148] Perez, H. E. (2016). *Model Based Optimal Control, Estimation, and Validation of Lithium-Ion Batteries*. PhD thesis, UC Berkeley.
- [149] Perez, H. E., Hu, X., Dey, S., & Moura, S. J. (2017). Optimal charging of Li-ion batteries with coupled electro-thermal-aging dynamics. *IEEE Transactions on Vehicular Technology*, 66(9), 7761–7770.
- [150] Perez, H. E. & Moura, S. J. (2015). Sensitivity-based interval PDE observer for battery SOC estimation. In *2015 American Control Conference* (pp. 323–328).
- [151] Petzl, M. & Danzer, M. A. (2013). Advancements in OCV measurement and analysis for lithium-ion batteries. *IEEE Transactions on Energy Conversion*, 28(3), 675–681.
- [152] Pinson, M. B. & Bazant, M. Z. (2013). Theory of SEI formation in rechargeable batteries: Capacity fade, accelerated aging and lifetime prediction. *J. Electrochem. Soc.*, 160(2), A243–A250.
- [153] Plett, G. L. (2004a). Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs: Part 2. Modeling and identification. *Journal of Power Sources*, 134(2), 262–276.
- [154] Plett, G. L. (2004b). Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs: Part 3. State and parameter estimation. *Journal of Power Sources*, 134(2), 277–292.
- [155] Plett, G. L. (2006). Sigma-point Kalman filtering for battery management systems of LiPB-based HEV battery packs: Part 1: Introduction and state estimation. *Journal of Power Sources*, 161(2), 1356–1368.
- [156] Plett, G. L. (2015). *Battery Management Systems, Volume 1: Battery Modeling*. Artech House.
- [157] Prasad, G. K. & Rahn, C. D. (2013). Model based identification of aging parameters in lithium ion batteries. *Journal of Power Sources*, 232, 79–85.
- [158] Rahimi-Eichi, H., Baronti, F., & Chow, M.-Y. (2014). Online adaptive parameter identification and state-of-charge coestimation for lithium-polymer battery cells. *IEEE Transactions on Industrial Electronics*, 61(4), 2053–2061.
- [159] Rahman, M. A., Anwar, S., & Izadian, A. (2016). Electrochemical model parameter identification of a lithium-ion battery using particle swarm optimization method. *Journal of Power Sources*, 307, 86–97.
- [160] Rahn, C. D. & Wang, C. (2013). *Battery Systems Engineering*. John Wiley & Sons.
- [161] Randles, J. E. B. (Mar. 1947). Kinetics of rapid electrode reactions. *Discussions of the Faraday Society*, 1, 11–19.

- [162] Rao, R., Vrudhula, S., & Rakhmatov, D. N. (2003). Battery modeling for energy aware system design. *Computer*, 36(12), 77–87.
- [163] Relan, R., Firouz, Y., Timmermans, J., & Schoukens, J. (2017). Data-driven nonlinear identification of Li-ion battery based on a frequency domain nonparametric analysis. *IEEE Transactions on Control Systems Technology*, 25(5), 1825–1832.
- [164] Rothenberger, M. J., Docimo, D. J., Ghanaatpishe, M., & Fathy, H. K. (2015). Genetic optimization and experimental validation of a test cycle that maximizes parameter identifiability for a Li-ion equivalent-circuit battery model. *Journal of Energy Storage*, 4, 156–166.
- [165] Rothgang, S., Nordmann, H., Schäper, C., & Sauer, D. U. (2012). Challenges in battery pack design. In *2012 Electrical Systems for Aircraft, Railway and Ship Propulsion* (pp. 1–6).: IEEE.
- [166] Schweighofer, B., Raab, K. M., & Brasseur, G. (2003). Modeling of high power automotive batteries by the use of an automated test system. *IEEE Transactions on Instrumentation and Measurement*, 52(4), 1087–1091.
- [167] Scott, M., Chaturvedi, N. A., & Miroslav, K. (2013). Adaptive partial differential equation observer for battery state-of-charge/state-of-health estimation via an electrochemical model. *ASME Journal of Dynamic Systems, Measurement, and Control*, 136(1).
- [168] Seaman, A., Dao, T.-S., & McPhee, J. (2014). A survey of mathematics-based equivalent-circuit and electrochemical battery models for hybrid and electric vehicle simulation. *Journal of Power Sources*, 256, 410–423.
- [169] Severson, K. A., Attia, P. M., Jin, N., Perkins, N., Jiang, B., Yang, Z., Chen, M. H., Aykol, M., Herring, P. K., Fraggedakis, D., et al. (2019). Data-driven prediction of battery cycle life before capacity degradation. *Nature Energy*, 4(5), 383–391.
- [170] Shi, Y., Smith, K., Wood, E., & Pesaran, A. (2015). A multi-node thermal system model for lithium-ion battery packs. In *2015 American Control Conference* (pp. 723–727).: IEEE.
- [171] Shiming, Y. & Wenquan, T. (2006). *Heat Transfer*. Beijing: Higher Education Press, 4 edition.
- [172] Sitterly, M., Wang, L. Y., Yin, G. G., & Wang, C. (2011). Enhanced identification of battery models for real-time battery management. *IEEE Transactions on Sustainable Energy*, 2(3), 300–308.
- [173] Smith, K. & Wang, C.-Y. (2006). Power and thermal characterization of a lithium-ion battery pack for hybrid-electric vehicles. *Journal of Power Sources*, 160(1), 662–673.

- [174] Smith, K. A., Rahn, C. D., & Wang, C. Y. (2010). Model-based electrochemical estimation and constraint management for pulse operation of lithium ion batteries. *IEEE Transactions on Control Systems Technology*, 18(3), 654–663.
- [175] Srinivasan, V. & Wang, C. (2003). Analysis of electrochemical and thermal behavior of Li-ion cells. *Journal of The Electrochemical Society*, 150(1), A98–A106.
- [176] Sun, J., Wei, G., Pei, L., Lu, R., Song, K., Wu, C., & Zhu, C. (2015). Online internal temperature estimation for lithium-ion batteries based on Kalman filter. *Energies*, 8(5), 4400–4415.
- [177] Suthar, B., Ramadesigan, V., De, S., Braatz, R. D., & Subramanian, V. R. (2014). Optimal charging profiles for mechanically constrained lithium-ion batteries. *Phys. Chem. Chem. Phys.*, 16(1), 277–287.
- [178] Szumanowski, A. & Chang, Y. (2008). Battery management system based on battery nonlinear dynamics modeling. *IEEE Transactions on Vehicular Technology*, 57(3), 1425–1432.
- [179] Tang, X., Mao, X., Lin, J., & Koch, B. (2011). Li-ion battery parameter estimation for state of charge. In *American Control Conference (ACC), 2011* (pp. 941–946).: IEEE.
- [180] Tian, N. & Fang, H. (2017). Distributed Kalman filtering-based three-dimensional temperature field reconstruction for a lithium-ion battery pack. In *2017 American Control Conference (ACC)* (pp. 905–910).: IEEE.
- [181] Tian, N., Fang, H., & Chen, J. (2018). A new nonlinear double-capacitor model for rechargeable batteries. In *Proc. of the 44th Annual Conf. of the IEEE Industrial Electronics Society* (pp. 1613–1618).
- [182] Tian, N., Fang, H., Chen, J., & Wang, Y. (2020a). Nonlinear double-capacitor model for rechargeable batteries: Modeling, identification, and validation. *IEEE Transactions on Control Systems Technology*.
- [183] Tian, N., Fang, H., & Wang, Y. (2019a). 3-D temperature field reconstruction for a lithium-ion battery pack: A distributed Kalman filtering approach. *IEEE Transactions on Control Systems Technology*, 27(2), 847–854.
- [184] Tian, N., Fang, H., & Wang, Y. (2019b). Real-time optimal charging for lithium-ion batteries via explicit model predictive control. In *2019 IEEE 28th International Symposium on Industrial Electronics (ISIE)* (pp. 2001–2006).: IEEE.
- [185] Tian, N., Fang, H., & Wang, Y. (2020b). Real-time optimal lithium-ion battery charging based on explicit model predictive control. *IEEE Transactions on Industrial Informatics*.

- [186] Tian, N., Wang, Y., Chen, J., & Fang, H. (2017). On parameter identification of an equivalent circuit model for lithium-ion batteries. In *Control Technology and Applications (CCTA), 2017 IEEE Conference on* (pp. 187–192).: IEEE.
- [187] Tian, N., Wang, Y., Chen, J., & Fang, H. (2020c). One-shot parameter identification of the Thevenin's model for batteries: Methods and validation. *Journal of Energy Storage*, 29, 101282.
- [188] Tian, Y., Xia, B., Sun, W., Xu, Z., & Zheng, W. (2014). A modified model based state of charge estimation of power lithium-ion batteries using unscented Kalman filter. *Journal of Power Sources*, 270, 619–626.
- [189] Tøndel, P., Johansen, T. A., & Bemporad, A. (2003). Evaluation of piecewise affine control via binary search tree. *Automatica*, 39(5), 945–950.
- [190] Torchio, M., Wolff, N. A., Raimondo, D. M., Magni, L., Krewer, U., Gopaluni, R. B., Paulson, J. A., & Braatz, R. D. (2015). Real-time model predictive control for the optimal charging of a lithium-ion battery. In *Proceedings of American Control Conference* (pp. 4536–4541).: IEEE.
- [191] Tran, M.-K. & Fowler, M. (2020). A review of lithium-ion battery fault diagnostic algorithms: Current progress and future challenges. *Algorithms*, 13(3), 62.
- [192] Van Doren, J. F., Douma, S. G., Van den Hof, P. M., Jansen, J. D., & Bosgra, O. H. (2009). Identifiability: from qualitative analysis to model structure approximation. *IFAC Proceedings Volumes of the 15th IFAC Symposium on System Identification*, 42(10), 664 – 669.
- [193] Vanbeylen, L., Pintelon, R., & Schoukens, J. (2009). Blind maximum-likelihood identification of Wiener systems. *IEEE Transactions on Signal Processing*, 57(8), 3017–3029.
- [194] Varaiya, P. & Kumar, P. (1986). *Stochastic Systems: Estimation, Identification and Adaptive Control*. Prentice Hall, Inc., 1 edition.
- [195] Wang, C. & Srinivasan, V. (2002). Computational battery dynamics (CBD)—electrochemical/thermal coupled modeling and multi-scale modeling. *Journal of Power Sources*, 110(2), 364–376.
- [196] Wang, Y., Fang, H., Sahinoglu, Z., Wada, T., & Hara, S. (2015). Adaptive estimation of the state of charge for lithium-ion batteries: Nonlinear geometric observer approach. *IEEE Transactions on Control Systems Technology*, 23(3), 948–962.
- [197] Wang, Y., Fang, H., Zhou, L., & Wada, T. (2017a). Revisiting the state-of-charge estimation for lithium-ion batteries: A methodical investigation of the extended Kalman filter approach. *IEEE Control Systems*, 37(4), 73–96.

- [198] Wang, Z., Hong, J., Liu, P., & Zhang, L. (2017b). Voltage fault diagnosis and prognosis of battery systems based on entropy and z-score for electric vehicles. *Applied Energy*, 196, 289–302.
- [199] Weng, C., Sun, J., & Peng, H. (2014). A unified open-circuit-voltage model of lithium-ion batteries for state-of-charge estimation and state-of-health monitoring. *Journal of Power Sources*, 258, 228–237.
- [200] Woodford IV, W. H. (2013). *Electrochemical shock: Mechanical degradation of ion-intercalation materials*. PhD thesis, Massachusetts Institute of Technology.
- [201] Wright, S. & Nocedal, J. (1999). *Numerical Optimization*, volume 35. New York, US: Springer-Verlag New York.
- [202] Xavier, M. A. & Trimboli, M. S. (2015). Lithium-ion battery cell-level control using constrained model predictive control and equivalent circuit models. *Journal of Power Sources*, 285, 374–384.
- [203] Xia, B., Chen, Z., Mi, C., & Robert, B. (2014). External short circuit fault diagnosis for lithium-ion batteries. In *2014 IEEE Transportation Electrification Conference and Expo (ITEC)* (pp. 1–7).: IEEE.
- [204] Xia, B., Zhao, X., De Callafon, R., Garnier, H., Nguyen, T., & Mi, C. (2016). Accurate lithium-ion battery parameter estimation with continuous-time system identification methods. *Applied Energy*, 179, 426–436.
- [205] Xiao, Y. (2015). Model-based virtual thermal sensors for lithium-ion battery in EV applications. *IEEE Transactions on Industrial Electronics*, 62(5), 3112–3122.
- [206] Xing, Y., He, W., Pecht, M., & Tsui, K. L. (2014). State of charge estimation of lithium-ion batteries using the open-circuit voltage at various ambient temperatures. *Applied Energy*, 113, 106–115.
- [207] Xiong, J., Banvait, H., Li, L., Chen, Y., Xie, J., Liu, Y., Wu, M., & Chen, J. (2012). Failure detection for over-discharged Li-ion batteries. In *2012 IEEE International Electric Vehicle Conference* (pp. 1–5).: IEEE.
- [208] Xiong, R. & Shen, W. (2019). *Advanced Battery Management Technologies for Electric Vehicles*. John Wiley & Sons.
- [209] Yan, J., Xu, G., Qian, H., Xu, Y., & Song, Z. (2011). Model predictive control-based fast charging for vehicular batteries. *Energies*, 4(8), 1178–1196.
- [210] Yang, J., Xia, B., Shang, Y., Huang, W., & Mi, C. (2016a). Improved battery parameter estimation method considering operating scenarios for HEV/EV applications. *Energies*, 10(1), 5.

- [211] Yang, N., Zhang, X., Shang, B., & Li, G. (2016b). Unbalanced discharging and aging due to temperature differences among the cells in a lithium-ion battery pack with parallel combination. *Journal of Power Sources*, 306, 733–741.
- [212] Ye, M., Guo, H., Xiong, R., & Yu, Q. (2018). A double-scale and adaptive particle filter-based online parameter and state of charge estimation method for lithium-ion batteries. *Energy*, 144, 789 – 799.
- [213] Yu, Z., Xiao, L., Li, H., Zhu, X., & Huai, R. (2017). Model parameter identification for lithium batteries using the coevolutionary particle swarm optimization method. *IEEE Trans. Ind. Electron*, 64(7), 5690–5700.
- [214] Zhang, C., Allafi, W., Dinh, Q., Ascencio, P., & Marco, J. (2018). Online estimation of battery equivalent circuit model parameters and state of charge using decoupled least squares technique. *Energy*, 142, 678 – 688.
- [215] Zhang, G., Ge, S., Xu, T., Yang, X.-G., Tian, H., & Wang, C.-Y. (2016). Rapid self-heating and internal temperature sensing of lithium-ion batteries at low temperatures. *Electrochimica Acta*, 218, 149–155.
- [216] Zhang, H., Ayoub, R., & Sundaram, S. (2017). Sensor selection for Kalman filtering of linear dynamical systems: Complexity, limitations and greedy algorithms. *Automatica*, 78, 202–210.
- [217] Zhang, L., Wang, L., Hinds, G., Lyu, C., Zheng, J., & Li, J. (2014). Multi-objective optimization of lithium-ion battery model using genetic algorithm approach. *Journal of Power Sources*, 270, 367–378.
- [218] Zhao, R., Zhang, S., Liu, J., & Gu, J. (2015). A review of thermal performance improving methods of lithium ion battery: Electrode modification and thermal management system. *Journal of Power Sources*, 299, 557–577.
- [219] Zheng, M., Qi, B., & Du, X. (2009). Dynamic model for characteristics of Li-ion battery on electric vehicle. In *2009 4th IEEE Conference on Industrial Electronics and Applications* (pp. 2867–2871).: IEEE.
- [220] Zheng, Y., Ouyang, M., Lu, L., & Li, J. (2015). Understanding aging mechanisms in lithium-ion battery packs: From cell capacity loss to pack capacity evolution. *Journal of Power Sources*, 278, 287–295.
- [221] Zia, A. I. & Mukhopadhyay, S. C. (2016). *Impedance Spectroscopy and Experimental Setup*, (pp. 21–37). Cham, Switzerland: Springer International Publishing.
- [222] Zou, C., Hu, X., Wei, Z., & Tang, X. (2017). Electrothermal dynamics-conscious lithium-ion battery cell-level charging management via state-monitored predictive control. *Energy*, 141, 250–259.
- [223] Zou, C., Manzie, C., & Nešić, D. (2018). Model predictive control for lithium-ion battery optimal charging. *IEEE/ASME Transactions on Mechatronics*, 23(2), 947–957.

- [224] Zou, C., Manzie, C., Nešić, D., & Kallapur, A. G. (2016). Multi-time-scale observer design for state-of-charge and state-of-health of a lithium-ion battery. *Journal of Power Sources*, 335, 121–130.
- [225] Zou, C., Manzie, C., & Nešić, D. (2016). A framework for simplification of PDE-based lithium-ion battery models. *IEEE Transactions on Control Systems Technology*, 24(5), 1594–1609.
- [226] Zou, Y., Hu, X., Ma, H., & Li, S. E. (2015). Combined state of charge and state of health estimation over lithium-ion battery cell cycle lifespan for electric vehicles. *Journal of Power Sources*, 273, 793–803.