

Techniques for Battery Management: Modeling, Estimation, Learning & Controls

by

Saehong Park

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Engineering - Civil and Environmental Engineering

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Scott Moura, Chair

Professor Claire Tomlin

Professor Alexandre Bayen

Fall 2020

The dissertation of Saehong Park, titled Techniques for Battery Management: Modeling, Estimation, Learning & Controls, is approved:

Chair	_____	Date	_____
	_____	Date	_____
	_____	Date	_____

University of California, Berkeley

Techniques for Battery Management: Modeling, Estimation, Learning & Controls

Copyright 2020
by
Saehong Park

Abstract

Techniques for Battery Management: Modeling, Estimation, Learning & Controls

by

Saehong Park

Doctor of Philosophy in Engineering - Civil and Environmental Engineering

University of California, Berkeley

Professor Scott Moura, Chair

This dissertation develops an advanced battery management system where lithium-ion battery dynamics are governed by electrochemical principles, and estimation & control techniques are applied to enhance the performance and longevity. The rapid progress in electrified transportation, electronic devices, and energy storage system needs cutting-edge management system that can monitor and control the electric battery system safely. We address this topic in three phases: i) modeling, ii) estimation, and iii) controls respectively.

First, we provide an overview of battery fundamentals and develop mathematical models for electrochemical battery phenomena. Electrochemical modeling is an important tool for simulating the performance of a battery. With the electrochemical model as a simulator, one can demonstrate prototypes of battery management system without conducting expensive experimental protocols. Furthermore, designing reduced-order electrochemical models helps to develop real-time estimation/control algorithms for large-scale battery systems.

Second, we propose a systematic framework to identify battery cell using the electrochemical model. To accurately represent the cell chemistry, the model parameters must be correctly identified. Good fitting parameters take a long time to identify. Often, it requires disassembling the cell and performing specialized electrochemical characterization tests, a costly and difficult process. The proposed parameter estimation framework for an electrochemical model enables to estimate electrochemical parameters by using input-output data.

Lastly, we then develop a novel battery charging strategy for the identified electrochemical model while accounting for the electrochemical constraints associated to aging mechanism. The model-based optimal solutions reveal that the characteristic of optimal charging protocol is *Bang-Ride* control, which provides theoretical supports on the existing charging strategy. Furthermore, learning-based controller design via reinforcement learning is highlighted as an adaptive controller for battery charging strategy. The performance of this learning-based feedback controller is demonstrated with other charging strategies.

To My Parents

For their infinite support and encouragement

Contents

Contents	ii
List of Figures	iv
List of Tables	vi
1 Introduction	1
1.1 Energy Systems: Li-ion Batteries	1
1.1.1 Battery Systems Overview	3
1.2 Outline	4
1.3 Contributions	4
2 Modeling	6
2.1 Electrochemical Full-Order Model	6
2.1.1 Doyle-Fuller-Newman (DFN) model	6
2.2 Electrochemical Reduced-Order Models	9
2.2.1 Single Particle Model (SPM)	9
2.2.2 Single Particle Model with Electrolyte dynamics (SPMe)	11
2.2.3 Single Particle Model with Electrolyte and Thermal dynamics (SPMeT)	16
2.3 Hybrid Electrochemical Modeling	17
2.3.1 Introduction	17
2.3.2 Neural Network Structure and Training Algorithm	19
2.3.3 Stability Analysis	21
2.3.4 Simulation Results	23
2.3.5 Summary	26
3 Estimation	27
3.1 Characterization of Electrochemical Model	27
3.1.1 Introduction	27
3.1.2 Electrochemical Parameters	29
3.1.3 System Generalization	30
3.1.4 Optimal Experimental Design	32

3.1.5	Experimental Results	46
3.1.6	Validation	49
3.1.7	Summary	53
3.2	State-Of-Health (SOH) Estimation	54
3.2.1	Introduction	54
3.2.2	The State of Health (SOH) for Electrochemical Model	55
3.2.3	SOH Parameter Estimation	57
3.2.4	Simulation Results	59
3.2.5	Summary	61
4	Learning & Optimal Controls	62
4.1	Model-based fast-charging control	62
4.1.1	Introduction	62
4.1.2	Preliminaries	64
4.1.3	Fast Charging Optimal Solution	66
4.1.4	Summary	75
4.2	Learning-based fast-charging control	76
4.2.1	Introduction	77
4.2.2	Reinforcement Learning Framework	79
4.2.3	Actor-Critic	80
4.2.4	Fast-Charging Problem	83
4.2.5	Results and Discussion	84
4.2.6	Summary	94
5	Conclusion	96
5.1	Contributions	96
5.2	Future Research Directions	97
A	Nomenclature	98
B	DFN Simulator	100
B.1	Numerical Implementation	100
B.2	Visualization	108
C	Sensitivity Validation	109
C.1	Sensitivity Dynamics vs Perturbation	110
	Bibliography	115

List of Figures

1.1	Battery Statistics	2
1.2	Typical battery cells construction adopted from [89, 140]	3
2.1	Schematic of the first principles electrochemical model, also known as the pseudo two-dimensional model or Doyle-Fuller-Newman (DFN) model.	7
2.2	Schematic of single particle model for Li-ion battery	9
2.3	Discharge curves of DFN and SPM at various C-rates	12
2.4	The comparison of proposed SPMe(-) with DFN(-) and previous SPMe(- -) model [79] for NCA electrochemical parameters identified in [86].	16
2.5	Proposed framework for hybrid modeling approach based on electrochemical model.	19
2.6	Elman network structure.	20
2.7	Voltage error comparison for UDDSx2 cycle: (a) Input current. (b) SPMe. (c) The special case of Elman network (d) The original Elman network.	24
2.8	Cumulative distribution function of absolute voltage error on UDDS profile, with 90% quantile errors annotated.	26
3.1	Battery parameter identification procedure in the literature	28
3.2	NCR 18650PF Li-ion battery used for parameter edentification.	29
3.3	The proposed model-based design of experiments framework for parameter identification in of the DFN model.	33
3.4	OCV comparison between identified model and experimental data for three averaged C/50 charge/discharge cycles.	36
3.5	The sorted average orthogonalized sensitivity magnitudes across all inputs in the input library.	39
3.6	The comparison of model output impact by changing parameters.	40
3.7	Experimental outputs and quantification.	44
3.8	Identification results for optimal designed inputs.	48
3.9	Parameter estimates and 95% confidence intervals versus the nominal parameter values.	50
3.10	Experimental validation results for the NCA 18650 battery with testing profiles.	52
3.11	The OCV trajectory for given mole of lithium in solid-phase. The solid line describes the trajectories of OCV curve.	56

3.12	The impact of SOH parameters $n_{Li,s}$, $n_{Li,e}$ on voltage output. Top: A decrease of $n_{Li,s}$ decreases capacity, i.e., capacity fade. Bottom: a decrease of $n_{Li,e}$ affects the instantaneous voltage, which contributes to power fade.	57
3.13	The simulation results on the voltage deviations as well as SOH estimation using Urban Dynamometer Driving Schedule (UDDS) input profile. The voltage RMSE reduces below 1 mV after 4 iterations using Levenberg-Marquardt algorithm. . .	60
4.1	Numerical constrained optimal control results using analytical solutions in Table 4.2.	76
4.2	Actor-Critic Structure.	81
4.3	PCA results on the electrochemical model.	86
4.4	States-based learning policy results with different initializations of the Actor-Critic networks at $T_{amb} = 25^{\circ}\text{C}$ and $T_{amb} = 15^{\circ}\text{C}$	87
4.5	Validation of states-based learning policy compared with modified reference governor at $T_{amb} = 25^{\circ}\text{C}$ and $T_{amb} = 15^{\circ}\text{C}$	89
4.6	Outputs-based learning policy results with different initialization of Actor-Critic networks at $T_{amb} = 25^{\circ}\text{C}$ and $T_{amb} = 15^{\circ}\text{C}$	90
4.7	Validation of outputs-based learning policy compared with CC-CT-CV at $T_{amb} = 25^{\circ}\text{C}$ and $T_{amb} = 15^{\circ}\text{C}$	91
4.8	Adaptive outputs-based learning policy results with different initialization of Actor-Critic networks at $T_{amb} = 25^{\circ}\text{C}$ and $T_{amb} = 15^{\circ}\text{C}$	92
4.9	Validation of adaptive outputs-based learning policy compared with CC-CT-CV at $T_{amb} = 25^{\circ}\text{C}$ and $R_f^- = 0.04\Omega$	93
B.1	The visualization of DFN simulator when a constant current is applied to the simulator.	108
C.1	Sensitivity validations for D_s^- , D_s^+ , R_s^- , and R_s^+	110
C.2	Sensitivity validations for σ^- , σ^+ , $D_e(\cdot)$, and ε_e^-	111
C.3	Sensitivity validations for ε_e^{sep} , ε_e^+ , $\kappa(\cdot)$, and t_c^0	112
C.4	Sensitivity validations for $\frac{d \ln f_{c/a}}{d \ln c_e}$, k^- , k^+ and R_f^-	113
C.5	Sensitivity validations for R_f^+ and $c_e(x, 0)$	114

List of Tables

2.1	Single Particle Model Parameters	10
2.2	The output voltage error with respect to DFN model.	15
2.3	RMSE comparison for different models (w/ and w/o learning) and input profiles in the training phase. The number in parentheses is the number of states in the model.	25
2.4	RMSE comparison for different models (w/ and w/o learning) and input profiles in the testing phase, using RNNs trained on UDDS. The number in parentheses is the number of states in the model.	25
3.1	Fixed geometric / thermal parameters.	30
3.2	Parameters of interest for identification.	31
3.3	DAE notation for the electrochemical model.	31
3.4	Identified equilibrium parameters.	36
3.5	Cell SOC to voltage mapping.	36
3.6	Pulse profile specification.	37
3.7	Sinusoid profile specification.	38
3.8	Driving cycle profile specification.	38
3.9	Dynamic Parameter Groups.	41
3.10	Input profiles from optimal experimental design.	46
3.11	Final identification results for NCA 18650 Li-ion battery.	49
3.12	RMSE between simulation and experimental measurement up to 90% capacity in constant discharge.	53
3.13	RMSE between simulation and experimental measurement with sinusoidal and pulse inputs.	53
3.14	RMSE between simulations and experimental measurements with driving cycle inputs.	53
4.1	A, B, and C matrices for three common battery chemistries.	68
4.2	The analytical solution of constrained optimal control problem (4.34) for $C_b \leq \bar{C}$. Unknown variables, ζ , $p(0)$, and $\eta(t_f)$ are derived in (4.79), (4.80), (4.72).	75
4.3	Actor-critic hyper parameters.	84
4.4	Summary of charging time [min.] and adaptability	94

A.1	List of symbols (1/2).	98
A.2	List of symbols (2/2).	99
B.1	DAE notation for DFN states.	101
C.1	Parameter list for sensitivity validation	109

Acknowledgments

I would like to thank people around me for their enormous support past five years of my doctoral studies and research. This dissertation is not just an outcome of research execution, but is the product of collective intelligence in the academic institution. The chain of students-staff-administrators-faculty members fosters intellectual research culture and inspires me as a scientific scholar. I am very pleased to take this opportunity to build up knowledge and contribute to academic society.

First of all, I am so fortunate to have Prof. Scott Moura as my academic advisor who always give me complimentary words and encouragement. His incredible devotion to students raises me up as an independent researcher with critical thinking and problem solving in academic research. I remember the moment when I first met him in Davis Hall where I delivered brief presentation of my research plans as a Ph.D. student. That led me to start graduate studies under his guidance at University of California, Berkeley. His advice and trust motivated me to pursue academic excellence in both classroom and research environment. He is my academic parents who fulfill his duties and present delightful memories to me.

I am extremely thankful to my dissertation committee members, Prof. Claire Tomlin and Prof. Alexandre Bayen for providing critical feedback on battery management system research and other collaborative works. Their insightful advice creates solid groundwork for the work presented in this dissertation.

Since I join the Energy, Controls, and Applications Lab (eCAL) in 2015, I have been through the lab is growing as it is of today. Dr. Xiaosong Hu and Dr. Satadru Dey mentored me how to conduct research as a team when I first join this lab. My senior lab-mates Hector Perez, Eric Burger, Eric Munsing, and Caroline Le Floch helped me to settle down at the start of the lab life. I also thank to Laurel Dunn, Zach Gima, Dong Zhang, Sangjae Bae, and Bertrand Travacca who joined the lab around the same time building the second generation of eCAL. It was nice to have you as co-workers in the lab having meals together, discussions on course works, retreats (lab tradition), and lab meetings. I also would like to thank Mathilde Badoual, Soomin Woo, Dylan Kato, Teng Zeng, Aaron Kandel, Patrick Keyantuo, Preet Gill, and Chitra Dangwal who are the next generation of eCAL and very smart students.

I would like to express gratitude to the undergraduate students I have mentored at eCAL for supporting battery controls projects: Raja Selvakumar, Michael Whitemeyer, Shiva Raisinghani, Sonia Martin, Vanessa Hernandez-Cruz.

I wish to acknowledge eCAL post-docs that have mentored me inside/outside of campus. Thanks Dr. Satadru Dey and Dr. Hector Perez for enjoying Korean bbq with me after project meetings. Thanks Dr. Hongcai Zhang for playing tennis with me. Thanks Dr. Milad Memarzadeh who selected me as an intern at NASA Ames Center and having interesting research discussion. Thanks Dr. Chao Sun for introducing delicious Chinese hotpot with Dong Zhang in Beijing .

I am also very thankful to visiting students in eCAL. They created diverse research environment and broaden my research horizons: Changfu Zhou, Leobardo Camacho-Solorio,

Hongcai Zhang, Zhou Zhe, Luis Couto, Ross Drummond, Huan Yu, and Andrea Pozzi. I thank them for their help and discussions.

I humbly thank industrial mentors who financially support the project and publish research papers together. I thank Reinhardt Klein for his friendship, witty humor and fruitful discussions on battery research. He hosted me as an intern at Bosch research technology center and gave invaluable career advice. I also thank Won Tae Joe and his team members: Hyoung Jun Ahn, Geumbee Kim, and Yohwan Choi in LG Chem. Their warm friendship and hospitality were shoring up my confidence.

I wish to thank graduate students in Korean Graduate Student Association at UC Berkeley. As a member of KGSA, we organize social activities and outreach together. I thank the graduate student committee: Kiwoo Shin, Hyundon Ha, Eric Choi, Taesung Park, Donghoon Kim, Yeojun Kim, Jinkyu Kim for their dedication and commitments.

I am lucky to have found Maribel Jaquez a few years ago here at Berkeley during a graduate social event. We became close friends by conversing about things that we like in common such as good Mexican food and outreach. I thank my loving girlfriend of a little over two years for being extremely supportive in addressing the many challenges that have come my way.

I am deeply indebted to Pistos Korean Presbyterian Church family, Pr. Daniel Cho, Maria Cho, Shinye Choi, Seongeun Jeong, Hye Jin Choi, and other members who support me both physically and spiritually. I am so fortunate to meet Seungyeon Baek in the church. I thank my lovely girlfriend over two years for being supportive and positive in any given situation.

Finally, I would like to thank my parents Ilkon Park and Youngkang Yoon, my brother Sojun Park for their endless love and support throughout my doctoral studies. I could not have started my doctoral studies without them.

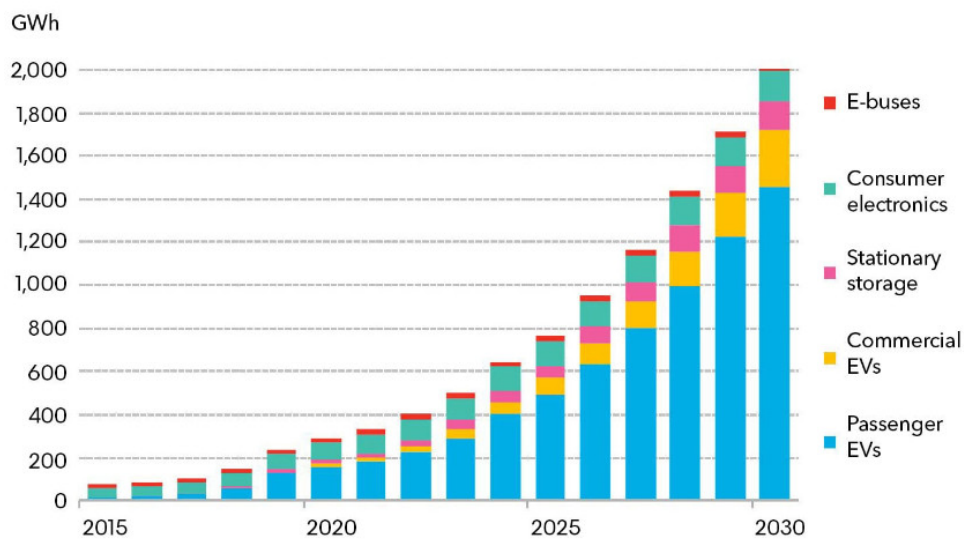
Chapter 1

Introduction

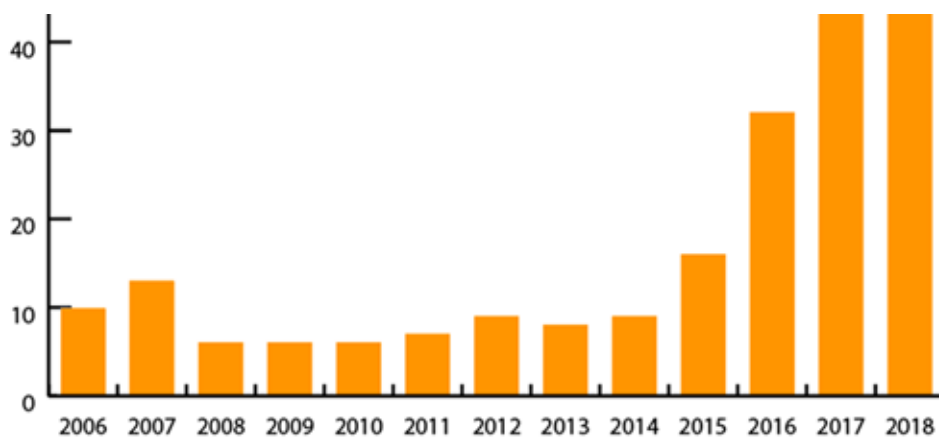
1.1 Energy Systems: Li-ion Batteries

As human civilization becomes more and more urban-centric, effective energy systems management is desired to handle uncertainties under both demand and supply sides. Furthermore, electrified transportation and energy storage have shifted the paradigm of energy systems and promised transition to zero-emission society. In this context, lithium-ion batteries play a significant role as a main component for energy system to drive green society. The overall lithium-ion battery demand is counted as 160 gigawatt hours (GWhs) in 2019, and expected to increase around 2,300 GWhs in 2030 according to market research shown in Fig. 1.1a. As more batteries are deployed to the urban systems, substantial battery accidents are reported and released to media recent years. In aerospace, the reported accidents involving lithium batteries in aircraft has risen sharply in recent years as shown in Fig. 1.1b according to federal aviation administration (FAA). In ground transportation, a number of battery fires in electric vehicle (EV) is steadily increasing, and the investigation for safety concerns of EV is discussed in [112]. To ensure safety and efficiency, better monitoring techniques for battery systems are inevitable.

With the perspective of engineering sciences, we recognize a battery as a physical system that can be described by various mathematical models, in which advanced control and estimation methodologies can be applied to improve the performance. Although models differ greatly in complexity, electrochemical models can provide physical interpretation of battery's behavior during operation. In this thesis, we will discuss mathematical models for lithium-ion batteries from electrochemistry perspectives. When the model is constructed, the model-based estimation and control techniques are utilized to characterize the battery cell behavior, and charge the battery safely. Lastly, a recent advancement of machine learning techniques is addressed to the energy systems. This thesis is all about techniques for battery management system with emphasis on modeling, estimation, controls & learning techniques to enhance the operation of lithium ion batteries with respect to model fidelity, charge time, and life time.



(a) Annual lithium-ion battery demand and expectation [8].



(b) Number of events with smoke, fire, and explosion involving lithium batteries in aircraft [33].

Figure 1.1: Battery Statistics

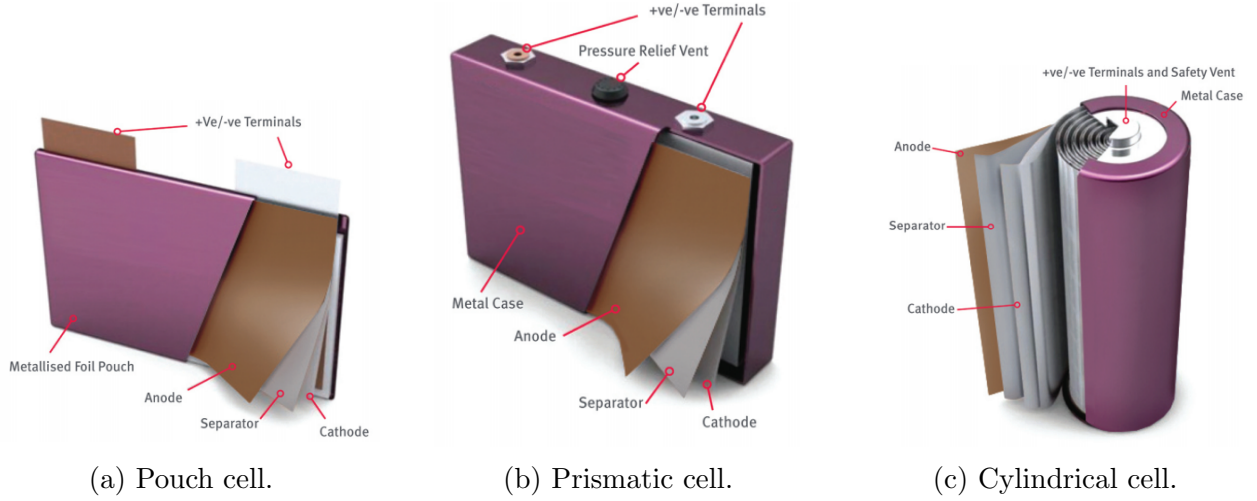


Figure 1.2: Typical battery cells construction adopted from [89, 140]

1.1.1 Battery Systems Overview

Commercial batteries are categorized by three types: cylindrical cells, prismatic cells, and pouch cells as illustrated in Fig. 1.2. These cells are assembled to support high current / high voltage in battery pack for energy system. For most of EVs, the number of 5,376 cells is required to provide “60” kWh energy . One of unique characteristics in battery is that we have limited information about the system. For instance we only measure input current, output voltage, and temperature. To understand how it is functioning during the operation, one needs to understand the fundamentals of the battery. For instance, the cylindrical battery is constructed spirally where the copper foil is used as the current collector for the anode, which is composed of graphite and attached to the negative terminal of the cell. The separator is an electrical insulator which keeps electrons from passing through but they let lithium ions through. The aluminum foil is used as the current collector for the cathode. Since every metallic materials have certain potential stability window, aluminum is stable at higher potentials, which is cathode, while copper is stable at lower potentials, which is anode. The electrodes are rolled up into a jelly roll, and then inserted into a cylindrical can. The liquid electrolyte serves as catalyst to make a battery conductive by promoting the movement of ions from cathode to the anode on charge and in reverse on discharge. When the battery is fully charged, the majority of the lithium in the cell exists within the solid phase particles (or the active material) in the anode lithiated carbon, Li_xC_6 . When the battery is discharged, the lithium is delithiated to the surface of spherical particles in the anode, and then the lithium ion is transported to the cathode while the electron flows through the circuit.



The electron and lithium ion are lithiated at the active material surface in the cathode, typically a lithium metal oxide $LiMO_2$. This electrochemical reaction can be described as:



The transport of lithium ion process is driven by electric potential difference between the anode and cathode, which implies that the reverse process occurs for charging phase. The configuration of Li-ion battery is determined which materials are used for each electrode. For the anode material, graphite and lithium-titanium-oxide (LTO) are widely used while the cathode can be made of lithium-cobalt-oxide (LCO), lithium-iron-phosphate (LFP), lithium nickel cobalt manganese oxide (NMC), and lithium nickel cobalt aluminium oxide (NCA). The separator can be made of polyolefin materials, i.e., polyethylene and polypropylene. The electrolyte can be made of lithium salts i.e., lithium hexafluorophosphate (LiPF₆). The electrochemical model can be constructed based on the understanding of electrochemical reaction with material properties. We will discuss the electrochemical models in Chapter 2.

1.2 Outline

The purpose of this thesis is to describe how electrochemical lithium ion battery models can be estimated and controlled for advanced battery management system. A background on the models and modeling techniques that are used in the thesis is provided in Chapter 2. The cell characterization of the battery model with experimental validation is presented in Chapter 3 as well as state-of-health estimation. With the identified cell information, the fast charging control problem is addressed in chapter 4. The key contributions of this dissertation and opportunities for future work are presented in Chapter 5.

1.3 Contributions

The goal of this dissertation is to provide various techniques for battery management system which enhances the performance of lithium ion batteries through model-based estimation/control approach and model-free learning approach. The new contributions to knowledge on battery systems can be summarized as:

1. Model Development of electrochemical battery system

- *First principles of electrochemical models:* The full-order electrochemical battery model and associated reduced order models are presented in Chapter 2. Based on the existing electrochemical models, the modeling for reduced-order models is studied, and the improvement for reduced-order model is proposed in Section 2.2.
- *Hybrid modeling approach for battery system:* Combining data-driven and first principles is proposed as hybrid modeling approach in Section 2.3. This grey-box model can capture the unknown dynamics of Li-ion battery, such as degradation process.

2. Cell characterization of battery system

- *Optimal experiment design:* Cell characterization is critical aspect of the battery management system to ensure safety and efficiency. The objective of optimal experimental design is to derive optimal inputs that maximize the identifiability of the system model. With proper objective function that represents parameter identifiability, the optimal control problem is transformed to experiment selection via convex programming in Section 3.1.
- *Sensitivity analysis for electrochemical model:* The sensitivity of electrochemical parameters with respect to voltage output is evaluated by deriving sensitivity dynamics from dynamical systems. This analysis demonstrates that some of parameters have huge impacts on the voltage output, which can be used for parameter estimation algorithm in Section 3.1.
- *Battery-health identification:* The battery health of electrochemical battery model is defined as the moles of lithium for both solid-phase and electrolyte phase. These battery health information is estimated by using input-output measurement via simulation in Section 3.2.

3. Fast-charging via Optimal control & Reinforcement learning

- *model-based controller design:* the optimal charging trajectory for battery fast-charging is determined via Pontryagin Maximum Principle. Control optimization is performed on a reduced battery model in Section 4.1.
- *learning-based controller design:* the learning-based controller for battery fast-charging is obtained via approximate dynamic programming (also called reinforcement learning). The learning-based controller plays as feedback controller for both states and outputs compared with existing model-based approaches in Section 4.2. This learning-based controller is also demonstrated as an adaptive controller that updates its control policy when the system behavior has changed. Such feature is highlighted from the battery management system perspective.

Chapter 2

Modeling

In this chapter, we will introduce electrochemical battery models and modeling techniques that are covered in the dissertation. First of all, the full-order electrochemical-thermal model is presented in Section 2.1. Then, various types of reduced-order model is derived from the full-order model in Section 2.2. Lastly, the hybrid modeling approach that combines electrochemical model and data-driven model is discussed in Section 2.3.

2.1 Electrochemical Full-Order Model

The electrochemical battery model provides physical interpretation of battery inside, that is the spatiotemporal dynamics of lithium ion concentration, electric potential, and the Butler-Volmer kinetics which govern the intercalation reactions. A schematic of the model is provided in Fig. 2.1. This cross section exhibits three regions: a negative electrode (anode), the separator, and a positive electrode (cathode). Each region is denoted by superscript $j \in \{n, s, p\}$ representing negative electrode, separator, and positive electrode respectively.

2.1.1 Doyle-Fuller-Newman (DFN) model

We consider the Doyle-Fuller-Newman (DFN) model to predict the evolution of lithium concentration in the solid $c_s^\pm(x, r, t)$, lithium concentration in the electrolyte $c_e(x, t)$, solid electric potential $\phi_s^\pm(x, t)$, electrolyte electric potential $\phi_e(x, t)$, ionic current $i_e^\pm(x, t)$, molar ion fluxes $j_n^\pm(x, t)$, battery core temperature $T_1(t)$, and surface temperature $T_2(t)$ [119, 64]. Notation is defined in Table A.2. The governing equations are given by

$$\frac{\partial c_s^\pm}{\partial t}(x, r, t) = \frac{1}{r^2} \frac{\partial}{\partial r} \left[D_s^\pm r^2 \frac{\partial c_s^\pm}{\partial r}(x, r, t) \right], \quad (2.1)$$

$$\varepsilon_e^j \frac{\partial c_e^j}{\partial t}(x, t) = \frac{\partial}{\partial x} \left[D_e^{\text{eff}}(c_e^j) \frac{\partial c_e^j}{\partial x}(x, t) + \frac{1 - t_c^0}{F} i_e^j(x, t) \right], j \in \{-, sep, +\}, \quad (2.2)$$

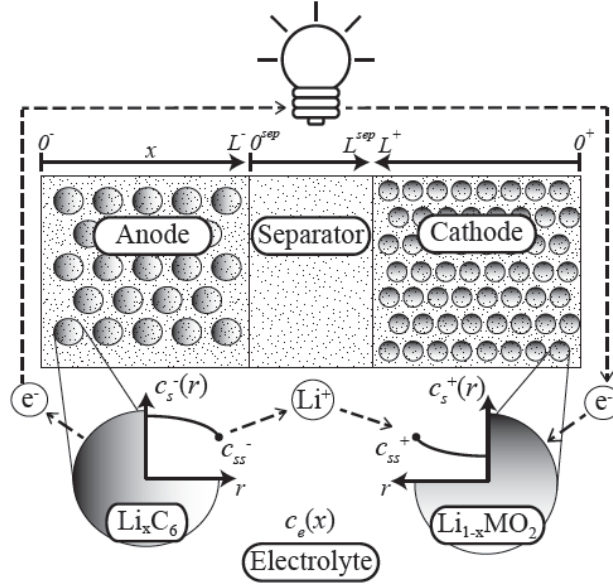


Figure 2.1: Schematic of the first principles electrochemical model, also known as the pseudo two-dimensional model or Doyle-Fuller-Newman (DFN) model.

$$\sigma^{\text{eff},\pm} \cdot \frac{\partial \phi_s^\pm}{\partial x}(x, t) = i_e^\pm(x, t) - I(t), \quad (2.3)$$

$$\begin{aligned} \kappa^{\text{eff}}(c_e) \cdot \frac{\partial \phi_e}{\partial x}(x, t) &= -i_e^\pm(x, t) \\ &+ \kappa^{\text{eff}}(c_e) \cdot \frac{2RT_1}{F}(1 - t_c^0) \cdot \left(1 + \frac{d \ln f_{c/a}}{d \ln c_e}(x, t)\right) \frac{\partial \ln c_e}{\partial x}(x, t), \end{aligned} \quad (2.4)$$

$$\frac{\partial i_e^\pm}{\partial x}(x, t) = a_s^\pm F j_n^\pm(x, t), \quad (2.5)$$

$$j_n^\pm(x, t) = \frac{1}{F} i_0^\pm(x, t) \left[e^{\frac{\alpha_a F}{RT_1} \eta^\pm(x, t)} - e^{-\frac{\alpha_c F}{RT_1} \eta^\pm(x, t)} \right], \quad (2.6)$$

$$C_1 \frac{dT_1}{dt}(t) = h_{12}[T_2(t) - T_1(t)] + \dot{Q}(t), \quad (2.7)$$

$$C_2 \frac{dT_2}{dt}(t) = h_{12}[T_1(t) - T_2(t)] + h_{2a}[T_{\text{amb}}(t) - T_2(t)], \quad (2.8)$$

$$\dot{Q}(t) = I(t) \left[V(t) - [U^+(\bar{c}_s^+) - U^-(\bar{c}_s^-)] + T_1(t) \left[\frac{\partial U^+}{\partial T_1}(\bar{c}_s^+) - \frac{\partial U^-}{\partial T_1}(\bar{c}_s^-) \right] \right], \quad (2.9)$$

where $D_e, \kappa, f_{c/a}$ are functions of $c_e(x, t)$ and $D_e^{\text{eff}} = D_e(c_e) \cdot (\varepsilon_e^j)^{\text{brug}}$, $\sigma^{\text{eff}} = \sigma \cdot (\varepsilon_s^j + \varepsilon_f^j)^{\text{brug}}$, $\kappa^{\text{eff}} = \kappa(c_e) \cdot (\varepsilon_e^j)^{\text{brug}}$ are the effective electrolyte diffusivity, effective solid conductivity, and effective electrolyte conductivity given by the Bruggeman relationship. We also have

$$i_0^\pm(x, t) = k^\pm [c_{ss}^\pm(x, t)]^{\alpha_c} [c_e(x, t) (c_{s,\text{max}}^\pm - c_{ss}^\pm(x, t))]^{\alpha_a}, \quad (2.10)$$

$$\eta^\pm(x, t) = \phi_s^\pm(x, t) - \phi_e(x, t) - U^\pm(c_{ss}^\pm(x, t)) - FR_f^\pm j_n^\pm(x, t), \quad (2.11)$$

$$c_{ss}^\pm(x, t) = c_s^\pm(x, R_s^\pm, t), \quad (2.12)$$

$$\Delta T_1(x, t) = U^\pm(\bar{c}_s^\pm(x, t)) - T_1(t) \frac{\partial U^\pm}{\partial T_1}(\bar{c}_s^\pm(x, t)), \quad (2.13)$$

$$\bar{c}_s^\pm(x, t) = \frac{3}{(R_s^\pm)^3} \int_0^{R_s^\pm} r^2 c_s^\pm(x, r, t) dr. \quad (2.14)$$

Along with these equations, boundary and initial conditions are required for well-posedness. The boundary conditions for the solid-phase diffusion PDE (2.1) are

$$\frac{\partial c_s^\pm}{\partial r}(x, 0, t) = 0, \quad \frac{\partial c_s^\pm}{\partial r}(x, R_s^\pm, t) = -\frac{1}{D_s^\pm} j_n^\pm. \quad (2.15)$$

The boundary conditions for the electrolyte-phase diffusion PDE (2.2) are given by

$$\frac{\partial c_e}{\partial x}(0^-, t) = \frac{\partial c_e}{\partial x}(0^+, t) = 0, \quad (2.16)$$

$$D_e^{\text{eff},-}(c_e(L^-)) \frac{\partial c_e}{\partial x}(L^-, t) = D_e^{\text{eff,sep}}(c_e(0^{\text{sep}})) \frac{\partial c_e}{\partial x}(0^{\text{sep}}, t), \quad (2.17)$$

$$D_e^{\text{eff,sep}}(c_e(L^{\text{sep}})) \frac{\partial c_e}{\partial x}(L^{\text{sep}}, t) = D_e^{\text{eff},+}(c_e(L^+)) \frac{\partial c_e}{\partial x}(L^+, t), \quad (2.18)$$

$$c_e(L^-, t) = c_e(0^{\text{sep}}, t), \quad (2.19)$$

$$c_e(L^{\text{sep}}, t) = c_e(0^+, t). \quad (2.20)$$

The boundary conditions for the solid-phase potential ODE (2.3) are given by

$$\frac{\partial \phi_s^-}{\partial x}(L^-, t) = \frac{\partial \phi_s^+}{\partial x}(L^+, t) = 0. \quad (2.21)$$

The boundary conditions for the electrolyte-phase potential ODE (2.4) are given by

$$\phi_e(0^-, t) = 0, \quad (2.22)$$

$$\phi_e(L^-, t) = \phi_e(0^{\text{sep}}, t), \quad (2.23)$$

$$\phi_e(L^{\text{sep}}, t) = \phi_e(L^+, t). \quad (2.24)$$

The boundary conditions for the ionic current ODE (2.5) are given by

$$i_e^-(0^-, t) = i_e^+(0^+, t) = 0, \quad (2.25)$$

and also note that $i_e(x, t) = I(t)$ for $x \in [0^{\text{sep}}, L^{\text{sep}}]$.

In addition, the parameters $D_s^\pm, D_e, \kappa, k^\pm$ vary with temperature via the Arrhenius relationship:

$$\psi = \psi_{\text{ref}} \exp \left[\frac{E_\psi}{R} \left(\frac{1}{T_1} - \frac{1}{T_{\text{ref}}} \right) \right], \quad (2.26)$$

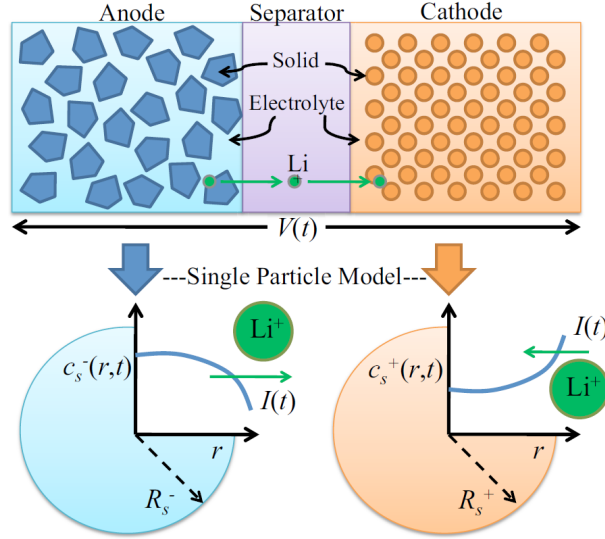


Figure 2.2: Schematic of single particle model for Li-ion battery

where ψ represents a temperature dependent parameter, E_ψ is the activation energy [J mol^{-1}], and ψ_{ref} is the reference parameter value at reference temperature T_{ref} .

The input to the model is the applied current density $I(t)$ [A m^{-2}], and the output is the voltage measured across the current collectors

$$V(t) = \phi_s^+(0^+, t) - \phi_s^-(0^-, t) - R_c I(t). \quad (2.27)$$

The definitions of electrochemical parameters and states are listed in Appendix A. The numerical implementations of the full-order model can be found in Appendix B.

2.2 Electrochemical Reduced-Order Models

The discretization of coupled partial differential equations (PDEs), ordinary differential equations (ODEs), and algebraic equations (AEs) results in the full-order model as differential algebraic equations (DAEs), whose differentials states and algebraic states contain hundreds of variables. This complexity makes control design difficult in real-time application. In this dissertation, we leverage the model reduction techniques to solve the electrochemical model equations orders of magnitude faster than real-time.

2.2.1 Single Particle Model (SPM)

In this section we describe the simplest reduced-order of electrochemical, called single particle model (SPM). This model idealizes each electrode as a single spherical porous particle while neglecting electrolyte dynamics, as shown in Fig. 2.2. Specifically, the electrolyte concentration is approximated as constant in space and time, and this restriction causes

errors which will be discussed later in this section. The intercalation process and mass transport is modeled by a linear diffusion PDE over spherical coordinates:

$$\frac{\partial c_s^-}{\partial t}(r, t) = D_s^- \left[\frac{2}{r} \frac{\partial c_s^-}{\partial r}(r, t) + \frac{\partial^2 c_s^-}{\partial r^2}(r, t) \right], \quad (2.28)$$

$$\frac{\partial c_s^+}{\partial t}(r, t) = D_s^+ \left[\frac{2}{r} \frac{\partial c_s^+}{\partial r}(r, t) + \frac{\partial^2 c_s^+}{\partial r^2}(r, t) \right], \quad (2.29)$$

with Neumann boundary conditions,

$$\frac{\partial c_s^-}{\partial r}(0, t) = 0, \quad \frac{\partial c_s^-}{\partial r}(R_s^-, t) = \frac{I(t)}{D_s^- F a^- A L^-}, \quad (2.30)$$

$$\frac{\partial c_s^+}{\partial r}(0, t) = 0, \quad \frac{\partial c_s^+}{\partial r}(R_s^+, t) = -\frac{I(t)}{D_s^+ F a^+ A L^+}. \quad (2.31)$$

The Neumann boundary conditions at $r = R_s^-$ and $r = R_s^+$ represent that the molar flux of lithium entering / exiting the electrode is proportional to the input current $I(t)$. For clarity, all parameters related to the SPM are listed in Table 2.1.

Symbol	Description	Units
A	Cell cross sectional area	m^2
a^j	Specific interfacial surface area	m^2/m^3
c_s^j	Concentration in solid phase	mol/m^3
c_{ss}^j	Concentration at particle surface	mol/m^3
$c_{s,max}^j$	Max concentration in solid phase	mol/m^3
D_s^j	Diffusion coefficient in solid phase	m^2/m^3
F	Faraday's constant	C/mol
I	Input current	A
i_0^j	Exchange current density	V
L^j	Electrode thickness	m
R	Universal gas constant	$\text{J}/\text{mol-K}$
R_f	Lumped current collector resistance	Ω
R_s^j	Particle radius	m
r	Radial coordinate	m
T	Cell temperature	K
t	Time	seconds
U^j	Equilibrium potential	V
V	Output voltage	V
α^j	Anodic/cathodic transfer coefficient	-

Table 2.1: Single Particle Model Parameters

Note that superscript j in Table 2.1 indicates anode, separator and cathode such as $j \in \{+, \text{sep}, -\}$. The terminal voltage output is governed by a combination of electric overpotential, electrode thermodynamics, and Butler-Volmer kinetics, yielding:

$$\begin{aligned} V(t) = & \frac{RT}{\alpha^+ F} \sinh^{-1} \left(\frac{-I(t)}{2a^+ AL^+ i_0^+(c_{ss}^+(t))} \right) \\ & + \frac{RT}{\alpha^- F} \sinh^{-1} \left(\frac{I(t)}{2a^- AL^- i_0^-(c_{ss}^-(t))} \right) \\ & + U^+(c_{ss}^+(t)) - U^-(c_{ss}^-(t)) + R_f I(t), \end{aligned} \quad (2.32)$$

where the exchange current density i_0^j and solid-electrolyte surface concentration c_{ss}^j are computed as:

$$i_0^j(c_{ss}^j) = k^j \sqrt{c_e^0 c_{ss}^j(t) (c_{s,\max}^j - c_{ss}^j(t))}, \quad (2.33)$$

$$c_{ss}^j(t) = c_s^j(R_s^j, t), \quad j \in \{+, -\}. \quad (2.34)$$

The model parameters used in this study come from DUALFOIL simulation packages which is publicly accessible [82]. These parameter values correspond to a lithium cobalt oxide cathode, graphite anode chemistry.

The drawback of using SPM is that predicted voltage accuracy decreases as C-rate increases. Discharge curves with different C-rates are plotted in Fig. 2.3. We use the DFN as our baseline or “truth data”. At low C-rates, i.e. 0.1C and 0.5C, we observe that the SPM voltage approximates the DFN voltage well. When C-rate increases, the prediction accuracy decreases because the electrolyte lithium concentration gradient becomes significant, and our key SPM assumption is violated. Hence, the SPM is no longer valid for high C-rate modeling. Refer to [75] for further discussion about electrolyte dynamics. This presents a significant challenge for BMS applications, such as electrified vehicles, where high C-rates are commonly experienced. Large voltage output mismatch could lead to misguided control policies. In response to the SPM’s limitations, the authors of [68] propose a modified SPM where an empirical current-dependent approximation of the electrolyte phase potential difference is augmented to the output voltage model. We address this challenge from the perspective of using data-driven modeling, i.e., recurrent neural networks (RNNs) to capture the un-modeled SPM dynamics in Section 2.3.

2.2.2 Single Particle Model with Electrolyte dynamics (SPMe)

Adding electrolyte dynamics into SPM can enhance the fidelity of reduced-order electrochemical model. This model is called Single Particle Model with Electrolyte dynamics (SPMe). From the previous work [79], the reduced-order model and associated state estimators are derived with the following assumptions:

- **[A1]:** The solid phase Li concentration in each electrode is constant in the spatial coordinate x , uniformly in time. Mathematically, $c_s^\pm(x, r, t)$ and $j_n^\pm(x, t)$ are constant in the x direction.

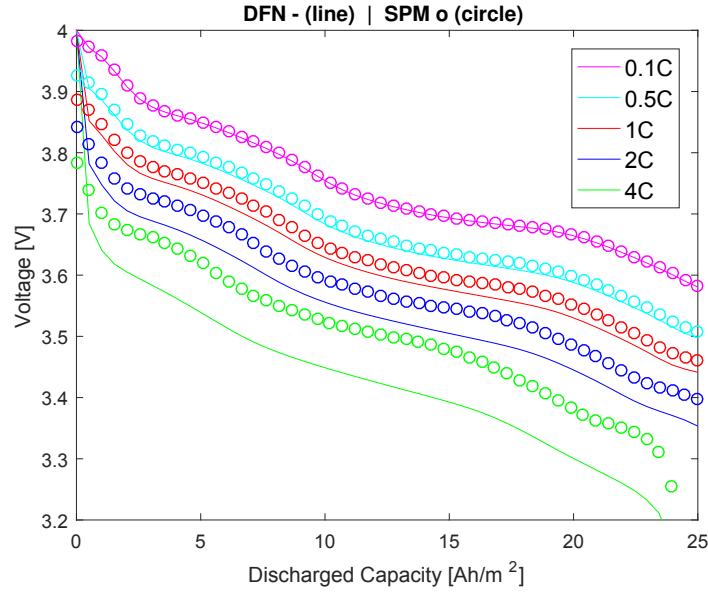


Figure 2.3: Discharge curves of DFN and SPM at various C-rates

- **[A2]:** The exchange current density term $i_0^\pm(x, t)$ can be approximated by its averaged value $\bar{i}_0^\pm(t)$, which is independent of x .
- **[A3]:** The constants $\alpha_a = \alpha_c$ (hereafter denoted simply by α). This assumption is almost always true in practice.
- **[A4]:** The term $k_f(x, t) \doteq \left(1 + \frac{d \ln f_{c/a}}{d \ln c_e}(x, t)\right)$ is approximately constant in x , i.e. $\bar{k}_f(t) \approx k_f(x, t)$.
- **[A5]:** The term $\kappa(c_e)$ is approximately constant in c_e , i.e. $\bar{\kappa} \approx \kappa(c_e)$.
- **[A6]:** The total moles of lithium in the electrolyte $n_{Li,e}$ and in the solid phase $n_{Li,s}$ are both conserved.
- **[A7]:** The total moles of lithium in the solid phase, $n_{Li,s}$, is known beforehand for state observer design.
- **[A8]:** The total moles of lithium in the electrolyte, $n_{Li,e}$, is known beforehand for state observer design.

These assumptions ultimately render a model consisting of: (i) two linear spherical diffusion PDEs modeling each electrode's solid concentration dynamics, (ii) a quasilinear diffusion equation (across three domains) modeling the electrolyte concentration dynamics, and (iii) a nonlinear output function mapping boundary values of solid concentration, electrolyte concentration, and current to voltage.

In this section, we aim to relax some of these assumptions in order to improve the model fidelity and state observer performance. The state-dependent parameters are preserved in the model output equation, which removes [A4] – [A5]. Next, the total moles of lithium in the solid-phase and electrolyte-phase are estimated by using sensitivity analysis, which eliminates assumptions [A6] – [A8]. The state-dependent parameters in the electrolyte-phase, i.e. diffusivity, conductivity, and activity coefficients are strongly dependent on the electrolyte concentration. Specifically, they vary by an order of magnitude [123] with c_e . Hence, it is of great importance to incorporate state-dependent electrochemical parameters, which is shown to have a greater impact on voltage prediction under high current.

To begin, applying assumptions [A1]-[A3] removes the spatial dependence of a variable, and thus an overline is added to the variable name to avoid confusion. The first step is to combine [A1] and ODE (2.5) to express molar ion flux as proportional to current,

$$\bar{j}_n^+(t) = -\frac{I(t)}{Fa^+L^+}, \quad \bar{j}_n^-(t) = \frac{I(t)}{Fa^-L^-}. \quad (2.35)$$

Note that the ionic current $i_e(x, t)$ has the trapezoidal shape [79]. Apply $\bar{j}_n^\pm$ in (2.35) to the solid-phase diffusion equations and boundary conditions:

$$\begin{aligned} \frac{\partial \bar{c}_s^\pm}{\partial t}(r, t) &= \frac{1}{r^2} \frac{\partial}{\partial r} \left[D_s^\pm r^2 \frac{\partial \bar{c}_s^\pm}{\partial r}(r, t) \right], \\ \frac{\partial \bar{c}_s^\pm}{\partial r}(0, t) &= 0, \quad \frac{\partial \bar{c}_s^\pm}{\partial r}(R_s^\pm, t) = \pm \frac{1}{D_s^\pm Fa^\pm L^\pm} I(t). \end{aligned} \quad (2.36)$$

Next, we derive the electrolyte diffusion equations by combining PDE (2.2) with (2.5), (2.35), and [A1]:

$$\frac{\partial c_e^-}{\partial t}(x, t) = \frac{\partial}{\partial x} \left[D_e(c_e^-) \frac{\partial c_e^-}{\partial x}(x, t) \right] + \frac{(1 - t_c^0)}{\varepsilon_e^- FL^-} I(t), \quad (2.37)$$

$$\frac{\partial c_e^{\text{sep}}}{\partial t}(x, t) = \frac{\partial}{\partial x} \left[D_e(c_e^{\text{sep}}) \frac{\partial c_e^{\text{sep}}}{\partial x}(x, t) \right], \quad (2.38)$$

$$\frac{\partial c_e^+}{\partial t}(x, t) = \frac{\partial}{\partial x} \left[D_e(c_e^+) \frac{\partial c_e^+}{\partial x}(x, t) \right] - \frac{(1 - t_c^0)}{\varepsilon_e^+ FL^+} I(t). \quad (2.39)$$

Note that the boundary conditions are the same as the full-order model. Next, we derive the nonlinear output function for terminal voltage, which is distinguished from previous work. From (2.27), we notice the voltage $V(t)$ depends on the solid potential at the current collectors $\phi_s^\pm(x, t)$. Therefore, we solve (4.17) in terms of ϕ_s and spatially averaged quantities,

$$\phi_s^\pm(x, t) = \bar{\eta}^\pm(t) + \phi_e^\pm(x, t) + U^\pm(\bar{c}_{ss}^\pm(t)) + FR_f^\pm \bar{j}_n^\pm(t). \quad (2.40)$$

Next we derive each term on the right hand side of (2.40). Overpotential $\bar{\eta}^\pm(t)$ is found by solving the Butler-Volmer equation (2.6) in terms of $\bar{\eta}^\pm(t)$, applying [A1], [A2], [A4], and substituting (2.35),

$$\bar{\eta}^\pm(t) = \frac{RT}{\alpha F} \sinh^{-1} \left(\frac{\mp I(t)}{2a^\pm L^\pm \bar{i}_0^\pm(t)} \right). \quad (2.41)$$

The electrolyte potential $\phi_e^\pm(x, t)$ is found by integrating ODE (2.4) w.r.t. x across the entire cell width,

$$\begin{aligned} \int_{0^-}^{0^+} \frac{\partial \phi_e}{\partial x}(x, t) dx &= \int_{0^-}^{0^+} \frac{-i_e^\pm(x, t)}{\kappa(c_e(x, t))} dx + \int_{0^-}^{0^+} \frac{2RT}{F} (1 - t_c^0) \\ &\times \left(1 + \frac{d \ln f_{c/a}}{d \ln c_e}(x, t) \right) \frac{\partial \ln c_e}{\partial x}(x, t) dx. \end{aligned} \quad (2.42)$$

Note that previous work [79] assumed c_e -dependent parameters such as conductivity, $\kappa(c_e)$ ¹ and activity coefficient, $\frac{d \ln f_{c/a}}{d \ln c_e}(x, t)$ in [A4], [A5] are constant in c_e . With the trapezoidal shape of ionic current, The first term in (2.42) can be expressed as:

$$\begin{aligned} \phi_e^+(0^+, t) - \phi_e^-(0^-, t) &= \frac{-I(t)}{L^-} \int_{0^-}^{L^-} \frac{x}{\kappa(c_e(x, t))} dx - I(t) \int_{L^-}^{L^{sep}} \frac{1}{\kappa(c_e(x, t))} dx \\ &+ \frac{I(t)}{L^+} \int_{L^{sep}}^{0^+} \frac{x - 0^+}{\kappa(c_e(x, t))} dx + \int_{0^-}^{0^+} \frac{2RT}{F} (1 - t_c^0) \\ &\times \left(1 + \frac{d \ln f_{c/a}}{d \ln c_e}(x, t) \right) \frac{\partial \ln c_e}{\partial x}(x, t) dx. \end{aligned} \quad (2.43)$$

Note that the activity coefficient is present in the anode, separator, and cathode respectively. We introduce $\kappa_A(c_e(x, t)) = 1 + \frac{d \ln f_{c/a}}{d \ln c_e}(x, t)$ for simplicity,

$$\begin{aligned} \phi_e^+(0^+, t) - \phi_e^-(0^-, t) &= \frac{-I(t)}{L^-} \int_{0^-}^{L^-} \frac{x}{\kappa(c_e(x, t))} dx - I(t) \int_{L^-}^{L^{sep}} \frac{1}{\kappa(c_e(x, t))} dx \\ &+ \frac{I(t)}{L^+} \int_{L^{sep}}^{0^+} \frac{x - 0^+}{\kappa(c_e(x, t))} dx \\ &+ \frac{2RT}{F} (1 - t_c^0) \left[\int_{0^-}^{L^-} \kappa_A(c_e(x, t)) \frac{\partial \ln c_e}{\partial x}(x, t) dx \right. \\ &\left. + \int_{L^-}^{L^{sep}} \kappa_A(c_e(x, t)) \frac{\partial \ln c_e}{\partial x}(x, t) dx + \int_{L^{sep}}^{L^+} \kappa_A(c_e(x, t)) \frac{\partial \ln c_e}{\partial x}(x, t) dx \right]. \end{aligned} \quad (2.44)$$

Using integration by parts, $\int f(x)g'(x)dx = f(x)g(x) - \int f'(x)g(x)dx$ and chain rule provides the following algebra,

$$\begin{aligned} \phi_e^+(0^+, t) - \phi_e^-(0^-, t) &= \frac{-I(t)}{L^-} \int_{0^-}^{L^-} \frac{x}{\kappa(c_e(x, t))} dx \\ &- I(t) \int_{L^-}^{L^{sep}} \frac{1}{\kappa(c_e(x, t))} dx + \frac{I(t)}{L^+} \int_{L^{sep}}^{0^+} \frac{x - 0^+}{\kappa(c_e(x, t))} dx \end{aligned}$$

¹The superscript {eff} is dropped in the notation for readability

$$\begin{aligned}
& + \frac{2RT}{F}(1 - t_c^0) \left[\kappa_A(c_e(x, t)) \ln c_e(x, t) \right]_{x=0^-}^{x=L^-} \\
& - \int_{c_e(0^-, t)}^{c_e(L^-, t)} \kappa'_A(c_e(x, t)) \ln c_e(x, t) dc_e(x, t) \\
& + \kappa_A(c_e(x, t) \ln c_e(x, t) \Big|_{x=L^-}^{x=L^{sep}} \\
& - \int_{c_e(L^-, t)}^{c_e(L^{sep}, t)} \kappa'_A(c_e(x, t)) \ln c_e(x, t) dc_e(x, t) \\
& + \kappa_A(c_e(x, t)) \ln c_e(x, t) \Big|_{x=L^{sep}}^{x=L^+} \\
& - \int_{c_e(L^{sep}, t)}^{c_e(L^+, t)} \kappa'_A(c_e(x, t)) \ln c_e(x, t) dc_e(x, t) \Big]. \tag{2.45}
\end{aligned}$$

We simply denote the electrolyte potential difference (2.45) by

$$\phi_e^+(0^+, t) - \phi_e^-(0^-, t) = \Delta\Phi_e(c_e(x, t), \kappa(c_e(x, t)), \kappa_A(c_e(x, t)), I(t)).$$

The voltage output for the SPMe model is written as:

$$\begin{aligned}
V(t) = & \frac{RT}{\alpha F} \sinh^{-1} \left(\frac{-I(t)}{2a^+ L^+ \bar{i}_0^+(t)} \right) \\
& - \frac{RT}{\alpha F} \sinh^{-1} \left(\frac{I(t)}{2a^- L^- \bar{i}_0^-(t)} \right) \\
& + U^+(\bar{c}_{ss}^+(t)) - U^-(\bar{c}_{ss}^-(t)) - \left(\frac{R_f^+}{a^+ L^+} + \frac{R_f^-}{a^- L^-} \right) I(t) \\
& + \Delta\Phi_e(c_e(x, t), \kappa(c_e(x, t)), \kappa_A(c_e(x, t)), I(t)). \tag{2.46}
\end{aligned}$$

The terminal voltage output equation can be described as a combination of electrochemical differences between two electrodes and the electrolyte,

$$V(t) = \Delta U(\bar{c}_{ss}^\pm) + \Delta\phi_e(c_e, I) + \Delta\bar{\eta}(\bar{c}_{ss}^\pm, I) + \Delta\bar{j}_n(I). \tag{2.47}$$

	0.5C	1C	2C
Proposed SPMe	16.1 mV	29.6 mV	46.5 mV
SPMe [79]	19.4 mV	40.3 mV	87.6 mV

Table 2.2: The output voltage error with respect to DFN model.

We validate the proposed SPMe model with the previous work [79] by comparing the voltage prediction accuracy. A 18650 Lithium nickel-cobalt-aluminum oxide (NCA) battery

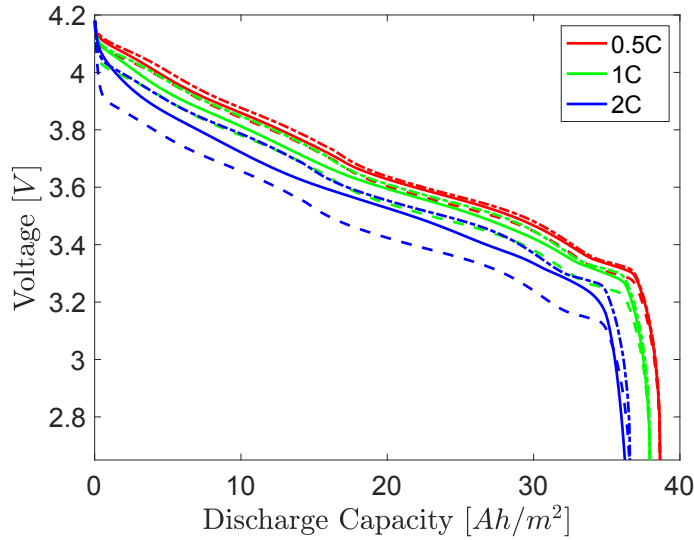


Figure 2.4: The comparison of proposed SPMe(-) with DFN(-) and previous SPMe(- -) model [79] for NCA electrochemical parameters identified in [86].

parameters identified previous work [86] are adopted to simulate the electrochemical models. Suppose the DFN model represents a truth battery model. We visualize the voltage predictions with different C-rates in Fig 2.4. The proposed model (-) outperforms the previous work (- -) compared to the truth model (-) in terms of voltage. The quantitative analysis for the voltage comparison is summarized in Table 2.2. The voltage root mean square error (RMSE) is computed up to 80 % of discharge capacity. We emphasize the inclusion of the state-dependent electrolyte parameters are critical to the model-order-reduction, as they are shown in the output voltage equation (2.46) directly. Furthermore, the voltage output is strongly affected by the electrolyte characteristics of the cell. For instance, the conventional SPMe model has lower voltage prediction error for the Lithium cobalt oxide (LiCoO_2) chemistry in [79]. However, the prediction gets worse for the NCA chemistry in [86].

2.2.3 Single Particle Model with Electrolyte and Thermal dynamics (SPMeT)

Lastly, the thermal dynamics of the battery can be added to the SPMe model, which is called SPMeT. This model takes temperature measurement into account, which directly affects to the electrochemical parameters governed by Arrhenius equation.

$$\psi = \psi_{\text{ref}} \exp \left[\frac{E_\psi}{R} \left(\frac{1}{T_1} - \frac{1}{T_{\text{ref}}} \right) \right], \quad (2.48)$$

where ψ represents a temperature dependent parameter, i.e., $\psi = [D_s^\pm, D_e, k^\pm, \kappa]$. E_ψ is the activation energy [J mol^{-1}], and ψ_{ref} is the reference parameter value at reference temperature

T_{ref} .

The nonlinear temperature dynamics are modeled with a simple heat transfer equation given by:

$$\frac{dT_{\text{cell}}}{dt}(t) = \frac{\dot{Q}(t)}{mc_{p,th}} - \frac{T_{\text{cell}}(t) - T_{\infty}}{mc_{p,th}R_{th}} \quad (2.49)$$

where T_{cell} represents cell temperature, T_{∞} is the ambient temperature, m is the mass of the cell, $c_{p,th}$ is the thermal specific heat capacity of the cell, R_{th} is the thermal resistance of the cell, and $\dot{Q}(t)$ is the heat added from the charging, which is governed by

$$\dot{Q}(t) = I(t)((U^+(SOC_p) - U^-(SOC_n)) - V_T(t)) \quad (2.50)$$

with the convention that a negative current is charging current, and $V_T(t)$ is the terminal voltage determined by model equations. Both nonlinear open circuit potential functions in (2.50) are functions of the bulk SOC in the anode and cathode, respectively. This heat generation term makes the temperature dynamics nonlinear. This thermal dynamics can be incorporated to any electrochemical model resulting in time-varying dynamical system.

2.3 Hybrid Electrochemical Modeling

Modeling battery system is an active research area as reader can see from previous section. We are particularly interested in combining first principle and data-driven modeling technique to capture the un-modeled dynamics by using input-output measurement in this section.

2.3.1 Introduction

Battery models for BMS applications can be categorized into two groups: equivalent circuit models (ECMs), and electrochemical models. ECMs have relatively simple structures to represent the input-output behavior of batteries using circuit elements, such as resistors and capacitors. However, ECMs do not directly capture the physical phenomena inside the battery, such as lithium transport, solid-electrolyte interphase dynamics, and degradation mechanisms. Electrochemical models directly incorporate diffusion, intercalation, and electrochemical kinetics. Although these models can accurately explain the internal behavior of the battery, their mathematical structure is relatively complex for observer designs, such as state of charge (SoC) and state of health (SoH) estimation. Consequently, reduced order models are generally selected for state estimator design. There are several widely studied reduced order electrochemical models in this field. The simplest is single particle model (SPM), where electrolyte concentration is assumed constant in space and time [103]. Papers by Santhanagopalan et al. [101] and Di Domenico et al. [30] study SPM-based Kalman filters for SOC estimation. More recently, [29] studies nonlinear observers for the SPM. The main limitation of the SPM is that it neglects electrolyte dynamics, which play important role at

high charge/discharge C-rates². For this reason, researchers have recently considered a SPM with electrolyte dynamics (SPMe) [96]. A provably convergent partial differential equation (PDE)-based state estimation technique based on the SPM can be found in [75]. Tanim et al. [117] augment the SPM with Arrhenius relations for the model parameters to account for temperature variation, abbreviated as SPM_eT. A Luenberger state observer based on this SPM_eT model is derived in [118].

In parallel, computational learning theory in artificial intelligence has been rapidly applied to real-world engineering applications, such as self-driving cars, pattern recognition, and robotics. Artificial neural networks are a fundamental technique in machine learning for modeling an environment from data. In this paper, we focus on *dynamic* neural networks, which were first proposed for dynamic system identification in the early 1990's [80]. Dynamic neural networks, a.k.a. recurrent neural networks (RNNs), can capture dynamical behavior by learning from input-output data. In battery applications, some published works have studied RNN models. The authors of [66] use an adaptive RNN to predict remaining useful life of lithium-ion batteries. They use the Levenberg-Marquardt algorithm to optimize the network parameters, then estimate the electrolyte resistance and charge transfer resistance in the ECM to infer power fade. In [21], the ECM is converted into an RNN model that captures SoC information and model parameters. The authors conclude that the RNN can accurately describe the charge-discharge phenomena of a lead-acid battery. Similarly, a RNN-based SoH indicator for lithium-ion batteries is proposed in [31]. They also predict variations in capacity and resistance of an ECM reformulated into a RNN. This literature suggests that RNNs can play a useful role in predicting battery dynamics, especially aging mechanisms. However, this set of literature is completely based on the ECM. Moreover, it lacks a theoretical stability analysis. This motivates us to explore the role of RNNs in combination with electrochemical models, and study stability for various classes of RNNs.

In this chapter, we propose a hybrid battery model that integrates the SPM and RNN. Note, the SPM is valid for low C-rates, but produces erroneous voltage predictions at high C-rates where electrolyte dynamics play an increased role [78]. This paper focuses on a simple RNN, known as the Elman network [32], which originally emerged in cognitive science. Contrary to the conventional electrochemical modeling approach of extending models based on first principles, we augment the SPM with Elman networks to improve voltage prediction accuracy. To the best of the authors' knowledge, the combination of electrochemical and data-driven models have not been previously explored. We summarize our key contributions as follows. First, we propose a hybrid modeling framework that incorporates electrochemical and RNN models – namely the SPM and Elman networks. We consider the real-time recurrent learning (RTRL) algorithm for learning Elman network weights online. Second, we suggest a special case of Elman networks that guarantees stability in the sense of Lyapunov. Third,

²C-rate is a normalized measure of electric current that enables comparisons between different sized batteries. Mathematically, the C-rate is defined as the ratio of current, I , in Amperes [A] to a cell's nominal capacity, in Ampere-hours [Ah]. For example, if a battery has a nominal capacity of 2.5 Ah, then C-rates of 2C, 1C, and C/2 correspond to 5 A, 2.5 A, and 1.25 A, respectively.

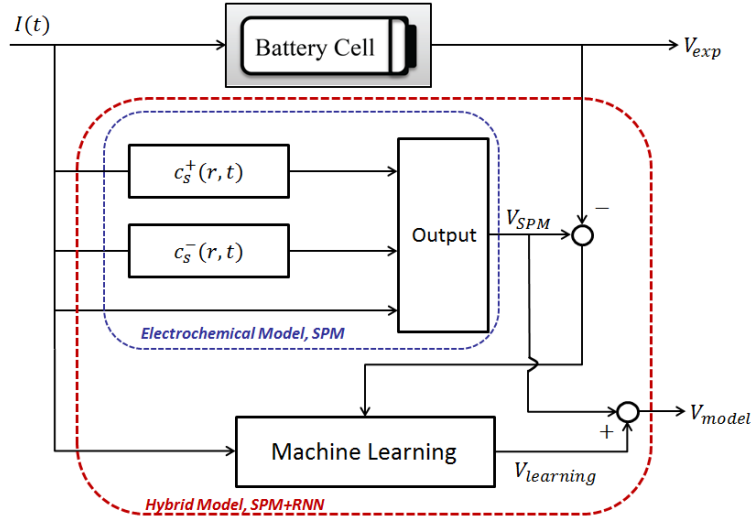


Figure 2.5: Proposed framework for hybrid modeling approach based on electrochemical model.

we demonstrate that our hybrid approach often outperforms other reduced electrochemical models that do not apply learning structures.

2.3.2 Neural Network Structure and Training Algorithm

In this section we describe the hybrid modeling framework, which applies a machine learning algorithm on top of the electrochemical model as depicted in Fig. 2.5. We augment the first principals model with an input-output learning model to enhance voltage prediction accuracy.

Consider the Elman network structure in Fig. 2.6. In general, there are n_I inputs, n_H hidden units, and n_O outputs in this network. The structure consists of output layer weight matrix $\mathbb{W}_3 \in \mathbf{R}^{n_O \times n_H}$, input layer-to-hidden layer weight matrix $\mathbb{W}_2 \in \mathbf{R}^{n_H \times n_I}$, and hidden layer-to-context layer weight matrix $\mathbb{W}_1 \in \mathbf{R}^{n_H \times n_H}$. Since we have few measurements in battery systems, this Elman network structure can be reduced to a single-input / single-output network, i.e., input current and output voltage. The network dynamic equations can be written as follows:

$$x(k) = f(\mathbb{W}_1 \cdot x(k-1) + \mathbb{W}_2 \cdot u(k)), \quad (2.51)$$

$$y(k) = g(\mathbb{W}_3 \cdot x(k)). \quad (2.52)$$

where $x(k)$ is called the “hidden layer” output vector, $y(k)$ is the output from the “output layer”, and $u(k)$ represents the input. Functions $f(\cdot)$ and $g(\cdot)$ are “activation functions” for each layer. We take

$$f(z) = \tanh(z), \quad (2.53)$$

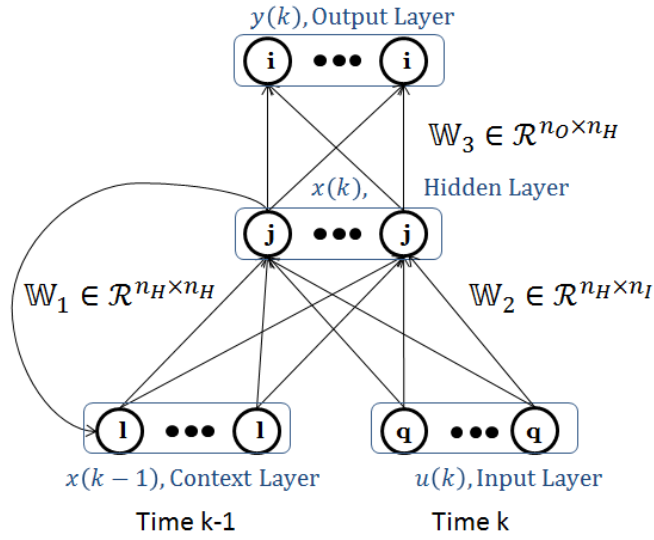


Figure 2.6: Elman network structure.

for reasons that emerge in the stability analysis. The function $g(\cdot)$ is often taken as the identity function, that is

$$y(k) = \mathbb{W}_3 \cdot x(k). \quad (2.54)$$

Note that (2.51)-(2.52) is a nonlinear system, with order given by user-selected parameter n_H .

The weight matrices, $\mathbb{W}_1$, $\mathbb{W}_2$, and $\mathbb{W}_3$ are updated to train the network. There are several methods for training RNNs, such as backpropagation through time, real-time recurrent learning (RTRL), and the extended Kalman filter [47]. In this work, we use the RTRL scheme which is a gradient-descent online learning algorithm that calculates the error gradient at every time step. The instantaneous squared error $E(k)$ at time step k is computed as:

$$E(k) = \frac{1}{2}e(k)^2 = \frac{1}{2}(y_d(k) - y(k))^2, \quad (2.55)$$

where $e(k)$ is the prediction error between desired output $y_d(k)$ and network output $y(k)$ at time step k . In RTRL, we update the weight matrix $\mathbb{W}$ via gradient-descent to minimize the squared error $E(k)$ such as:

$$\begin{aligned} W_{3,ij}(k+1) &= W_{3,ij}(k) - \eta_3 \cdot \frac{\partial E(k)}{\partial W_{3,ij}(k)} \\ &= W_{3,ij}(k) + \eta_3 \cdot e(k) \cdot x_j(k), \end{aligned} \quad (2.56)$$

$$\begin{aligned} W_{2,jq}(k+1) &= W_{2,jq}(k) - \eta_2 \cdot \frac{\partial E(k)}{\partial W_{2,jq}(k)} \\ &= W_{2,jq}(k) + \eta_2 \cdot e(k) \cdot W_{3,ij} \cdot \gamma_j(k) \cdot u(k), \end{aligned} \quad (2.57)$$

$$W_{1,jl}(k+1) = W_{1,jl}(k) - \eta_1 \cdot \frac{\partial E(k)}{\partial W_{1,jl}(k)} \quad (2.58)$$

$$= W_{1,jl}(k) + \eta_1 \cdot e(k) \cdot W_{3,ij} \cdot \gamma_j(k) \cdot x_j(k-1), \quad (2.59)$$

where $\eta_1, \eta_2, \eta_3 \in \mathbb{R}$ are user-selected learning rates for each weight, and i, j, q, l represent the unit index in each layer. Parameter γ_j is the derivative of activation function:

$$\gamma_j(k) = f'_j(W_{1,jl} \cdot x(k) + W_{2,jq} \cdot u(k)).$$

Compared to typical feedforward neural networks that build approximators of nonlinear functions by using static input-output mappings [44], the Elman network is able to approximate arbitrary dynamical systems with arbitrary precision according to the *universal approximation property* [25]. We consider online training of (2.51)-(2.52), i.e. recursive estimation of $\mathbb{W}_1, \mathbb{W}_2, \mathbb{W}_3$, to capture the error between measured voltage and SPM-predicted voltage.

Special Case of Elman Network

A long-standing and un-solved challenge is proving the convergence of weights $\mathbb{W}_1, \mathbb{W}_2, \mathbb{W}_3$, or even convergence of $e(k)$ to zero via online learning [110]. The key issue is nonlinear term $\mathbb{W}_3 x(k)$. In the neural network community, a number of papers have discussed this issue. For instance, in [48], the stability of the dynamic backpropagation algorithm is thoroughly discussed based on a dynamic learning process and RNN suggested by Narendra [80]. The authors propose two learning schemes: a multiplier method and a constrained learning rate algorithm to avoid unstable phenomenon during the learning process. The author in [110] proves that the Elman network's weights converge in the sense of Lyapunov functions under a backpropagation algorithm with adaptive dead zones. In this paper, we examine a special case of Elman networks for which weight convergence can be proven with RTRL update laws.

2.3.3 Stability Analysis

This section analyzes stability for the proposed learning structure. First we derive the weight error dynamics,

$$\tilde{\mathbb{W}}(k) = \mathbb{W}(k) - \mathbb{W}^*, \quad (2.60)$$

where $\mathbb{W}^*$ is the “true” weight matrix that represents the desired output given input data. The output error at time k is given by:

$$\begin{aligned} e(k) &= y_d(k) - y(k) \\ &= \bar{\mathbb{W}}_3 f(\mathbb{W}_1^* x(k-1) + \mathbb{W}_2^* u(k)) \\ &\quad - \bar{\mathbb{W}}_3 f(\mathbb{W}_1(k) x(k-1) + \mathbb{W}_2(k) u(k)) \\ &= -\alpha \bar{\mathbb{W}}_3 \left(\tilde{\mathbb{W}}_1(k) x(k-1) + \tilde{\mathbb{W}}_2(k) u(k) \right), \end{aligned} \quad (2.61)$$

where the last equality is derived by the Mean Value Theorem, which requires function f to be continuous on a closed interval and differentiable on an open interval, which activation function $f(z) = \tanh(z)$ satisfies. Therefore $\exists z \in \mathbb{R}$ such that $\alpha = f'(z)$ and $0 < \alpha \leq 1$. Then, we can re-write (2.57)-(2.58) as follows:

$$\begin{aligned}\mathbb{W}_1(k+1) &= \mathbb{W}_1(k) + \eta_1(k)e(k)\gamma(k)x(k-1)^T \\ &= \mathbb{W}_1(k) - \eta_1(k)\alpha\bar{\mathbb{W}}_3\left(\tilde{\mathbb{W}}_1(k) \cdot x(k-1) + \tilde{\mathbb{W}}_2(k) \cdot u(k)\right)\gamma(k)x(k-1)^T, \quad (2.62)\end{aligned}$$

$$\begin{aligned}\mathbb{W}_2(k+1) &= \mathbb{W}_2(k) + \eta_2(k)e(k)\gamma(k)u(k) \\ &= \mathbb{W}_2(k) - \eta_2(k)\alpha\bar{\mathbb{W}}_3\left(\tilde{\mathbb{W}}_1(k) \cdot x(k-1) + \tilde{\mathbb{W}}_2(k) \cdot u(k)\right)\gamma(k)u(k). \quad (2.63)\end{aligned}$$

Assume the true weight matrices $\mathbb{W}_1^*, \mathbb{W}_2^*$ are constant, and $\eta_1(k) = \eta_2(k) = \eta(k)$. Then the weight error dynamics are:

$$\tilde{\mathbb{W}}_1(k+1) = \tilde{\mathbb{W}}_1(k) - \eta(k)\alpha\bar{\mathbb{W}}_3\left(\tilde{\mathbb{W}}_1(k)x(k-1) + \tilde{\mathbb{W}}_2(k)u(k)\right)\gamma(k)x(k-1)^T, \quad (2.64)$$

$$\tilde{\mathbb{W}}_2(k+1) = \tilde{\mathbb{W}}_2(k) - \eta(k)\alpha\bar{\mathbb{W}}_3\left(\tilde{\mathbb{W}}_1(k)x(k-1) + \tilde{\mathbb{W}}_2(k)u(k)\right)\gamma(k)u(k). \quad (2.65)$$

The following theorem guarantees the convergence of $\mathbb{W}_1, \mathbb{W}_2$ to their true values under update laws (2.62)-(2.63).

Theorem 1. The zero equilibrium of weight error dynamics (2.64), (2.65) are stable in the sense of Lyapunov functional, $V(k) = \|\tilde{\mathbb{W}}_1(k)\|^2 + \|\tilde{\mathbb{W}}_2(k)\|^2$ if

$$\eta(k) \leq 2 \left[\alpha \|x(k-1)\|^2 + u(k)^2 \bar{\mathbb{W}}_3^T \gamma(k)^T \right]^{-1}. \quad (2.66)$$

Proof. Lyapunov's indirect theorem requires that the difference in the Lyapunov functional denoted by ΔV is negative-definite along the trajectories $\forall \tilde{\mathbb{W}}_1, \tilde{\mathbb{W}}_2$, i.e.:

$$\|\tilde{\mathbb{W}}_1(k+1)\|^2 - \|\tilde{\mathbb{W}}_1(k)\|^2 + \|\tilde{\mathbb{W}}_2(k+1)\|^2 - \|\tilde{\mathbb{W}}_2(k)\|^2 \leq 0.$$

Squaring both sides equations (2.62) and (2.63), we can construct ΔV as:

$$\begin{aligned}\Delta V &= -2\eta(k)\alpha\bar{\mathbb{W}}_3\left(\|\tilde{\mathbb{W}}_1\|^2 x(k-1) + \tilde{\mathbb{W}}_1^T \tilde{\mathbb{W}}_2 u(k)\right) \times \gamma(k)x(k-1)^T \\ &\quad + \left\| \eta(k)\alpha\bar{\mathbb{W}}_3\left(\tilde{\mathbb{W}}_1 x(k-1) + \tilde{\mathbb{W}}_2 u(k)\right)\gamma(k)x(k-1)^T \right\|^2 \\ &\quad - 2\eta(k)\alpha\bar{\mathbb{W}}_3\left(\tilde{\mathbb{W}}_2^T \tilde{\mathbb{W}}_1 x(k-1) + \|\tilde{\mathbb{W}}_2\| u(k)\right)\gamma(k)u(k) \\ &\quad + \left\| \eta(k)\alpha\bar{\mathbb{W}}_3\left(\tilde{\mathbb{W}}_1 x(k-1) + \tilde{\mathbb{W}}_2 u(k)\right)\gamma(k)u(k) \right\|^2. \quad (2.67)\end{aligned}$$

After a few steps for re-arranging ΔV , we can finally obtain

$$\begin{aligned} \Delta V &\leq -2\eta(k)\alpha\bar{\mathbb{W}}_3\left\|\left(\tilde{\mathbb{W}}_1x(k-1) + \tilde{\mathbb{W}}_2u(k)\right)\right\|^2\gamma(k) \\ &\quad + \eta(k)^2\alpha^2\left\|\left(\tilde{\mathbb{W}}_1x(k-1) + \tilde{\mathbb{W}}_2u(k)\right)\right\|^2 \times \left\|x(k-1)^T + u(k)\right\|^2\|\bar{\mathbb{W}}_3\|^2\|\gamma(k)\|^2 \\ &\leq 0, \end{aligned} \tag{2.68}$$

where the last inequality is derived from the definition of the adaptive learning rate in (2.66). $\square$

Remark 1. The adaptive learning rate, $\eta(k)$ defined in (2.66) can be re-written in terms of the previous desired output $y_d(k-1)$, since we adopt specialized Elman network structure expressed by $y_d(k-1) = \bar{\mathbb{W}}_3 \cdot x(k-1)$. That is:

$$\eta(k) \leq 2 \left[\alpha \left\| \frac{1}{n_H} \mathbf{1}_{n_H} y_d(k-1) + u(k) \right\|^2 \bar{\mathbb{W}}_3^T \gamma(k)^T \right]^{-1}, \tag{2.69}$$

where $\mathbf{1}_{n_H}$ is a $n_H \times 1$ vector of ones. Assuming a constant weight matrix $\bar{\mathbb{W}}_3$ plays the critical role for calculating the adaptive learning rates from the previous output data $y_d(k-1)$ and current input data $u(k)$.

Remark 2. Since the activation function is strictly monotonically increasing and has bounded derivative, we can state that when $\alpha = 1$, the following condition

$$\eta(k) \leq 2 \left[\left\| \frac{1}{n_H} \mathbf{1}_{n_H} y_d(k-1) + u(k) \right\|^2 \bar{\mathbb{W}}_3^T \gamma(k)^T \right]^{-1}, \tag{2.70}$$

becomes sufficient condition for (2.69) as $\alpha \in (0, 1]$. Practically, we calculate the learning rates based on (2.70) with equality as long as the computed value does not exceed 1, since $0 < \eta(k) \leq 1$ is used to regularize the learning rate.

2.3.4 Simulation Results

This section presents simulations that demonstrate the hybrid approach performance compared to several electrochemical models. The measured voltage data is generated from the DFN model, which plays the role of “true” voltage. Consequently, the model voltage error arises from model reduction. We notate Elman 1 as the special case of the Elman network and Elman 2 as the original Elman network. We use 4 hidden units for the Elman networks. Elman 1 uses adaptive learning parameter $\eta(k)$ given by (2.70), and Elman 2 uses fixed learning parameter $\eta = 0.3$ in the simulation. We train the RNN to identify an uncertainty model with various input profiles such as 1C/2C/5C discharge cycles and dynamic current profiles. This enables us to investigate how the identified RNN captures the uncertainty dynamics. In all cases, we require 10 epochs or less for the output error to converge. To evaluate the resulting learned RNN, our primary performance metric is the root mean square error (RMSE) between the DFN voltage and various model voltages.

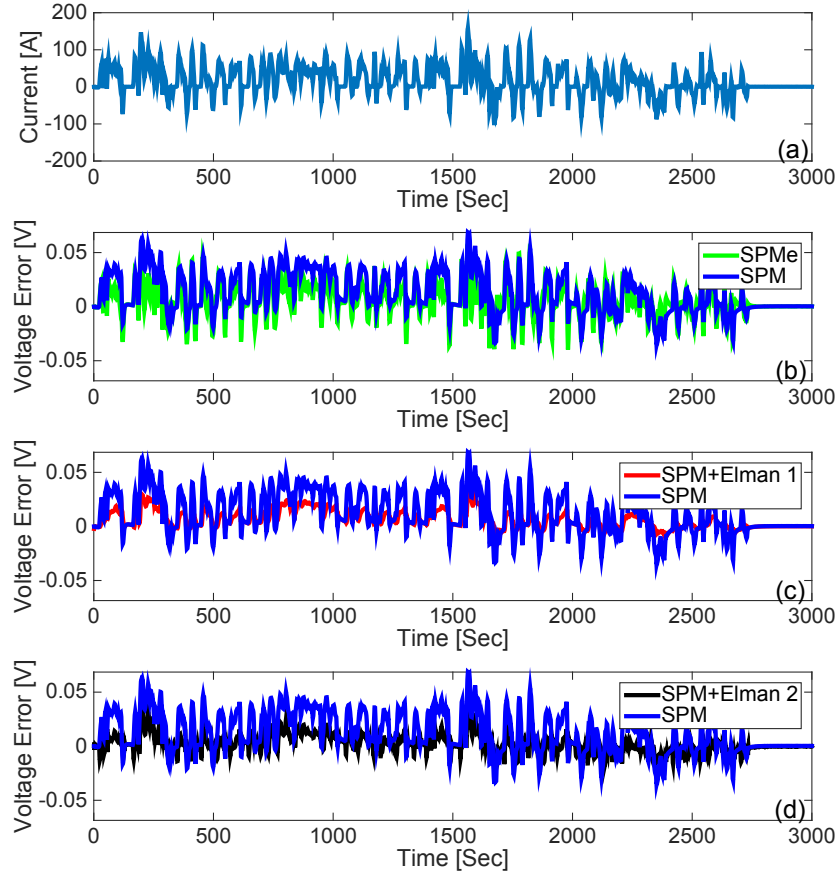


Figure 2.7: Voltage error comparison for UDDSx2 cycle: (a) Input current. (b) SPMe. (c) The special case of Elman network (d) The original Elman network.

Training Phase

The numerical training results are displayed in Table 2.3, for various input profiles and models that do and do not incorporate learning structures. Indeed, our proposed hybrid approach resolves the limitations of the SPM. To explore this, we also consider the SPMe from [75] as a baseline comparison to quantify the error between pure electrochemical models and hybrid electrochemical models. As shown in Table 2.3, the SPMe – which includes electrolyte dynamics – has about 50 % lower error than SPM in all scenarios. The hybrid modeling approach exhibits even better performance than SPMe in many cases. Interestingly, the SPM+Elman 2 consistently outperforms its simpler SPM+Elman 1 counterpart. This is not surprising, as SPM+Elman 2 has more parameters in W_3 that it may tune to reduce output error. However, as discussed before, analytic weight error convergence guarantees are not available.

Consider the Urban Dynamometer Driving Schedule (UDDS), which is a standard fuel economy test cycle. The input current data in Fig. 2.7(a) is generated from two concatenated

Input Profiles (Training Data)	SPM (60) (No learning)	SPMe (90) (No learning)	SPM+Elman 1 (64) (Learning)	SPM+Elman 2 (64) (Learning)
1C discharge	20.0mV	10.0mV	1.8mV	1.8mV
2C discharge	39.4mV	19.0mV	3.0mV	5.2mV
5C discharge	107.2mV	54.6mV	67.6mV	38.7mV
Sine wave [†]	67.6mV	31.0mV	34.7mV	29.3mV
UDDS	20.8mV	14.7mV	10.1mV	7.6mV

[†]min:1C, max:5C

Table 2.3: RMSE comparison for different models (w/ and w/o learning) and input profiles in the training phase. The number in parentheses is the number of states in the model.

Input Profiles (Testing Data)	SPM (60) (No learning)	SPMe (90) (No learning)	SPM+Elman 1 (64) (Learning)	SPM+Elman 2 (64) (Learning)
1C discharge	20.0mV	10.0mV	10.5mV	9.2mV
2C discharge	39.4mV	19.0mV	20.9mV	18.5mV
5C discharge	107.2mV	54.6mV	73.8mV	66.5mV
Sine wave [†]	67.6mV	31.0mV	42.7mV	37.9mV

[†]min:1C, max:5C

Table 2.4: RMSE comparison for different models (w/ and w/o learning) and input profiles in the testing phase, using RNNs trained on UDDS. The number in parentheses is the number of states in the model.

UDDS cycles. Fig. 2.7(b) shows the voltage error for the SPM and SPM relative to DFN, while Fig. 2.7(c),(d) presents the voltage errors for the proposed hybrid modeling scheme and SPM only, relative to the DFN. Note that we use the fixed weights after learning to verify the training performance. In addition, the cumulative distribution function (CDF) of the absolute voltage error in UDDS is computed for measuring model output precision, as shown in Fig. 2.8. We comment that applying adaptive learning rate $\eta(k)$ from (2.70) to the Elman 2 model yields divergent weights for UDDS. This corroborates our analysis that suggests a condition on $\eta(k)$ is not available to guarantee convergent weights for Elman 2, however such a condition is derived for Elman 1. This is a fundamental tradeoff balanced by the increased error reduction potential of the Elman 2 model.

Testing Phase

In this phase, we use the fixed weights obtained from the training phase to test the accuracy of hybrid electrochemical model. Here, we simply illustrate the robustness of the hybrid modeling framework. Specifically, we select the RNN models trained on UDDS data and test the performance with different input profiles. The UDDS data is selected for training, since it contains the richest frequency content. The numerical results in Table 2.4 illustrate two important observations: (i) The hybrid models do not degrade the voltage prediction accuracy compared to a stand-alone SPM model. (ii) The hybrid models yield comparable

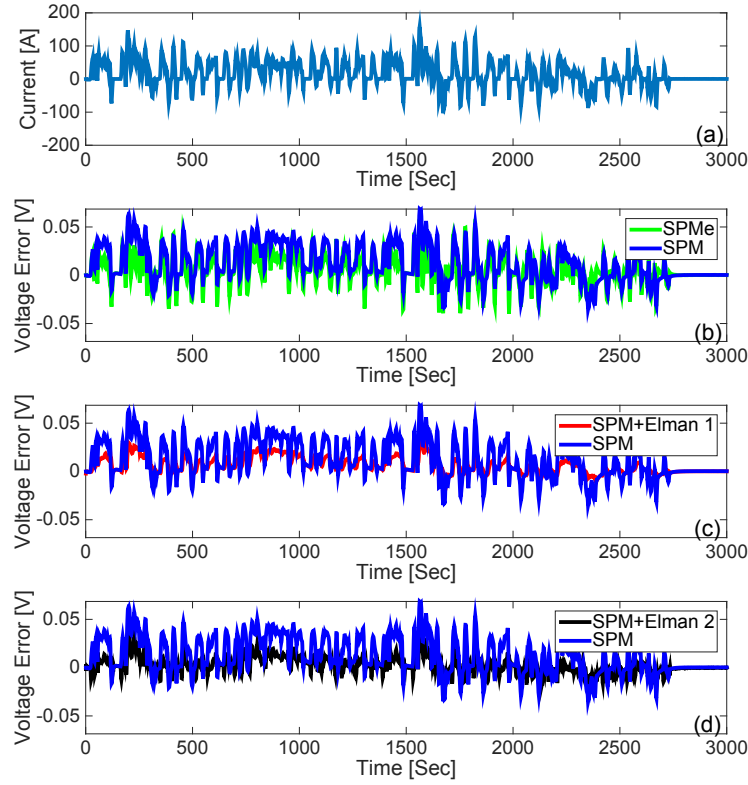


Figure 2.8: Cumulative distribution function of absolute voltage error on UDDS profile, with 90% quantile errors annotated.

accuracy to the SPM_e.

2.3.5 Summary

In this chapter, a hybrid electrochemical modeling framework that integrates recurrent neural networks (RNNs) with a reduced electrochemical model is proposed. Contrary to conventional approaches of integrating more physics into the electrochemical models, we examine the potential of data-driven models to recover model-measurement mismatch. Specifically, we study a simple RNN, called the Elman network, which mimics dynamical phenomena from observed data. Additionally, we prove convergence of estimated weights to their true values for a special case of the Elman network. This stability analysis yields an adaptive learning rate for the real-time recurrent learning (RTRL) update law. Simulation results demonstrate that our hybrid battery modeling significantly improves the predicted voltage accuracy compared to other reduced electrochemical models. On-going work involves validating our proposed hybrid approach experimentally with different types of Li-ion batteries.

Chapter 3

Estimation

In this chapter, we present parameter estimation framework for electrochemical models. Proposed parameter estimation method, a.k.a system identification is crucial aspect of battery management system in order to derive model-based control strategy. In Section 3.1, the system identification framework for lithium-ion battery cell is presented via optimal experimental design. In Section 3.2, the electrochemical parameters that determine battery health are estimated.

3.1 Characterization of Electrochemical Model

3.1.1 Introduction

Accurate electrochemical battery models are critical for a variety of tasks, such as designing high-performance battery management systems, battery pack design, and analysis. However, identifying parameters in electrochemical battery models from measured voltage, current, and temperature data is notoriously difficult. Recently, non-invasive parameter identification of electrochemical models has become an emergent research topic. Schmidt *et al.* [106] conducted combined parameter analysis and identification by using a Fisher information matrix approach in combination with sensitivity analysis. They use a reduced electrochemical model – a single particle model with electrolyte potential. Recently, Bizeray *et al.* [13] analyze parameter sensitivity in a single particle model and show it is fully parameterized with six subgroups, under certain mild conditions on the electrode potentials. In contrast, Forman *et al.* [36] use an “all-in-one” approach to identifying 88 parameters from driving cycle data using a genetic algorithm and the Doyle-Fuller-Newman model. They validate the identified parameter values with experiments and perform Fisher information analysis *ex post facto*. Similarly, authors in Zhang *et al.* [134] use a multi-objective genetic algorithm called NSGA-II for a LiFePo₄ cell. They use terminal voltage and surface temperature as identification objectives. Recently, both Jobman *et al.* [50] and Jin *et al.* [49] propose a two-step procedure that sequences parameter identification. The thermodynamic parameters

are identified first, and then the kinetic parameters are identified second via pulses and/or electrochemical impedance spectroscopy (EIS). In the published literature, we can categorize parameter identification process: 1) Battery models 2) Optimization techniques 3) Algorithms 4) Experimental setup as shown in Fig. 3.1.

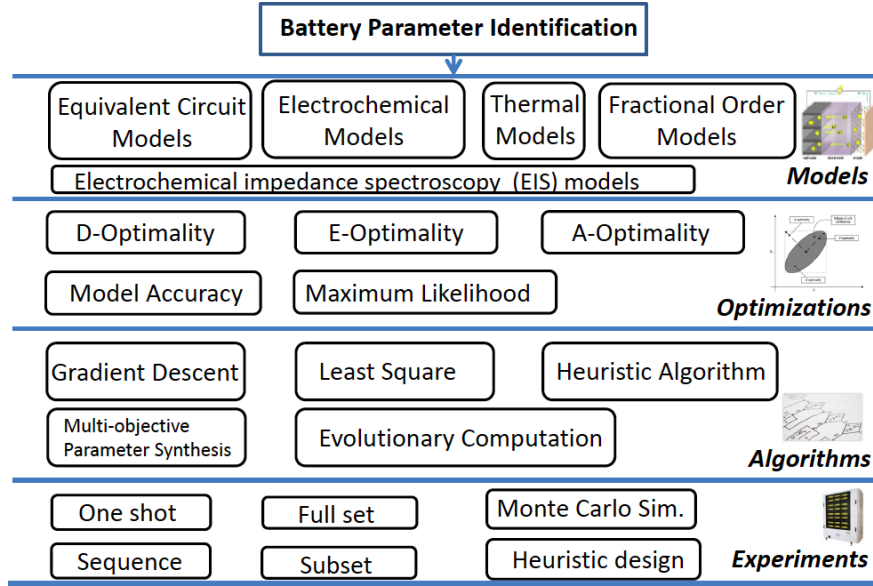


Figure 3.1: Battery parameter identification procedure in the literature

Rather than identify all the parameters, some researchers focus on identifying specific subsets of parameters, such as battery health-related or kinetic parameters. For example, the physical parameters, such as diffusion coefficients and activation energies are targeted in [134]. In Marcicki *et al.* [72], the authors focus on the electrochemical parameters related to power and capacity fade, as well as their temperature dependence under a variety of charge sustaining and depleting experiments. In Vazquez-Arenas *et al.* [125], the researchers analyze sensitivity of certain key parameters using Analysis of Variations (ANOVA), and identify the kinetic and transport parameters with standard test cycles.

Most existing literature on battery parameter identification focuses on parameter fitting, namely, matching model output to experimental data. However, it is unclear if the experimental data is “sufficiently rich” to identify the parameters. A small set of publications in the battery parameter identification literature directly address this problem by formulating an input trajectory optimization problem [100, 67]. This work optimizes the amplitude and frequency of a sinusoidal input signal to maximize the Fisher information matrix, for an equivalent circuit model and single particle model, respectively. One could exploit a series of inputs that excite specific parameter sensitivity, however, collecting the required data from experiments can be cost and time intensive. This motivates the following question: Which inputs should be applied to maximize parameter identifiability in a systematic way?



Figure 3.2: NCR 18650PF Li-ion battery used for parameter edentification.

In addition, the estimated parameters should be characterized by confidence intervals. These questions motivate *optimal experimental design* (OED), which provides an important link between experimental design and modeling [38]. In this paper, we propose an electrochemical model-based optimal experiment design framework that yields parameter estimates with statistical information. Instead of formulating a nonlinear trajectory optimization problem, we propose a convex input selection problem. We summarize our unique key contributions as follows.

1. A sensitivity analysis for the full-order electrochemical model, as known as Doyle-Fuller-Newman (DFN) model, is executed by *analytically* deriving the sensitivity differential algebraic equations. To the best of the authors' knowledge, an *analytic sensitivity analysis* for all the electrochemical parameters has never been executed before. We place emphasis on this sensitivity analysis, as it plays a crucial role in computing the Fisher information matrix, and the parameter estimation algorithm.
2. We formulate an *optimal experimental design via convex optimization*. Rather than solving a large-scale nonlinear optimal control problem, we propose an input selection problem where the optimal inputs are selected from a (large) discrete set. Notably, we quantify the experimental output measurement variance for our battery tester to trade off Fisher information with measurement variance.
3. We provide the *confidence intervals* along with the parameter estimates via nonlinear least squares with the Levenberg-Marquardt algorithm [70].

3.1.2 Electrochemical Parameters

The electrochemistry-based battery model combined with two state thermal dynamics in Sec. 2.1.1 is capable of high-fidelity simulations. However, it has a large number of parameters that must be identified for experimental data. Next, we distinguish fixed parameters from those we seek to identify from data. We classify parameters into two categories: fixed geometric / thermal parameters in Table 3.1 and electrochemical parameters listed in Table 3.2.

Geometric parameters	
Symbol	Description [SI units]
L^-	Thickness of negative electrode [m]
L^{sep}	Thickness of separator [m]
L^+	Thickness of positive electrode [m]
A	Electrode current collector area [m ²]
Thermal parameters	
Symbol	Description [SI units]
C_1	Heat capacity of battery core [J (m ² K) ⁻¹]
C_2	Heat capacity of battery surface [J (m ² K) ⁻¹]
h_{12}	Heat transfer coefficients from core to surface [W (m ² K) ⁻¹]
h_{2a}	Heat transfer coefficients from surface to ambient [W (m ² K) ⁻¹]
$E_{\{D_s^\pm, D_e, \kappa, k^\pm\}}$	Activation energy for Arrhenius temperature dependence [kJ mol ⁻¹]

Table 3.1: Fixed geometric / thermal parameters.

For the fixed parameter category, we directly measure the geometric parameters and adopt the thermal parameters for a cylindrical 18650 cell shown in Fig 3.2 from the literature [134]. We then divide the electrochemical parameters in Table 3.2 into two groups: equilibrium parameters and dynamics parameters. There are 21 to-be-identified parameters in total. The equilibrium parameters are closely related to the cell charge capacity. The dynamics parameters characterize the internal dynamics of the battery, e.g. diffusion, ion transport, ohmic overpotential and electrochemical reactions. Due to the wide range of dynamics parameter values, we apply normalization. Parameter θ_i is normalized to $\bar{\theta}_i$ according to:

$$\text{Logarithmic scale: } \bar{\theta}_i = \frac{\log \theta_i - \log \theta_{i,\min}}{\log \theta_{i,\max} - \log \theta_{i,\min}}, \quad \text{Linear scale: } \bar{\theta}_i = \frac{\theta_i - \theta_{i,\min}}{\theta_{i,\max} - \theta_{i,\min}}, \quad (3.1)$$

where each parameter's upper and lower bound is determined from the existing literature values, and possibly any *a priori* knowledge of the physically meaningful parameter values.

3.1.3 System Generalization

Next, we abstract the electrochemical battery model into a general dynamical system format to formulate our optimal experimental design approach. The dynamical system notation is shown in Table B.1. In particular, the DFN model is represented by differential algebraic equations (DAEs) after discretizing (2.1)-(2.27) in space via a suitable method, e.g. finite differences, Padé approximation, spectral methods (see e.g. [111, 37, 34, 12]):

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{z}, u, \boldsymbol{\theta}), \quad \mathbf{x}(t_0) = \mathbf{x}_0, \quad (3.2)$$

$$\mathbf{0} = \mathbf{g}(\mathbf{x}, \mathbf{z}, u, \boldsymbol{\theta}), \quad \mathbf{z}(t_0) = \mathbf{z}_0, \quad (3.3)$$

$$\mathbf{y} = \mathbf{h}(\mathbf{x}, \mathbf{z}, u, \boldsymbol{\theta}). \quad (3.4)$$

Equilibrium parameters		
Symbol	Description [SI units]	Normalization scheme
ϵ_s^-	Solid-phase volume fraction [-]	-
ϵ_s^+	Solid-phase volume fraction [-]	-
$n_{\text{Li},s}$	Moles of cyclable lithium in solid phase [mol]	-
Dynamics parameters		
Symbol	Description [SI units]	Normalization scheme
D_s^-	Solid-phase diffusion coefficients of the anode [$\text{m}^2 \text{s}^{-1}$]	Logarithmic
D_s^+	Cathode solid-phase diffusion coefficients of the cathode [$\text{m}^2 \text{s}^{-1}$]	Logarithmic
R_s^-	Solid-phase particle radii of the anode [m]	Linear
R_s^+	Solid-phase particle radii of the cathode [m]	Linear
σ^-	Solid-phase conductivity of the anode [S m^{-1}]	Logarithmic
σ^+	Solid-phase conductivity of the cathode [S m^{-1}]	Logarithmic
$D_e(\cdot)$	Electrolyte diffusion coefficient [$\text{m}^2 \text{s}^{-1}$]	Linear
ϵ_e^-	Electrolyte volume fraction [-]	Linear
ϵ_e^{sep}	Electrolyte volume fraction [-]	Linear
ϵ_e^+	Electrolyte volume fraction [-]	Linear
$\kappa(\cdot)$	Electrolyte conductivity [S m^{-1}]	Linear
t_c^0	Transference number [-]	Linear
$\frac{d \ln f_{c/a}}{d \ln c_e}(\cdot)$	Activity coefficient [-]	Linear
k^-	Kinetic rate constants [$(\text{A m}^{-2})(\text{m}^3 \text{mol}^{-1})^{(1+\alpha)}$]	Logarithmic
k^+	Kinetic rate constants [$(\text{A m}^{-2})(\text{m}^3 \text{mol}^{-1})^{(1+\alpha)}$]	Logarithmic
R_f^-	Film resistance [Ωm^2]	Linear
R_f^+	Film resistance [Ωm^2]	Linear
$c_e(x, 0)$	Initial Li-ion concentration in electrolyte [mol m^{-3}]	Linear

Table 3.2: Parameters of interest for identification.

DAE Variables	DFN Variables
$\mathbf{x}$	$c_s^-, c_s^+, c_e, T_1, T_2$
$\mathbf{z}$	$\phi_s^-, \phi_s^+, i_e^-, i_e^+, \phi_e, j_n^-, j_p^+$
u	I
$\boldsymbol{\theta}$	dynamic parameters in Table 3.2

Table 3.3: DAE notation for the electrochemical model.

Denote $\mathbf{x} = [c_s^-, c_s^+, c_e, T_1, T_2]^T \in \mathbb{R}^{n_x}$ as the state vector, $\mathbf{z} = [\phi_s^-, \phi_s^+, i_e^-, i_e^+, \phi_e, j_n^-, j_p^+]^T \in \mathbb{R}^{n_z}$ as the algebraic variable vector, $\mathbf{y} = V(t)$ as the output variable defined in (2.27). Importantly, $\boldsymbol{\theta} = [D_s^-, D_s^+, \dots, c_e^0] \in \mathbb{R}^{n_p}$ is the vector of dynamics parameters in Table 3.2 which we seek to identify.

Sensitivity analysis is used to understand how a model's output depends on variations in the parameter values [132]. Based on nominal parameter values, local sensitivity analysis measures the effects of small changes in the parameters have on the output. For continuous dynamical systems, the local sensitivities are defined as the first-order partial derivatives of the system output with respect to the parameters. We briefly introduce how to derive local sensitivities in dynamical systems described by (3.49)-(3.51). Subsequently, we develop this

approach toward a parameter estimation framework via Fisher information.

Define sensitivity variables as follows:

$$S_{\mathbf{x}} = \frac{\partial \mathbf{x}}{\partial \boldsymbol{\theta}}, \quad S_{\mathbf{z}} = \frac{\partial \mathbf{z}}{\partial \boldsymbol{\theta}}, \quad S_{\mathbf{y}} = \frac{\partial \mathbf{y}}{\partial \boldsymbol{\theta}}, \quad (3.5)$$

where $S_{\mathbf{x}} \in \mathbb{R}^{n_x \times n_p}$, $S_{\mathbf{z}} \in \mathbb{R}^{n_z \times n_p}$, $S_{\mathbf{y}} \in \mathbb{R}^{n_y \times n_p}$ are sensitivity matrices. The i, j matrix element is defined as the partial derivative of the i -th variable to the j -th normalized parameter, e.g.

$$[S_{\mathbf{x}}]_{i,j}(t) = \frac{\partial x_i(t)}{\partial \theta_j}. \quad (3.6)$$

The evolution of the sensitivity variables is governed by the sensitivity differential algebraic equations (SDAEs), which can be derived following the procedure in [51]:

$$\frac{d}{dt} S_{\mathbf{x}} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}} S_{\mathbf{x}} + \frac{\partial \mathbf{f}}{\partial \mathbf{z}} S_{\mathbf{z}} + \frac{\partial \mathbf{f}}{\partial \boldsymbol{\theta}}, \quad S_{\mathbf{x}}(0) = S_{\mathbf{x}0}, \quad (3.7)$$

$$0 = \frac{\partial \mathbf{g}}{\partial \mathbf{x}} S_{\mathbf{x}} + \frac{\partial \mathbf{g}}{\partial \mathbf{z}} S_{\mathbf{z}} + \frac{\partial \mathbf{g}}{\partial \boldsymbol{\theta}}, \quad S_{\mathbf{z}}(0) = S_{\mathbf{z}0}, \quad (3.8)$$

$$S_{\mathbf{y}} = \frac{\partial \mathbf{h}}{\partial \mathbf{x}} S_{\mathbf{x}} + \frac{\partial \mathbf{h}}{\partial \mathbf{z}} S_{\mathbf{z}} + \frac{\partial \mathbf{h}}{\partial \boldsymbol{\theta}}. \quad (3.9)$$

The advantage of SDAEs is that they provide a rigorous mathematical computation of the sensitivities compared to a perturbation method where sensitivities are obtained by perturbing each parameter slightly and calculating the output difference with respect to nominal parameters. Note that SDAEs are linear time-varying DAEs, where the Jacobians are computed at each time step. The Jacobians can be derived analytically by-hand, or calculated numerically via finite differences. In this work, we utilize automatic differentiation since it provides accurate, automated, and fast Jacobian calculations. In particular, we use CasADi [3], which efficiently computes the first and second-order derivatives. In this work, the battery model DAEs and the corresponding SDAEs are simulated by using the IDAS integrator provided by SUNDIALS via the CasADi interface [43]. Besides convenience for simulation, the automatic calculation of Jacobians provides advantages for optimal experiment design and parameter estimation, as described in the following sections.

3.1.4 Optimal Experimental Design

In this section, we propose a systematic framework for Li-ion battery parameter identification, as summarized in Fig. 3.3. First, we briefly summarize the approach.

As a first step, one should select a mathematical model to represent the system under study. The mathematical model considered here is the DFN model. The DFN model mathematically takes the form of DAEs. Once the battery model is selected, then nominal parameter values must be determined. It is important to note that finding nominal parameter values near the true values is beneficial, since the optimal parameter fitting algorithms are based on

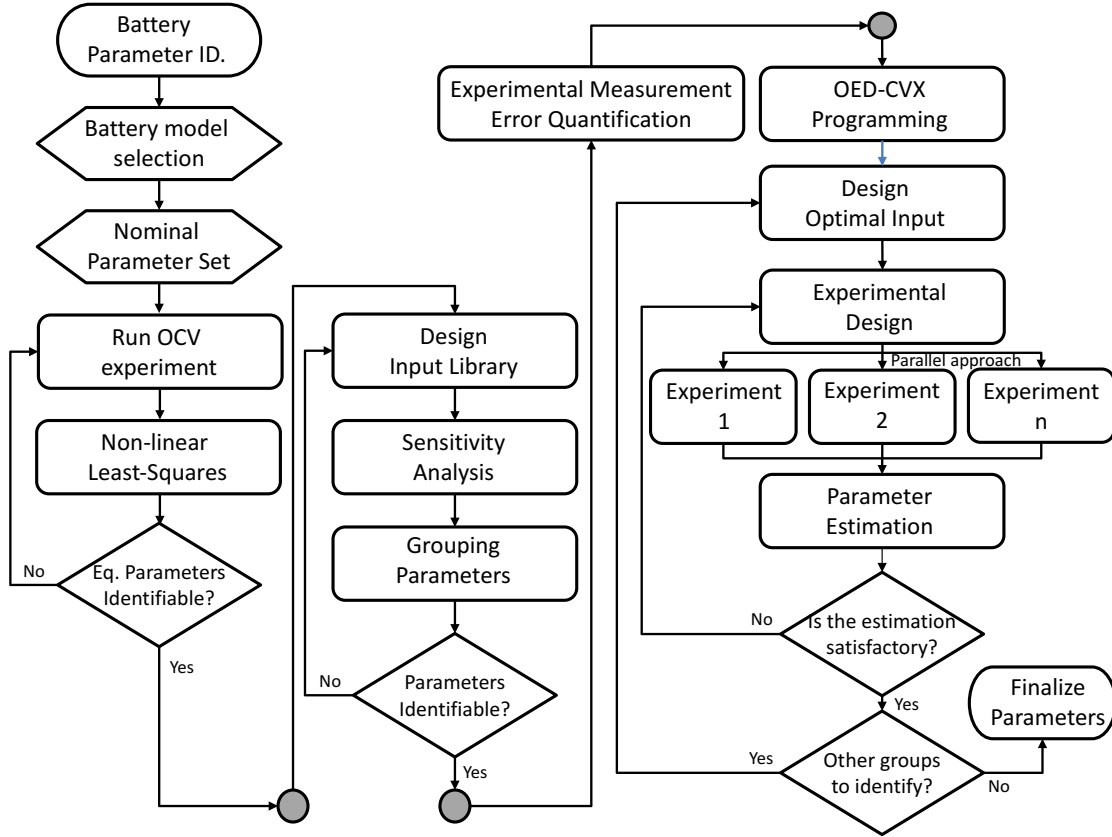


Figure 3.3: The proposed model-based design of experiments framework for parameter identification in of the DFN model.

gradient-descent approach and therefore converge to local minima. We searched relevant papers to choose credible nominal values.

After setting nominal parameter values, we separate the parameters into two groups: equilibrium parameters and dynamical parameters. For equilibrium parameter identification, we adopt the method in [24]. Note that it is possible to derive sensitivity values for equilibrium parameters, however, these values turned out to be dependant on the initial states, resulting in non-zero values for zero currents. For this reason, we solve the non-linear least square problem using data generated from an open circuit voltage (OCV) experiment, which applies very low rate charge/discharge cycles to characterize the equilibrium state. To estimate the dynamical parameters, we generate an input library of possible input trajectories. The sensitivities for each input trajectory are calculated in the library. Then we partition the dynamical parameters into groups, based on their sensitivity magnitudes. High sensitivity parameter groups should be identified first, followed by less sensitive parameters groups.

After grouping the parameters, we run optimal experimental design via convex programming to select optimal inputs for parameter estimation. We use the open-source `cvx` solver to

select optimal inputs. Then, corresponding experiments are executed to collect experimental measurements. For the experimental setup, we utilize a PEC Corp. SBT2050 series tester and Espec environmental chamber. Once experimental measurements are acquired, the parameter estimation algorithm is applied to fit the simulation result to experimental measurements. In this work, we utilize the Levenberg-Marquardt algorithm for parameter updates at each iteration. We repeat this process until each parameter group is complete. Once we finalize the identified parameters, we compare the identified model output predictions versus a testing data set. For comparison, we benchmark our proposed approach against simple discharge and charge current profiles, which we refer to as a “conventional approach”. The following subsections contain detailed analysis for each block of the proposed model-based design of experiments.

Equilibrium structure parameter identification

We formulate a procedure to identify parameters in the equilibrium structure of electrochemical models. In words, these parameters correspond to cyclable lithium in the solid phase $n_{Li,s}$ [mol], electrode capacity $Q^\pm$ [Ah], and the stoichiometric points $\theta^\pm$ [-]. Note, in this subsection we overload the symbol θ to represent stoichiometric points to remain consistent with the literature. To identify these parameters, we require experimentally obtained open circuit voltage (OCV) data and known open circuit potential functions for each electrode $U^\pm(\cdot)$. We remark that *a priori* knowledge of $U^\pm(\cdot)$ appears to be a necessary assumption for equilibrium parameter identification, a finding that is consistent with existing literature [13, 50, 49]. We adopt the procedure from a patent by one of the co-authors [24].

Consider the equilibrium structure of the electrochemical model described in [119]. The relationship between cyclable lithium $n_{Li,s}$ and normalized electrode concentrations $\theta^\pm$ is given by:

$$n_{Li,s} = \varepsilon_s^- L^- A c_{s,\max}^- \cdot \theta^-(k) + \varepsilon_s^+ L^+ A c_{s,\max}^+ \cdot \theta^+(k), \quad \forall k, \quad (3.10)$$

where k indexes stoichiometric points parameterized by time or Ah-processed. For convenience, let us define the lumped parameters $H^\pm$ as

$$H^\pm = \varepsilon_s^\pm L^\pm A \cdot c_{s,\max}^\pm. \quad (3.11)$$

Dimensional analysis reveals that $H^\pm$ have dimensions of [mol]. Consequently, we define $H^\pm$ to be the molar capacity of the electrodes. The charge capacity of the electrodes is therefore given by:

$$Q^\pm = F \cdot H^\pm, \quad (3.12)$$

where F is Faraday’s constant. Then, let us define a recursive relationship that relates the steady-state normalized anode concentrations before and after injecting current I for Δt time units.

$$\theta^-(k+1) = \theta^-(k) - \frac{\Delta t}{F H^-} \cdot I, \quad \forall k. \quad (3.13)$$

The equilibrium voltage, i.e. OCV, is given by:

$$V(k) = U^+(\theta^+(k)) - U^-(\theta^-(k)), \quad \forall k. \quad (3.14)$$

Solving (3.10) for $\theta^+(k)$ and substituting into (3.14) gives:

$$V(k) = U^+ \left(\frac{n_{Li,s} - H^-\theta^-(k)}{H^+} \right) - U^-(\theta^-(k)), \quad \forall k. \quad (3.15)$$

To summarize, we have unknowns $x = [\theta^-(0), H^+, H^-, n_{Li,s}]^T$ and equations (3.13), (3.15).

Optimization formulation

We formulate the following nonlinear optimization problem to find the unknown variables x given experimentally obtained cell OCV data $V_{ocv}(k)$ and known open circuit potential functions $U^\pm(\theta^\pm)$:

$$\underset{x}{\text{minimize}} \quad \sum_{k=0}^N [V(k) - V_{ocv}(k)]^2, \quad (3.16)$$

$$\text{subject to:} \quad V(k) = U^+ \left(\frac{n_{Li,s} - H^-\theta^-(k)}{H^+} \right) - U^-(\theta^-(k)), \quad \forall k, \quad (3.17)$$

$$V^{\max} = U^+ \left(\frac{n_{Li,s} - H^-\theta^-(0)}{H^+} \right) - U^-(\theta^-(0)), \quad (3.18)$$

$$\theta^-(k+1) = \theta^-(k) - \frac{\Delta t}{FH^-} \cdot I, \quad \forall k = 0, 1, \dots, N-1, \quad (3.19)$$

where N is the total number of experimental data points, and $V^{\max}$ corresponds to the maximum OCV and the first index $k = 0$. Note the last constraint is initialized with optimization variable $\theta^-(0)$. This optimization program is nonlinear and non convex in the optimization variables, requiring a nonlinear optimization solver, such as `fmincon` in Matlab.

Suppose that $\theta^\pm(0)^*$ and $\theta^\pm(N)^*$ correspond to the optimized normalized concentrations at the maximum and minimum voltage limits, respectively, according to the cell's datasheet. Then the stoichiometric points are given by:

$$\underline{\theta}^{\pm,*} = \theta^\pm(0)^*, \quad \bar{\theta}^{\pm,*} = \theta^\pm(N)^*. \quad (3.20)$$

Suppose that $H^{\pm,*}$ correspond to the optimized molar capacities of each electrode. Then the charge capacity of each electrode $Q^{\pm,*}$ and the cell Q^* can be calculated as:

$$Q^{\pm,*} = F \cdot N^{\pm,*}, \quad Q^* = \min \{ Q^{+,*}, Q^{-,*}, F \cdot n_{Li,s}^* \}. \quad (3.21)$$

By solving the optimization problem (3.16)-(3.19) using OCV data from three averaged C/50 charge/discharge cycles, we obtain the following equilibrium parameters as follows:

Figure 3.4 compares the identified equilibrium model against experimental data. We achieve an overall root mean square error (RMSE) of less than 5 mV in 98% of the operating range. If one uses faster OCV tests, e.g. C/25, C/10, or C/5, then the resulting model identification accuracy will degrade. Based on the identified structure, we define the cell state-of-charge (SOC) in association with the equilibrium voltage as listed in Table 3.5, which is used for simulation and experiment design in subsequent sections.

Parameter	Estimated values
ε_s^-	5.438895e-01
ε_s^+	6.663649e-01
$n_{Li,s}$	0.1406 moles

Table 3.4: Identified equilibrium parameters.

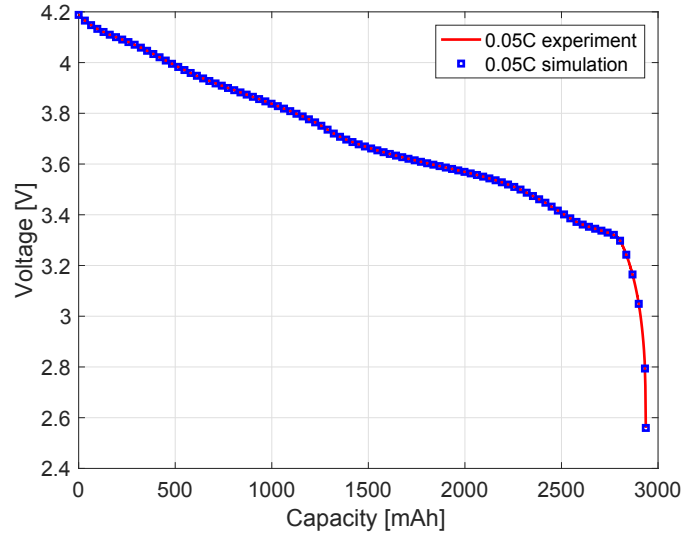


Figure 3.4: OCV comparison between identified model and experimental data for three averaged C/50 charge/discharge cycles.

SOC	Voltage mapping [V]
80%	3.9481
60%	3.7689
40%	3.5982
20%	3.4562

Table 3.5: Cell SOC to voltage mapping.

Sensitivity analysis & Grouping for dynamic parameters

Next we identify the dynamic parameters in Table 3.2. A fundamental challenge is linear dependence between the parameter sensitivities, resulting in non-uniqueness between estimated parameter values. For this reason, we adopt a *parameter grouping*-based approach [106]. In our approach, each parameter's sensitivity is analyzed across a large number of input profiles.

Input library design

In order to design the optimal set of experiments for identifying battery parameters, a suite of candidate input profiles has been generated. Broadly, these candidate profiles can be categorized as (i) pulse, (ii) sinusoid, and (iii) driving cycle inputs. A total of 738 profiles

exists across these three input categories. Across all categories, profiles have been normalized to 1 Ah of charge processed in order to fairly evaluate candidate profiles. Setting constraints that define the feasible set of input profiles is the first step in evaluating the candidate input profiles. Across all profile categories, the following constraints must be upheld:

1. $T_1 \leq 50^\circ C$
2. $|I(t)| \leq 5C$
3. $600 \text{ sec} \leq t_f \leq 3600 \text{ sec}$

Additionally, all input profiles are generated from the following initial SOC: $SOC_0 \in \{0.8, 0.6, 0.4, 0.2\}$. The initial voltage V_0 values associated with initial SOC are determined according to the Table 3.5. On the other hand, input profiles need to keep the SOC within its upper and lower limit. To check that these conditions hold, the absolute ΔSOC for each profile is calculated and added to SOC_0 . The immediate conclusions are that driving cycle profiles, all of which are net discharging, cannot be applied when the battery SOC is 20%. Similarly, discharging pulse and sinusoid profiles cannot be applied when the battery SOC is 20%. Charging pulse and sinusoid profiles cannot be applied for $SOC \geq 80\%$. Consequently, these inputs are parsed out of the input library. The remainder of this section will describe the parameters of each profile category and its corresponding normalization scheme.

Pulse profiles are generated with six input variables for a total of 540 pulse profiles as shown in Table 3.6:

Input variable	Setting
Amplitude	0.5C to 5C, in 0.5C increments
Pulse width	1 sec to 1000 sec
Duty cycle	$n * (\text{pulse width})$, $n \in \{1, 2, 3\}$
Total time	600 sec (10 min) to 3600 sec (1 hr)
Initial voltage	$V_0 = \{3.9481 \text{ } 3.7689 \text{ } 3.5982 \text{ } 3.4562\}$
Current	charge only, discharge only, or both

Table 3.6: Pulse profile specification.

Note that the pulse width upper bound is set to be commensurate with the characteristic diffusion time for a given battery chemistry. The current values are normalized using an L_1 norm, i.e. the integral of the current magnitude over the total time of the profile: $I(t)/\int |I(t)|dt$. Therefore, the total normalized amount of charge processed in each pulse is equal to 1 Ah.

Sinusoid profiles are generated with four input variables for a total of 180 sinusoid profiles, as detailed in Table 3.7. Note that the frequency range was set in order to: (i) avoid exciting dynamics occurring at un-modeled frequencies, and (ii) not violate the sampling rate limit of a standard battery tester. The current values are also normalized with the L_1 norm, i.e. the integral of the current magnitude over the total time of the profile: $I(t)/\int |I(t)|dt$. Therefore,

the total normalized amount of charge processed in each pulse is equal to 1 Ah. Charge only, discharge only, and charge-discharge profiles follow the same structure as the pulse profiles.

Input variable	Setting
Frequency	$\{0.01, 0.05, 0.1\}$ Hz
Total time	600 sec (10 min) to 3600 sec (1 hr)
Initial voltage	$V_0 = \{3.9481 \ 3.7689 \ 3.5982 \ 3.4562\}$
Current	charge only, discharge only, or both

Table 3.7: Sinusoid profile specification.

The dynamic drive cycle profiles summarized in Table 3.8 are evaluated at three different initial SOC_s, for a total of 18 dynamic drive cycle profiles. Note that US06 is the most aggressive cycle in terms of peak current applied. The current values are normalized to the L_1 norm so that the total normalized amount of charge processed in each pulse is equal to 1 Ah.

Name	Description
DC1	naturalistic morning driving test profile [57]
DC2	nautralistic evening driving test profile [57]
LA92	Unified driving schedule for emission inventory [1]
SC04	Speed correction driving schedule [1]
UDDS	Urban dynamometer driving schedule [1]
US06	High acceleration aggressive driving schedule [1]

Table 3.8: Driving cycle profile specification.

There are 540 pulse inputs, 180 sinusoidal inputs, and 18 driving cycle inputs, yielding a total of 738 inputs and 329.65 hours in the input library. Note that short duration input profiles tend to have large input magnitudes due to normalization while long duration input profiles have relatively small input magnitudes. High performance computing cluster is used to parallelize model simulation and sensitivity calculations.

Parameter grouping

In this section, we introduce the parameter grouping methodology for parameter identification. It is well known (see e.g. [36]) that the entire electrochemical parameter vector θ is weakly identifiable from the measured output, since the system is nonlinear in the parameters. This is due to linear dependence between the parameter sensitivity vectors [106]. Therefore, it is necessary to analyze the linear dependency among electrochemical parameters, and rank/organize them into groups to avoid non-unique solutions during the parameter identification process. Suitable linear transformations of $S_y^T S_y$ can reveal properties such as norm and linear dependence [69]. For parameter grouping, we first perform sensitivity analysis across the library of input profiles.

After calculating the sensitivities (3.54)-(3.56) for 738 profiles through parallel computing, we apply the Gram-Schmidt process on $S_y^T S_y$ to reveal the orthogonalized sensitivity magni-

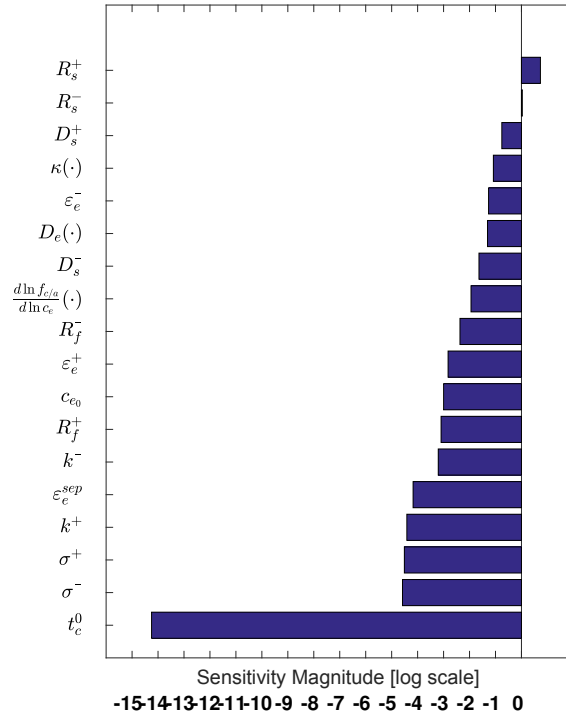


Figure 3.5: The sorted average orthogonalized sensitivity magnitudes across all inputs in the input library.

tudes and linear dependence [14]. Figure 3.5 visualizes the average sensitivity magnitudes via Graham-Schmidt orthogonalization over 738 profiles.

Based on this sensitivity analysis, we group the parameters based on their orthogonalized sensitivity magnitudes. The resultant groups are shown in Table 3.9. It is evident that some of the parameters have strong identifiability, i.e. particle radii $R_s^\pm$. Other parameters, e.g. the transference number t_c^0 and solid phase conductivities $\sigma^\pm$, are weakly identifiable. To validate these conclusions, we apply a parameter perturbation approach to a parameter from each of the four groups. Figure 3.6(a) visually demonstrates that Group 1 parameter R_s^- has the largest impact on voltage, followed by representative parameters in Group 2 (D_s^-), Group 3 (R_f^-), and Group 4 (σ^-). For fair comparison, we perturb each representative parameter according to its normalized value, denoted by upper bars in Fig. 3.6. Note, one should not expect these conclusions to be generally true across all cell chemistries, models, and manufacturers. These conclusions are specific to the cell under study and normalization.

In the parameter identification framework, Group 1 is identified first, while the other parameters are fixed to their nominal values. Next, Group 2 parameters are identified while fixing the remaining unidentified parameters, and so forth.

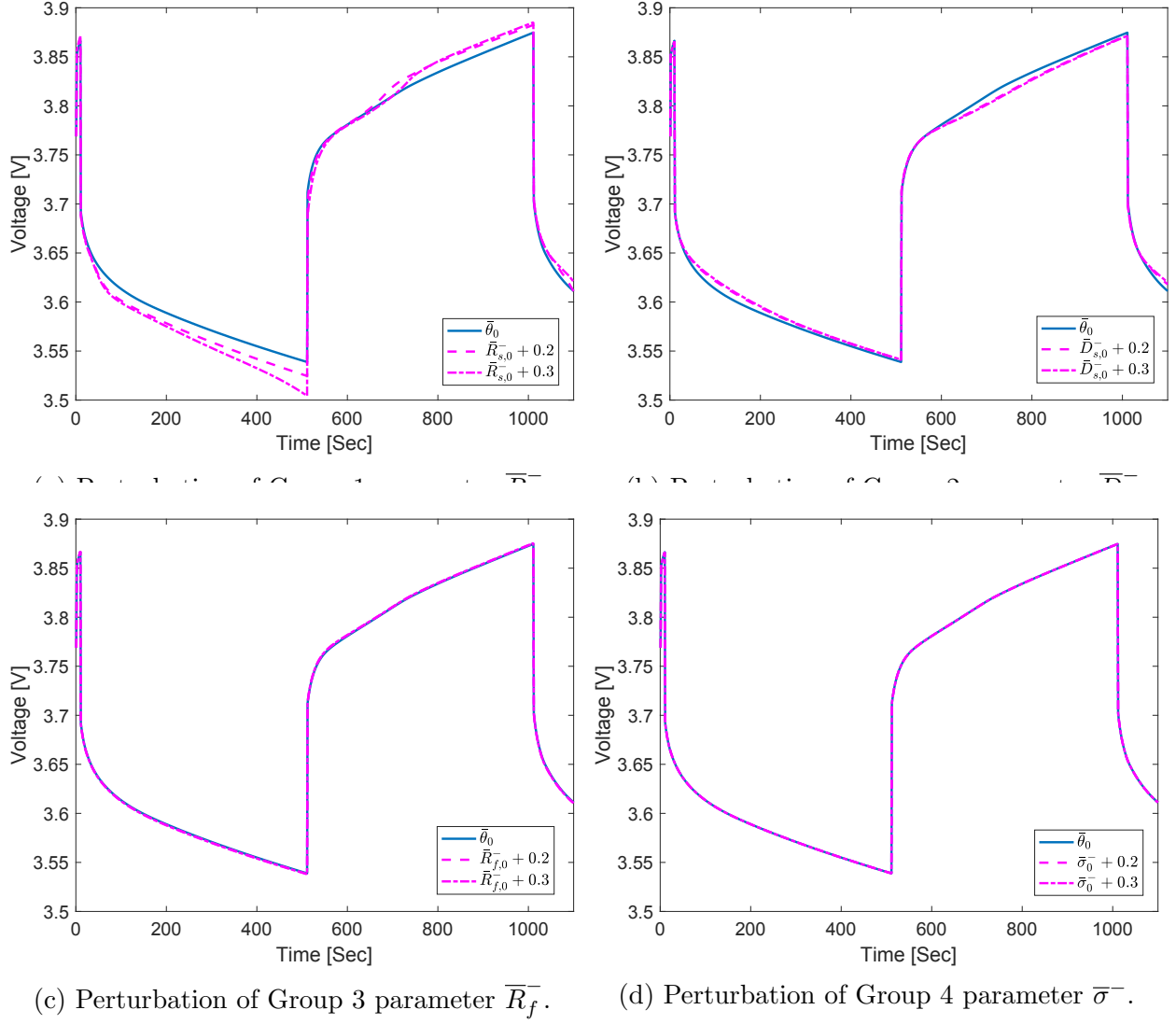


Figure 3.6: The comparison of model output impact by changing parameters.

Group 1	Group 2	Group 3	Group 4
R_s^-	D_s^-	R_f^-	σ^-
R_s^+	D_s^+	k^-	σ^+
	ε_e^-	ε_e^+	ε_e^{sep}
	$\kappa(\cdot)$	c_{e0}	k^+
	$D_e(\cdot)$	R_f^+	t_c^0
	$\frac{d \ln f_{c/a}}{d \ln c_e}(\cdot)$		

Table 3.9: Dynamic Parameter Groups.

Optimal input design

In statistical experiment design, the amount of “information” on parameter vector $\boldsymbol{\theta}$ contained in the observation $\mathbf{y}$ from an experiment is calculated by the Fisher information matrix, $\mathbf{F}$ [129]. The Fisher information matrix is mathematically defined as:

$$\mathbf{F} = \int_0^{t_f} S_{\mathbf{y}}^T(t) Q(t)^{-1} S_{\mathbf{y}}(t) dt, \quad (3.22)$$

where $t \in [0, t_f]$, and $Q(t)$ is the covariance matrix of the measurement error. Since the true parameters $\boldsymbol{\theta}^*$ are unknown, the sensitivity is calculated around nominal parameter values $\boldsymbol{\theta}_0$. The deviation of the parameter estimates from their true values can be expressed as the covariance matrix Σ . According to the Cramer-Rao bound [28], [22], the inverse of the Fisher information matrix provides a lower bound on Σ ,

$$\mathbf{F}^{-1} \leq \Sigma. \quad (3.23)$$

Our goal is to find inputs that minimize the lower bound of the parameter estimation error, thus improving the parameter estimation quality. To optimize the amount of information, a proper scalarization of $\mathbf{F}$ should be considered. Several scalarization criteria are commonly used in the literature such as:

- D-optimality : $\log \det(\mathbf{F}^{-1})$.
- A-optimality : $\text{trace}(\mathbf{F}^{-1})$.
- E-optimality : $\lambda_{\max}(\mathbf{F})$.

Each optimality criterion has a different geometrical interpretation. For instance, D-optimal design minimizes the geometric mean of the errors in the parameters, while E-optimal design minimizes the largest eigenvalue of the confidence region of parameters. In this work, we use D-optimal design as it is the most commonly used [38]. However, the other criteria are equally applicable in this framework.

We formulate a procedure to optimize experiment design to produce inputs that are *maximally informative* for parameter estimation. A natural and mathematically elegant

approach is to formulate an optimal control problem. Namely, one may seek an input trajectory that maximizes D-optimality subject to the system dynamics. This concept has been applied in [100, 67]. However, solving a large-scale optimal control problem with thousands of states and nonlinear dynamics given by (3.49)-(3.51), (3.54)-(3.56) is computationally intractable. For example, we found it requires two weeks of wall clock time to optimize a 100 second input trajectory for the electrochemical model. The previous literature [100, 67] side-steps this challenge using control vector parameterization and simplified models, such as an equivalent circuit model or single particle model.

To bypass the challenge of solving a large-scale nonlinear optimal control problem, we pursue a different approach. Specifically, we seek the set of inputs from an input library which maximizes the Fisher information matrix. This process yields a convex optimization program, which can be rapidly solved with polynomial complexity open-source solvers, such as `cvx` [39].

We now detail the theory behind optimal experiment design via convex optimization. Suppose we have a set of L experimental inputs $u_i(t), i = 1, 2, \dots, L$. For each input profile $u_i(t)$, we obtain a corresponding sensitivity vector $S_{\mathbf{y},i}(t)$ by solving (3.49)-(3.51) and (3.54)-(3.56) simultaneously. Amongst these L inputs $u_i(t), i = 1, 2, \dots, l$, we select M inputs that are maximally informative as measured by the Fisher information matrix F , where $M < L$. Let $m_j \in \{0, 1\}$ be a binary value that indicates if experiment j is executed from the input library. Then, the total number of experiments is given by

$$m_1 + m_2 + \dots + m_l = M. \quad (3.24)$$

We then rewrite the Fisher information matrix as:

$$\mathbf{F} = \sum_{i=1}^L m_i S_{\mathbf{y},i}^T Q_i^{-1} S_{\mathbf{y},i}. \quad (3.25)$$

We now formulate a combinatorial optimization problem to maximize the D-optimality criterion of F :

$$\underset{m_i}{\text{minimize}} \quad \log \det \left(\sum_{i=1}^L m_i S_{\mathbf{y},i}^T Q_i^{-1} S_{\mathbf{y},i} \right)^{-1}, \quad (3.26)$$

$$\text{subject to} \quad m_i \in \{0, 1\}, \quad (3.27)$$

$$m_1 + \dots + m_l = M. \quad (3.28)$$

This problem is a binary integer program, where the optimal number of experiments m_i is the solution. In general, large-scale combinatorial problems are NP-hard. In this work, we relax the integer constraint (3.27) into $0 \leq m_i \leq 1$, yielding a relaxed optimization problem that is convex. Let $\eta_i = m_i/M$ be the fraction of experiment type i to execute. Then the Fisher information (3.25) can be re-written as

$$\mathbf{F} = M \sum_{i=1}^L \eta_i S_{\mathbf{y},i}^T Q_i^{-1} S_{\mathbf{y},i}, \quad (3.29)$$

where $\eta \in \mathbb{R}^L$, $\mathbf{1}^T \eta = 1$. Thus, our final convex optimal experiment design problem is:

$$\underset{\eta}{\text{minimize}} \quad \log \det \left(\sum_{i=1}^L \eta_i S_{\mathbf{y},i}^T Q_i^{-1} S_{\mathbf{y},i} \right)^{-1}, \quad (3.30)$$

$$\text{subject to} \quad 0 \leq \eta_i \leq \frac{1}{M}, \quad \forall i \quad \mathbf{1}^T \eta = 1, \quad (3.31)$$

where M can be dropped without affecting the minimizer. One can show this is a convex problem with respect to η [15]. Additionally, the solutions will be integral; that is, the solution to relaxed optimization problem (3.30)-(3.31) is also the solution to (3.26)-(3.27).

Remark: Suppose that associated with each experiment $u_i(t)$ is a cost c_i , which can represent time required, economic/labor cost, or battery degradation. The total cost is:

$$c_1 m_1 + \dots + c_l m_l = m c^T \eta. \quad (3.32)$$

Suppose we have a cost budget of B . One can then add a budget constraint as a scalar linear inequality, yielding convex program:

$$\underset{\eta}{\text{minimize}} \quad \log \det \left(\sum_{i=1}^L \eta_i S_{\mathbf{y},i}^T Q_i^{-1} S_{\mathbf{y},i} \right)^{-1}, \quad (3.33)$$

$$\text{subject to} \quad 0 \preceq \eta \preceq \frac{1}{M}, \quad \mathbf{1}^T \eta = 1, \quad (3.34)$$

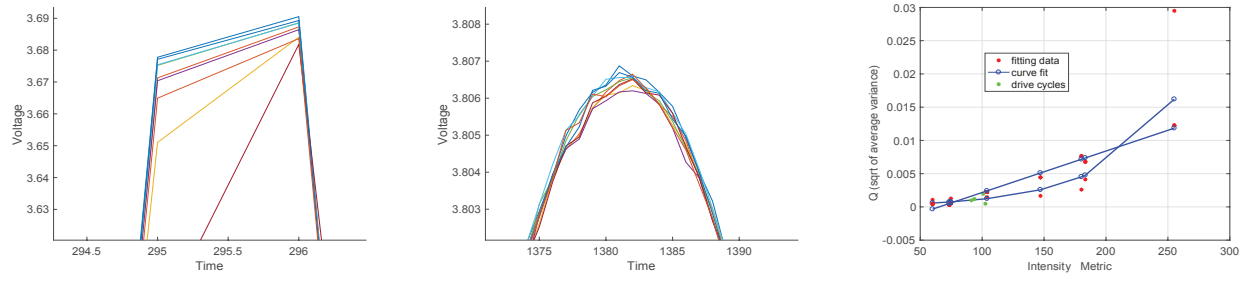
$$m c^T \eta \leq B. \quad (3.35)$$

Output measurement error quantification

Consider the output measurement covariance Q_i in (3.30). This quantity is critically important to include in the optimization formulation. To motivate this point, consider figure 3.7a and 3.7b which visualize the battery tester's measured voltage for ten repeated pulse inputs. Figure 3.7a contains time steps in which the measured voltage has variance larger than 30 mV. This variance is uncorrelated with the order in which the experiments are performed. In contrast, the sinusoidal tests in Fig. 3.7b have variance consistently around 0.3 mV. Consequently, there can be non-trivial variation in the battery tester's measured voltage, depending on the input profile. Note, our SBT2050 test system manual claims a voltage resolution of 100 μ V. The precise source of the measurement variance is unknown.

This motivates quantification of the output measurement covariance. Namely, (3.30)-(3.31) requires Q_i for each input in the library. However, experimentally characterizing Q_i for each input requires running every experiment in the input library multiple times (e.g. 10 times) – an intractable task. Therefore, we propose to predict the covariance Q_i using a regression model trained from a small number of experiments. This enables us to account for variance in experimental trials without running a large number of experiments.

The size of matrix Q is based on the number of samples taken during an experiment. Due to this, we scalarize Q in order to compare how Q varies across experiments that have



(a) Zoom of multiple, superimposed pulse experimental trials. This experiment has up to 30 mV of measured voltage variance.

(b) Zoom of multiple, superimposed sinusoidal experimental trials. This experiment has 0.3 mV of measured voltage variance.

(c) Scalarized variance from training data, testing data, and regression model (curve fit).

Figure 3.7: Experimental outputs and quantification.

different lengths. For a given experiment and its corresponding covariance matrix Q^{exp} , we scalarize Q^{exp} using the square root of the mean across the diagonal entries of Q^{exp} :

$$q = \sqrt{\frac{1}{n} \sum_i^n Q_{i,i}^{\text{exp}}}, \quad (3.36)$$

where n is the dimension of the matrix Q . Next, we seek a regression model to predict q from the current input profile. To this end, we consider a metric of “input intensity” as the feature in our regression model. Namely, we propose to use the L_2 norm, $\tilde{I}$, to characterize input intensity where

$$\tilde{I} = \sqrt{\int_0^{t_f} I(t)^2 dt}. \quad (3.37)$$

Using $\tilde{I}$ as a regression model feature, we fit the following regression models to q :

$$\text{for charge inputs: } \hat{q}_c = a_c \tilde{I} + b_c, \quad \text{for discharge inputs: } \hat{q}_d = a_d \cdot e^{b_d \tilde{I}}, \quad (3.38)$$

where $a_c = 6.2382 \times 10^{-5}$, $b_c = -4.0739 \times 10^{-3}$, $a_d = 2.1045 \times 10^{-4}$, $b_d = 1.7020 \times 10^{-2}$. For inputs that are not purely charge or discharge, the two models are averaged:

$$\hat{q}_{cd} = \frac{1}{2} (q_c + q_d). \quad (3.39)$$

The scalar predictions $\hat{q}$ are then squared and constructed into a matrix according to:

$$\hat{Q} = \hat{q}^2 \cdot \mathbb{I}_n, \quad (3.40)$$

where $\mathbb{I}_n$ is an $n \times n$ identity matrix. For training, 5 drive cycle profiles are used to estimate the parameters a_c, b_c, a_d, b_d in (3.38). We test the predicted variance on drive cycle inputs. Data for training and testing is provided in Fig. 3.7c, along with the curve fit regression model predictions.

Parameter estimation

After obtaining experimental data, we now seek to optimally fit the parameter vector $\boldsymbol{\theta}$ to this data. The optimization problem for parameter identification can be formulated as a nonlinear least squares problem:

$$\underset{\hat{\boldsymbol{\theta}}}{\text{minimize}} \quad \sum_{i=1}^M \sum_{k=0}^{k_f} \frac{\mathbf{y}_i(k) - \hat{\mathbf{y}}_i(k; \hat{\boldsymbol{\theta}})}{Q_i(k)}, \quad (3.41)$$

where k indexes the timed samples, M is the number of optimized input profiles obtained from optimal input design, $\mathbf{y}_i(k)$ is the experimentally measured voltage, $\hat{\mathbf{y}}_i(k; \hat{\boldsymbol{\theta}})$ is the model's voltage prediction, and $Q_i(k)$ is the measurement variance for input profile i at time step k . The Levenberg-Marquardt algorithm is used to update the parameters $\hat{\boldsymbol{\theta}}$ iteratively to solve the nonlinear optimization problem (3.57). This algorithm adaptively updates the parameter estimates via a hybridization of the gradient descent update and the Gauss-Newton update [59]:

$$\left[\mathbf{J}^T \mathbf{W} \mathbf{J} + \gamma \cdot \text{diag}(\mathbf{J}^T \mathbf{W} \mathbf{J}) \right] \Delta \boldsymbol{\theta} = \mathbf{J}^T \mathbf{W} (\mathbf{y} - \hat{\mathbf{y}}), \quad (3.42)$$

where $\mathbf{J} = \partial \hat{\mathbf{y}} / \partial \hat{\boldsymbol{\theta}}$ is the local sensitivity of the output $\hat{\mathbf{y}}$, and $\mathbf{W}$ is the inverse of the measurement error covariance matrix, $\mathbf{W} = \mathbf{Q}^{-1}$. The value of γ weighs gradient descent update against Gauss-Newton update. Conveniently, the Levenberg-Marquardt algorithm directly utilizes the Jacobians computed via automatic differentiation, as discussed in the sensitivity analysis in Section 3.1.3. Then, the parameter updates are iteratively updated according to:

$$\boldsymbol{\theta}(n+1) = \boldsymbol{\theta}(n) + \Delta \boldsymbol{\theta}, \quad (3.43)$$

where n is the iteration index.

After optimally fitting the parameters, estimation error statistics are calculated according to

$$\rho_{\boldsymbol{\theta}} = \mathbf{J}^T \mathbf{W} \mathbf{J}, \quad (3.44)$$

$$\sigma_{\boldsymbol{\theta}} = \sqrt{\text{diag}[\mathbf{J}^T \mathbf{W} \mathbf{J}]^{-1}}, \quad (3.45)$$

where $\rho_{\boldsymbol{\theta}}$ is the parameter covariance matrix, and $\sigma_{\boldsymbol{\theta}}$ is the standard parameter error. Lastly, 95% confidence intervals for the parameter estimates are calculated as follows:

$$\hat{\boldsymbol{\theta}} - \mathbf{t}_{(1-0.05, N)} \frac{\sigma_{\boldsymbol{\theta}}}{\sqrt{N}} \leq \boldsymbol{\theta}^* \leq \hat{\boldsymbol{\theta}} + \mathbf{t}_{(1-0.05, N)} \frac{\sigma_{\boldsymbol{\theta}}}{\sqrt{N}}, \quad (3.46)$$

where N is the number of observations, and $\mathbf{t}$ is the upper critical value for the t -distribution with $n - 1$ freedom [126].

3.1.5 Experimental Results

In this section the proposed optimal experimental design is applied to a 18650 Lithium nickel-cobalt-aluminum oxide (NCA) battery cell. To assess the benefits of our proposed approach, we compare results against parameter identification using a conventional approach with standard test cycles.

Parameter identification results

With consideration for output measurement error covariance Q , the results for optimal input selections for parameter identification are listed in the Table 3.10. The optimized inputs have moderate magnitude and do not increase cell temperature more than 5°C. Based on the sensitivity analysis and parameter grouping, we estimate parameters for each group in a cumulative fashion. For example, we allow the Group 1 parameters to be refined with Group 2 parameters while fitting the model parameter to Group 2 experimental data. Similarly, Group 1 & 2 parameters are refined with Group 3 parameters and so on. Since we do not have information on the true parameters, it is useful to identify the parameters in a cumulative way to prevent overfitting and increase the degrees of freedom during optimization. The authors' previous work also showed that an optimized input reduces the condition number of the objective function's Hessian with respect to the parameters [87]. This accelerates gradient descent methods for parameter estimation.

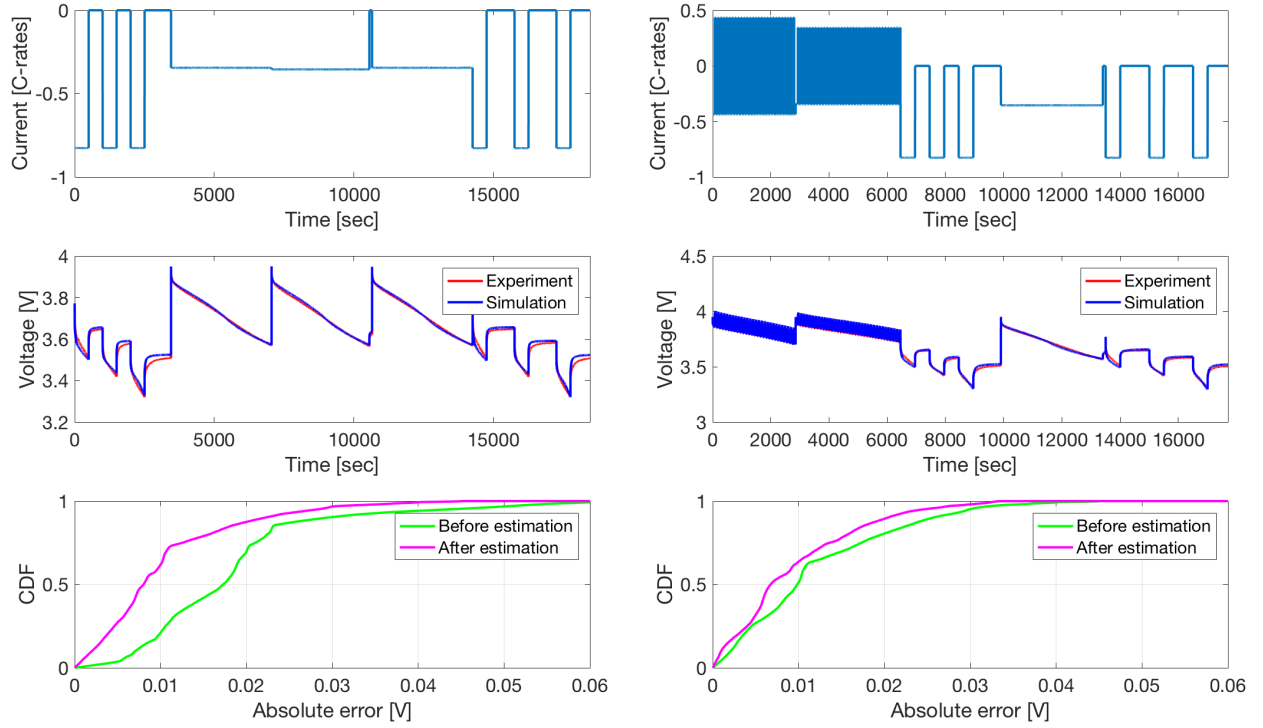
Group 1	Group 2	Group 3	Group 4
#339	#90	#292	#120
#342	#120	#365	#343
#345	#339	#467	#362
#348	#345	#503	#503
#365	#365	#507	#507

Table 3.10: Input profiles from optimal experimental design.

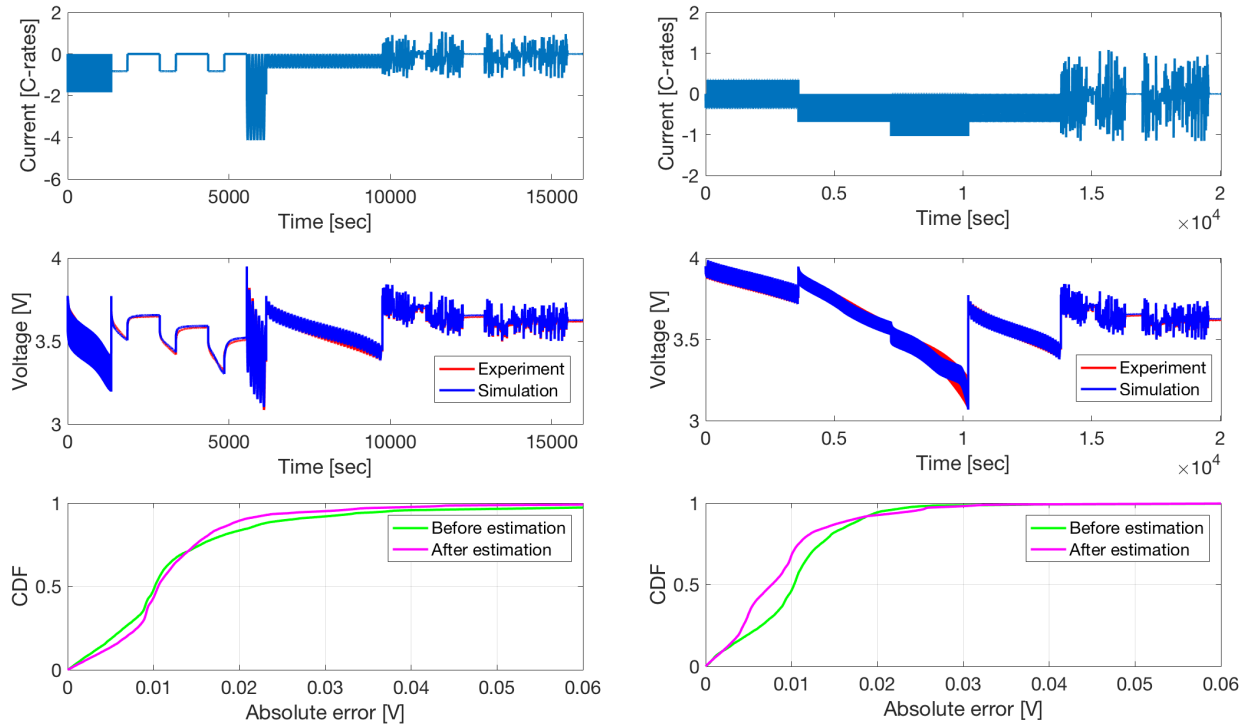
Figure 3.8 displays the parameter identification results for each group. For instance, Group 1's optimal input consists of a series of concatenated input profiles with id #339, 342, 345, 348, 365, as shown in Fig. 3.8a. We also plot the identified model's voltage prediction versus the experimentally measured voltage. Finally, the cumulative distribution function of the absolute voltage prediction error before and after applying parameter estimation are displayed in the bottom subplot. Note the most significant improvements relative to the nominal parameter values come after applying the Group 1 optimized inputs. This is intuitive, as the Group 1 parameters have the greatest sensitivity, by definition.

The final parameter estimates along with their 95% confidence intervals are listed in Table 3.11. The identified parameter values are also visualized in Fig. 3.9 on a normalized axis, along with the nominal values for comparison. The 95% confidence intervals are also visualized by error bars for each parameter, and correspond to the optimized input for that

parameter. Note that parameter bounds are obtained from the literature [35, 9, 122, 5] to ensure that the identified values are physically meaningful. The results indicate that strongly identifiable parameters (e.g. $R_s^\pm$) have narrow confidence intervals while weakly identifiable parameters (e.g. $\sigma^\pm$) have relatively wide confidence intervals. This finding is intuitive, since large variations in the weakly identifiable parameters produce trivial changes in the voltage output trajectory, as demonstrated in Fig.3.6.



(a) Identification results for Group 1 parameters. (b) Identification results for Group 2 parameters.



(c) Identification results for Group 3 parameters. (d) Identification results for Group 4 parameters.

Figure 3.8: Identification results for optimal designed inputs.

Parameter	Unit	Lower	Upper	Nominal	Result	95% C.I. [†]
D_s^-	$[\text{m}^2 \text{ s}^{-1}]$	2.25e-16	1.05e-12	3.90e-14	2.634e-14	2.407e-18
D_s^+	$[\text{m}^2 \text{ s}^{-1}]$	2.00e-16	1.00e-12	1.00e-13	6.625e-14	1.548e-17
R_s^-	$[\Omega]$	1.00e-06	1.00e-04	10.9e-06	20.235e-06	8.362e-10
R_s^+	$[\Omega]$	1.00e-06	1.00e-04	10.9e-06	17.163e-06	7.494e-10
ϵ_s^-	$[-]$	-	-	-	0.5438	-
ϵ_s^+	$[-]$	-	-	-	0.6663	-
σ^-	$[\text{S m}^{-1}]$	50	500	100	500	1.740e+03
σ^+	$[\text{S m}^{-1}]$	50	500	100	500	1.334e+02
$D_e(\cdot)$	$[\text{m}^2 \text{ s}^{-1}]$	0.5	1.5	1	1.195	1.176e-03
ϵ_e^-	$[-]$	0.18	0.45	0.3	0.289	3.687e-04
ϵ_e^{sep}	$[-]$	0.45	0.5	0.5	0.468	5.801e-04
ϵ_e^+	$[-]$	0.18	0.33	0.3	0.307	5.523e-04
$\kappa(\cdot)$	$[\text{S m}^{-1}]$	0.5	1.5	1	1.398	1.308e-03
t_c^0	$[-]$	0.36	0.363	0.363	0.36	5.585e-04
$\frac{d \ln f_{e/a}}{d \ln c_e}(\cdot)$	$[-]$	0.5	1.5	1	0.573	9.971e-04
k^-	$[(\text{A m}^{-2})(\text{m}^3 \text{ mol}^{-1})^{(1+\alpha)}]$	7.5e-05	7.5e-03	7.5e-04	7.5e-05	4.457e-07
k^+	$[(\text{A m}^{-2})(\text{m}^3 \text{ mol}^{-1})^{(1+\alpha)}]$	2.3e-04	2.3e-02	2.3e-03	2.3e-04	4.614e-06
R_f^-	$[\Omega \text{ m}^2]$	1.0e-05	1.0e-03	5.0e-04	8.719e-05	6.773e-06
R_f^+	$[\Omega \text{ m}^2]$	1.0e-04	1.0e-03	1.0e-03	4.619e-04	6.963e-06
$n_{\text{Li},s}$	$[\text{mol}]$	-	-	-	0.1406	-
$c_e(x, 0)$	$[\text{mol m}^{-3}]$	500	1500	1000	1500	8.191e-01

[†]Confidence interval calculated at the end of identification process in Fig. 3.8d

Table 3.11: Final identification results for NCA 18650 Li-ion battery.

3.1.6 Validation

In this section, the proposed optimal experimental design (OED) is tested on various input profiles not used for identification, including driving cycles, sinusoidal inputs, and pulse inputs. In order to investigate the performance of the proposed approach, a “conventional” experimental design approach is considered for comparison. It consists of 7 standard constant charge / discharge profiles, along with a driving cycle. Unlike the proposed approach, the conventional approach does not group parameters, nor design inputs to maximize identifiability. All parameters of interest are identified from concatenating experimental data. The Levenberg-Marquardt algorithm is also utilized to fit parameter values. The parameters are fit using an “all-in-one” approach, meaning they are not grouped nor fit sequentially. All parameters are fit simultaneously.

Figure 3.10a and 3.10b compare the proposed and conventional approach to experimental constant discharge data with respect to battery capacity. These figures demonstrate that our proposed approach outperforms the conventional approach in terms of voltage accuracy. Table 3.12 provides the RMSE on up to 90% of capacity range for the 0.5C and 1C discharge cases. Both experimental design approaches improve the model accuracy compared to nominal parameters. However, our proposed approach outperforms the conventional approach. Since

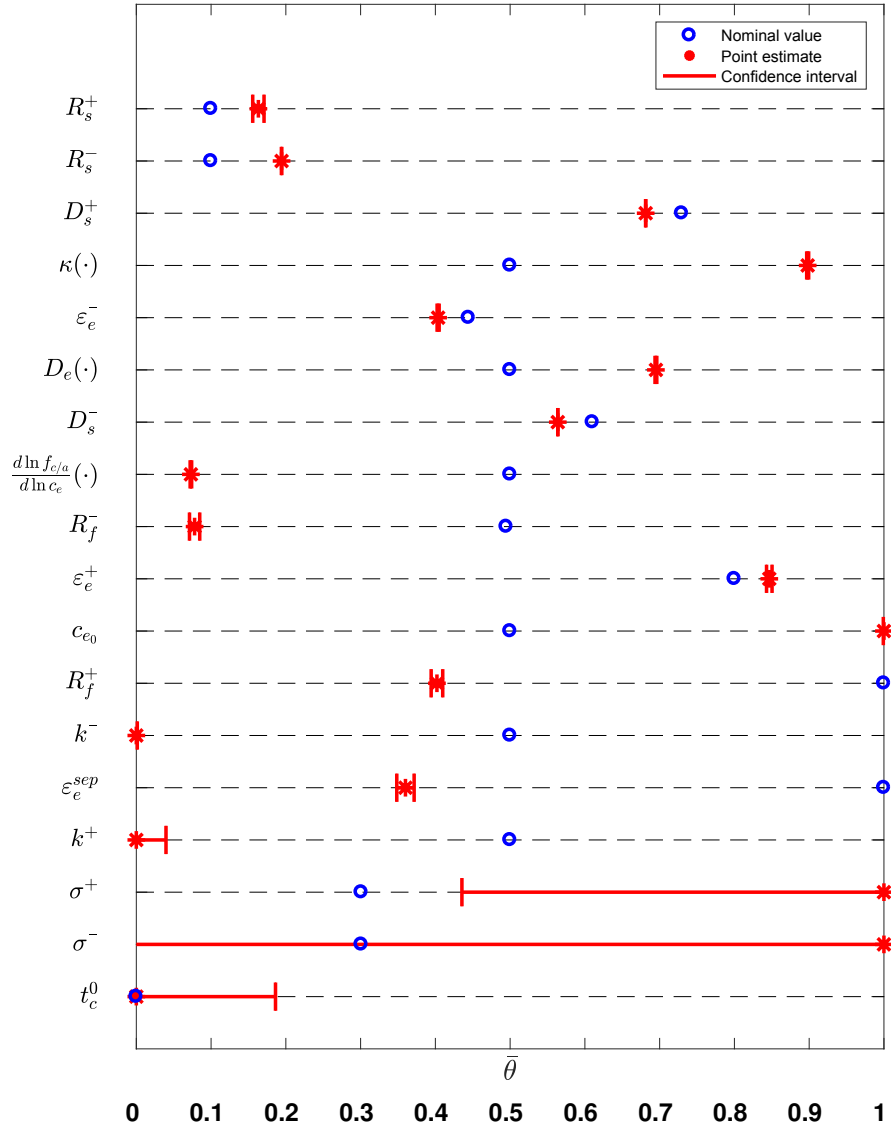


Figure 3.9: Parameter estimates and 95% confidence intervals versus the nominal parameter values.

we do not consider thermal parameters in this paper, the voltage prediction accuracy decreases as temperature evolves away from 25°C. Specifically, constant discharge at 1C is enough to produce non-trivial heat generation and cause surface temperature to nearly reach 30°C. The voltage prediction in the extremely low SOC region, such as below 3V, is difficult to match with experimental data. In this region voltage drops dramatically and we have insufficient data sampling granularity to accurately identify the equilibrium structure. On the other hand, battery cells rarely operate in this region in practice.

Besides constant discharge tests, validation is also conducted on sinusoidal and pulse inputs to assure that the OED approach is able to capture a comprehensive set of input profiles. Figure 3.10c and 3.10d show the applied current profile and output voltage for sample sinusoidal and pulse inputs profiles. The voltage accuracy is calculated and summarized in Table 3.13. Again, the OED approach outperforms the conventional approach.

Lastly, we use the driving cycle inputs described in Table.3.8 for validation. These driving cycles originate from automotive fuel emission testing. The RMSE calculations are summarized in Table. 3.14. The results show that the OED approach outperforms the conventional approach in every case.

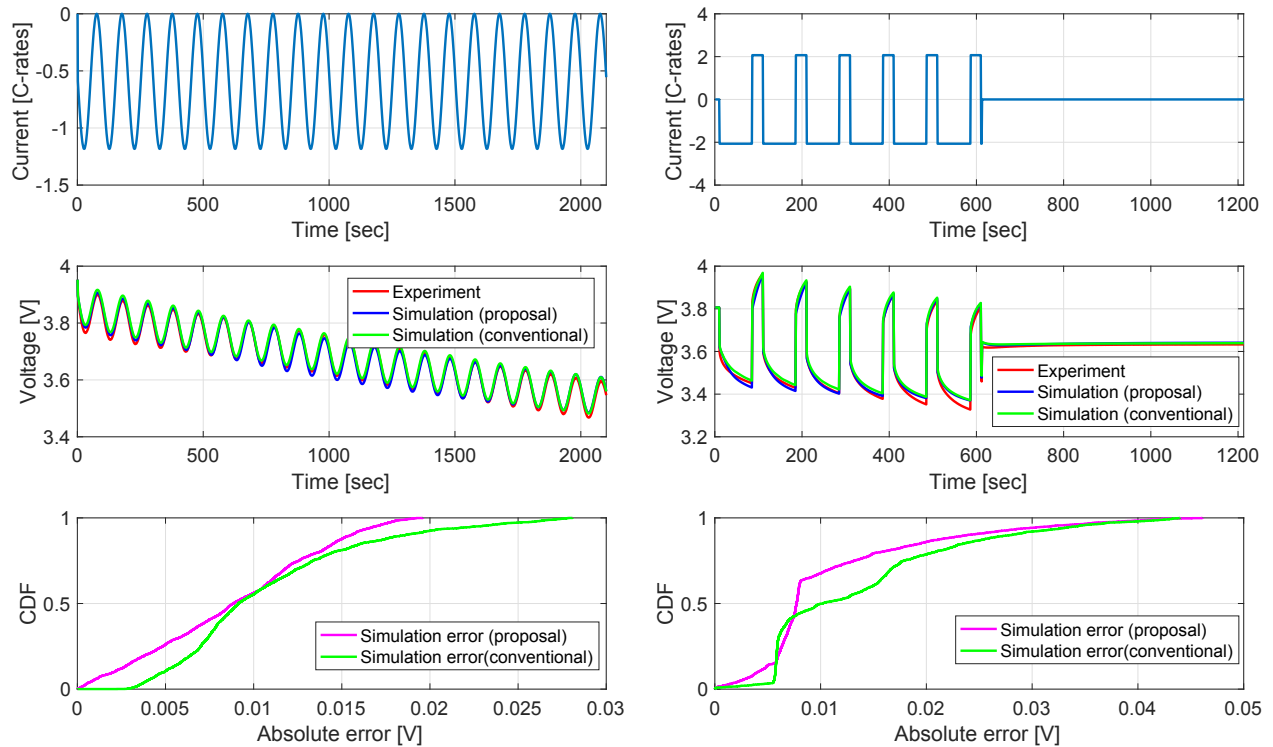
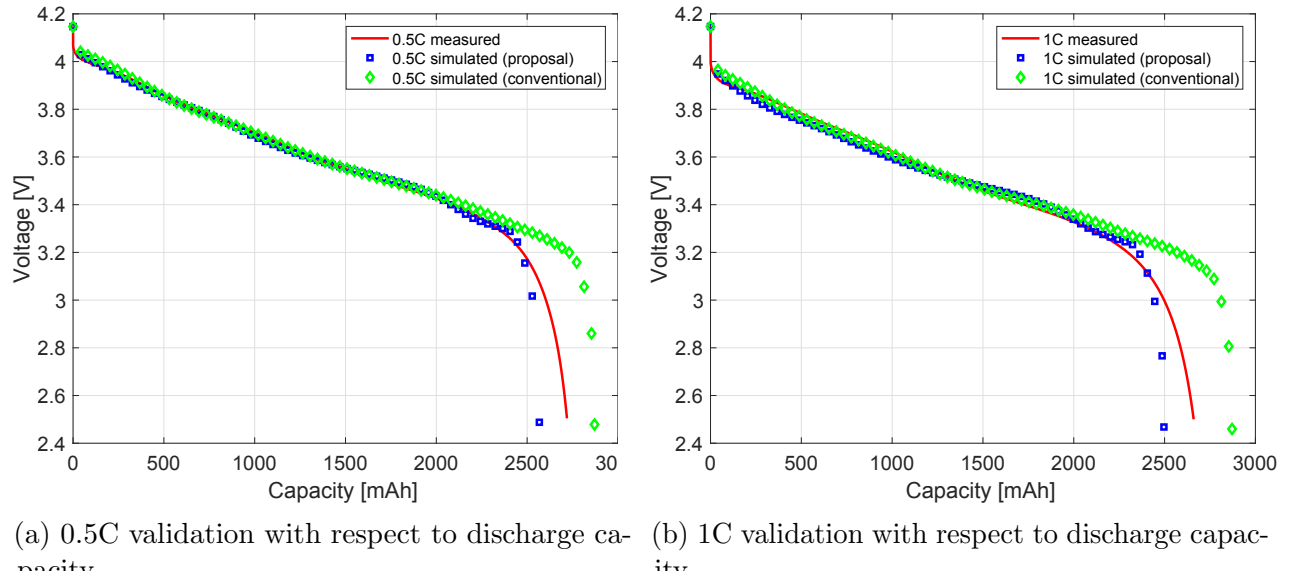


Figure 3.10: Experimental validation results for the NCA 18650 battery with testing profiles.

Parameters	0.5C discharge	1C discharge
Nominal parameters	33.7 mV	55.7 mV
Conventional approach	19.9 mV	36.1 mV
Proposed approach	11.8 mV	25.5 mV

Table 3.12: RMSE between simulation and experimental measurement up to 90% capacity in constant discharge.

Input profiles	Errors in OED [mV]	Errors in conventional [mV]
Sinusoidal input	10.13	12.01
Pulse input	14.06	16.63

Table 3.13: RMSE between simulation and experimental measurement with sinusoidal and pulse inputs.

Input profiles	Errors in OED [mV]	Errors in conventional [mV]
DC 1	6.89	7.96
DC 2	8.49	12.50
LA 92	11.90	13.70
SC 04	14.50	17.20
UDDS	12.40	14.70

Table 3.14: RMSE between simulations and experimental measurements with driving cycle inputs.

3.1.7 Summary

In this work we have established an optimal experimental design (OED) framework for systematically identifying the parameters of an electrochemical battery model. The key novelty is optimizing the input trajectory to maximize the Fisher information matrix. Specifically, we leverage sensitivity analysis and convex optimization theory to rank parameters according to orthogonalized sensitivity and select the optimal experiments for identifying these parameters. A sensitivity-based parameter grouping enables us to focus on the most sensitive electrochemical parameters. For parameter estimation, the Levenberg-Marquardt algorithm is adopted to solve a nonlinear least square problem. Additionally, we provide statistical confidence intervals for the parameter estimates. We experimentally validate the performance of the proposed OED approach compared to conventional approach on a 18650 lithium nickel-cobalt-aluminum oxide (NCA) cell. We quantify the performance in terms of voltage accuracy with respect to experimentally measured voltage. We use a set of 9 validation profiles ranging from constant current pulses to driving cycles. The proposed approach achieves less than 15 mV RMSE in all validation scenarios, and outperforms the conventional approach in terms of voltage accuracy in all cases.

3.2 State-Of-Health (SOH) Estimation

Previous Section 3.1 focused on electrochemical parameters in the full-order electrochemical model while some of parameters are assumed to be known or directly measured. In this section, further analysis on battery state of health (SoH) estimation is conducted for reduced-order model. Specifically the moles of lithium for solid-phase and electrolyte phase are defined as indicators of battery health. This section examines estimation of lithium in the solid-phase and electrolyte-phase of electrochemical battery models, for state-of-health estimation.

3.2.1 Introduction

Electrochemical models are often reduced in complexity to balance prediction accuracy with a control-oriented structure that enables online algorithm design. The simplest electrochemical model is single particle model (SPM) where a single particle replaces the spatial distribution of active particles in each electrode, and the electrolyte concentration is treated as constant in space and time. State estimation based on SPM is conducted by using a Kalman filter in [101, 131]. Furthermore, nonlinear and backstepping state estimation techniques based on the SPM can be found in [29, 116]. The limitation of SPM is that it neglects the electrolyte dynamics, which play an important role during high charge and discharge events. For this reason, researchers consider SPM with electrolyte dynamics (SPMe) to understand the internal behavior more accurately than SPM. A state-of-the-art backstepping state estimation approach based on SPM can be found in [79]. Tanim et al. [117] augments the SPM with Arrhenius relations for the model parameters to account for temperature variation, abbreviated as SPM_{eT}. In addition, the author uses Luenberger state estimator based on this model [118]. While SOC estimation can be considered as a state estimation design problem, the SOH estimation problem is not clearly defined. There are many aging mechanisms to describe degradation [11], and they can be difficult to integrate with the electrochemical models and validate experimentally.

SOH estimation for electrochemical models is still a nascent research topic in the literature. One of the most dominant sources for the loss of cell capacity is the loss of cyclable lithium induced by anodic solid-electrolyte interphase (SEI) layer growth [63, 26]. SEI layer growth was found to be the dominant cause of capacity fade during the first 200 cycles (also known as the acceleration stage) [102]. Side reactions, such as SEI layer growth and lithium plating/stripping, can alter the amount of lithium/lithium-ions in either the solid-phase ($n_{\text{Li},s}$) or electrolyte-phase ($n_{\text{Li},e}$). Many previous works have addressed capacity estimation by tracking model parameters [58, 45, 133], but very few have focused on battery SOH estimation by estimating the moles of cyclable lithium. Previously, an adaptive PDE observer coupled with a least-squares algorithm was proposed to identify solid-phase lithium as well as internal resistance [77]. The extended Kalman filter (EKF) was also employed to estimate the amount of cyclable Li-ions as an unknown model parameter using a single particle model [136]. In [46], the total number of lithium in solid-phase in a reduced SPM was estimated though estimation of the states. Although these works illuminate the importance of cyclable lithium, none of

the aforementioned literature (i) simultaneously estimates solid-phase and electrolyte-phase lithium, and (ii) mathematically analyzes the sensitivities of these two parameters to the output voltage.

In light of the previously highlighted works, this section addresses these challenges by proposing a parameter estimation framework driven by parameter sensitivity analysis for a reduced-order electrochemical model. In particular, this work is motivated by the authors' previous work [77, 79, 86]. First, we improve the output voltage prediction accuracy by considering state-dependent parameters, i.e. conductivity and the activity coefficient in the electrolyte dynamics. The performance of the proposed reduced-order model is demonstrated by comparing it against a full-order model and the previous reduced-order model in [79]. Secondly, we formulate the SOH estimation problem as a parameter estimation of cyclable lithium in both the solid-phase and electrolyte-phase of an electrochemical model. Drifting SOH parameters leads to voltage deviations compared to a fresh cell, which imply capacity and power fade. Lastly, we construct a parameter estimation scheme using sensitivity analysis that exploits the choice of cyclable lithium as the unknown parameters. We verify the proposed parameter estimation technique in the simulation.

3.2.2 The State of Health (SOH) for Electrochemical Model

The *moles of lithium* in the solid phase and electrolyte phase impact Li-ion battery state of health (SOH). They are defined as:

$$n_{Li,s}(t) = \sum_{j \in \{+, -\}} \frac{\varepsilon_s^j L^j A}{\frac{4}{3} \pi (R_s^j)^3} \int_0^{R_s^j} 4\pi r^2 c_s^j(r, t) dr. \quad (3.47)$$

$$n_{Li,e}(t) = \sum_{j \in \{-, sep, +\}} \varepsilon_e^j A \int_{0^j}^{L^j} c_e^j(x, t) dx. \quad (3.48)$$

These quantities are initially determined during the manufacturing process by the Li metal in the active materials and lithium ions dissolved in the electrolyte. In realistic scenarios, the moles of lithium are not conserved due to aging mechanisms, e.g. SEI layer growth, Li-plating, and solvent oxidation. The loss of total lithium contained in the solid phase is known to cause capacity fade, and the loss of total lithium in the electrolyte phase contributes to power fade. For instance, the change of total lithium in the solid-phase results in different trajectories of open circuit voltage (OCV) during the operation. Figure 3.11 describes two different OCV trajectory when the total moles of lithium has changed.

Figure 3.12 illustrates this phenomenon, where we have reduced the amount of cyclable lithium for each phase. The moles of lithium is decreased by 10% to represent an aged cell, and compared with the fresh cell in simulation. Mathematically, these changes only affect the initial conditions of the electrochemical states. The battery is fully discharged to the minimum voltage threshold using constant current in order to analyze the capacity fade in Fig. 3.12a, while high C-rate pulse current is applied to the battery for power fade

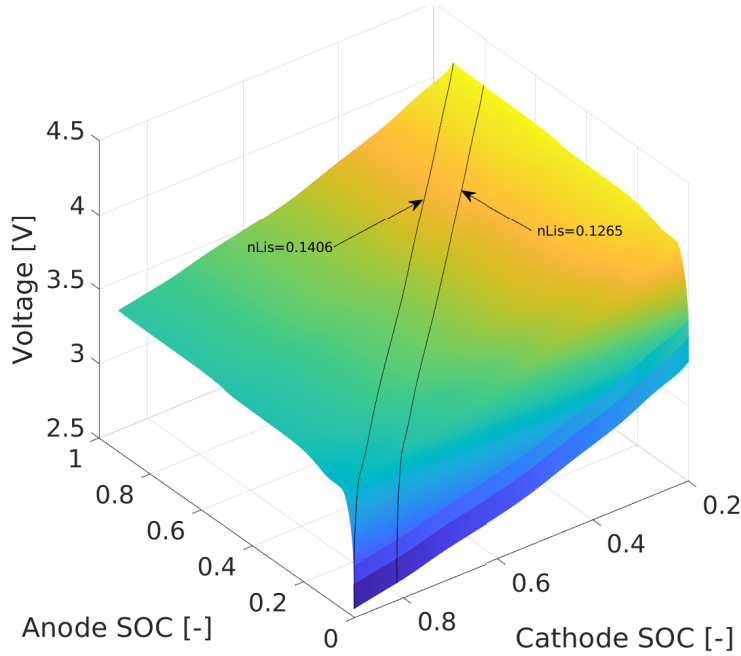


Figure 3.11: The OCV trajectory for given mole of lithium in solid-phase. The solid line describes the trajectories of OCV curve.

demonstration in Fig. 3.12b. We observe that *the moles of lithium* contribute to degrading state of health in Li-ion batteries.

In order to estimate these SOH parameters, we first combine the system dynamics (2.36) – (2.39) with the algebraic equations (3.47) – (3.48), which results in differential algebraic equations (DAEs),

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{z}, u), \quad \mathbf{x}(t_0) = \mathbf{x}_0, \quad (3.49)$$

$$\mathbf{0} = \mathbf{g}(\mathbf{x}, \mathbf{z}, u), \quad \mathbf{z}(t_0) = \mathbf{z}_0, \quad (3.50)$$

$$\mathbf{y} = \mathbf{h}(\mathbf{x}, \mathbf{z}, u), \quad (3.51)$$

after discretizing (2.36) – (2.39) and (3.47) – (3.48) in space via a suitable methods, e.g. finite differences, Padé approximation, spectral methods, etc. [37, 34]. Denote $\mathbf{x} = [c_s^-, c_s^+, c_e]^T \in \mathbb{R}^{n_x}$ as state vectors, $\mathbf{z} = [n_{\text{Li,s}}, n_{\text{Li,e}}]^T \in \mathbb{R}^{n_z}$ as algebraic variables, $\mathbf{y} = V(t)$ as output variable defined in (2.46). The system is a semi-explicit DAE of index 1 as $[\partial \mathbf{g} / \partial \mathbf{z}]^{-1}$ exists and the SOH parameters, $\mathbf{z}$, only appear in the initial conditions of time-derivative states $\mathbf{x}_0$.

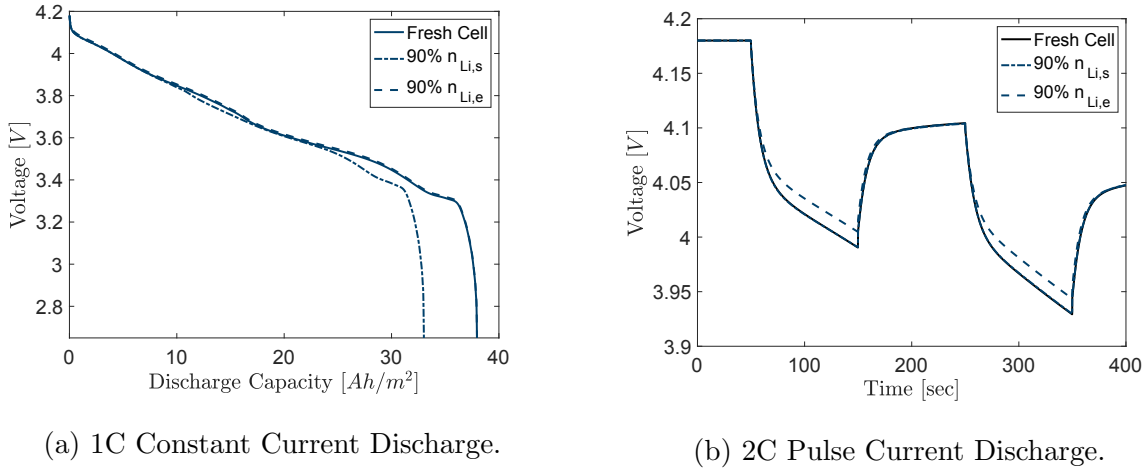


Figure 3.12: The impact of SOH parameters $n_{Li,s}$, $n_{Li,e}$ on voltage output. Top: A decrease of $n_{Li,s}$ decreases capacity, i.e., capacity fade. Bottom: a decrease of $n_{Li,e}$ affects the instantaneous voltage, which contributes to power fade.

3.2.3 SOH Parameter Estimation

This section derives estimators for SOH parameters $n_{Li,s}$, $n_{Li,e}$ by leveraging the proposed SPM model in Section 2.2.2 and (3.49)-(3.51).

Sensitivity Analysis

Sensitivity analysis is used to understand how a model's output depends on variations in parameter values, or initial conditions [86]. For time-continuous dynamic systems, the local sensitivities are defined as the first-order partial derivatives of the system output with respect to the parameters around nominal parameter values. In this section, we briefly introduce how to derive local sensitivity of SOH parameters in dynamical system described in (3.49)-(3.51), and develop this approach toward an SOH parameter estimation framework.

Let's define sensitivity variables as follows:

$$S_{\mathbf{x}} = \frac{\partial \mathbf{x}}{\partial \mathbf{z}}, \quad S_{\mathbf{y}} = \frac{\partial \mathbf{y}}{\partial \mathbf{z}}, \quad (3.52)$$

where $S_{\mathbf{x}}$ and $S_{\mathbf{y}}$ are sensitivity matrices of the local sensitivity vector where $s_{i,j}$ is defined as the partial derivative of the i -th state to the j -th algebraic variable such as:

$$s_{i,j}(t) = \frac{\partial x_i(t)}{\partial z_j}. \quad (3.53)$$

Then, we can formulate the sensitivity differential equations (SDEs) for the system (3.49) – (3.51) via multivariate calculus as follows:

$$\dot{S}_{\mathbf{x}} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \cdot S_{\mathbf{x}} + \frac{\partial \mathbf{f}}{\partial \mathbf{z}}, \quad S_{\mathbf{x}}(0) = S_{\mathbf{x}0}, \quad (3.54)$$

$$S_{\mathbf{y}} = \frac{\partial \mathbf{h}}{\partial \mathbf{x}} \cdot S_{\mathbf{x}} + \frac{\partial \mathbf{h}}{\partial \mathbf{z}}, \quad S_{\mathbf{y}}(0) = S_{\mathbf{y}0}, \quad (3.55)$$

where

$$0 = \frac{\partial \mathbf{g}}{\partial \mathbf{x}} \cdot S_{\mathbf{x}}(0) + \frac{\partial \mathbf{g}}{\partial \mathbf{z}}. \quad (3.56)$$

Note that the sensitivity dynamic equations are formulated as ordinary differential equations (ODEs), not differential algebraic equations (DAEs) because the sensitivity vectors are obtained by differentiating the states and outputs with respect to the algebraic variables. It is important to notice that the algebraic equations (3.56) are only used for sensitivity initial conditions, $S_{\mathbf{x}}(0)$ as the algebraic variables only appear in the initial conditions of the states in (3.50). This is key difference compared to the sensitivity analysis discussed in Section 3.1. The advantage of SDEs lies in fundamental mathematical derivation and computational efficiency compared to a perturbation method whose sensitivity is obtained by perturbing each parameter slightly, finding the corresponding initial conditions, and then calculating the output difference with nominal parameters. Notice that SDEs are time-varying ODEs where Jacobians of state transition functions, algebraic equations with respect to the states, and algebraic variables should be computed at each time step. Such intricate ODEs encourage researchers to utilize automatic differentiation tools, such as CasADi [3], which efficiently computes the first and second-order derivatives of this intricate model. In this work, the battery model DAEs and its SDEs are simulated by using the IDAS/CVODES integrator provided by SUNDIALS via CasADi interfaces [43].

Nonlinear Least Squares

The SOH parameter estimation problem can be formulated as a nonlinear optimization problem, i.e., nonlinear least squares. The objective is to minimize the output voltage difference between measured output, $\mathbf{y}$, and predicted output, $\hat{\mathbf{y}}$:

$$\min_{\hat{\mathbf{z}}} \sum_{t=0}^{t_f} [\mathbf{y}(t) - \hat{\mathbf{y}}(t; \hat{\mathbf{z}})]^2. \quad (3.57)$$

The Levenberg-Marquardt algorithm is used to update the parameters $\hat{\mathbf{z}}$ iteratively by solving the nonlinear optimization problem (3.57) above. This algorithm adaptively chooses parameter updates between the gradient descent update and the Gauss-Newton update [59] via equation:

$$[\mathbf{J}^T \mathbf{J} + \gamma \text{diag}(\mathbf{J}^T \mathbf{J})] h_{\hat{\mathbf{z}}} = \mathbf{J}^T (\mathbf{y} - \hat{\mathbf{y}}), \quad (3.58)$$

where $\mathbf{J} = \partial \hat{\mathbf{y}} / \partial \hat{\mathbf{z}}$ is the local sensitivity of the output $\hat{\mathbf{y}}$. Conveniently, the Levenberg-Marquardt algorithm utilizes the sensitivities derived in Section 3.2.3. The value of γ trades off between gradient descent update and Gauss-Newton update. Then the parameters are updated iteratively by

$$\hat{\mathbf{z}}_{k+1} = \hat{\mathbf{z}}_k + h_{\hat{\mathbf{z}}}. \quad (3.59)$$

Once the parameters are fit to the data, the least squares estimator provides the parameter estimation statistics evaluated at the final estimates denoted by $\underline{\mathbf{z}}$. Namely,

$$\hat{\mathbf{z}} \sim \mathcal{N}(\underline{\mathbf{z}}, \rho^2(\underline{\mathbf{z}})), \quad (3.60)$$

where ρ^2 is the asymptotic covariance matrix obtained explicitly by

$$\rho^2(\underline{\mathbf{z}}) = \sigma^2 [\mathbf{J}(\underline{\mathbf{z}})^T \mathbf{J}(\underline{\mathbf{z}})]^{-1}, \quad (3.61)$$

where $\mathbf{J}(\underline{\mathbf{z}})$ is the local sensitivity at the final estimates. The error variance σ^2 is computed as:

$$\sigma^2 = \frac{1}{N - p} \sum_{t=0}^{t_f} [\mathbf{y}(t) - \hat{\mathbf{y}}(t; \underline{\mathbf{z}})]^2, \quad (3.62)$$

where N is the number of observations and p is the number of parameters. The 95 % confidence intervals are derived by using t -distributions:

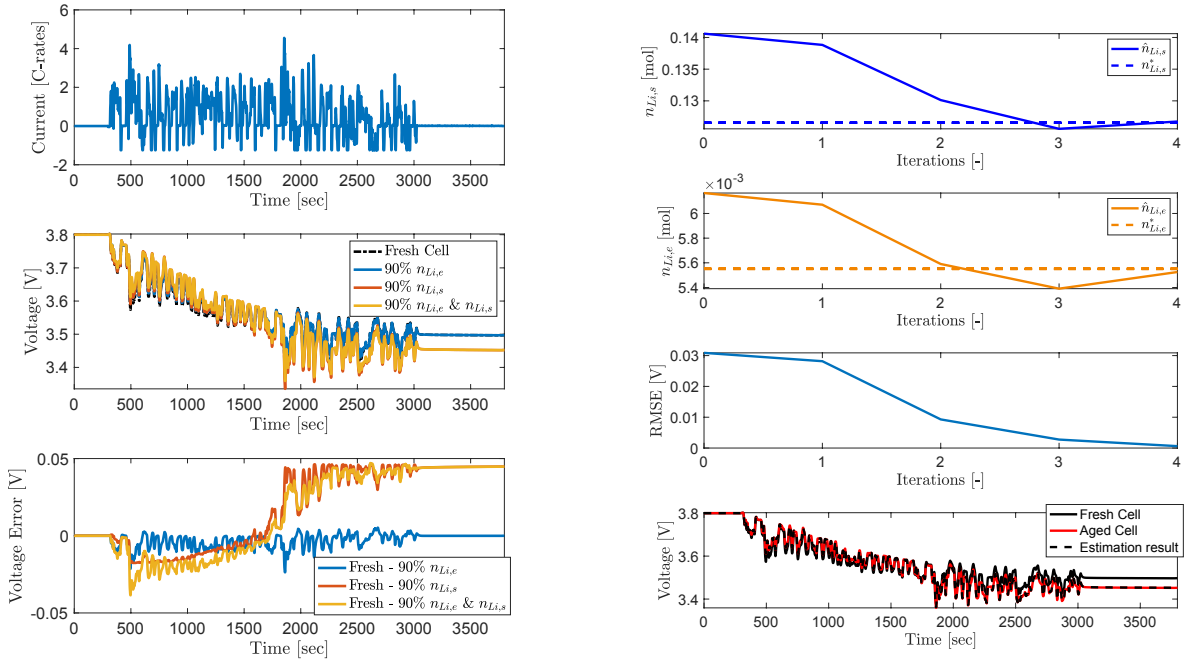
$$\underline{\mathbf{z}} - \mathbf{t}_{(1-0.025)} \frac{\rho}{\sqrt{N}} \leq \mathbf{z}^* \leq \underline{\mathbf{z}} + \mathbf{t}_{(1-0.025)} \frac{\rho}{\sqrt{N}}, \quad (3.63)$$

where $\mathbf{t}$ is the upper critical value for the t -distribution with $N - 1$ freedom [98].

3.2.4 Simulation Results

In this section, we present SOH parameter estimation results to validate the proposed approach via simulation. Suppose the aging mechanisms consume moles of lithium in solid and electrolyte phase after a number of cycles. Then the voltage starts to deviated compared to a fresh cell. Even though the exact underlying ageing mechanisms are not considered in this study, many of them ultimately lead to a decrease of these quantities. Our goal is to track these SOH parameters by nonlinear least squares using input/output measurements.

Figure 4.1a exhibits the impact of parameter changes to the output voltage. The Urban Dynamometer Driving Schedule (UDDS) input profile is considered as an input current profile applied to the enhanced SPM. One can notice that the perturbation of $n_{Li,s}$ affects the voltage variation in the whole period even after cycling, i.e. the relaxation period. This is because $n_{Li,s}$ is related to the equilibrium structure of the electrochemical model determining the window of the open-circuit voltage (OCV) curve [86]. Consequently, a change of $n_{Li,s}$ shifts



(a) The voltage output error between fresh cell and aged cell.

(b) The SOH parameter estimation results.

Figure 3.13: The simulation results on the voltage deviations as well as SOH estimation using Urban Dynamometer Driving Schedule (UDDS) input profile. The voltage RMSE reduces below 1 mV after 4 iterations using Levenberg-Marquardt algorithm.

the equilibrium states during the relaxation period. On the other hand, the $n_{Li,e}$ parameter is linked to the electrolyte concentrations by definition. This fact implies that a change of $n_{Li,e}$ influences on the level of electrolyte concentration, which alters the instantaneous voltage output during operation. Interestingly, $n_{Li,s}$ and $n_{Li,e}$ are linearly independent with each other, with respect to the output voltage variation since they shown up in separate terms of (2.47). We confirm that the voltage errors are linearly independent at the bottom in Fig. 4.1a. For instance, the voltage error (Verr) is linearly independent on each other, $Verr(90\% n_{Li,s} \& 90\% n_{Li,e}) = Verr(90\% n_{Li,s}) + Verr(90\% n_{Li,e})$. This fact makes the SOH parameter estimation problem simpler and straightforward.

In these simulations, we adopt the electrochemical parameters for a fresh cell identified in the authors' previous work [86]. Specifically, the "true" SOH parameters are set as $n_{Li,s} = 0.1406$ [mol], $n_{Li,e} = 0.0062$ [mol]. To represent the aged cell behavior, we assume that the aged cell retains 90 % moles from the fresh cell, i.e., $n_{Li,s} = 0.1265$ [mol], $n_{Li,e} = 0.0056$ [mol]. The parameter updates at each iteration compared to the true parameters (aged) are shown at the top in Fig. 4.1b. We observe that the voltage RMSE decreases every iteration by updating the SOH parameters. After 4 iterations the parameter estimates approach the correct aged

cell values. Moreover, the voltage RMSE reaches less than 1 mV. The final parameter estimates with 95% confidence interval are obtained as $\hat{n}_{Li,s} = 0.1267 \pm 2.09 \times 10^{-5}$ [mol], and $\hat{n}_{Li,e} = 0.0055 \pm 7.80 \times 10^{-6}$ [mol].

3.2.5 Summary

This section addresses state-of-health (SOH) parameter identification for an electrochemical battery model. We consider the SOH estimation problem for general electrochemical models, and propose a parameter estimation framework via sensitivity analysis. Once the output voltage measurement starts to diverge from the fresh cell behavior due to ageing mechanisms, the Levenberg-Marquardt algorithm is used to solve a nonlinear least squares problem for SOH parameter fitting. In order to validate our proposed framework, we simulate the overall process using a model-to-model comparison. These results demonstrate how the SOH parameters are monitored by proposed approach. On-going work involves validation of the proposed SOH parameter estimation framework with experimental data.

Chapter 4

Learning & Optimal Controls

The objective of battery control is to derive optimal fast-charging strategy based on the identified electrochemical model. In this chapter, we formulate battery fast charging control problem and analyze the control strategies from controls & machine learning perspectives. First of all, the analytical solution of battery fast charging strategy is discussed via model-based optimal control theory in Section 4.1. The contribution is to support the existing charging protocols via optimality of input trajectory. Secondly, reinforcement learning-based controller is proposed as a tool for fast charging control in Section 4.2. This learning-based controller is validated with other existing control strategies with several remarks.

4.1 Model-based fast-charging control

4.1.1 Introduction

Battery fast charging performs an essential role in electrified transportation to reduce driver's waiting time and range anxiety. This can be easily achieved by using aggressive current profiles, however, it also accelerates battery degradation effects, such as solid electrolyte interface layer growth, and lithium plating deposition. For this reason, batteries should be carefully monitored and controlled during fast charging as it may cause cells to crack, leak, and lose capacity. The role of battery management systems is to charge the batteries as fast as possible while not violating the safety constraints. In the literature, several model-based optimal control techniques have been proposed in consideration of providing fast-charging while guaranteeing safety constraints.

The authors in [55] formulate a minimum-time charging problem and use nonlinear model predictive control. Similarly, authors in [121] propose quadratic dynamic matrix control formulation to design an optimal charging strategy for real-time model predictive control. In the context of aging mechanism, the authors of [92] have studied the trade-off between charging speed and degradation, based on an electro-thermal-aging model. The authors in [94] consider minimizing film layer growth of the electrochemical model. Authors in [113] derive an optimal

current profile using a single particle model with intercalation-induced stress generation. The key novelty here is incorporating mechanical fracture, which can be a dominant mechanism in degradation and capacity fade. To ensure safety, a proportional–integral–derivative controller is proposed. In [65], the authors exploit differential flatness properties of the single particle model to yield a computationally efficient optimal control problem, solved via pseudospectral methods. On the other hand, the authors in [138] synthesize a state estimation and model predictive control scheme for a reduced electrochemical-thermal model, in order to design health-aware fast charging strategy. The problem is formulated as a linear time-varying model predictive control scheme, with a moving horizon state estimation framework. Similarly, authors of [99] propose reference governor approach for equivalent hydraulic battery model, which integrates state estimation with a reference governor scheme.

Nearly all existing model-based control strategies apply computational methods that yield the following: apply maximal charging current until the battery reaches one of the constraints, and then ride this constraint. There are a number of research papers showing this rule empirically. However, a theoretical analysis that derives and characterizes this solution has not been executed.

To solve a deterministic optimal control problem, there are two related methods: (i) the Hamilton-Jacobi-Bellman (HJB), which is based on Bellman’s principle of optimality [2], and (ii) Pontryagin Minimum Principle (PMP), which originates from *Calculus of Variations*. The HJB approach guarantees the global optimal solution. Often, the solution to the HJB PDE is computed by the level-set method [84], and the computational complexity is exponential in the dimension of the state. This leads to intractable computation for high-dimensional systems. On the other hand, PMP provides necessary conditions for optimality. The solution of PMP is either numerically computed by, e.g. shooting methods [56], or analytically derived. With shooting methods, computing the solutions to the PMP conditions is more tractable for higher-dimensional systems than the HJB approach. Furthermore, PMP yields sufficient conditions for convex optimal control problems [41, 71]. In the literature, PMP has been widely adopted to solve various constrained optimal control problems. It was first applied to aircraft control and fuel management [16, 18], and expanded to other areas such as hybrid electric vehicle energy management [53], train operation [52], fluid structures [85], and persistent monitoring in multi-agent systems [130].

In this paper, we focus on the PMP approach to analytically derive the optimal solution for the battery fast charging problem. This allows us to explicitly relate model parameters to the optimal input and state trajectories. We examine a reduced-order electrochemical model, as this represents a physical interpretation of state constraints. Our contribution is to prove that battery fast charging, for a class of models, yields a solution that maximizes its allowable charging current until a safety constraint is active. When the safety constraint becomes active, then a feedback control law is applied to ensure safety. We derive analytical solutions from PMP and validate the proposed solutions by numerical simulation.

4.1.2 Preliminaries

This section introduces the Pontryagin Minimum Principle for optimal control problems [93, 54, 17, 41]. For a given dynamical system,

$$\dot{x} = f(x(t), u(t), t), \quad x(0) = x_0, \quad (4.1)$$

where $x(t) \in \mathbb{R}^n$ and $u(t) \in \mathbb{R}^m$, and $f : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \rightarrow \mathbb{R}^n$ is assumed to be continuously differentiable in its arguments. The control input $u(t)$ is *admissible* in a given time horizon if

$$u(t) \in \Omega(t) \subset \mathbb{R}^m, \quad t \in [0, t_f]. \quad (4.2)$$

The objective function is defined as

$$\mathcal{J} = h(x(t_f), t_f) + \int_0^{t_f} g(x(t), u(t), t) dt, \quad (4.3)$$

where g is the running cost, $g : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \rightarrow \mathbb{R}$ and h is terminal cost, $h : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}$. Both functions are continuously differentiable and t_f denotes the terminal time. The objective of the optimal control problem is to design $u(t)$ to minimize the overall cost, mathematically,

$$\min_{u(\cdot)} \mathcal{J} : \dot{x} = f(x(t), u(t), t), x(0) = x_0, u(t) \in \Omega(t), \quad (4.4)$$

the optimal control problem is to find an admissible control u^* , which minimizes the objective function (4.3) subject to the state equation and the associated constraints. The control u^* is called an optimal control and x^* is called the *optimal trajectory*.

Unconstrained Optimal Control

Without constraints on the inputs and states, we first derive the necessary conditions for (4.4). First, the Hamiltonian function, $\mathcal{H} : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}$ is defined as:

$$\mathcal{H}(x, u, p, t) \triangleq g(x, u, t) + p^\top f(x, u, t), \quad (4.5)$$

where $p \in \mathbb{R}^n$ is called the adjoint vector or the costate vector. The necessary conditions for u^* to be an optimal control are obtained by *Calculus of Variations*: for all $u \in \Omega(t), t \in [0, T]$

$$\dot{x}^*(t) = \frac{\partial \mathcal{H}}{\partial p}(x^*(t), u^*(t), p(t), t), \quad (4.6)$$

$$\dot{p}(t) = -\frac{\partial \mathcal{H}}{\partial x}(x^*(t), u^*(t), p(t), t), \quad (4.7)$$

$$0 = \frac{\partial \mathcal{H}}{\partial u}(x^*(t), u^*(t), p(t), t), \quad (4.8)$$

with corresponding transversality or boundary conditions,

$$0 = \frac{\partial h}{\partial x}(x^*(t_f), t_f) - p(t_f), \quad (4.9)$$

$$0 = \mathcal{H}(x^*(t_f), u^*(t_f), p(t_f), t_f) + \frac{\partial h}{\partial t}(x^*(t_f), t_f). \quad (4.10)$$

With eqs. (4.7) and (4.9), we can determine the costate trajectory. Note that $p(t_f) = 0$ when the terminal state is free or the terminal cost function $h(\cdot)$ is zero. The conditions for optimality are also sufficient if $\mathcal{H}$ is convex in (x, u) for each $(p, t) \in \mathbb{R}^n \times [0, T]$ and $h(x, t_f)$ is convex in x [41, 71].

Constrained Optimal Control

Most control systems have constraints on inputs and states inherently. Based on the Hamiltonian function in (4.5), we add some of features of constraint functions to construct the constrained optimal control problem. First of all, the input $u(t)$ is *admissible* if it is piecewise continuous and satisfies the mixed constraints, such as

$$l(x(t), u(t), t) \leq 0, \quad t \in [0, t_f], \quad (4.11)$$

where $l : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \rightarrow \mathbb{R}^q$ is continuously differentiable. In addition, we can consider pure state variable inequality constraints, namely,

$$c(x(t), t) \leq 0, \quad t \in [0, t_f], \quad (4.12)$$

where $c : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^{n_c}$. For example, suppose states are required to be positive, $x(t) \geq 0$. In any interval where $x(t) = 0$, we must have $\dot{x}(t) \geq 0$ so that x does not become negative. Lastly, the terminal state is constrained by inequality and equality constraints, namely

$$a(x(t_f), t_f) \leq 0, \quad (4.13)$$

$$b(x(t_f), t_f) = 0, \quad (4.14)$$

where $a : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^{n_a}$ and $b : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^{n_b}$ are continuously differentiable. The Lagrange function, $\mathcal{L} : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n \times \mathbb{R}^q \times \mathbb{R}^{n_c} \times \mathbb{R} \rightarrow \mathbb{R}$ is defined as:

$$\mathcal{L}(x, u, p, \mu, \eta, t) \triangleq \mathcal{H}(x, u, p, t) + \mu^\top l(x, u, t) + \eta^\top c(x, t), \quad (4.15)$$

where μ and η are Lagrange multipliers. When convenient, we omit the time argument for readability. Since the constraints are adjoined directly to form the Lagrangian, this method is called direct adjoining method. The Lagrange multipliers satisfy the complimentary slackness conditions, namely,

$$\mu(t) \geq 0, \quad \mu(t)^\top l(x, u, t) = 0, \quad (4.16)$$

$$\eta(t) \geq 0, \quad \eta(t)^\top c(x, t) = 0. \quad (4.17)$$

Pontryagin's Minimum Principle allows for a jump in $p(t)$ at a point in time when the state $x(t)$ enters its constraint boundary. The jump must satisfy the conditions

$$p(\tau^-) = p(\tau^+) + \zeta(\tau) \frac{\partial c}{\partial x}(x^*(\tau), \tau), \quad (4.18)$$

where $\zeta(\tau) \geq 0$ and

$$\begin{aligned} \mathcal{H}(x^*(\tau), u^*(\tau^-), p(\tau^-), \tau) = \\ \mathcal{H}(x^*(\tau), u^*(\tau^+), p(\tau^+), \tau) - \zeta(\tau) \frac{\partial c}{\partial t}(x^*(\tau), \tau), \end{aligned} \quad (4.19)$$

at a time τ at which one of the state variables has just reached its boundary value. An instant τ is called *entry time* if there is an interior interval ending at $t = \tau$ and a boundary interval starting at τ . Similarly τ is called an *exit time* if a boundary interval ends and an interior interval starts. If the trajectory just touches the boundary, then τ is called *contact time*. Taken together, *entry*, *exit*, *contact* times are called *junction times*.

The necessary conditions for the optimality of u^* by using the *Calculus of Variations* [54] are obtained as follows:

$$\dot{x}^*(t) = \frac{\partial \mathcal{L}}{\partial p}(x^*(t), u^*(t), p(t), \mu(t), \eta(t), t), \quad (4.20)$$

$$\dot{p}(t) = -\frac{\partial \mathcal{L}}{\partial x}(x^*(t), u^*(t), p(t), \mu(t), \eta(t), t), \quad (4.21)$$

$$0 = \frac{\partial \mathcal{L}}{\partial u}(x^*(t), u^*(t), p(t)), \quad (4.22)$$

with corresponding transversality conditions,

$$\begin{aligned} 0 = \frac{\partial h}{\partial x}(x^*(t_f), t_f) + \alpha \frac{\partial a}{\partial x}(x^*(t_f), t_f) + \beta \frac{\partial b}{\partial x}(x^*(t_f), t_f) \\ + \gamma \frac{\partial c}{\partial x}(x^*(t_f), t_f) - p(t_f), \quad \alpha \geq 0 \end{aligned} \quad (4.23)$$

$$0 = \gamma^\top c(x^*(t_f), t_f), \quad \gamma \geq 0, \quad (4.24)$$

the complimentary slackness conditions hold

$$\mu(t) \geq 0, \quad \mu(t)^\top l(x^*, u^*, t) = 0, \quad (4.25)$$

$$\eta(t) \geq 0, \quad \eta(t)^\top c(x^*(t), t) = 0, \quad (4.26)$$

$$\zeta(\tau) \geq 0, \quad \zeta(\tau) c(x^*(\tau), \tau) = 0, \quad (4.27)$$

and jump conditions (4.18)–(4.19) should hold at any entry/contact time τ . Similar to the unconstrained case, if the Hamiltonian is convex with respect to (x, u) , then the conditions are also sufficient [41, 71].

4.1.3 Fast Charging Optimal Solution

SPM model discretization

There are numbers of techniques to discretize PDE (2.28) – (2.29) in the literature [19, 34, 76, 73, 37]. Among them, we apply Padé approximation [37] to the solid-phase diffusion

PDE (2.1). We briefly explain Padé approximation next, since the optimal control solution is based on this discretized model. By taking the Laplace transform from input j_n to $c_s^\pm(R_s^\pm, s)$, the PDE (2.28)–(2.31) is written as a transcendental transfer function:

$$G(s) = \frac{c_s^\pm(R_s^\pm, s)}{j_n(s)} = \frac{R_s^\pm}{D_s^\pm} \frac{\sinh\left(\sqrt{\frac{s}{D_s^\pm}} R_s^\pm\right)}{R_s^\pm \sqrt{\frac{s}{D_s^\pm}} \cosh\left(\sqrt{\frac{s}{D_s^\pm}} R_s^\pm\right) - \sinh\left(\sqrt{\frac{s}{D_s^\pm}} R_s^\pm\right)}. \quad (4.28)$$

The third-order Padé approximation of (4.28) is:

$$G(s) \approx \frac{\frac{3}{R_s^\pm} + \frac{4R_s^\pm}{11D_s^\pm}s + \frac{R_s^{\pm 3}}{165D_s^{\pm 2}}s^2}{s\left(1 + \frac{3R^2}{55D_s}s + \frac{R^4}{3465D_s^2}s^2\right)}. \quad (4.29)$$

Then, the controllable canonical state-space form is:

$$\begin{bmatrix} \dot{c}_{s1}^\pm \\ \dot{c}_{s2}^\pm \\ \dot{c}_{s3}^\pm \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -\frac{3465D_s^{\pm 2}}{R_s^{\pm 4}} & -\frac{189D_s^\pm}{R_s^{\pm 2}} \end{bmatrix} \begin{bmatrix} c_{s1}^\pm \\ c_{s2}^\pm \\ c_{s3}^\pm \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{3465D_s^{\pm 2}}{R_s^{\pm 4}} \end{bmatrix} j_n^\pm \quad (4.30)$$

$$c_s^\pm(R_s^\pm, s) = \left[\frac{3}{R_s^\pm} \quad \frac{4R_s^\pm}{11D_s^\pm} \quad \frac{R_s^{\pm 3}}{165D_s^{\pm 2}} \right]. \quad (4.31)$$

Lastly, we can derive the state-space realization where the bulk concentration, $\bar{c}_s^- = SOC_n \cdot c_{s,\max}^-$ is expressed as a state in a Jordan-form. We perform *linear transformation* such that the system matrix has diagonal form,

$$\dot{x} = Ax + Bu,$$

where

$$A = \begin{bmatrix} a_1 & 0 & 0 \\ 0 & a_2 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}, \quad (4.32)$$

and we define surface concentration, $c_{ss}^- = C^\top x$, as an output function of the states, $x = [x_1, x_2, x_3]^\top$, and bulk concentration for the anode is a state, $\bar{c}_s^- = x_3$.

$$\bar{c}_{ss}^- = \left[\frac{3}{R_s^-} \quad \frac{4R_s^-}{11D_s^-} \quad \frac{(R_s^-)^3}{165(D_s^-)^2} \right] x = C^\top x. \quad (4.33)$$

Table 4.1 shows the values of the A, B, C matrices for several common Li-ion chemistries. Note that the sign of the elements are consistent, which is important for generalizing the optimal control solutions across chemistries.

	LCO [†]	NCA [‡]	NMC ^{‡‡}
a_1	-7.3E-02	-1.2E-02	-3.4E-01
a_2	-8.9E-03	-1.47E-03	-4.2E-02
B	$\begin{bmatrix} 6.5\text{E-}07 \\ -8.0\text{E-}08 \\ -1.7\text{E-}01 \end{bmatrix}$	$\begin{bmatrix} 2.4\text{E-}05 \\ -3.0\text{E-}06 \\ -3.2 \end{bmatrix}$	$\begin{bmatrix} 2.2\text{E-}07 \\ -2.7\text{E-}08 \\ -1.2\text{E-}01 \end{bmatrix}$
C	$\begin{bmatrix} -1.3\text{E+}06 \\ 1.5\text{E+}06 \\ 1 \end{bmatrix}$	$\begin{bmatrix} -6.8\text{E+}05 \\ 7.7\text{E+}05 \\ 1 \end{bmatrix}$	$\begin{bmatrix} -2.9\text{E+}06 \\ 3.3\text{E+}06 \\ 1 \end{bmatrix}$

[†]Lithium Cobalt Oxide [81], [‡] Nickel Cobalt Aluminum Oxide [86], ^{‡‡} Nickel Manganese Cobalt Oxide.

Table 4.1: A, B, and C matrices for three common battery chemistries.

PMP analysis

In this section, we derive the optimal solution for the fast charging problem. First, we start with the Single Particle Model (SPM) optimal control problem formulated as follows:

$$\begin{aligned}
& \max_{u(t)} \int_{t_0}^{t_f} x_3(t) dt & (4.34) \\
& \text{subject to} \quad \dot{x}_1 = a_1 x_1 + b_1 u, & x_1(0) = x_{10}, \\
& \quad \dot{x}_2 = a_2 x_2 + b_2 u, & x_2(0) = x_{20}, \\
& \quad \dot{x}_3 = b_3 u, & x_3(0) = x_{30}, \\
& \quad -u_b \leq u \leq u_b, \\
& \quad C^\top x \leq C_b,
\end{aligned}$$

where $-u_b$ represents the maximum charging rate, and the state constraint places an upper bound on the surface concentration in (4.33), denoted by C_b . The surface concentration cannot exceed its maximum concentration level, $c_{s,max}^-$, which is determined by the electrode's electrochemical properties. The objective is to maximize the bulk concentration of the anode, in a fixed time horizon, in the presence of input and state constraints. Note that $h(x(t_f), t_f) = a(x(t_f), t_f) = b(x(t_f), t_f) = 0$ in (4.3), (4.13), (4.14). Our objective is to characterize the battery fast charging problem by using Pontryagin Minimum Principle. The necessary optimal conditions discussed in Section 4.1.2 are computed as follows. The Lagrangian is:

$$\begin{aligned}
\mathcal{L} = & x_3 + p^\top (Ax + Bu) \\
& + \mu_1(u + u_b) + \mu_2(u_b - u) + \eta(C_b - C^\top x).
\end{aligned} \tag{4.35}$$

where $p = [p_1, p_2, p_3]^\top$ is the co-state vector, and μ_1, μ_2, η are Lagrange multipliers associated with input and state constraints. Note that μ_1, μ_2 do not affect the sign of $\mathcal{L}$ due to the

complimentary slackness. To find the optimal control solution, we focus on the part of $\mathcal{L}$ that depends on u

$$\tilde{\mathcal{L}} = (b_1 p_1 + b_2 p_2 + b_3 p_3) u + (\mu_1 - \mu_2) u, \quad (4.36)$$

The costate dynamics (4.21) and input optimality (4.22) are computed as

$$\dot{p}_1(t) = -a_1 p_1 + \eta c_1, \quad (4.37)$$

$$\dot{p}_2(t) = -a_2 p_2 + \eta c_2, \quad (4.38)$$

$$\dot{p}_3(t) = -1 + \eta c_3, \quad (4.39)$$

$$0 = p_1 b_1 + p_2 b_2 + p_3 b_3 + \mu_1 - \mu_2. \quad (4.40)$$

Next, we state the two main results of this paper. They describe the optimal solution with inactive constraints and active constraints, which respectively give *Bang* and *Bang-Ride* solutions. First of all, the state trajectory for maximum charging current can be derived analytically:

$$x_1(t) = e^{a_1 t} x_1(0) + \frac{b_1 u_b}{a_1} (1 - e^{a_1 t}), \quad (4.41)$$

$$x_2(t) = e^{a_2 t} x_2(0) + \frac{b_2 u_b}{a_2} (1 - e^{a_2 t}), \quad (4.42)$$

$$x_3(t) = -b_3 u_b t + x_3(0). \quad (4.43)$$

The range of C_b that yields an unconstrained optimal control solution to (4.34) is given by:

$$c_1 x_1(t_f) + c_2 x_2(t_f) + c_3 x_3(t_f) \triangleq \bar{C} < C_b. \quad (4.44)$$

Theorem 1. If $C_b > \bar{C}$, then the optimal solution to (4.34) is to maximize the current input, $-u_b$, so-called *Bang* control, and the state constraint is not active during the control horizon.

Proof. Inactive state constraint implies that $\gamma = 0$ in transversality conditions (4.23), (4.24) and the Lagrange multiplier, $\eta(\cdot) \equiv 0$. Then, we can find the final conditions of co-states $p(t)$ at terminal time by solving co-state dynamical equations in (4.37) – (4.39), such as:

$$\begin{aligned} p_1(t_f) &= p_2(t_f) = p_3(t_f) = 0, \\ \iff p_1(0) &= p_2(0) = 0, p_3(0) = t_f. \end{aligned}$$

Therefore, the co-state trajectories are:

$$p_1(t) = p_2(t) = 0, \quad p_3(t) = -t + t_f.$$

We find that (4.36) is maximized when $u = -u_b$, which implies that the optimal current is the maximum charging rate. $\square$

Theorem 2. If $C_b < \bar{C}$, then the optimal solution to (4.34) is to maximize the current input, $-u_b$, until the inequality constraint becomes active. Then it rides the constraint bound – Bang-Ride control. Furthermore, the switching time¹, σ , does not exist for $C_b < \bar{C}$ but the Junction time, τ , exists.

Proof. To prove this theorem, we consider two cases: 1) when the state constraint becomes active at the terminal time, 2) when the state constraint becomes active prior to the terminal time.

Case 1: Consider the case when the state constraint becomes active at the terminal time, which implies that $\gamma \neq 0$. Then the transversality condition (4.23) yields

$$p_1(t_f) = -c_1\gamma,$$

$$p_2(t_f) = -c_2\gamma,$$

$$p_3(t_f) = -c_3\gamma,$$

with the final conditions above, the co-state trajectories become a function of γ , such as,

$$p_1(t) = \frac{-c_1\gamma}{e^{-a_1 t_f}} e^{-a_1 t}, \quad (4.45)$$

$$p_2(t) = \frac{-c_2\gamma}{e^{-a_2 t_f}} e^{-a_2 t}, \quad (4.46)$$

$$p_3(t) = -t - c_3\gamma + t_f. \quad (4.47)$$

Plugging this into $b_1 p_1 + b_2 p_2 + b_3 p_3$ in (4.36), then the switching time occurs when the sign of (4.36) changes. That is, the optimal current input switches from $-u_b$ to $+u_b$. The existence of a switching time σ is equivalent to the existence of a root of $b_1 p_1 + b_2 p_2 + b_3 p_3 = 0$ w.r.t. t , namely:

$$\gamma \left(\frac{-b_1 c_1}{e^{-a_1 t_f}} e^{-a_1 t} + \frac{-b_2 c_2}{e^{-a_2 t_f}} e^{-a_2 t} - b_3 c_3 \right) = b_3(t - t_f). \quad (4.48)$$

The range of γ that yields a solution $t = \sigma(\gamma)$ to (4.48) is

$$0 \leq \gamma \leq \frac{-b_3 t_f}{\frac{-b_1 c_1}{e^{-a_1 t_f}} + \frac{-b_2 c_2}{e^{-a_2 t_f}} - b_3 c_3} \approx t_f. \quad (4.49)$$

Then the *Bang-Bang* control law is obtained as follows:

$$u^*(t) = \begin{cases} -u_b, & t \leq \sigma(\gamma), \\ +u_b, & t > \sigma(\gamma), \end{cases} \quad (4.50)$$

¹switching refers to shift other extreme value, e.g., $-u_b \rightarrow u_b$.

and the state trajectory is computed as:

$$x_1(t) = \begin{cases} e^{a_1 t} x_1(0) + \frac{b_1 u_b}{a_1} (1 - e^{a_1 t}), & t \leq \sigma(\gamma), \\ e^{a_1(t-\sigma(\gamma))} x_1(\sigma(\gamma)) - \frac{b_1 u_b}{a_1} (1 - e^{a_1(t-\sigma(\gamma))}), & t > \sigma(\gamma), \end{cases} \quad (4.51)$$

$$x_2(t) = \begin{cases} e^{a_2 t} x_2(0) + \frac{b_2 u_b}{a_2} (1 - e^{a_2 t}), & t \leq \sigma(\gamma), \\ e^{a_2(t-\sigma)} x_2(\sigma(\gamma)) - \frac{b_2 u_b}{a_2} (1 - e^{a_2(t-\sigma)}), & t > \sigma(\gamma), \end{cases} \quad (4.52)$$

$$x_3(t) = \begin{cases} -b_3 u_b t + x_3(0), & t \leq \sigma(\gamma), \\ b_3 u_b (t - \sigma(\gamma)) + x_3(\sigma(\gamma)), & t > \sigma(\gamma). \end{cases} \quad (4.53)$$

We claim that $\sigma(\gamma)$ does not exist as the state constraint is violated before t reaches $\sigma(\gamma)$. That is $C^\top x(\sigma) > C^\top x(t_f), \forall \sigma(\gamma)$. Note that $\sigma(\gamma)$ is determined by γ in (4.49), then the constraint bound, C_b is obtained as $C^\top x(t_f)$, where $x(t_f)$ follows from (4.51) – (4.53). We can check that the following statement holds for given ranges of electrochemical values in Table 4.1 :

$$\begin{aligned} & c_1[(e^{a_1(t_f-\sigma)} - 1)x_1(\sigma) - \frac{b_1 u_b}{a_1}(1 - e^{a_1(t_f-\sigma)})] \\ & + c_2[(e^{a_2(t_f-\sigma)} - 1)x_2(\sigma) - \frac{b_2 u_b}{a_2}(1 - e^{a_2(t_f-\sigma)})] \\ & + c_3[b_3 u_b(t_f - \sigma)] < 0. \end{aligned} \quad (4.54)$$

We conclude that the switching time does not exist when the state constraint becomes active at the terminal time. The switching time occurs only if $C_b = \bar{C}$, and is equivalent to terminal time, t_f . Note that we speculated that the state constraint becomes active when the constraint C_b is less than $\bar{C}$, which is the scenario discussed in Case 2 next.

Case 2: Consider the case when the state constraint becomes active prior to the terminal time. Then transversality condition (4.23) and junction condition (4.18) state that

$$p(t_f) = -\gamma C, \quad (4.55)$$

$$p(\tau^-) = p(\tau^+) + \zeta(-C), \quad (4.56)$$

where $t = \tau$ is the time when the state hits the state constraint bound,

$$C^\top x(\tau) = C_b. \quad (4.57)$$

Then the input u can be found such that state does not violate the constraint further, i.e., $C^\top \dot{x}(t) = 0 \forall t \in [\tau^+, t_f]$. We can find such an input by taking time derivative of the state constraint function,

$$0 = C^\top \dot{x},$$

$$= C^\top (Ax + Bu), \quad (4.58)$$

then the input $u(t)$ becomes a state feedback control law:

$$u(t) = -\frac{C^\top A}{C^\top B}x \triangleq -Kx, \quad (4.59)$$

where $K \in \mathbb{R}^{1 \times 3}$. Note that the input u is potentially discontinuous at $t = \tau$. Plugging the feedback control law into the dynamical system results in the following autonomous system:

$$\dot{x}(t) = (A - BK)x(t), \quad t \in [\tau^+, t_f]. \quad (4.60)$$

We claim that the feedback control law (4.59) is bounded for the input constraint, namely

$$-u_b < -Kx(t) < u_b. \quad (4.61)$$

The system is autonomous for $t \in [\tau^+, t_f]$ such that $x(t)$ can be analytically derived,

$$-u_b < -Ke^{(A-BK)t}x_0 < u_b, \quad t \in [\tau^+, t_f]. \quad (4.62)$$

Notice that $|-Ke^{(A-BK)t}x_0| \leq |-Kx_0|, \forall t \in [\tau^+, t_f]$ as $\text{eig}(A - BK) \leq 0$ for given electrochemical model. A bounded feedback control law ensures the Lagrange multipliers associated with input constraints are zero, i.e., $\mu_1 = 0, \mu_2 = 0$ due to complimentary slackness. Then (4.40) becomes

$$p^\top(t)B = 0, \quad t \in [\tau, t_f]. \quad (4.63)$$

We aim to find γ at terminal time $t = t_f$,

$$0 = p^\top(t_f)B = -\gamma C^\top B, \quad (4.64)$$

since $C^\top B \neq 0$, $\gamma = 0$. For the constrained optimal control case, we divide the time horizon into two cases i) state constraint is not active, i.e., $t \in [0, \tau]$ ii) state constraint becomes active, i.e., $t \in [\tau^+, t_f]$.

The next step is to compute the co-state trajectory for the following input,

$$u^*(t) = \begin{cases} -u_b, & t \in [0, \tau], \\ -Kx, & t \in [\tau^+, t_f], \end{cases} \quad (4.65)$$

then the costate trajectory is obtained as follows:

$$p(t) = \begin{cases} e^{-At}p(0) + [0, 0, -t]^\top, & t \in [0, \tau], \\ [e^{\bar{A}(t_f-t)} - \mathbb{I}] \theta, & t \in [\tau^+, t_f], \end{cases} \quad (4.66)$$

where

$$\bar{A} = \left(I - \frac{CB^\top}{B^\top C} \right) A, \quad \theta = \begin{bmatrix} -\frac{c_1}{a_1 c_3} \\ -\frac{c_2}{a_2 c_3} \\ 0 \end{bmatrix}. \quad (4.67)$$

The derivation of the co-state trajectory for $t \in [\tau, t_f]$ is not trivial. The analytical solution of costate dynamics for $t \in [\tau, t_f]$ is derived briefly as follows: The ODEs for the costates are

$$\dot{p}_1 = -p_1 a_1 + \eta c_1, \quad \dot{p}_2 = -p_1 a_2 + \eta c_2, \quad \dot{p}_3 = -1 + \eta c_3, \quad (4.68)$$

where $\eta \geq 0$. It is obvious that $\eta(t) = 0$ if $t \leq \tau^-$ due to complimentary slackness. We also have $\gamma = 0$ as described in (4.64) Then,

$$p(t_f) = 0 \in \mathbb{R}^3. \quad (4.69)$$

From (4.63), (4.68) can be written as:

$$0 = B^\top \dot{p}(t) = B^\top \left(-Ap(t) + \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix} + \eta C \right), \quad (4.70)$$

thus we can compute $\eta(t)$ such as:

$$\eta(t) = \frac{B^\top A}{B^\top C} p(t) + \eta(t_f), \quad (4.71)$$

where

$$\eta(t_f) := \frac{B^\top [0 \ 0 \ 1]^\top}{B^\top C}. \quad (4.72)$$

Substituting (4.71) to (4.68), we obtain

$$\begin{aligned} \dot{p} &= -Ap + \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix} + \left(\frac{B^\top A}{B^\top C} p(t) + \eta(t_f) \right) C. \\ &= -(I - \frac{CB^\top}{B^\top C}) Ap(t) + \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix} + \eta(t_f) C \\ &=: -\bar{A}p(t) + \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix} + \eta(t_f) C. \end{aligned} \quad (4.73)$$

We can design a constant vector

$$\theta = \left[-\frac{c_1}{a_1 c_3} \quad -\frac{c_2}{a_2 c_3} \quad 0 \right]^\top, \quad (4.74)$$

such that the following costate equation is derived.

$$p(t) = [e^{\bar{A}(t_f-t)} - I]\theta. \quad (4.75)$$

where $\bar{A}$ is defined in (4.67).

Given an allowable maximum current, i.e. $u_b \geq |-Kx_0|$, we find the junction time τ when the state constraint becomes active, i.e.

$$c_1 x_1(\tau) + c_2 x_2(\tau) + c_3 x_3(\tau) = C_b. \quad (4.76)$$

Then we construct algebraic equations for unknowns $\zeta, p(0)$ from 1) jump condition form of co-states in (4.18), 2) jump condition for Hamiltonian in (4.19), such as,

$$e^{-A\tau}p(0) + \begin{bmatrix} 0 \\ 0 \\ -\tau \end{bmatrix} = [e^{\bar{A}(t_f-\tau)} - \mathbb{I}] \theta + \zeta(\tau)C, \quad (4.77)$$

$$\begin{aligned} & \left[e^{-A\tau}p(0) + \begin{bmatrix} 0 \\ 0 \\ -\tau \end{bmatrix} \right]^\top (Ax(\tau) + Bu(\tau^-)) \\ &= \left[(e^{\bar{A}(t_f-\tau)} - \mathbb{I}) \theta \right]^\top (Ax(\tau) + Bu(\tau^+)), \end{aligned} \quad (4.78)$$

where

$$u(\tau^-) = -u_b, \quad u(\tau^+) = -\frac{C^\top Ax(\tau)}{C^\top B}.$$

Then, the analytical solution of ζ is obtained as follows:

$$\zeta(\tau) = \frac{\left[(e^{\bar{A}(t_f-\tau)} - \mathbb{I}) \theta \right]^\top B \left(-\frac{C^\top Ax(\tau)}{C^\top B} + u_b \right)}{C^\top (Ax(\tau) - Bu_b)}. \quad (4.79)$$

With $\zeta(t)$ computed above, the boundary condition for costate, p_0 can be obtained as:

$$p(0) = e^{A\tau} \left([e^{\bar{A}(t_f-\tau)} - \mathbb{I}] \theta + \zeta(\tau)C + \begin{bmatrix} 0 \\ 0 \\ \tau \end{bmatrix} \right). \quad (4.80)$$

Lastly, we check whether a switching time exists for $t \in [0, \sigma]$. First, τ is chosen between $\tau \in [0, t_f]$, then C_b is obtained by $C_b = C^\top x(\tau)$. By solving the algebraic equations (4.77) – (4.78), we find p_0 , and then check that $b_1 p_1(t) + b_2 p_2(t) + b_3 p_3(t) \leq 0$ for $\tau, t \in [0, t_f]$ for given electrochemical parameters described in Table 4.1.

□

Therefore, when the state constraint becomes active prior to the terminal time, the optimal control input is to apply state feedback control law (4.59), which results in riding the constraints boundary.

Remark 1. The optimal solution for $u(t)$ in (4.34) is the global optimal solution as the necessary conditions become necessary and sufficient conditions because (4.34) is convex.

The analytical solution of the optimal input, state trajectory, co-state trajectory, and associated Lagrange multipliers for the constrained optimal control problem are summarized in Table 4.2.

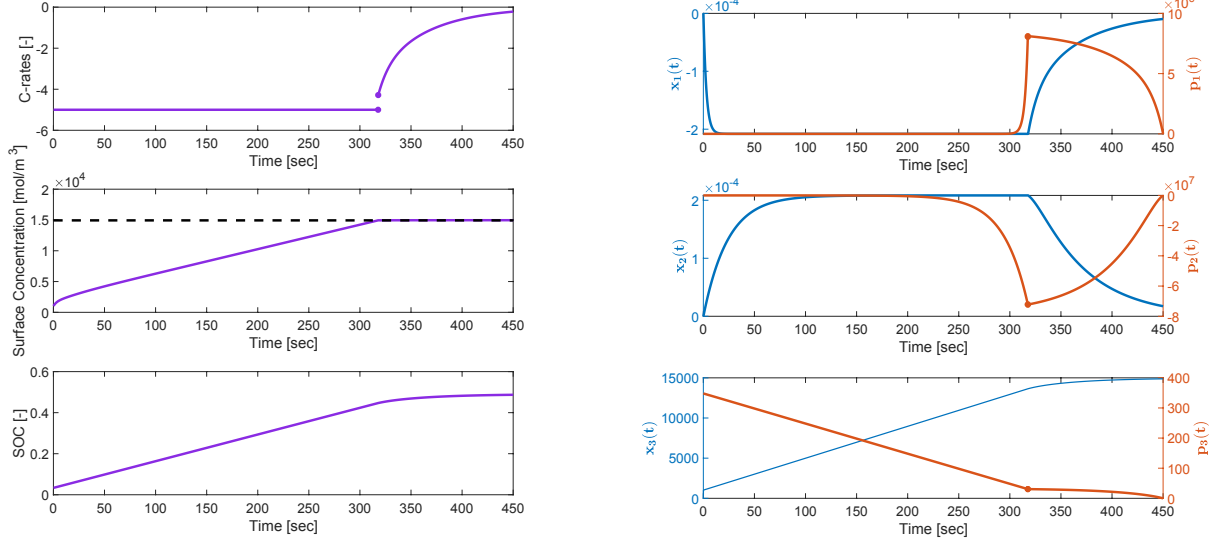
Interval	$t \in [0, \tau]$	$t \in [\tau, t_f]$
$u(t)$	$-u_b$	$-Kx$
$x(t)$	$e^{At}x_0 - \int_0^t e^{A(t-s)}Bu_b ds$	$e^{(A-BK)(t-\tau)}x(\tau)$
$p(t)$	$e^{-At}p(0) + [0 \ 0 \ -t]^\top$	$[e^{\bar{A}(t_f-t)} - \mathbb{I}] \theta$
$\mu_1(t)$	$-p^T(t)B$	0
$\mu_2(t)$	0	0
$\eta(t)$	0	$\frac{B^\top A}{B^\top C}p(t) + \eta(t_f)$

Table 4.2: The analytical solution of constrained optimal control problem (4.34) for $C_b \leq \bar{C}$. Unknown variables, $\zeta, p(0)$, and $\eta(t_f)$ are derived in (4.79), (4.80), (4.72).

Numerical simulation results for the optimal fast charging problem (4.34) are presented in Fig. 4.1 to validate the proposed analytical solution. The terminal time, t_f , is chosen as 450 seconds, allowable maximum current, u_b , is 5 C-rate, and constraint bound, C_b is chosen as 50% of maximum concentration level (for illustration). The constrained optimal control solution satisfies $C_b < \bar{C}$ from (4.44), and thus the junction time occurs at $t = \tau = 317$. Figure 4.1a displays the optimal input, state constraint activation, and the normalized objective function, which is SOC from (3.13). Figure 4.1b exhibits the corresponding trajectories for the states and co-states. Although difficult to see in the plots, the co-states are discontinuous at the junction time, and the trajectories satisfy the optimality conditions.

4.1.4 Summary

In this section, we analyze optimal fast charging control of Li-ion batteries. Pontryagin's Minimum Principle is used to derive the optimality conditions. We prove that the optimal fast-charging control is a *Bang-Ride* control, which applies maximum allowable charging current until the state constraint becomes active, and then rides the constraint bound to ensure safety. PMP analysis provides theoretical evidence why optimal fast-charging protocols are Bang-Ride control in the literature, at least for single particle models. Furthermore, our theoretical proofs can explain why specific constant current-constant voltage (CCCV) protocols are candidates for optimal for experimental design in fast charging problem [6],



(a) Input, state constraint, and normalized objective function.

(b) States and co-states trajectories.

Figure 4.1: Numerical constrained optimal control results using analytical solutions in Table 4.2.

different experimental current pulse profiles [105], and fast charging method design [131]. Due to this analysis, the search space for optimal control solutions can be reduced to *Bang-Ride* trajectories without loss of optimality. On-going work involves extending PMP analysis for more sophisticated electrochemical models that are nonlinear with multiple constraints.

4.2 Learning-based fast-charging control

The model-based optimal control provides theoretical tools for the optimality of derived input currents as discussed in previous section. However, batteries are generally partial-observable system which requires expensive-computational power to estimate the parameters and the internal states. This can be alleviated by using learning-based controller, such as reinforcement learning. This learning-based approach does not require knowledge of the system dynamics, but can achieve the control objective. In this section, we are going to investigate this learning-based controller using deep reinforcement learning techniques and perform comparative analysis with other existing model-based controller.

4.2.1 Introduction

The most common charging procedure for Li-ion batteries is the well known Constant-Current Constant-Voltage method (CC-CV). This is employed in industry due to its ability to provide reasonable performance with a relatively simple implementation [27]. However, such a simple charging algorithm is often based on excessively conservative constraints which reduce the probability of safety hazards at the expense of higher charging times. Therefore, it does not constitute an optimal policy for the problem that we aim to solve - at least not in all cases. For these reasons, several advance battery management strategies have been employed. In particular we can classify them as (i) Model-based strategies, and (ii) Model-free strategies. The former seek to find an optimal input trajectory based on a specified battery model, while the latter interact directly with the battery (denoted as the “environment” in the language of RL).

The use of mathematical models for battery control is a large topic in the literature. Equivalent circuit models are simple, intuitive, and mimic the battery behaviors through lumped electrical parameters which can be easily identified [42]. Electrochemical Models (EM) exhibit higher accuracy than equivalent circuit models, as well as the ability of describing internal battery phenomena from the perspective of electrochemistry, and therefore are usually preferred for simulation purposes [104]. There are a number of research to investigate the health-aware fast charging strategy for EMs. The authors in [55] formulate a minimum-time charging problem with health-related constraints and use nonlinear model predictive control. Similarly, the authors in [91] propose a reference governor approach to solve a minimum-time charging problem, and the authors in [121] propose a quadratic dynamic matrix control formulation to design an optimal charging strategy for real-time model predictive control. The authors in [137] synthesize a state estimation and model predictive control scheme for a reduced electrochemical-thermal model, in order to design health-aware fast charging strategy. The problem is formulated as a linear time-varying model predictive control scheme, with a moving horizon state estimation framework. In the context of ageing mechanism for EMs, the authors of [92] studied the trade-off between charging speed and degradation, based on an electro-thermal-ageing model. The authors in [94] minimizes the film layer growth of an electrochemical model by formulating shrinking-horizon nonlinear model predictive control. However, the exploitation of model-based charging procedure has to face some crucial challenges. (i) Every model is subject to uncertainties and modeling mismatches, which affect its accuracy. Since the controller’s performance depends on the model accuracy, a proper parameter identification procedure has to be conducted based on experimentally collected data. In the case of electrochemical models, there are typically dozens of parameters to be identified. This motivates sophisticated optimally designed experiments for parameter estimation [95, 86]. (ii) Electrochemical models usually consist of a large number of states, thus leading to a large-scale optimization problem. Moreover, most states are not measurable in a realistic scenario and, therefore, the presence of an observer is required to reconstruct the full state information from the available measurements [127]. (iii) The model parameters drift as the battery ages. It is important to notice that none of the model-based strategies

proposed in the literature considers the adaptability of the control strategy to variations in the parameters.

In order to overcome all the limitations of the model-based approach, there has been substantial effort in the literature to design fast-charging strategies that do not rely on a mathematical model. Some of them rely on rule-based adaptation of the CC-CV protocol [83, 135, 62]. The authors of [131] propose a charging algorithm that incorporates CC-CV charging profile and battery health estimation using an extended Kalman filter, where the magnitude of CC and threshold for CV are updated from the estimator. The authors of [6] propose an optimal design of fast-charging procedures relying on machine learning. In particular, the current profile is parametrized by six-steps of ten-minutes each, and Bayesian optimization is used to select the optimal sequence which maximizes battery cycle life. The authors of [90] propose closed-loop charging techniques called Constant-Current Constant-Temperature Constant Voltage (CC-CT-CV), where a rule-based proportional-integral-derivative (PID) controller is employed. This closed-loop charging strategy constitutes an output-feedback control law which enables the CC-CV protocol to consider temperature constraints. It is important to notice that the difficulty of the observability and identifiability is no longer an issue as this strategy relies only on the output measurements. The main issues of this outputs-based strategy are represented by the fact that (i) the optimality of the resulting charging policy is no longer guaranteed, (ii) the controller gain should be obtained by trial and error, and finally (iii) the controller does not adapt to parameter changes.

All these challenges can be addressed by using a charging procedure based on the Reinforcement Learning (RL) framework [114]. An RL framework consists of an agent (the battery management system) which interacts with the environment (the battery) by taking specific actions (the applied current) according to the environment configuration (charging time). The main idea is that the agent learns the feedback control policy directly from interactions with the environment, namely observations of the reward and state. The control policy is iteratively updated to maximize the expected long-term reward. Notice that the reward has to be properly designed so the agent learns how to accomplish the required task. Most RL algorithms can be classified in two different groups: tabular methods, e.g., Q-learning, SARSA, and approximate solutions methods which is also called Approximate Dynamic Programming (ADP). While the former performs well only in the presence of small and discrete sets of actions and states, the latter can be used even with continuous state and action spaces, thus solving the so-called “curse of dimensionality”. On the other hand, the convergence of the former is proven under mild assumptions, while no proof of convergence exists for the approximate methods in the general case. The recent success in several applications of RL using deep neural networks as function approximators has greatly increased expectations in the scientific community [74, 107, 124, 7]. From a control systems perspective, the design of RL algorithms involves the computation of feedback control laws for dynamical systems via optimal adaptive control methods [60]. RL can be regarded as an indirect adaptive controller wherein the parameters of the value function are estimated, and then the controller is improved based on the estimated value function.

In this paper, a fast-charging strategy subject to safety constraints, using a deep rein-

forcement learning framework, is proposed as an extension of the authors' previous work [88]. In particular, Deep Deterministic Policy Gradient (DDPG) [61] is adopted. DDPG is an actor-critic method that deals with continuous state and action spaces. The safety constraints are considered as soft constraints where the agent (controller) receives penalties in the reward function in case of violation.

We summarize our novel contributions to the relevant literature as follows. First of all, reinforcement learning is adopted to overcome the challenges of the fast-charging problem in battery management systems: computational complexity, observability, and adaptation. To mitigate the computational complexity associated with feedback controller design for a high-dimensional state space, we project the state onto a lower-dimensional feature space via Principal Component Analysis (PCA). This approach, however, requires a state observer to estimate the states in a real-world application, which motivates an output feedback controller using voltage, temperature, and current measurements only. The validity of the state-feedback and output-feedback controllers are tested in simulation and compared to other existing charging algorithms, namely the reference governor and CC-CT-CV. Lastly, we show that RL policies are capable of adaptation when the battery parameters are changing over the cycles. This demonstrates that the use of the proposed methodology enables adaptation to uncertain and drifting parameters, which is a distinct advantage over other conventional approaches.

4.2.2 Reinforcement Learning Framework

Markov Decision Process, Policy and Value Functions

In the following, we briefly review the Markov Decision Process (MDP) to provide the critical background. A thorough exposition can be found in [115, 10]. In the MDP setting, we seek the best policy that maximizes the total rewards received from the environment, E (i.e. the plant). At each time step $t \in \mathbb{R}^+$ the environment conditions are described by a state vector, $\mathbf{s}_t \in \mathcal{S}$, where $\mathcal{S}$ is the state space while the control policy picks an action $\mathbf{a}_t \in \mathcal{A}$, with $\mathcal{A}$ being the action space, which is based on the observation of the state $\mathbf{s}_t$. The action is therefore applied to the environment, whose state evolves to $\mathbf{s}_{t+1} \in \mathcal{S}$, according to the state-transition probability $p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t)$, and the agent receives a scalar reward $\mathbf{r}_{t+1} = r(\mathbf{s}_t, \mathbf{a}_t)$. The policy is represented by π which maps the state to the action and can be either deterministic or stochastic. The total discounted reward from time t onward can be expressed as:

$$R_t = \sum_{k=0}^{\infty} \gamma^k r(\mathbf{s}_{t+k}, \mathbf{a}_{t+k}) \quad (4.81)$$

where $\gamma \in [0, 1]$ is the discounting factor.

The *state value function*, $V^\pi(\mathbf{s}_t)$ is the expected total discounted reward starting from state $\mathbf{s}_t$. In the controls community, this is sometimes called the cost-to-go, or reward-to-go. Importantly, note that the value function depends on the control policy. If the agent uses a given policy π to select actions starting from the state $\mathbf{s}_t$, the corresponding value function is

given by:

$$V^\pi(\mathbf{s}_t) = \mathbb{E}[R_t \mid \mathbf{s}_t] \quad (4.82)$$

Then, the optimal policy π^* is the policy that corresponds to the maximum value $V^*(\mathbf{s}_t)$ of the value function

$$\pi^* = \arg \max_{\pi} V^\pi(\mathbf{s}_t) \quad (4.83)$$

The solution of (4.83) is pursued by those methods which follow the Dynamic Programming (DP) paradigm. Such paradigm, however, assumes a perfect knowledge of the environment E (i.e. the state-transition probability as well as the reward function).

The next definition, known as the “Q-function”, is fundamental since it enables the concept of model-free reinforcement learning. Consider the *state-action value function*, $Q^\pi(\mathbf{s}_t, \mathbf{a}_t)$, which is a function of the state-action pair that returns a real value. This latter corresponds to the expected total discounted reward when the action $\mathbf{a}_t$ is taken in state $\mathbf{s}_t$, and then the policy π is followed henceforth. Mathematically,

$$Q^\pi(\mathbf{s}_t, \mathbf{a}_t) = \mathbb{E}[R_t \mid \mathbf{s}_t, \mathbf{a}_t] \quad (4.84)$$

The optimal Q-function is given by

$$Q^*(\mathbf{s}_t, \mathbf{a}_t) = \arg \max_{\pi} Q^\pi(\mathbf{s}_t, \mathbf{a}_t) \quad (4.85)$$

and represents the expected total discounted reward received by an agent that starts in $\mathbf{s}_t$, picks (possibly non-optimal) action $\mathbf{a}_t$, and then behaves optimally afterwards. Since $V^*(\mathbf{s}_t)$ is the maximum expected total discounted reward starting from state $\mathbf{s}_t$, it will also be the maximum of $Q^*(\mathbf{s}_t, \mathbf{a}_t)$ over all possible actions $\mathbf{a}_t \in \mathcal{A}$

$$V^*(\mathbf{s}_t) = \max_{\mathbf{a}_t \in \mathcal{A}} Q^*(\mathbf{s}_t, \mathbf{a}_t) \quad (4.86)$$

If the optimal Q-function is known, then the optimal action $\mathbf{a}_t^*$ can be extracted by choosing the action $\mathbf{a}_t$ that maximizes $Q^*(\mathbf{s}_t, \mathbf{a}_t)$ for state $\mathbf{s}_t$ (i.e. the optimal policy π^* is retrieved),

$$\mathbf{a}_t^* = \arg \max_{\mathbf{a}_t \in \mathcal{A}} Q^*(\mathbf{s}_t, \mathbf{a}_t) \quad (4.87)$$

without requiring the knowledge of the environment dynamics.

4.2.3 Actor-Critic

Actor-critic is an ADP method which solves dynamic programming heuristically, as depicted in Fig. 4.2. The actor-critic is a policy gradient approach, i.e. it seeks to model and optimize the policy directly. Importantly, the actor-critic approach allows for continuous state/action spaces by using a function approximator, e.g., a neural network. In RL, as well as in dynamic

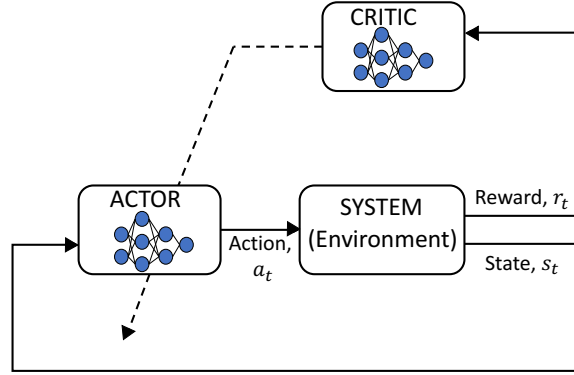


Figure 4.2: Actor-Critic Structure.

programming, the action is taken by a *policy* to maximize the expected total discounted reward. By following a given policy and processing the rewards, one should estimate the expected return given states from the *value function*. In the actor-critic approach, the *actor* improves the policy based on the value function that is estimated by the *critic*. We specifically focus on the policy gradient-based actor-critic algorithm in this work, and, in particular, on the deep deterministic policy gradient (DDPG) [61]. This algorithm is an extension of deep Q-network (DQN) [74] to continuous actions, maintaining the importance of features such as: (i) random sampling from a replay buffer where tuples are saved, (ii) the presence of target networks for stabilizing the learning process. The algorithm is characterized by the parametrization of both the critic, $Q(\mathbf{s}_t, \mathbf{a}_t | \boldsymbol{\theta}^Q)$, and the actor, $\pi(\mathbf{s}_t | \boldsymbol{\theta}^\pi)$, as deep neural network. In addition, the correspondent target networks are defined as $Q'(\mathbf{s}_t, \mathbf{a}_t | \boldsymbol{\theta}^{Q'})$ and $\pi'(\mathbf{s}_t | \boldsymbol{\theta}^{\pi'})$, where the parameters $\boldsymbol{\theta}^{Q'}$ and $\boldsymbol{\theta}^{\pi'}$ are slowly updated in order to track and filter the ones of the actual network ($\boldsymbol{\theta}^Q$ and $\boldsymbol{\theta}^\pi$) thus reducing the chattering due to the learning process and enhancing its convergence. In the next paragraphs the concepts of critic and actor are explained in details. Both the actual and target parameters are initialized to the random vectors $\boldsymbol{\theta}_0^\pi$ and $\boldsymbol{\theta}_0^Q$.

Critic

The role of the critic is to evaluate the current policy prescribed by the actor. The action is taken as the sum of the actor network output, which is a function of the environment states, and an exploration noise $\mathcal{N}_t$, namely

$$\mathbf{a}_t = \pi(\mathbf{s}_t | \boldsymbol{\theta}^\pi) + \mathcal{N}_t \quad (4.88)$$

where the exploration noise is a random variable, for instance an Ornstein-Uhlenbeck process is considered in this paper. After applying the action, we observe the reward $\mathbf{r}_{t+1}$ and the next state $\mathbf{s}_{t+1}$. For each time step t , the tuple $(\mathbf{s}_t, \mathbf{a}_t, \mathbf{r}_{t+1}, \mathbf{s}_{t+1})$ is stored in the replay buffer memory. As soon as the number of tuples stored in the memory reaches a default threshold

N , at each time step a random mini-batch of N transitions is sampled from the buffer and for each of them we set

$$y_i = r_{i+1} + \gamma Q'(\mathbf{s}_{i+1}, \pi'(\mathbf{s}_{i+1}|\boldsymbol{\theta}^{\pi'}))|\boldsymbol{\theta}^{Q'} \quad i = 1, \dots, N \quad (4.89)$$

where superscript $'$ denotes the target network, whose parameters are slowly updated in (4.95). The mini-batch, which is randomly extracted at each time step from the buffer if enough tuples are stored in the memory, is exploited for updating the networks. In particular, the critic is updated to minimize the loss function $\mathcal{L}(\boldsymbol{\theta}^Q)$:

$$\mathcal{L}(\boldsymbol{\theta}^Q) = \frac{1}{N} \sum_i (y_i - Q(\mathbf{s}_i, a_i|\boldsymbol{\theta}^Q))^2 \quad (4.90)$$

$$\boldsymbol{\theta}_{k+1}^Q = \boldsymbol{\theta}_k^Q - \eta_Q \nabla_{\boldsymbol{\theta}^Q} \mathcal{L}(\boldsymbol{\theta}^Q) \quad (4.91)$$

where index- k denotes the gradient descent algorithm iterates, and η_Q denotes the learning rate of the critic network. Note that the subscript k in the network parameters $\boldsymbol{\theta}_k^Q$, $\boldsymbol{\theta}_k^\pi$, $\boldsymbol{\theta}_k^{Q'}$ and $\boldsymbol{\theta}_k^{\pi'}$ is omitted when clear from the context.

Actor

The parameters of the actor network are updated in order to maximize the cumulative expected reward $V^\pi(\mathbf{s}_t)$. In this paragraph we refer to the cumulative reward with the variable $\mathcal{J}(\theta^\pi)$, in order to highlight its dependency on the actor parametrization. The update of the actor parameters is done as follows

$$\boldsymbol{\theta}_{k+1}^\pi = \boldsymbol{\theta}_k^\pi + \eta_\pi \nabla_{\boldsymbol{\theta}^\pi} \mathcal{J}(\theta^\pi) \quad (4.92)$$

where index- k denotes the gradient ascent algorithm iterates, and η_π denotes the learning rate of the actor network. Notice that according to the proof in [108], the policy gradient in (4.92) can be expressed as

$$\nabla_{\boldsymbol{\theta}^\pi} \mathcal{J}(\theta^\pi) \approx \mathbb{E} [\nabla_{\mathbf{a}} Q(\mathbf{s}_t, \mathbf{a}_t|\boldsymbol{\theta}^Q) \nabla_{\boldsymbol{\theta}^\pi} \pi(\mathbf{s}_t|\boldsymbol{\theta}^\pi)] \quad (4.93)$$

which is then approximated by samples as follows

$$\nabla_{\boldsymbol{\theta}^\pi} \mathcal{J}(\theta^\pi) \approx \frac{1}{N} \sum_i [\nabla_{\mathbf{a}} Q(\mathbf{s}_i, \mathbf{a}_i|\boldsymbol{\theta}^Q) \nabla_{\boldsymbol{\theta}^\pi} \pi(\mathbf{s}_i|\boldsymbol{\theta}^\pi)] \quad (4.94)$$

Once the parameters of critic and actor network given samples are updated, then the target networks are also updated as follows:

$$\begin{aligned} \boldsymbol{\theta}^{Q'} &\leftarrow \tau \boldsymbol{\theta}^Q + (1 - \tau) \boldsymbol{\theta}^{Q'} \\ \boldsymbol{\theta}^{\pi'} &\leftarrow \tau \boldsymbol{\theta}^\pi + (1 - \tau) \boldsymbol{\theta}^{\pi'} \end{aligned} \quad (4.95)$$

where τ is the level of “soft-update”. Equation (4.95) improves the stability of the learning procedure. Note that convergence is no longer guaranteed, in general, when a value function approximator is used. Since convergence of the critic network is not guaranteed, it is important to note that these target networks should update slowly to avoid divergence. Thus one should choose a small value of τ . This is a challenging point when the action space becomes continuous unlike tabular Q-learning.

4.2.4 Fast-Charging Problem

The crucial role in battery management system is to the trade-off between fast-charging and ageing while satisfying safety constraints. In this context, the fast charging problem can be described as reaching the final SOC in minimum time without violating constraints. The input current is limited by the hardware configuration of the battery charger, namely,

$$-I_{\max} \leq I(t) \leq 0, \quad (4.96)$$

with the convention that a negative current is charging the battery. In order to limit the different degradation mechanism of the cell, we consider constraints that have to be satisfied during the whole charging process. First, the cell temperature is not allowed to exceed a maximum temperature, $T_{\max}$, since the temperature is closely related to the SEI layer growth [120]

$$T(t) \leq T_{\max}. \quad (4.97)$$

Furthermore, we aim to avoid Lithium plating deposition by constraining the side-reaction overpotential to be positive, which is defined as:

$$\eta_{sr}(x, t) = \phi_s^-(x, t) - \phi_e^-(x, t) - U_{sr}. \quad (4.98)$$

where U_{sr} denotes the equilibrium potential of the side reaction and is assumed to be zero [4]. It is known that main battery degradation mechanism, i.e., lithium plating [23, 120, 4], are related to the side reaction overpotential η_{sr} . Lithium plating is a particularly harmful phenomenon which happens when it becomes thermodynamically favorable for lithium to plate onto the surface of the negative electrode particles instead of intercalating [120]. If this phenomenon persists, then dendrites can form, grow, and pierce through the separator, causing a short-circuit. Note that this degradation mechanism is aggravated when the battery operates at low temperature. We impose the side-reaction overpotential constraint as follows:

$$\eta_{sr}(L^-, t) \geq 0. \quad (4.99)$$

In consideration of the preceding characteristics, we formulate an optimal control problem as follows:

$$\max_{I(t)} -t_f \quad (4.100)$$

subject to

$$\begin{aligned} & \text{battery dynamics, (2.1) – (2.9),} \\ & \text{input constraint, (4.96)} \\ & \text{state constraints, (4.97) – (4.99)} \\ & V(t_0) = V_0, T(t_0) = T_0 \\ & SOC^-(t_f) = SOC_{\text{ref}}, \end{aligned}$$

where $t_0 = 0$ and t_f are the initial and final time of the charging procedure, V_0 and T_0 are the initial value for voltage and temperature respectively, and SOC_{ref} is the reference SOC at which the charging is considered to completed.

For the case in which only output measurements are available (output feedback policy), we cannot consider the positivity constraints on the side-reaction overpotential, since it is based on unmeasured states. Therefore, (4.99) is replaced with a more conservative constraint that limits the voltage below a predefined threshold V_{max} , as follows:

$$V(t) \leq V_{\text{max}}. \quad (4.101)$$

4.2.5 Results and Discussion

In this section, we assess in simulation the performance of the RL when applied to the battery charging problem (see Section 4.2.4) for both states-based and outputs-based configurations. The objective is to compare the proposed actor-critic approach with some benchmark algorithms that have been discussed in the literature and exhibit satisfying results in accomplishing the required task. We consider the modified reference governor technique [91] and CC-CT-CV protocol based on PI controller. As previously stated, we rely on the electrochemical model presented in Section 2.1.1 as a battery simulator. The battery considered here consists of a cell with a graphite anode and a LiNiMnCoO_2 (NMC) cathode chemistry.

Notice that the structure of the actor-critic networks remain consistent for different charging problems. The actor-critic networks are based on neural network architectures [61] with different numbers of neurons. Specifically, the actor network uses two hidden layers with 20 - 20 neurons while the critic network uses two hidden layers with 100 - 75 neurons. Hyper parameters are detailed in Table 4.3.

Variable	Description	Value
γ	Discount factor	0.99
η_π, η_Q	Learning rate of actor, critic	$10^{-4}, 10^{-3}$
τ	Soft update of target networks	10^{-3}

Table 4.3: Actor-critic hyper parameters.

The reward function is designed according to the optimization problem in (4.100) with the aim of achieving fast charging while guaranteeing safety,

$$r_{t+1} = r_{\text{fast}} + r_{\text{safety}}(\mathbf{s}_t, \mathbf{a}_t), \quad (4.102)$$

where r_{fast} is an instantaneous penalty for each time step which passes before the reference SOC is achieved. In addition, a penalty is also introduced at each time step when the safety constraints are violated by means of linear penalty functions [109]. As discussed in Section 4.2.4, the safety constraints enforced in the case of a states-based framework and in the case of a outputs-based one are different, due to the fact that the side-reaction overpotential measurement is not available in the latter case. Therefore, in the states-based configuration we have the following safety term in the reward function

$$r_{\text{safety}}(\mathbf{s}_t, \mathbf{a}_t) = r_{\eta_{\text{sr}}}(\mathbf{s}_t, \mathbf{a}_t) + r_{\text{temp}}(\mathbf{s}_t, \mathbf{a}_t), \quad (4.103)$$

where

$$r_{\eta_{\text{sr}}}(\mathbf{s}_t, \mathbf{a}_t) = \begin{cases} \lambda_{\text{sr}} \eta_{\text{sr}}(t), & \text{if } \eta_{\text{sr}}(t) < 0 \\ 0, & \text{otherwise} \end{cases} \quad (4.104)$$

$$r_{\text{temp}}(\mathbf{s}_t, \mathbf{a}_t) = \begin{cases} \lambda_{\text{temp}}(T(t) - T_{\text{max}}), & \text{if } T(t) \geq T_{\text{max}} \\ 0, & \text{otherwise} \end{cases} \quad (4.105)$$

where the temperature threshold is set to $T_{\text{max}} = 40^\circ\text{C}$. The fast charging term and penalty function coefficients have been tuned as $r_{\text{fast}} = -0.1$, $\lambda_{\text{sr}} = 10$ and $\lambda_{\text{temp}} = -5$.

For the outputs-based setting we substitute the limit on the side-reaction overpotential with the more conservative constraint on the voltage, such as

$$r_{\text{safety}}(\mathbf{s}_t, \mathbf{a}_t) = r_{\text{volt}}(\mathbf{s}_t, \mathbf{a}_t) + r_{\text{temp}}(\mathbf{s}_t, \mathbf{a}_t), \quad (4.106)$$

where r_{temp} is equivalent to (4.105), and

$$r_{\text{volt}}(\mathbf{s}_t, \mathbf{a}_t) = \begin{cases} \lambda_{\text{volt}}(V(t) - V_{\text{max}}), & \text{if } V(t) \geq V_{\text{max}} \\ 0, & \text{otherwise} \end{cases} \quad (4.107)$$

where $V_{\text{max}} = 4.2\text{V}$ as specified in the datasheet and the coefficient in the voltage constraint is set to $\lambda_{\text{volt}} = -100$.

The current is limited within the range $[-2.5C, 0]$, where C is the C-rate related to the considered cell. In particular, the current constraint is imposed by considering the agent's action is inherently bounded within the open-interval $(-1, 1)$ due to the choice of its last layer as an hyperbolic tangent operator.

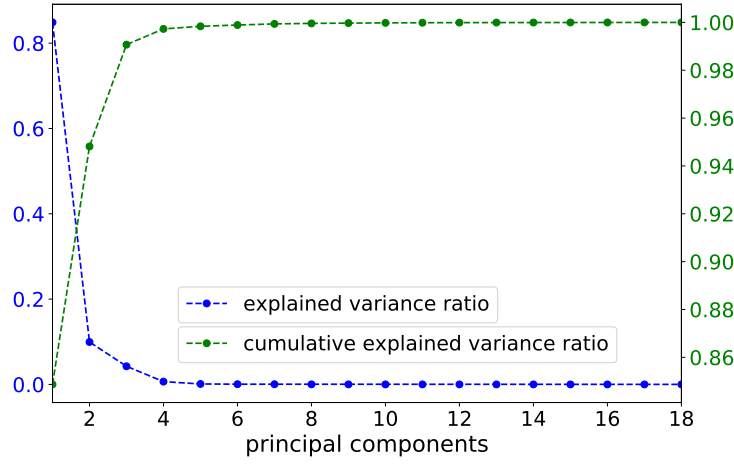


Figure 4.3: PCA results on the electrochemical model.

States-based Learning Policy

We first consider the states-based RL approach. Although deep RL has the distinct advantage of side-stepping Bellman’s curse of dimensionality for tabular optimal control, it is also known in the literature that a proper features selection procedure can significantly increase the performance [40]. Therefore, due to the fact that the electrochemical model presented in Section 2.1.1 presents a large number of states after discretization, we transform the state-space into a reduced observation space through principal component analysis (PCA) [139].

Principal Component Analysis

Suppose we have m time-series data samples for the states, $\mathbf{s} \in \mathbb{R}^n$, represented as matrix $\mathbf{S} \in \mathbb{R}^{n \times m}$. Consider a so-called “principal component” which can be expressed as:

$$P = w^T \mathbf{S}, \quad (4.108)$$

where $w \in \mathbb{R}^{n \times 1}$ is a vector of weights, $P \in \mathbb{R}^{1 \times m}$ is an arbitrary principal component. If we consider $\mathbf{S}$ as a random matrix, then we seek to choose w to maximize the variance of P

$$\text{var}(P) = w^T \mathbf{S} \mathbf{S}^T w. \quad (4.109)$$

We then formulate the following optimization problem while constraining w to have unit length:

$$\begin{aligned} \max_w \quad & w^T \mathbf{S} \mathbf{S}^T w \\ \text{subject to} \quad & w^T w = 1. \end{aligned} \quad (4.110)$$

whose solution w^* yields the first principal component according to $P = (w^*)^T \mathbf{S}$. This method can be extended to compute q principal components, in which the original data

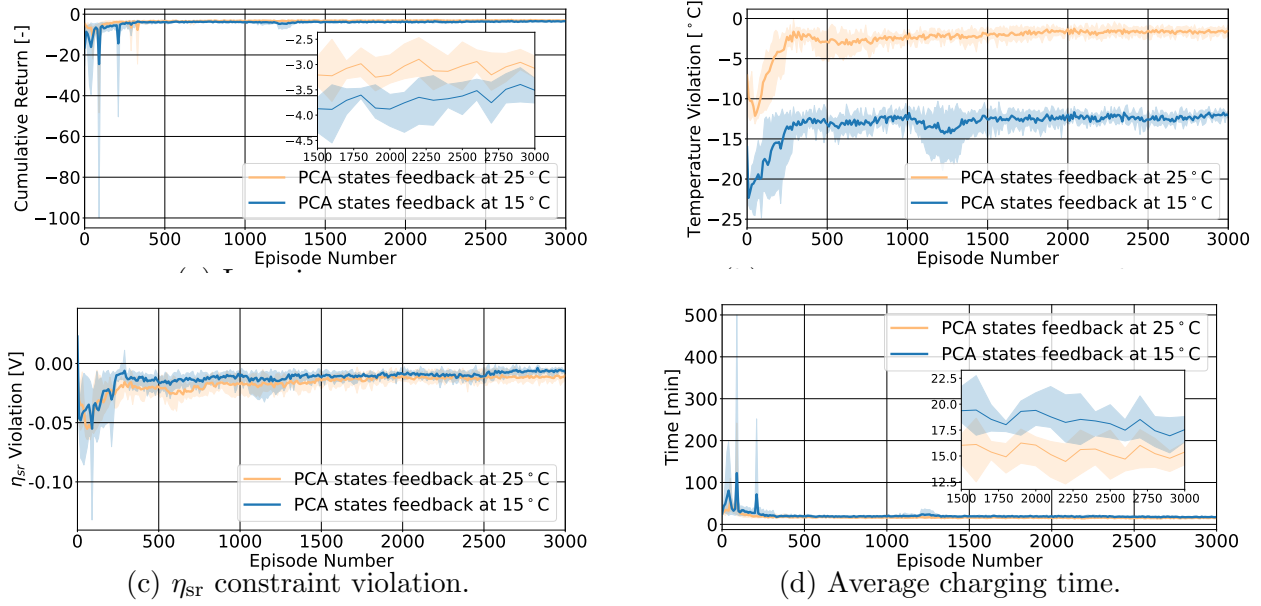


Figure 4.4: States-based learning policy results with different initializations of the Actor-Critic networks at $T_{amb} = 25^\circ\text{C}$ and $T_{amb} = 15^\circ\text{C}$.

$\mathbf{s} \in \mathbb{R}^n$ is projected onto a reduced basis of dimension q that maximizes variance [20]. The number of principal components is determined by calculating the ratio of each eigen-value to the sum of eigen-values. The most significant principal component is the one with the largest eigenvalue, which is the most informative. To see how much information we retain through PCA, we calculate the *explained variance ratio* of the principal components as follows:

$$\text{Explained variance ratio} = \frac{\lambda_j}{\sum_j \lambda_j}. \quad (4.111)$$

Figure 4.3 describes the PCA results on the electrochemical model. We are specifically interested in the dynamical states in differential equations (2.1) – (2.9) that are comprised of hundreds of states after spatial discretization. In this manuscript, we choose five principal components that explain 99.8% of the original states' evolution data. Based on the states reduction technique, the learned policy maps the PCA states to input current by interacting with the environment.

Training Results of States-based Policy

In the following we discuss the performance of the states-based RL approach applied in two different settings of environment temperature, $T_{amb} = 25^\circ\text{C}$ and $T_{amb} = 15^\circ\text{C}$. Figure 4.4 shows the training results of the actor-critic approach, where the shaded region represents the variance of the different variables for each episode over 5 different initializations of the networks parameters. The solid lines describe their average value. During the training an exploration noise is added to the agent's action and the electrochemical model is randomly initialized with $SOC^-(t_0) \in [0.2, 0.4]$ and $T(t_0) \in [25^\circ\text{C}, 32^\circ\text{C}]$, thus increasing the exploration capability.

Figure 4.4a depicts the cumulative reward by evaluating the policy every 10 episodes without the exploration noise. As it can be noticed, the cumulative reward approaches -3.08 for $T_{\text{amb}} = 25^\circ\text{C}$ and -3.51 for $T_{\text{amb}} = 15^\circ\text{C}$. It is important to consider that, although 3000 training episodes are considered, the cumulative reward converges already after 500 cycles. This number of episodes however constitutes a large part of the life-time of a standard Li-ion cell (which is about 1000 cycle). For this reason, it is impossible to conduct a complete training process directly on a real cell, but, as it is common in RL, the training phase is carried out on a very detailed simulator and then only a fine tuning of the network parameters is done on-line on the real cell during the first few cycles. Figures 4.4b – 4.4c describe the constraint violations during training. The constraint violation scores are computed according to $\max\{T(t) - T_{\text{max}}, \forall t \in [0, t_f]\}$, $\max\{0 - \eta_{\text{sr}}(t), \forall t \in [0, t_f]\}$, for each episode. Positive values imply that the constraints are violated during that particular episode. We can see the constraint violation scores approach zero as the episodes increase, which implies that the optimal control policy involves a boundary solution in which these constraints are active (i.e. true with equality during segments of the optimal trajectory). Another interesting interpretation of these figures comes by analyzing which constraint has a greater impact in achieving a fast charging protocol. For instance, the cell temperature constraint is the dominant factor when the ambient temperature is 25°C while side-reaction overpotential dominates the charging at $T_{\text{amb}} = 15^\circ\text{C}$. This fact is also reported by the literature stating that low temperatures lead to lithium plating and subsequent lithium dendritic growth [128]. Lastly, Fig. 4.4d displays the charging time versus episode number. The charging time decreases to around 15 minutes at $T_{\text{amb}} = 25^\circ\text{C}$ and 17 minutes at $T_{\text{amb}} = 15^\circ\text{C}$, on average. Note that lower ambient temperature takes more time to reach the target SOC as the side reaction overpotential constraint is prone to violation when the battery is in charging. Note, upon convergence the proposed actor-critic approach achieves the minimum time goal without violating the safety constraints.

It is also informative to examine the charging profiles of SOC, voltage, temperature and side-reaction overpotential that are obtained by testing the agent after the training process. We consider the initial conditions $\text{SOC}^-(t_0) = 0.2$ and $T(t_0) = T_{\text{amb}}$ (for both the cases of $T_{\text{amb}} = 25^\circ\text{C}$ and $T_{\text{amb}} = 15^\circ\text{C}$). The results of this analysis are shown in Fig. 4.5. Moreover, a comparison with a modified reference governor (MRG) approach [91] is provided. In the MRG approach, the applied current $I(t)$ and reference current $I^r(t)$ are related according to

$$I(t+1) = \beta(t)I^r(t), \quad \beta \in [0, 1]$$

where $\beta(t)$ is the ratio of reference value that maintains the states in the admissible set. Note that the reference value is equivalent to the allowable charging current in the RL framework, i.e., -2.5C for comparative analysis. For 25°C ambient temperature case, MRG/RL take 19/20 minutes to reach the target $\text{SOC}_{\text{ref}} = 0.8$. For 15°C ambient temperature case, MRG/RL take 21.5/22.5 minutes to reach the 0.8 target. The proposed actor-critic approach performs similar to this MRG benchmark. Note that the MRG assumes perfect knowledge of the model, and computes the forward dynamics in the optimization to achieve the goal. In contrast the states-based actor-critic method is completely model-free. Now, this requires careful

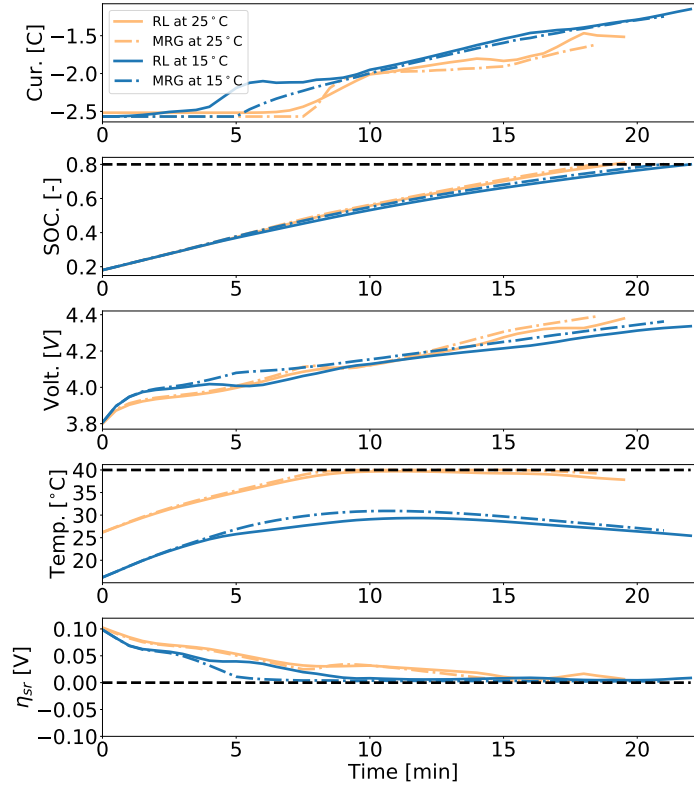


Figure 4.5: Validation of states-based learning policy compared with modified reference governor at $T_{\text{amb}} = 25^{\circ}\text{C}$ and $T_{\text{amb}} = 15^{\circ}\text{C}$.

qualification. The actor-critic method in this study obviously interacts with a simulation model via iterative episodes. In practice, the actor-critic method could interact with a physical battery, thus avoiding the modeling step. In either case, the results in Fig. 4.5 assume full state measurement. In a realistic scenario in which only outputs – voltage, temperature and, in the case of coulomb-counting approximation, also SOC – can be measured, the use of a model-based observer becomes necessary if the states-based RL configuration is adopted. Due to the fact that the development of such a model-based states estimator is a challenging task, we provide in the subsequent section an alternative, a completely model-free RL strategy which relies on outputs measurements only.

Outputs-based Learning Policy

The previously discussed states-based RL charging strategy requires a state estimator to estimate the internal states from the output measurements. However, this is a formidable task as the mathematical structure of DFN model is formulated as nonlinear DAEs. For this reason, most of the relevant works in the literature on design estimators and controllers are based on reduced order models. As an alternative we present, for the first time to the authors' knowledge, a model-free RL charging algorithm based on output measurements only.

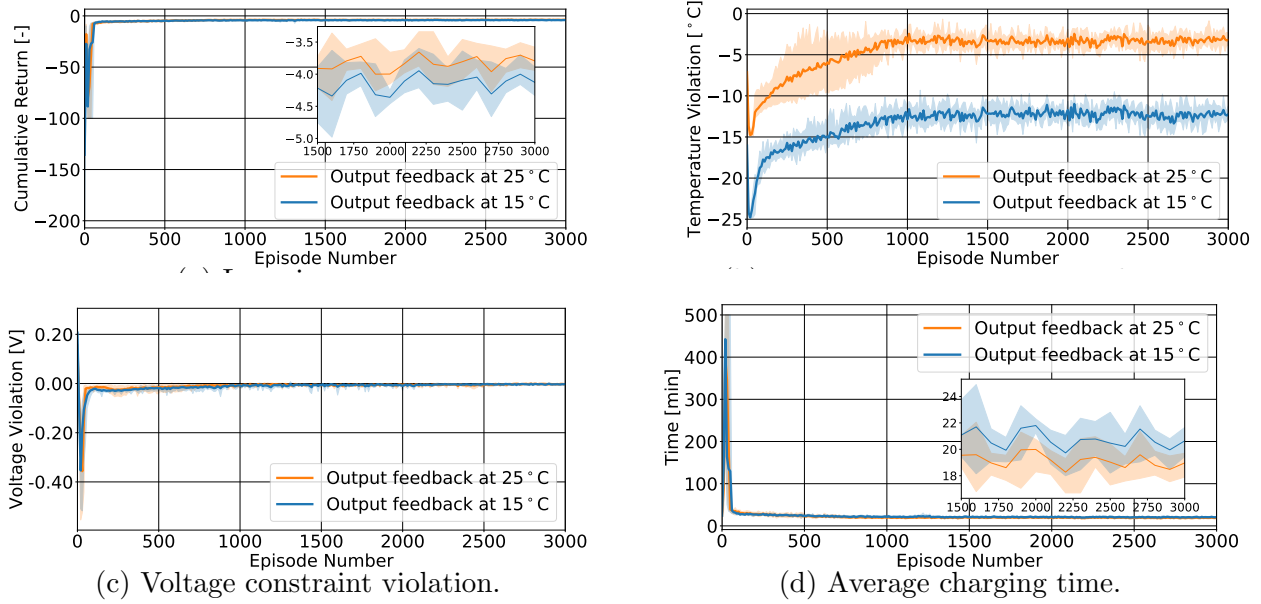


Figure 4.6: Outputs-based learning policy results with different initialization of Actor-Critic networks at $T_{\text{amb}} = 25^\circ\text{C}$ and $T_{\text{amb}} = 15^\circ\text{C}$.

In particular, in this section we provide voltage, temperature and the SOC (retrieved through coulomb-counting) as observations to the deep RL agent.

Figure 4.6 presents the training results for outputs-based learning policy using the actor-critic approach at 25°C and 15°C ambient temperature. All the settings are consistent with previous case studies, except for the observations provided to the actor-critic networks, that here consist of output measurements only. Also, it is important to notice that the safety constraint which enforces a positive side-reaction overpotential is substituted with the more conservative one that limits the terminal voltage below a predefined threshold, as discussed in Section 4.2.4. In Fig. 4.6a the performance of the learned policy are tested every 10 episodes with exploration noise removed. The cumulative reward approaches -3.79 and -4.13 for the two ambient temperatures respectively. Simulation results confirm that the proposed actor-critic approach can be applied for outputs-based controller design without safety violations.

In particular, Fig. 4.6b – 4.6c describe the constraint violations during training. The violation scores are calculated according to

$$\begin{aligned} & \max \{V(t) - V_{\text{max}}, \forall t \in [0, t_f]\} \\ & \max \{T(t) - T_{\text{max}}, \forall t \in [0, t_f]\} \end{aligned}$$

for each episode. For the outputs-based policy, the voltage constraint is dominant, at least for the battery model and parameters considered here. The agent is able to learn to ride the voltage constraint while minimizing the charging time. This is not surprising, since it is known that the voltage constraint is particularly conservative. Lastly, Fig. 4.6d displays the

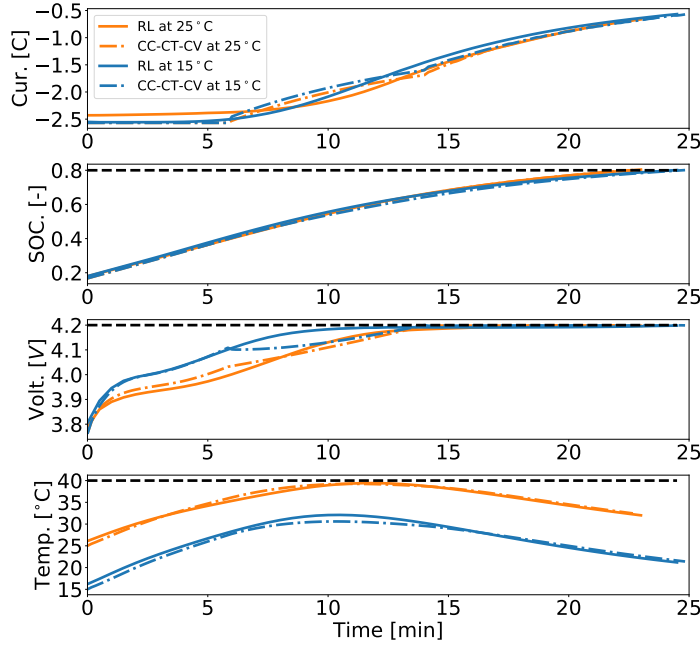


Figure 4.7: Validation of outputs-based learning policy compared with CC-CT-CV at $T_{\text{amb}} = 25^\circ\text{C}$ and $T_{\text{amb}} = 15^\circ\text{C}$.

charging time for achieving the reference SOC. The charging time decreases to around 19 minutes at $T_{\text{amb}} = 25^\circ\text{C}$ and 21 minutes at $T_{\text{amb}} = 15^\circ\text{C}$ ambient temperature, on average. We also observe that the charging time takes longer when the ambient temperature is lower, which is consistent with the states-based policy.

In the following, we focus on the profiles of the key battery variables that are obtained by applying the optimal policy achieved at the end of the training process. Moreover, we compare to another output-feedback model-free scheme that is based on PI controller and implements a CC-CT-CV protocol. The CC-CT-CV algorithm is defined as

$$I(t) = \begin{cases} I_{ref}, & t \leq t_{pk} \\ I_{ref} + K_p e(t) + K_i \sum_{\tau=0}^t e(\tau), & t_{pk} \leq t \leq t_{cv} \\ I_{cv}, & t > t_{cv} \end{cases}$$

where $e(t)$ is the controller error, i.e., $e(t) = T(t) - T^*$, t_{pk} is the period where constant peak current is applied, i.e., $-2.5C$, and t_{cv} is the time when the voltage reaches its maximum. In CV mode, the current is decreasing in an exponential-like fashion, while the voltage is kept constant until the target SOC is reached. The control gains, K_p and K_i are obtained by human trial and error. The main defining feature of CC-CT-CV method is the fact that a temperature constraint is considered, thus enabling a safer charging procedure compared to the conventional CC-CV approach [90]. In this study, we are interested in assessing the RL

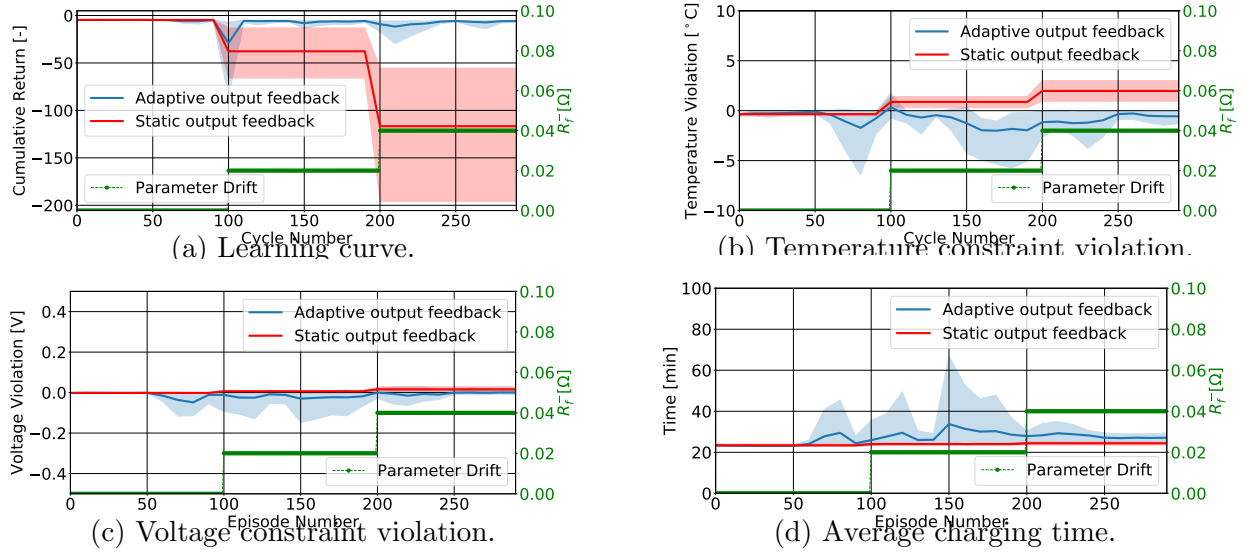


Figure 4.8: Adaptive outputs-based learning policy results with different initialization of Actor-Critic networks at $T_{\text{amb}} = 25^\circ\text{C}$ and $T_{\text{amb}} = 15^\circ\text{C}$.

performance in terms of charging time and state violations when compared to the CC-CT-CV approach.

We use the same configuration of the previous case study for the initial conditions of the battery. The comparison is shown in Fig. 4.7. For 25°C ambient temperature case, CC-CT-CV/RL take 23/23.5 minutes to reach $SOC_{\text{ref}} = 0.8$, while for the case of $T_{\text{amb}} = 15^\circ\text{C}$ the charging time increases to 25/25 minutes. We conclude that the outputs-based actor-critic approach can be used to design an output feedback control for battery fast-charging with similar performance to CC-CT-CV. This is due to the fact that the CC-CT-CV profile seems to be the optimal charging procedure in presence of temperature and voltage constraints. However, the main limitation of the CC-CT-CV relies in the fact that an accurate tuning of the controller is required. Therefore, a possible solution may be the exploitation of RL strategies for the automatic tuning of the PID coefficients in the CC-CT-CV framework. Although we have shown that RL only provided equivalent performance to existing methods when addressing the charging of a fresh battery, we aim to highlight in the next section its ability to adapt the charging control policy in the face of battery aging.

Adaptive Outputs-based Learning Policy

One of the main challenges facing battery management system development is the controller's adaptability to changes in the cell behavior as it cycles. The proposed deep reinforcement learning framework exhibits the advantage of learning the optimal policy by directly interacting with the surrounding environment. Specifically, it can adapt to slow changes in the battery parameters due to aging. First, we analyze the adaptability of RL by comparing the fixed

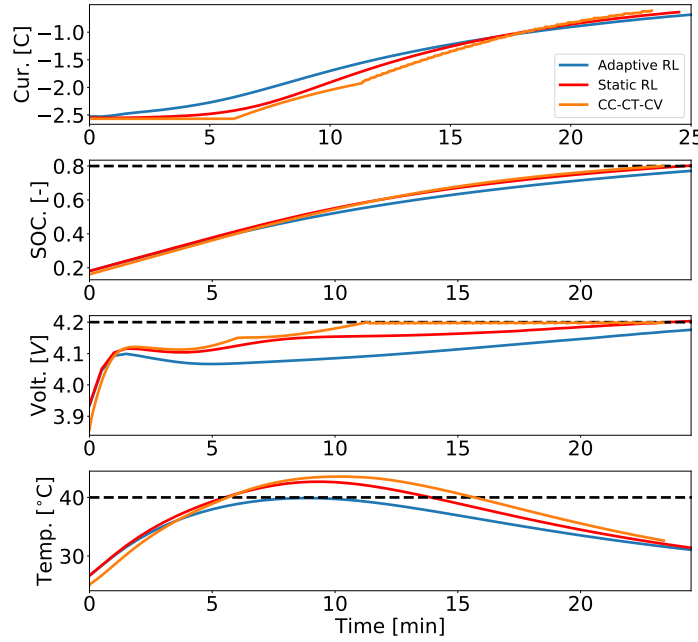


Figure 4.9: Validation of adaptive outputs-based learning policy compared with CC-CT-CV at $T_{\text{amb}} = 25^\circ\text{C}$ and $R_f^- = 0.04\Omega$.

parameters of actor-critic called *static* RL policy with an *adaptive* RL policy that continues to adjust the actor-critic network parameters as the cell ages. Note that exploration noise is not considered in this adaptive study because the controller is updated from the estimates of the critic, not an exploration.

Figure 4.8 describes the results in terms of rewards, state violation, and average charging time. To emulate the ageing mechanism, we increase the film resistance in the anode, R_f^- , every 100 cycles. The actor-critic networks are initialized by the last training results from previous study. Both the static and adaptive RL policies achieve the same cumulative return for the first 100 cycles, since there is no parameter changes in the environment. The static outputs-based controller fails as the battery is aged, while the adaptive one adapts its control policy through learning as shown in Fig. 4.8a. The drops in the reward can be explained by states violation in figures 4.8b – 4.8c. We can observe that the RL policy is capable of adapting by updating the parameters of actor-critic from the reward (penalty). Figure 4.8d indicates that the charging time increases from 23.4 minutes to 27.1 minutes on average due to battery aging. Secondly, we validate the adaptability of RL method by comparing with CC-CT-CV, as shown in Fig. 4.9. From the figure, we can clearly see that adaptive RL can satisfy the constraints while static RL and CC-CT-CV violate the temperature constraint during the charging process. Consequently, RL (actor-critic) demonstrates a distinct advantage as an adaptive battery fast-charging controller that adjust as the cell ages.

Discussion and Remarks

We show that the proposed RL framework can be utilized for solving the battery fast charging problem using both states-based and outputs-based frameworks. The overall results are summarized in Table 4.4. We make several remarks in this section for readers. First, the availability of full state information enables faster charge times than output feedback only, since one can consider the side-reaction overpotential constraint instead of the more conservative voltage constraint. This motivates the design of state/parameter estimators as a very important component for reducing the charging time, and carefully monitoring immeasurable degradation mechanisms such as lithium plating. Second, temperature plays a significant role in the battery management system, as it affects both side-reaction overpotential and terminal voltage. Thermal management is another key component for battery management systems. Lastly, deep reinforcement learning, commonly associated with artificial intelligence, can be applied to battery management systems. This work demonstrates that RL is comparable with other existing approaches and its adaptability to aging is well-suited for battery fast-charging application.

Strategy	Condition	States-based		Outputs-based	
		RL	MRG	RL	CC-CT-CV
Model-based	25°C	20 min	19 min	-	-
	15°C	22.5 min	21.5 min	-	-
Model-free	25°C	-	-	23.5 min	23 min
	15°C	-	-	25 min	25 min
Adaptability	25°C	YES	-	YES	NO

Table 4.4: Summary of charging time [min.] and adaptability

4.2.6 Summary

In this paper, we propose a model-free deep reinforcement learning framework for solving the battery fast-charging problem in the presence of safety constraints, when a detailed electrochemical model is used as a battery simulator. Among the RL paradigms, the actor-critic scheme and specifically the DDPG algorithm has been adopted due to its ability to deal with continuous state and action spaces. To address the state constraints, the reward function has been designed such that the agent learns constraint violations. First, we assume full state measurements and compare a states-based RL algorithm with a reference governor approach considered as state-of-the-art benchmark. Subsequently, a more realistic scenario in which only output measurements are available is taken into account. In this case, instead of relying on a state estimator, which is challenging to design, we formulate an outputs-based configuration of the same RL approach. This relies on measurements of voltage, temperature, and SOC (via coulomb counting). This latter strategy is compared against a CC-CT-CV approach, which can be considered a benchmark state-of-the-art output feedback model-free

approach. For both the RL formulations, simulation results show that RL performs similarly to the state-of-the-art benchmarks. Finally, the main advantage of RL is adaptation. Namely, we analyze the outputs-based RL strategy in the presence of changing battery parameters that mimic battery aging. The results highlight that the proposed approach is able to achieve reasonable performance as the environment changes according to ageing throughout the whole battery life, while the CC-CT-CV approach eventually violates safety constraints. Future work involves experimental validation of proposed framework using hardware-in-the-loop testing environment with different objectives.

Chapter 5

Conclusion

This dissertation presents electrochemical model-based estimation, control, and learning techniques for lithium-ion batteries. The main contribution of this dissertation and opportunities for future work are discussed in this chapter.

5.1 Contributions

- The full-order electrochemical model provides highly reliable and accurate modeling scheme for lithium-ion batteries which comprises of coupled partial differential equations, ordinary differential equations, and algebraic equations. The model gives physical insights to understand the mechanisms inside the battery, however, it is too complicated to apply micro controller in real time. The reduced-order models can be constructed for control/estimation purposes with mild assumptions on the electrochemical model. Furthermore the gap between reduced-order model and actual battery can be filled with data-driven modeling techniques using dynamical neural networks.
- The system identification of lithium-ion battery cell is crucial aspect to the battery system management. The convergence speed of identification for electrochemical parameters can be accelerated by designing optimal input profiles that maximize Fisher information matrix. The sensitivity analysis for given dynamical system ranks the parameter's identifiability and provides confidence interval for estimated parameters via nonlinear regression. The batter cell characterization framework is experimentally validated with the conventional characterization process. This sensitivity analysis can be further expanded to battery health, namely, capacity and power fade due to degradation mechanisms during operation.
- Optimal charging strategy can prolong the battery's life span. For identified electrochemical models, the optimal control problem for battery fast charging is formulated. The goal of battery fast charging is to reach to the target state of charge as soon as possible while critical electrochemical variables are not violated. The model-based

optimal control provides theoretical optimality of derived charging current, however, the model needs to be simple. The learning-based controller facilitates states/outputs feedback controller scheme under the model uncertainty. It has unique feature of adaptability for which model-based controller can not afford.

5.2 Future Research Directions

- Hybrid electrochemical modeling can be further extended to capture the degradation mechanisms by learning from data. The initial model can be identified by the proposed characterization framework, however, the model accuracy is depreciated throughout the cycles. This is because the electrochemical properties of the lithium-ion battery have changed due to degradation mechanism. There are many factors to describe the aging mechanisms, but it's difficult to predict the remaining of useful lifetime for the battery because the speed of aging process is usually determined by user behavior. For instance, heavy user might experience faster battery degradation because the battery is used aggressively. So, this user-oriented battery degradation mechanisms can be explained by the historical data.
- Thermal management of lithium ion battery is another key aspect of advanced battery management system. It is highly recommended considering the thermal model parameters simultaneously with the dynamical electrochemical parameters, since their dynamics are tightly coupled. To further advance battery characterization work discussed in Chapter 3, thermal parameters, i.e., activation energy, heat capacity, etc should be estimated by designing optimal thermal & current inputs.
- State of charge estimation has been active research topic for previous decade, now the research focus has moved on to the state of health estimation. The battery degradation mechanisms are complicated and hard to predict. Electrochemical model-based state of health (SoH) estimation is the future of battery management system. This can be captured by data-driven models or by estimating the key parameters that constitute battery health.

Appendix A

Nomenclature

Table A.1: List of symbols (1/2).

Symbol	Description [SI units]
a_s	Specific interfacial surface area [$\text{m}^2 \text{ m}^{-3}$]
A	Electrode Area [m^2]
c_s	Lithium concentration in solid phase [mol m^{-3}]
c_{ss}	Lithium concentration at surface in solid phase [mol m^{-3}]
c_e	Lithium concentration in electrolyte phase [mol m^{-3}]
C	Heat capacity [$\text{J}/(\text{m}^2\text{-K})$]
$\frac{d \ln f_{c/a}}{d \ln c_e}$	Activity coefficient [-]
D_s	Diffusion coefficient in solid phase [$\text{m}^2 \text{ s}^{-1}$]
D_e	Diffusion coefficient in electrolyte phase [$\text{m}^2 \text{ s}^{-1}$]
E	Activation energy [kJ/mol]
F	Faraday constant [$96487 \text{ Coulomb mol}^{-1}$]
h	Heat transfer coefficients [$\text{W}/(\text{m}^2\text{-K})$]
i	Ionic current [A m^{-2}]
I	Applied current density [A m^{-2}]
j	Molar ion flux [$\text{mol} (\text{m}^2 \text{ s})^{-1}$]
k	Kinetic rate constants [$(\text{A m}^{-2})(\text{m}^3 \text{ mol}^{-1})^{(1+\alpha)}$]
L	Thickness of electrode [m]
$n_{Li,s}$	cyclable lithium in the solid phase [mol]
$\dot{Q}$	Heat generation [$\text{W} (\text{m}^{-2})$]
$Q^\pm$	Electrode capacity [Ah]
Q	Voltage output variance [V^2]
R	Universal gas constant [$8.3145 \text{ J} (\text{mol K})^{-1}$]
R_c	Resistance of connectors [$\Omega \text{ m}^2$]
R_s	Radius of solid particles [m]
R_f	Solid-electrolyte inter-phase film resistance [$\Omega \text{ m}^2$]
t	Time [seconds]

Table A.2: List of symbols (2/2).

Symbol	Description [SI units]
t_c^0	Transference number [-]
T	Temperature [K]
T_1	Temperature at core [K]
T_2	Temperature at surface [K]
T_{amb}	Ambient temperature [K]
u	Input magnitude [-]
U	Open circuit potential of solid material [V]
V	Terminal voltage [V]
α	Charge transfer coefficients [-]
ε_s	Volume fraction in solid phase [-]
ε_e	Volume fraction in electrolyte phase [-]
η	Overpotential of an electrode [V]
θ^-	Negative electrode stoichiometry [-]
θ^+	Positive electrode stoichiometry [-]
κ	Electrolyte conductivity [S m^{-1}]
σ	Solid phase conductivity [S m^{-1}]
ϕ	Electric potential [V]
$\mathbf{x}$	State variable vector [-]
$\mathbf{y}$	Output vector [-]
$\mathbf{z}$	Algebraic variable vector [-]
$\mathbf{S}$	Sensitivity vector w.r.t parameters [-]
$\boldsymbol{\theta}$	Parameter vector [-]

Appendix B

DFN Simulator

In this section, we describe the process how to solve DFN model numerically, and exhibit the states evolution over time.

B.1 Numerical Implementation

The DFN model described in Section 2.1 can be formulated as DAE system such as:

$$\dot{x} = f(x, z, u), \quad (\text{B.1})$$

$$0 = g(x, z, u) \quad (\text{B.2})$$

with initial conditions $x(0), z(0)$ that are consistent. That is, they verify (B.2). The function $f(x, z, u)$ and $g(x, z, u)$ are computed in Matlab/Python function. The time-stepping is done by solving the nonlinear equation

$$0 = F(x(t + \Delta t), z(t + \Delta t)), \quad (\text{B.3})$$

$$0 = \begin{bmatrix} x(t) - x(t + \Delta t) + \frac{1}{2}\Delta t [f(x(t), z(t), u(t)) + f(x(t + \Delta t), z(t + \Delta t), u(t + \Delta t))] \\ g(x(t + \Delta t), z(t + \Delta t), u(t + \Delta t)) \end{bmatrix} \quad (\text{B.4})$$

for $x(t + \Delta t), z(t + \Delta t)$. The function returns the solution $(x(t + \Delta t), z(t + \Delta t))$ of (B.3)-(B.4), given $x(t), z(t), u(t), u(t + \Delta t)$. Note that we solve (B.3)-(B.4) using Newton's method, meaning analytic Jacobians of $F(\cdot, \cdot)$ are required w.r.t. x, z .

$$J = \begin{bmatrix} F_x^1 & F_z^1 \\ F_x^2 & F_z^2 \end{bmatrix} \quad (\text{B.5})$$

$$= \begin{bmatrix} -I + \frac{1}{2}\Delta t \cdot \frac{\partial f}{\partial x}(x(t + \Delta t), z(t + \Delta t), u(t + \Delta t)) & \frac{1}{2}\Delta t \cdot \frac{\partial f}{\partial z}(x(t + \Delta t), z(t + \Delta t), u(t + \Delta t)) \\ \frac{\partial g}{\partial x}(x(t + \Delta t), z(t + \Delta t), u(t + \Delta t)) & \frac{\partial g}{\partial z}(x(t + \Delta t), z(t + \Delta t), u(t + \Delta t)) \end{bmatrix} \quad (\text{B.6})$$

The analytic Jacobian can be computed by hands or automatic differential toolbox.

DAE System

To perform the time-stepping in the previous section, we must compute functions $f(x, z, u)$ and $g(x, z, u)$. These functions, which represent the RHS of (B.1)-(B.2), are calculated given the inputs x, z, u . The role of variables x, z, u are played by the DFN variables shown in Table B.1.

Table B.1: DAE notation for DFN states.

DAE Variable	DFN Variable
x	c_s^-, c_s^+, c_e, T
z	$\phi_s^-, \phi_s^+, i_e^-, i_e^+, \phi_e, j_n^-, j_p^+$
u	I

In the subsequent sections, we go through each DFN variable listed in Table B.1 and document its numerical implementation.

Solid Concentration, c_s^-, c_s^+

The PDEs (2.1) and boundary conditions (2.15) governing Fickian diffusion in the solid phase are implemented using third order Padé approximations of the transfer function from $j_n^\pm$ to $c_{ss}^\pm$ [37, 97]:

$$\frac{C_{ss}^\pm(s)}{J_n^\pm(s)} = \frac{-\frac{(R_s^\pm)^3}{165(D_s^\pm)^2}s^2 - \frac{4R_s^\pm}{11D_s^\pm}s - \frac{3}{R_s^\pm}}{\frac{(R_s^\pm)^4}{3465(D_s^\pm)^2}s^3 + \frac{3(R_s^\pm)^2}{55D_s^\pm}s^2 + s}, \quad (\text{B.7})$$

which we notate more simply by

$$\frac{C_{ss}^\pm(s)}{J_n^\pm(s)} = \frac{b_2s^2 + b_1s + b_0}{a_3s^3 + a_2s^2 + a_1s + a_0}. \quad (\text{B.8})$$

Next we multiply top and bottom by $\frac{1}{a_3}$ to yield a unity coefficient on the highest-order term in the denominator, yielding

$$\frac{C_{ss}^\pm(s)}{J_n^\pm(s)} = \frac{\bar{b}_2s^2 + \bar{b}_1s + \bar{b}_0}{s^3 + \bar{a}_2s^2 + \bar{a}_1s + \bar{a}_0}, \quad (\text{B.9})$$

where $\bar{b}_i = b_i/a_3$ and $\bar{a}_i = a_i/a_3$.

The transfer function (B.9) is converted into controllable canonical state-space form, thus producing the subsystem:

$$\frac{d}{dt} \begin{bmatrix} c_{s1}^\pm(t) \\ c_{s2}^\pm(t) \\ c_{s3}^\pm(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -\bar{a}_0 & -\bar{a}_1 & -\bar{a}_2 \end{bmatrix} \begin{bmatrix} c_{s1}^\pm(t) \\ c_{s2}^\pm(t) \\ c_{s3}^\pm(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} j_n^\pm(t) \quad (\text{B.10})$$

$$\begin{bmatrix} c_{ss}^\pm(t) \end{bmatrix} = \begin{bmatrix} \bar{b}_0 & \bar{b}_1 & \bar{b}_2 \end{bmatrix} \begin{bmatrix} c_{s1}^\pm(t) \\ c_{s2}^\pm(t) \\ c_{s3}^\pm(t) \end{bmatrix} \quad (\text{B.11})$$

for each discrete point in x .

To simplify analytical applications for the DFN model, we seek a different state-space realization in which bulk concentration $\bar{c}_s^\pm(t)$ is expressed as a state. This can be found by pursuing a Jordan-form state-space realization, which diagonalizes the system matrix. The diagonal elements represent the system eigenvalues. The zero eigenvalue corresponds to bulk concentration. All the aforementioned calculations are performed in Matlab function `c_s_mats.m`. The final result produces system matrices:

$$\frac{d}{dt} \begin{bmatrix} c_{s1}^\pm(t) \\ c_{s2}^\pm(t) \\ c_{s3}^\pm(t) \end{bmatrix} = A_{cs}^\pm \begin{bmatrix} c_{s1}^\pm(t) \\ c_{s2}^\pm(t) \\ c_{s3}^\pm(t) \end{bmatrix} + B_{cs}^\pm j_n^\pm(t) \quad (\text{B.12})$$

$$\begin{bmatrix} c_{ss}^\pm(t) \\ \bar{c}_s^\pm(t) \end{bmatrix} = C_{cs}^\pm \begin{bmatrix} c_{s1}^\pm(t) \\ c_{s2}^\pm(t) \\ c_{s3}^\pm(t) \end{bmatrix} \quad (\text{B.13})$$

where the second row of $C_{cs}^\pm$ is $[C_{cs}^\pm]_2 = [0, 0, 1]$. Note we have abused notation. The states in (B.12)-(B.13) are not the same as those in (B.10)-(B.11), due to the different realizations. Also, these matrices must be computed online, due to the temperature dependence of $D_s^\pm$ described in (2.48).

Electrolyte Concentration, c_e

The electrolyte concentration PDE (2.2) combined with (2.5), and BCs (2.16)-(2.20) are implemented using the central difference method, which ultimately produces the matrix differential equation:

$$\frac{d}{dt} c_e^j(t) = \frac{dD_e^{\text{eff},j}}{dc_e}(c_e^j) [M_{ce}^{j,1} c_e^j(t) + M_{ce}^{j,2} c_{e,bc}^j(t)]^2 + D_e^{\text{eff},j}(c_e^j) [M_{ce}^{j,3} c_e^j(t) + M_{ce}^{j,4} c_{e,bc}^j(t)] + M_{ce}^{j,5} j_n^j(t) \quad (\text{B.14})$$

where c_e, j_n^j are vectors whose elements represent discrete points along the x -dimension of the DFN model. The variable $c_{e,bc}^j$ represents the boundary values for region j , namely $c_{e,bc}^- = [c_{e,bc,1}, c_{e,bc,2}]^T$, $c_{e,bc}^{\text{sep}} = [c_{e,bc,2}, c_{e,bc,3}]^T$, $c_{e,bc}^+ = [c_{e,bc,3}, c_{e,bc,4}]^T$. The boundary values are computed by: $c_{e,bc}(t) = C_{ce} c_e(t)$ where $c_{e,bc} = [c_{e,bc,1}, c_{e,bc,2}, c_{e,bc,3}, c_{e,bc,4}]^T$.

Note that the effective diffusivity, $D_e^{\text{eff},j}(c_e^j)$, and its derivative, $\frac{dD_e^{\text{eff},j}}{dc_e}(c_e^j)$, are state-dependent and must be computed online. The matrices $M_{ce}^{j,1}, M_{ce}^{j,2}, M_{ce}^{j,3}, M_{ce}^{j,4}, M_{ce}^{j,5}, C_{ce}$ are

computed offline by Matlab/Python function block. These matrices are given by

$$M_{ce}^{j,1} = \frac{1}{2L^j \Delta x^j} \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ -1 & 0 & 1 & \dots & 0 \\ 0 & -1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & & \\ 0 & 0 & \dots & \dots & 1 \\ 0 & 0 & \dots & -1 & 0 \end{bmatrix}, \quad M_{ce}^{j,2} = \frac{1}{2L^j \Delta x^j} \begin{bmatrix} -1 & 0 \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ 0 & 1 \end{bmatrix}, \quad (\text{B.15})$$

$$M_{ce}^{j,3} = \frac{1}{(L^j \Delta x^j)^2} \begin{bmatrix} 1 & -2 & 1 & \dots & 0 \\ 0 & 1 & -2 & \dots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & 1 & -2 & 1 \\ 0 & 0 & \dots & 0 & 1 & -2 \end{bmatrix}, \quad M_{ce}^{j,4} = \frac{1}{(L^j \Delta x^j)^2} \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ 0 & 1 \end{bmatrix}, \quad (\text{B.16})$$

$$M_{ce}^{j,5} = \frac{(1 - t_c^0) a_s^j}{\varepsilon_e^j} \mathbb{I}. \quad (\text{B.17})$$

for $j \in \{-, \text{sep}, +\}$. The matrix C_{ce} is given by

$$C_{ce} = -(N_{ce}^2)^{-1} (N_{ce}^1) \quad (\text{B.18})$$

where

$$N_{ce}^1 = \left[\begin{array}{ccccc|ccccc} 4 & -1 & \dots & 0 & 0 & 0 & 0 & \dots & \dots \\ 0 & 0 & \dots & \frac{(\varepsilon_e^-)_{\text{brug}}}{2L^- \Delta x^-} & -4 \frac{(\varepsilon_e^-)_{\text{brug}}}{2L^- \Delta x^-} & -4 \frac{(\varepsilon_e^{\text{sep}})_{\text{brug}}}{2L^{\text{sep}} \Delta x^{\text{sep}}} & \frac{(\varepsilon_e^{\text{sep}})_{\text{brug}}}{2L^{\text{sep}} \Delta x^{\text{sep}}} & \dots & \dots \\ 4 & 0 & \dots & 0 & 0 & 0 & 0 & \dots & \dots \\ 0 & 0 & \dots & 0 & 0 & 0 & 0 & \dots & \dots \\ \dots & 0 & & 0 & & 0 & 0 & \dots & 0 & 0 \\ \dots & 0 & & 0 & & 0 & 0 & \dots & 0 & 0 \\ \dots & \frac{(\varepsilon_e^{\text{sep}})_{\text{brug}}}{2L^{\text{sep}} \Delta x^{\text{sep}}} & & -4 \frac{(\varepsilon_e^{\text{sep}})_{\text{brug}}}{2L^{\text{sep}} \Delta x^{\text{sep}}} & & -4 \frac{(\varepsilon_e^+)_{\text{brug}}}{2L^+ \Delta x^+} & \frac{(\varepsilon_e^+)_{\text{brug}}}{2L^+ \Delta x^+} & \dots & 0 & 0 \\ \dots & 0 & & 0 & & 0 & 0 & \dots & 1 & -4 \end{array} \right] \quad (\text{B.19})$$

$$N_{ce}^2 = \left[\begin{array}{ccccc} -3 & 0 & 0 & 0 & 0 \\ 0 & 3 \frac{(\varepsilon_e^-)_{\text{brug}}}{2L^- \Delta x^-} + 3 \frac{(\varepsilon_e^{\text{sep}})_{\text{brug}}}{2L^{\text{sep}} \Delta x^{\text{sep}}} & 0 & 0 & 0 \\ 0 & 0 & 3 \frac{(\varepsilon_e^{\text{sep}})_{\text{brug}}}{2L^{\text{sep}} \Delta x^{\text{sep}}} + 3 \frac{(\varepsilon_e^+)_{\text{brug}}}{2L^+ \Delta x^+} & 0 & 0 \\ 0 & 0 & 0 & 0 & 3 \end{array} \right]. \quad (\text{B.20})$$

Note, we have used second-order accurate finite difference approximations for the boundary conditions.

Temperature, T

For lumped thermal dynamics, the temperature is scalar, so the ODE is directly implemented as:

$$\rho^{\text{avg}} c_P \frac{dT}{dt}(t) = h_{\text{cell}} [T_{\text{amb}}(t) - T(t)] + I(t)V(t) - \int_{0^-}^{0^+} a_s F j_n(x, t) \Delta T(x, t) dx, \quad (\text{B.21})$$

$$\Delta T(x, t) = U^\pm(\bar{c}_s^\pm(x, t)) - T(t) \frac{\partial U^\pm}{\partial T}(\bar{c}_s^\pm(x, t)), \quad (\text{B.22})$$

$$\bar{c}_s^\pm(x, t) = \frac{3}{(R_s^\pm)^3} \int_0^{R_s^\pm} r^2 c_s^\pm(x, r, t) dr \quad (\text{B.23})$$

Solid Potential, ϕ_s^-, ϕ_s^+

The solid potential ODE (2.3) and BCs (2.21) are implemented using the central difference method, which ultimately produces the matrix equation:

$$\frac{d}{dt} \phi_s^-(t) = F_{psn}^1 \phi_s^-(t) + F_{psn}^2 i_{e, \text{aug}}^-(t) + G_{psn} I(t) = 0 \quad (\text{B.24})$$

$$\frac{d}{dt} \phi_s^+(t) = F_{psp}^1 \phi_s^+(t) + F_{psp}^2 i_{e, \text{aug}}^+(t) + G_{psp} I(t) = 0. \quad (\text{B.25})$$

where $i_{e, \text{aug}}^\pm$ are

$$i_{e, \text{aug}}^-(t) = \begin{bmatrix} 0 \\ i_e^-(x, t) \\ I(t) \end{bmatrix}, \quad i_{e, \text{aug}}^+(t) = \begin{bmatrix} I(t) \\ i_e^+(x, t) \\ 0 \end{bmatrix} \quad (\text{B.26})$$

This section also computes the terminal voltage $V(t)$ from (2.27) using matrix equations

$$\phi_{s, bc}^-(t) = C_{psn} \phi_s^-(t) + D_{psn} I(t), \quad (\text{B.27})$$

$$\phi_{s, bc}^+(t) = C_{psp} \phi_s^+(t) + D_{psp} I(t), \quad (\text{B.28})$$

$$V(t) = \phi_{s, bc, 2}^+(t) - \phi_{s, bc, 1}^-(t) - R_c I(t) \quad (\text{B.29})$$

where the following matrices are computed a priori by Matlab/Python function

$$(F1n) = (M1n) - (M2n)(N2n)^{-1}(N1n), \quad (\text{B.30})$$

$$(F2n) = (M3n), \quad (\text{B.31})$$

$$(Gn) = (M4n) - (M2n)(N2n)^{-1}(N3n), \quad (\text{B.32})$$

$$(F1p) = (M1p) - (M2p)(N2p)^{-1}(N1p), \quad (\text{B.33})$$

$$(F2p) = (M3p), \quad (\text{B.34})$$

$$(Gp) = (M4p) - (M2p)(N2p)^{-1}(N3p), \quad (\text{B.35})$$

$$(Cn) = -(N2n)^{-1}(N1n), \quad (\text{B.36})$$

$$(Dn) = -(N2n)^{-1}(N3n), \quad (\text{B.37})$$

$$(Cp) = -(N2p)^{-1}(N1p), \quad (\text{B.38})$$

$$(Dp) = -(N2p)^{-1}(N3p), \quad (\text{B.39})$$

where the (Mij) and $N(ij)$ matrices result from central difference approximations of the ODE in space (2.3) and boundary conditions (2.21).

$$(M1j) = \begin{bmatrix} 0 & \alpha_j & 0 & \dots & 0 \\ -\alpha_j & 0 & \alpha_j & \dots & 0 \\ 0 & -\alpha_j & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & & \\ 0 & 0 & \dots & \dots & \alpha_j \\ 0 & 0 & \dots & -\alpha_j & 0 \end{bmatrix}, \quad (M2j) = \begin{bmatrix} -\alpha_j & 0 \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ 0 & \alpha_j \end{bmatrix}, \quad (\text{B.40})$$

$$(M3j) = \frac{1}{\sigma_{\text{ref},\pm}} \begin{bmatrix} 0 & -1 & 0 & \dots & 0 \\ 0 & 0 & -1 & \dots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & -1 & 0 & 0 \\ 0 & 0 & \dots & 0 & -1 & 0 \end{bmatrix}, \quad (M4j) = \frac{1}{\sigma_{\text{ref},\pm}} \mathbb{I} \quad (\text{B.41})$$

$$(N1j) = \begin{bmatrix} 4\alpha_j & -\alpha_j & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & \dots & 1\alpha_j & -4\alpha_j \end{bmatrix}, \quad (N2j) = \begin{bmatrix} -3\alpha_j & 0 \\ 0 & 3\alpha_j \end{bmatrix}, \quad (\text{B.42})$$

$$(N3n) = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad (N3p) = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad (\text{B.43})$$

for $j \in \{n, p\}$, $\alpha_j = 1/(2L^j \Delta x^j)$. Note, we have used second-order accurate finite difference approximations for the boundary conditions.

Electrolyte Current, i_e^-, i_e^+

The electrolyte current ODE (2.5) and BCs (2.25) are implemented using the central difference method, which ultimately produces the matrix equation:

$$\frac{d}{dt} i_e^-(t) = F_{ien}^{1-} i_e^-(t) + F_{ien}^{2-} j_n^-(t) + F_{ien}^{3-} I(t) \quad (\text{B.44})$$

$$\frac{d}{dt} i_e^+(t) = F_{iep}^{1+} i_e^+(t) + F_{iep}^{2+} j_n^+(t) + F_{iep}^{3+} I(t) \quad (\text{B.45})$$

where the following matrices are computed a priori by Matlab/Python function

$$F_{ien}^{1-} = (M1n) - (M2n)(N2n)^{-1}(N1n), \quad (\text{B.46})$$

$$F_{ien}^{2-} = (M3n) - (M2n)(N2n)^{-1}(N3n), \quad (\text{B.47})$$

$$F_{ien}^{3-} = (M2n)(N2n)^{-1}(N4n), \quad (\text{B.48})$$

$$F_{iep}^{1+} = (M1p) - (M2p)(N2p)^{-1}(N1p), \quad (\text{B.49})$$

$$F_{iep}^{2+} = (M3p) - (M2p)(N2p)^{-1}(N3p), \quad (\text{B.50})$$

$$F_{iep}^{3+} = (M2p)(N2p)^{-1}(N4p) \quad (\text{B.51})$$

where the (Mij) and $N(ij)$ matrices result from central difference approximations of the ODE in space (2.5) and boundary conditions (2.25).

$$(M1j) = \begin{bmatrix} 0 & \alpha_j & 0 & \dots & 0 \\ -\alpha_j & 0 & \alpha_j & \dots & 0 \\ 0 & -\alpha_j & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & & \\ 0 & 0 & \dots & \dots & \alpha_j \\ 0 & 0 & \dots & -\alpha_j & 0 \end{bmatrix}, \quad (M2j) = \begin{bmatrix} -\alpha_j & 0 \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ 0 & \alpha_j \end{bmatrix}, \quad (M3j) = -\beta_j \mathbb{I}, \quad (\text{B.52})$$

$$(N1j) = \begin{bmatrix} 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix}, \quad (N2j) = \mathbb{I}, \quad (N3j) = (N1j), \quad (\text{B.53})$$

$$(N4n) = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad (N4p) = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad (\text{B.54})$$

for $j \in \{n, p\}$, $\alpha_j = (2L^j \Delta x^j)^{-1}$, $\beta_j = a_s^j F$.

Electrolyte Potential, ϕ_e

The electrolyte potential is implemented using the central difference method, which ultimately produces the matrix equation:

$$\frac{d}{dt} \phi_e^-(t) = F_{pe}^1(c_{e,x}) \cdot \phi_e(t) + F_{pe}^2(c_{e,x}) \cdot i_{e,x}(t) + F_{pe}^3(c_{e,x}) \cdot \ln(c_{e,x}(t)) \quad (\text{B.55})$$

where vectors $i_{e,x}$ and $c_{e,x}$ represent the entire electrolyte current and concentration, respectively, across the entire battery, including boundary values,

$$i_{e,x}(t) = [0, i_e^-(x, t), I(x, t), i_e^+(x, t), 0]^T, \quad (\text{B.56})$$

$$c_{e,x}(t) = [c_{e,bc,1}(t), c_e^-(x, t), c_{e,bc,2}(t), c_e^{sep}(x, t), c_{e,bc,3}(t), c_e^+(x, t), c_{e,bc,4}(t)]^T, \quad (\text{B.57})$$

$$c_{e,bc}(t) = C_{ce} c_e(t) \quad (\text{B.58})$$

Note that the system matrices $F_{pe}^1, F_{pe}^2, F_{pe}^3$ are state-varying. These state matrices are computed online as follows:

$$F_{pe}^1 = \kappa^{\text{eff}}(c_e) \cdot M_{pe,1} + M_{pe,2} C_{pe}, \quad (\text{B.59})$$

$$F_{pe}^2 = M_{pe,3}, \quad (\text{B.60})$$

$$F_{pe}^3 = \kappa_D^{\text{eff}}(c_e) M_{pe,4} \quad (\text{B.61})$$

where

$$\kappa^{\text{eff}}(c_e) = \kappa(c_e) \cdot (\varepsilon_e^j)^{\text{brug}}, \quad (\text{B.62})$$

$$\kappa_D^{\text{eff}}(c_e) = \kappa^{\text{eff}}(c_e) \frac{2RT}{F} (t_c^0 - 1) \left(1 + \frac{d \ln f_{c/a}}{d \ln c_e}(x, t) \right). \quad (\text{B.63})$$

The matrices $M_{pe,1}, M_{pe,2}, M_{pe,3}, M_{pe,4}, C_{pe}$ are computed offline by Matlab function `phi_e_mats.m` as follows. Let

$$\alpha^j = \frac{1}{2L^j \Delta x^j} \quad (\text{B.64})$$

Molar ion fluxes, i.e. Butler-Volmer Current, j_n^-, j_n^+

Since the Butler-Volmer equation (2.6) is algebraic, and we always assume $\alpha_a = \alpha_c = 0.5 = \alpha$, it is trivially implemented as:

$$\frac{d}{dt} j_n^-(t) = \frac{2}{F} i_0^-(t) \sinh \left[\frac{\alpha F}{RT} \eta^-(t) \right] - j_n^-(t), \quad (\text{B.65})$$

$$\frac{d}{dt} j_n^+(t) = \frac{2}{F} i_0^+(t) \sinh \left[\frac{\alpha F}{RT} \eta^+(t) \right] - j_n^+(t) \quad (\text{B.66})$$

where

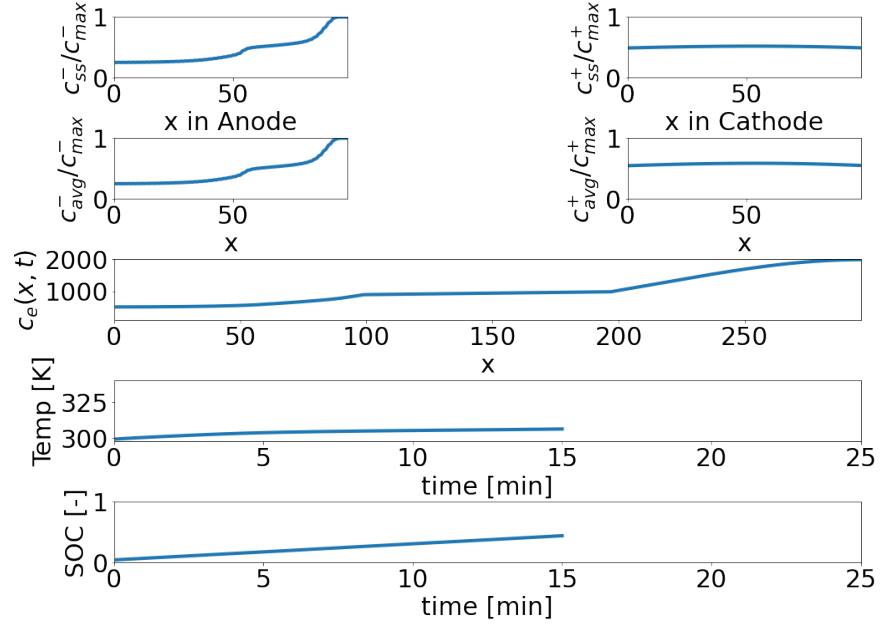
$$i_0^\pm(t) = k^\pm [c_{ss}^\pm(t) c_e(t) (c_{s,\max}^\pm - c_{ss}^\pm(t))]^\alpha, \quad (\text{B.67})$$

$$\eta^\pm(t) = \phi_s^\pm(t) - \phi_e(t) - U^\pm(c_{ss}^\pm(t)) - F R_f^\pm j_n^\pm(t) \quad (\text{B.68})$$

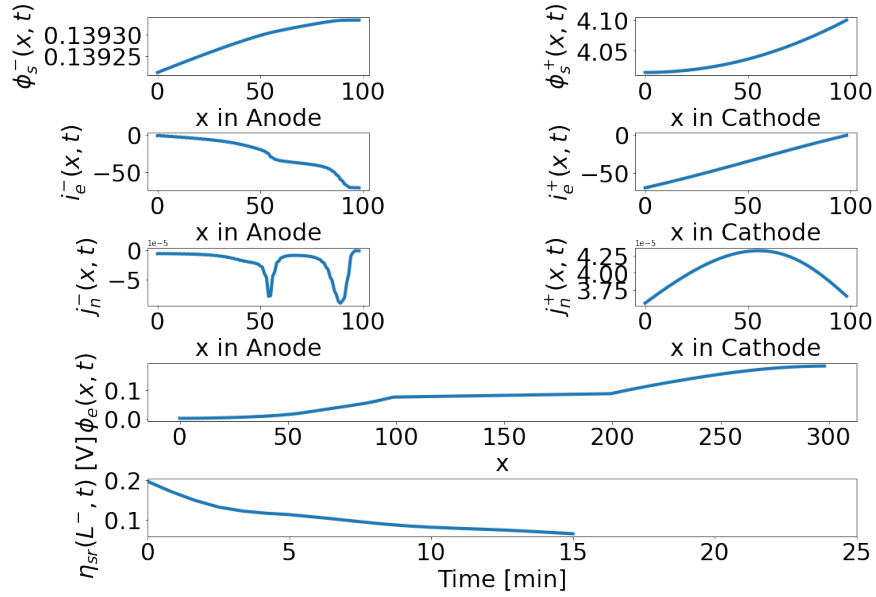
for each discrete point in x , in the electrodes only. Note that $\frac{d}{dt} j_n^\pm(t)$ is a dummy variable used to save the corresponding element of vector $g(x, z, t)$. Note that the surface concentration is not allowed to increase more than its maximum, otherwise it's not feasible to solve the whole dynamical system equations.

In the next section, we visualize the evolution of the system when a simple input current, i.e., constant current is applied to this model.

B.2 Visualization



(a) Time-derivative states and associated state trajectory.



(b) Algebraic state evolution and associated algebraic state trajectory.

Figure B.1: The visualization of DFN simulator when a constant current is applied to the simulator.

Appendix C

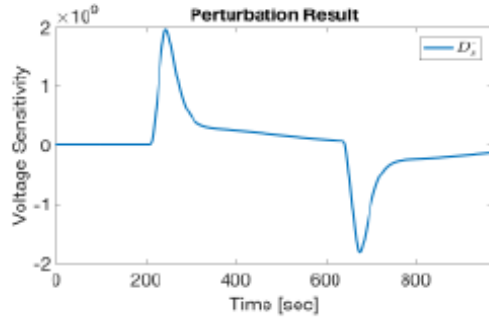
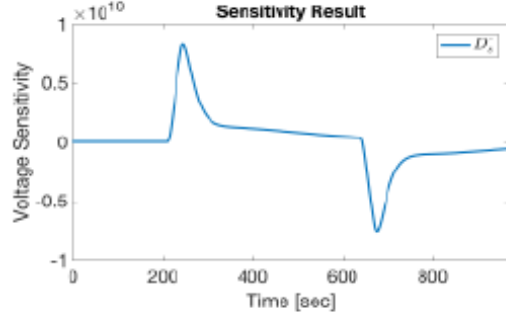
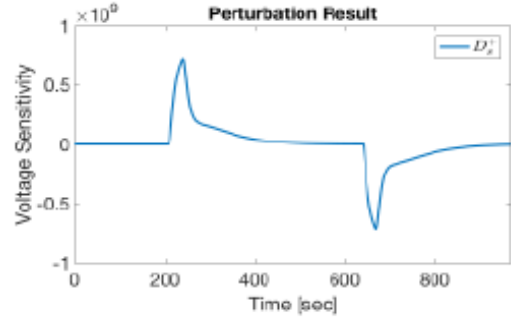
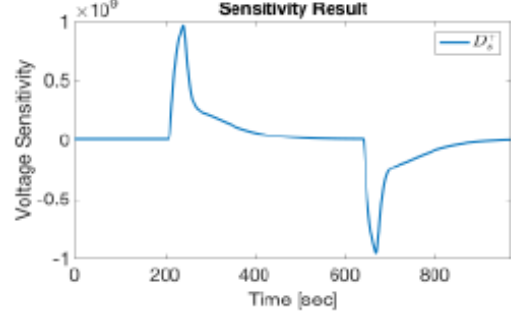
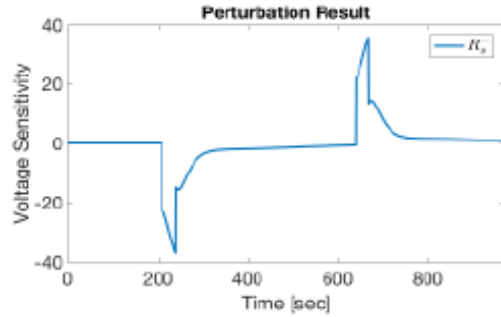
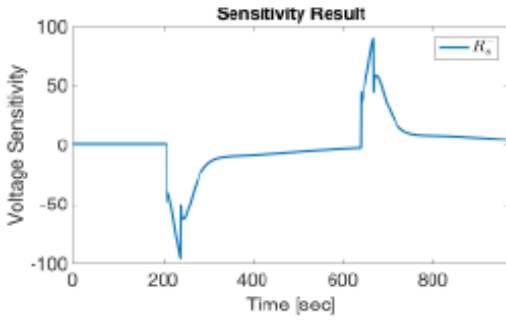
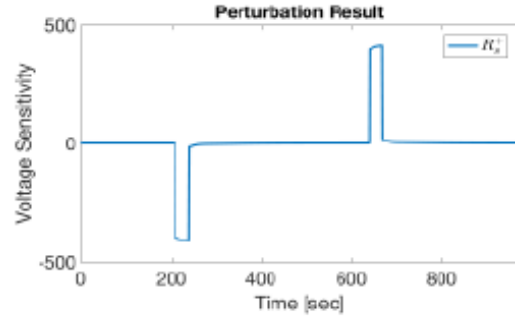
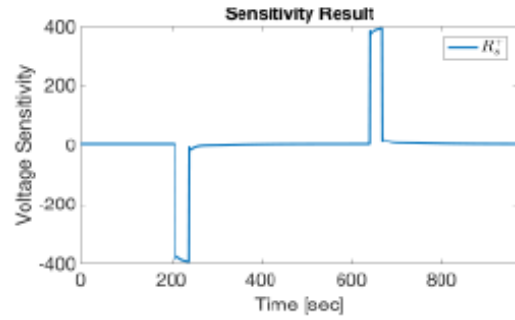
Sensitivity Validation

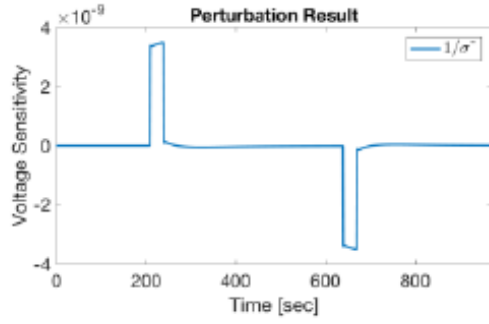
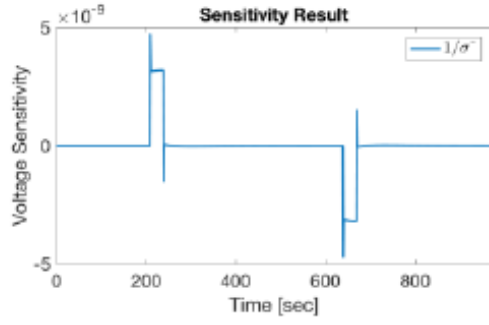
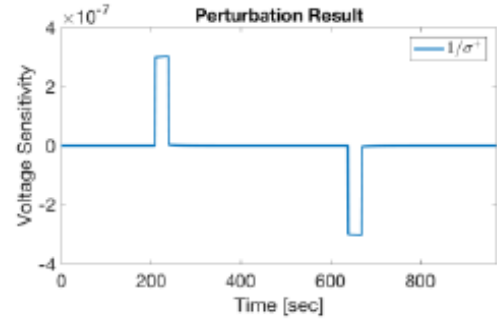
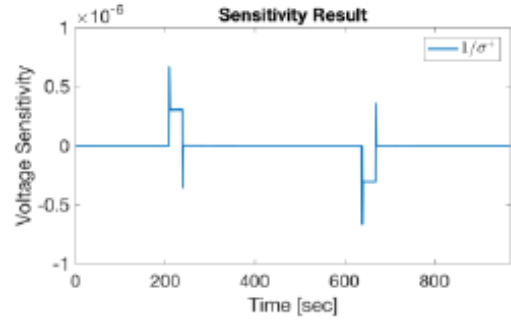
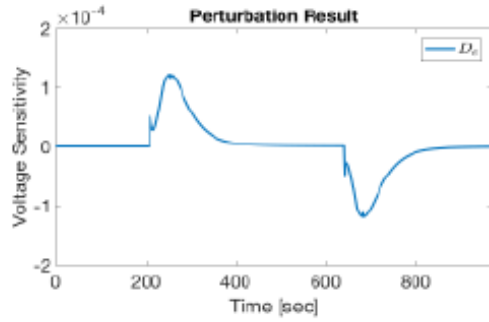
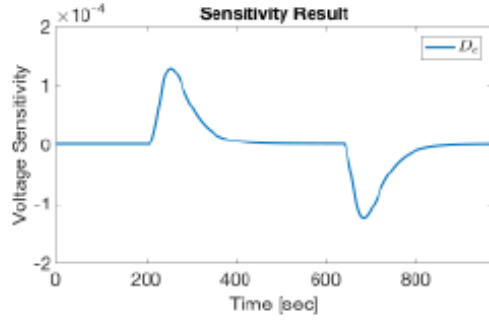
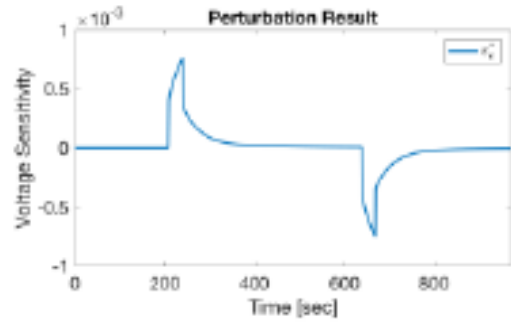
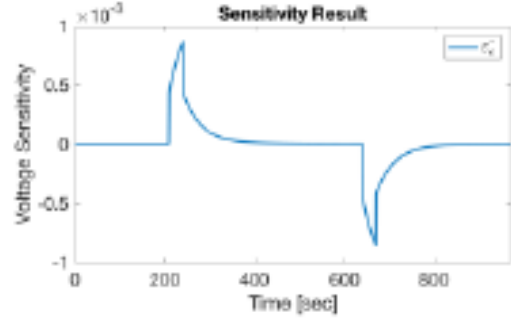
In this section, we plot the sensitivity results in comparison to perturbation results. First, below Table C.1 is the parameter list to calculate the sensitivity. In this test, we give a simple pulse current.

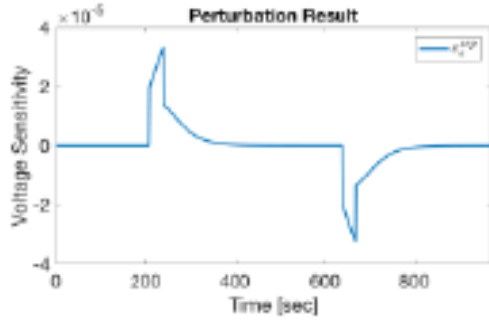
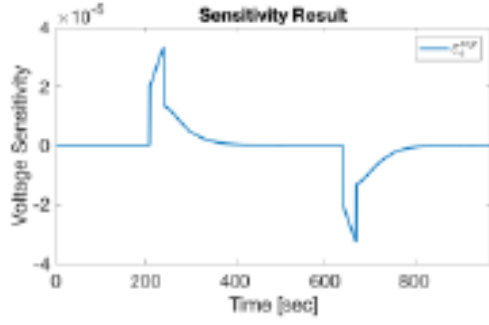
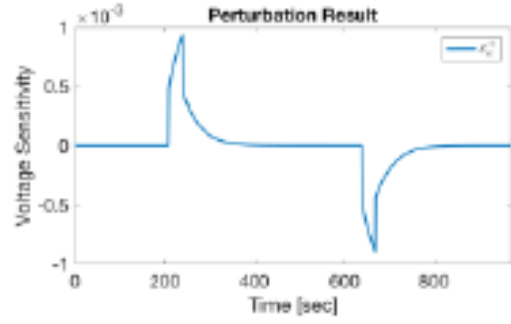
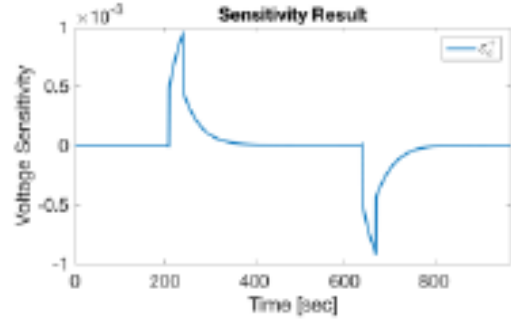
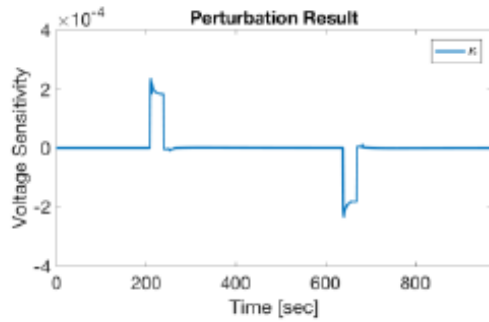
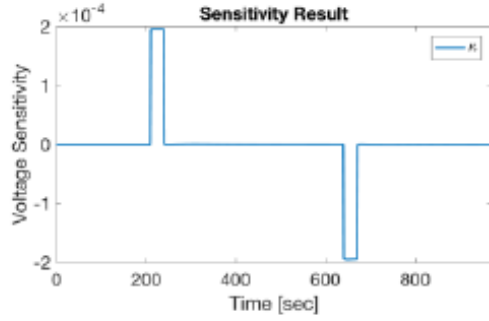
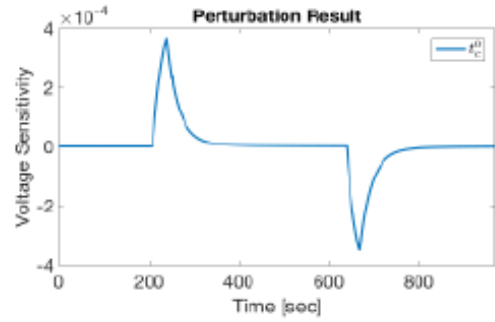
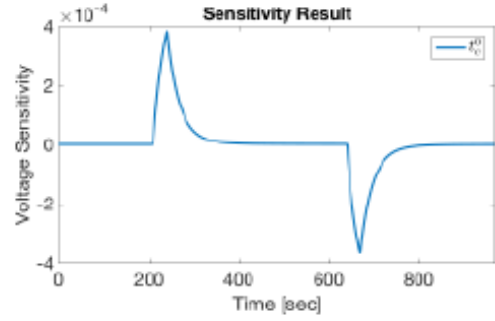
Params of Interest			
ID #	Symbol	Description	Unit
1	D_s^-	Solid-phase diffusion coefficients	$[m^2/sec]$
2	D_s^+	Solid-phase diffusion coefficients	$[m^2/sec]$
3	R_s^-	Solid-phase particle radii	$[m]$
4	R_s^+	Solid-phase particle radii	$[m]$
5	ϵ_s^-	Solid-phase volume fraction	$[-]$
6	ϵ_s^+	Solid-phase volume fraction	$[-]$
7	$1/\sigma^-$	Solid-phase conductivity	$[\Omega\ m]$
8	$1/\sigma^+$	Solid-phase conductivity	$[-]$
9	$D_e(\cdot)$	Electrolyte diffusion coefficient	$[m^2/sec]$
10	ϵ_e^-	Electrolyte volume fraction	$[-]$
11	ϵ_e^{sep}	Electrolyte volume fraction	$[-]$
12	ϵ_e^+	Electrolyte volume fraction	$[-]$
13	$\kappa(\cdot)$	Electrolyte conductivity	$[\Omega\ m]$
14	t_c^0	Transference number	$[-]$
15	$\frac{d \ln f_{c/a}}{d \ln c_e}(\cdot)$	Activity coefficient	$[-]$
16	k^-	Kinetic rate constants	$[(A/m^2)(mol^3/mol^{(1+\alpha)})]$
17	k^+	Kinetic rate constants	$[(A/m^2)(mol^3/mol^{(1+\alpha)})]$
18	R_f^-	Film resistance	$[\Omega]$
19	R_f^+	Film resistance	$[\Omega]$
20	$n_{Li,s}$	Moles of cyclable lithium in solid phase	$[mol]$
21	$c_e(x, 0)$	Initial Li-ion concentration in electrolyte	$[mol/m^3]$

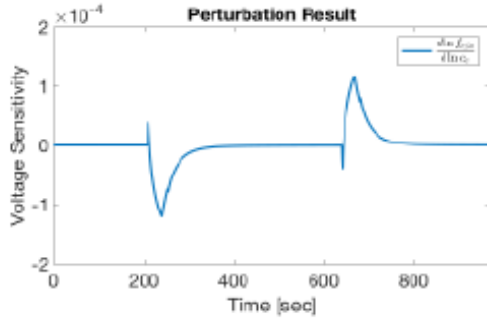
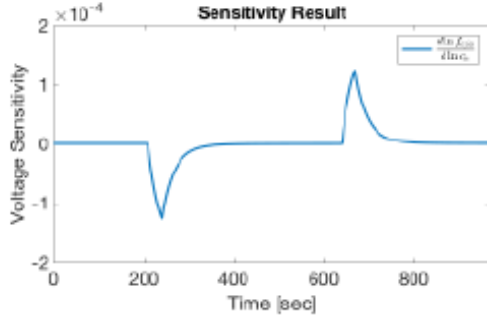
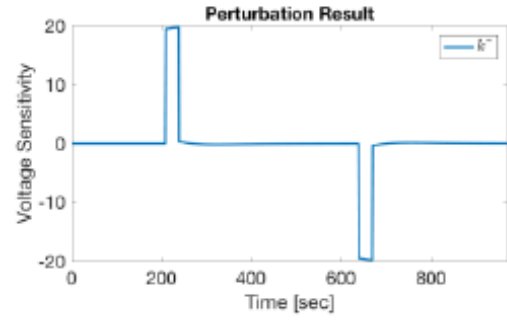
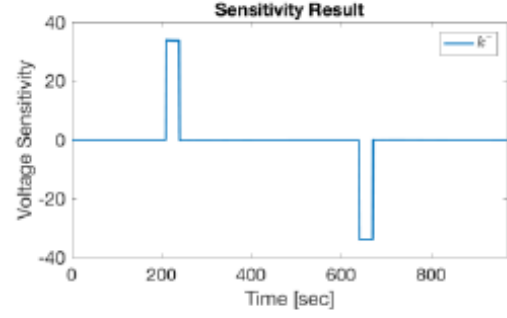
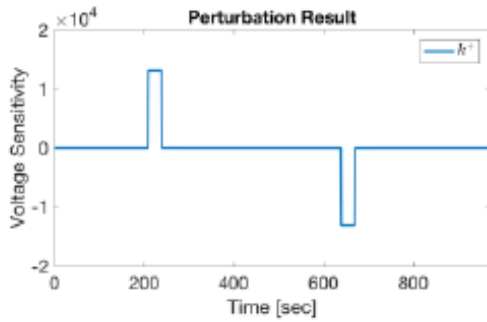
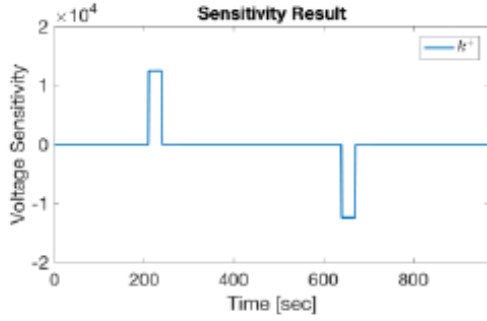
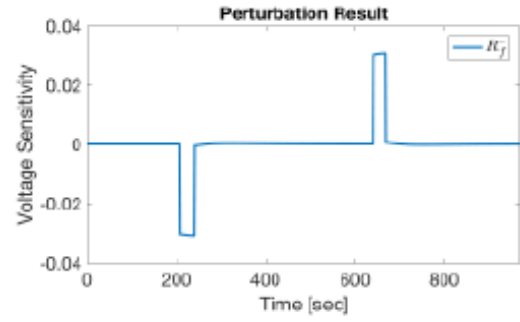
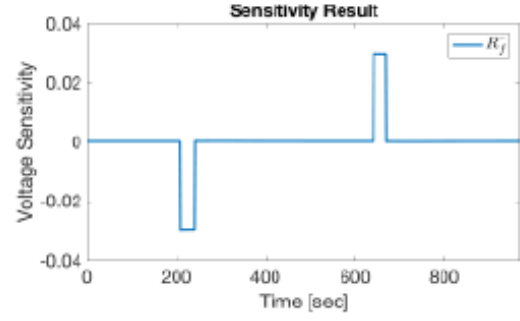
Table C.1: Parameter list for sensitivity validation

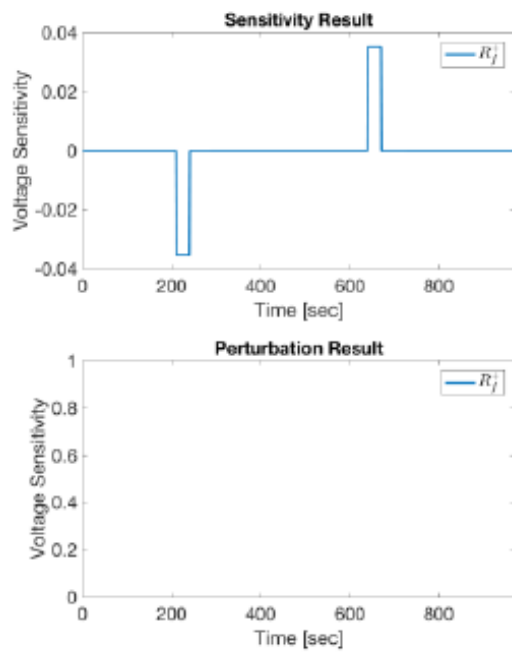
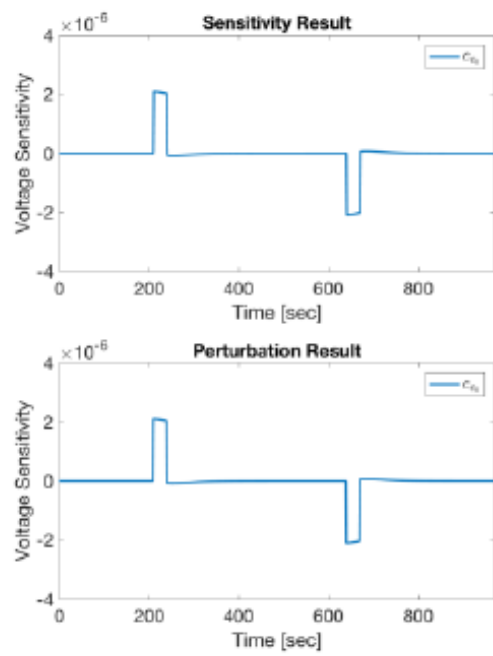
C.1 Sensitivity Dynamics vs Perturbation

(a) D_s^- (b) D_s^+ (c) R_s^- (d) R_s^+ Figure C.1: Sensitivity validations for D_s^- , D_s^+ , R_s^- , and R_s^+ .

(a) σ^- (b) σ^+ (c) $D_e(\cdot)$ (d) ε_e^- Figure C.2: Sensitivity validations for σ^- , σ^+ , $D_e(\cdot)$, and ε_e^- .

(a) ε_e^{sep} (b) ε_e^+ (c) $\kappa(\cdot)$ (d) t_c^0 Figure C.3: Sensitivity validations for ε_e^{sep} , ε_e^+ , $\kappa(\cdot)$, and t_c^0 .

(a) $\frac{d \ln f_{c/a}}{d \ln c_e}$ (b) k^- (c) k^+ (d) R_f^- Figure C.4: Sensitivity validations for $\frac{d \ln f_{c/a}}{d \ln c_e}$, k^- , k^+ and R_f^- .

(a) R_f^+ (b) $c_e(x, 0)$ Figure C.5: Sensitivity validations for R_f^+ and $c_e(x, 0)$.

Bibliography

- [1] United States Environmental Protection Agency. *Vehicle and Fuel Emission Testing*. Lastly accessed: 2017-12-16. URL: <https://www.epa.gov/vehicle-and-fuel-emissions-testing/dynamometer-drive-schedules>.
- [2] Albert Altarovici, Olivier Bokanowski, and Hasnaa Zidani. “A general Hamilton-Jacobi framework for non-linear state-constrained control problems”. In: *ESAIM: Control, Optimisation and Calculus of Variations* 19.2 (2013), pp. 337–357.
- [3] Joel Andersson. “A General-Purpose Software Framework for Dynamic Optimization”. PhD thesis. Department of Electrical Engineering (ESAT/SCD) and Optimization in Engineering Center, Kasteelpark Arenberg 10, 3001-Heverlee, Belgium: Arenberg Doctoral School, KU Leuven, Oct. 2013.
- [4] Pankaj Arora, Marc Doyle, and Ralph E White. “Mathematical modeling of the lithium deposition overcharge reaction in lithium-ion batteries using carbon-based negative electrodes”. In: *Journal of The Electrochemical Society* 146.10 (1999), p. 3543.
- [5] TR Ashwin et al. “Modified electrochemical parameter estimation of NCR18650BD battery using implicit finite volume method”. In: *Journal of Power Sources* 341 (2017), pp. 387–395.
- [6] Peter M Attia et al. “Closed-loop optimization of fast-charging protocols for batteries with machine learning”. In: *Nature* 578.7795 (2020), pp. 397–402.
- [7] Francois Belletti et al. “Expert level control of ramp metering based on multi-task deep reinforcement learning”. In: *IEEE Transactions on Intelligent Transportation Systems* 19.4 (2017), pp. 1198–1207.
- [8] “Benchmark Minerals’ Report”. In: (2020).
- [9] Dawn M Bernardi and Joo-Young Go. “Analysis of pulse and relaxation behavior in lithium-ion batteries”. In: *Journal of Power Sources* 196.1 (2011), pp. 412–427.
- [10] Dimitri P Bertsekas. *Dynamic programming and optimal control*. Vol. 1. 3. Athena scientific Belmont, MA, 2005.
- [11] Christoph R Birkl et al. “Degradation diagnostics for lithium ion cells”. In: *Journal of Power Sources* 341 (2017), pp. 373–386.

- [12] A M Bizeray et al. “Lithium-ion battery thermal-electrochemical model-based state estimation using orthogonal collocation and a modified extended Kalman filter”. In: *Journal of Power Sources* 296 (2015), pp. 400–412.
- [13] Adrien M. Bizeray et al. “Identifiability and parameter estimation of the single particle lithium-ion battery model”. In: *CoRR* abs/1702.02471 (2017). arXiv: 1702.02471. URL: <http://arxiv.org/abs/1702.02471>.
- [14] Åke Björck. “Solving linear least squares problems by Gram-Schmidt orthogonalization”. In: *BIT Numerical Mathematics* 7.1 (1967), pp. 1–21.
- [15] Stephen Boyd and Lieven Vandenbergh. *Convex optimization*. Cambridge university press, 2009.
- [16] Arthur E Bryson Jr, Walter F Denham, and Stewart E Dreyfus. “Optimal programming problems with inequality constraints”. In: *AIAA journal* 1.11 (1963), pp. 2544–2550.
- [17] Arthur Earl Bryson. *Applied optimal control: optimization, estimation and control*. Routledge, 2018.
- [18] James W Burrows. “Fuel-optimal aircraft trajectories with fixed arrival times”. In: *Journal of Guidance, Control, and Dynamics* 6.1 (1983), pp. 14–19.
- [19] Long Cai and Ralph E White. “Reduction of model order based on proper orthogonal decomposition for lithium-ion battery simulations”. In: *Journal of The Electrochemical Society* 156.3 (2009), A154–A161.
- [20] Giuseppe C Calafiore and Laurent El Ghaoui. *Optimization models*. Cambridge university press, 2014.
- [21] Giacomo Capizzi, Francesco Bonanno, and Giuseppe M Tina. “Recurrent neural network-based modeling and simulation of lead-acid batteries charge–discharge”. In: *IEEE Transactions on Energy Conversion* 26.2 (2011), pp. 435–443.
- [22] Kathryn Chaloner and Isabella Verdinelli. “Bayesian experimental design: A review”. In: *Statistical Science* (1995), pp. 273–304.
- [23] N. A. Chaturvedi et al. “Algorithms for Advanced Battery-Management Systems”. In: *IEEE Control Systems* 30.3 (June 2010), pp. 49–68. ISSN: 1066-033X.
- [24] N. Chaturvedi et al. *Method and system for estimating a capacity of individual electrodes and the total capacity of a lithium-ion battery system*. EP Patent App. EP20,140,707,912. Mar. 2016. URL: <https://google.com/patents/EP2989675A1?cl=zh>.
- [25] Tianping Chen and Hong Chen. “Universal approximation to nonlinear operators by neural networks with arbitrary activation functions and its application to dynamical systems”. In: *IEEE Transactions on Neural Networks* 6.4 (1995), pp. 911–917.
- [26] J Christensen and J Newman. “Effect of anode film resistance on the charge/discharge capacity of a lithium-ion battery”. In: *Journal of the Electrochemical Society* 150.11 (2003), A1416.

- [27] Richard C Cope and Yury Podrazhansky. “The art of battery charging”. In: *Fourteenth Annual Battery Conference on Applications and Advances. Proceedings of the Conference (Cat. No. 99TH8371)*. IEEE. 1999, pp. 233–235.
- [28] Thomas M Cover and Joy A Thomas. *Elements of information theory*. John Wiley & Sons, 2012.
- [29] Satadru Dey, B Ayalew, and P Pisu. “Nonlinear adaptive observer for a lithium-ion battery cell based on coupled electrochemical–thermal model”. In: *Journal of Dynamic Systems, Measurement, and Control* 137.11 (2015).
- [30] Domenico Di Domenico, Anna Stefanopoulou, and Giovanni Fiengo. “Lithium-ion battery state of charge and critical surface charge estimation using an electrochemical model-based extended Kalman filter”. In: *Journal of dynamic systems, measurement, and control* 132.6 (2010).
- [31] Akram Eddahech et al. “Remaining useful life prediction of lithium batteries in calendar ageing for automotive applications”. In: *Microelectronics Reliability* 52.9 (2012), pp. 2438–2442.
- [32] Jeffrey L. Elman. “Finding structure in time”. In: *Cognitive Science* 14.2 (1990), pp. 179–211.
- [33] “Events with smoke, fire, extreme heat or explosion involving lithium batteries”. In: (2020). URL: <http://people.maths.ox.ac.uk/trefethen/pdetext.html>.
- [34] Guodong Fan, Ke Pan, and Marcello Canova. “A comparison of model order reduction techniques for electrochemical characterization of lithium-ion batteries”. In: *Decision and Control (CDC), 2015 IEEE 54th Annual Conference on*. IEEE. 2015, pp. 3922–3931.
- [35] Weifeng Fang, Ou Jung Kwon, and Chao-Yang Wang. “Electrochemical–thermal modeling of automotive Li-ion batteries and experimental validation using a three-electrode cell”. In: *International journal of energy research* 34.2 (2010), pp. 107–115.
- [36] Joel C. Forman et al. “Genetic identification and fisher identifiability analysis of the Doyle-Fuller-Newman model from experimental cycling of a LiFePO₄ cell”. In: *Journal of Power Sources* (2012), pp. 263–275.
- [37] Joel C Forman et al. “Reduction of an electrochemistry-based li-ion battery model via quasi-linearization and pade approximation”. In: *Journal of the Electrochemical Society* 158.2 (2011), A93–A101.
- [38] Gaia Franceschini and Sandro Macchietto. “Model-based design of experiments for parameter precision: State of the art”. In: *Chemical Engineering Science* 63.19 (2008), pp. 4846–4872.
- [39] Michael Grant and Stephen Boyd. *CVX: Matlab Software for Disciplined Convex Programming, version 2.1*. <http://cvxr.com/cvx>. Mar. 2014.

- [40] Mark A Hall and Lloyd A Smith. “Practical feature subset selection for machine learning”. In: (1998).
- [41] Richard F Hartl, Suresh P Sethi, and Raymond G Vickson. “A survey of the maximum principles for optimal control problems with state constraints”. In: *SIAM review* 37.2 (1995), pp. 181–218.
- [42] Hongwen He, Rui Xiong, and Jinxin Fan. “Evaluation of lithium-ion battery equivalent circuit models for state of charge estimation by an experimental approach”. In: *energies* 4.4 (2011), pp. 582–598.
- [43] Alan C Hindmarsh et al. “SUNDIALS: Suite of nonlinear and differential/algebraic equation solvers”. In: *ACM Transactions on Mathematical Software (TOMS)* 31.3 (2005), pp. 363–396.
- [44] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. “Multilayer feedforward networks are universal approximators”. In: *Neural networks* 2.5 (1989), pp. 359–366.
- [45] Chao Hu, Byeng D Youn, and Jaesik Chung. “A multiscale framework with extended Kalman filter for lithium-ion battery SOC and capacity estimation”. In: *Applied Energy* 92 (2012), pp. 694–704.
- [46] Meng Huang et al. “Aging estimation of lithium-ion battery cell using an electrochemical model-based extended Kalman filter”. In: *AIAA Scitech 2019 Forum*. 2019, p. 0785.
- [47] Herbert Jaeger. *Tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and the “echo state network” approach*. GMD-Forschungszentrum Informationstechnik, 2002.
- [48] Liang Jin and Madan M Gupta. “Stable dynamic backpropagation learning in recurrent neural networks”. In: *IEEE Transactions on Neural Networks* 10.6 (1999), pp. 1321–1334.
- [49] N. Jin et al. *Parameter estimation of an electrochemistry-based Lithium-ion battery model using a two-step procedure and a parameter sensitivity analysis*. Tech. rep. Technische Universiteit Eindhoven (TUE), 2017, pp. 1–12.
- [50] Ryan Jobman, M. Scott Trimboli, and Gregory L. Plett. “Identification of Lithium-Ion Physics-Based Model Parameter Values”. In: *Journal of Energy Challenges and Mechanics* 2.2 (2015), pp. 45–55.
- [51] Hassan Khalil. *Nonlinear Systems*. 3rd ed. Prentice Hall, 2002. ISBN: 013067530X.
- [52] Eugene Khmelnitsky. “On an optimal control problem of train operation”. In: *IEEE transactions on automatic control* 45.7 (2000), pp. 1257–1266.
- [53] Namwook Kim, Sukwon Cha, and Huei Peng. “Optimal control of hybrid electric vehicles based on Pontryagin’s minimum principle”. In: *IEEE Transactions on control systems technology* 19.5 (2010), pp. 1279–1287.

- [54] Donald E Kirk. *Optimal control theory: an introduction*. Courier Corporation, 2004.
- [55] Reinhardt Klein et al. “Optimal charging strategies in lithium-ion battery”. In: *American Control Conference (ACC), 2011*. IEEE. 2011, pp. 382–387.
- [56] GJ Lastman. “A shooting method for solving two-point boundary-value problems arising from non-singular bang-bang optimal control problems”. In: *International journal of control* 27.4 (1978), pp. 513–524.
- [57] David LeBlanc. “Road departure crash warning system field operational test: methodology and results. volume 1: technical report”. In: (2006).
- [58] Seongjun Lee et al. “State-of-charge and capacity estimation of lithium-ion battery using a new open-circuit voltage versus state-of-charge”. In: *Journal of power sources* 185.2 (2008), pp. 1367–1373.
- [59] Kenneth Levenberg. “A method for the solution of certain non-linear problems in least squares”. In: *Quarterly of applied mathematics* 2.2 (1944), pp. 164–168.
- [60] Frank L Lewis, Draguna Vrabie, and Kyriakos G Vamvoudakis. “Reinforcement learning and feedback control: Using natural decision methods to design optimal adaptive controllers”. In: *IEEE Control Systems Magazine* 32.6 (2012), pp. 76–105.
- [61] Timothy P Lillicrap et al. “Continuous control with deep reinforcement learning”. In: *arXiv preprint arXiv:1509.02971* (2015).
- [62] Chia-Hsiang Lin, Chun-Yu Hsieh, and Ke-Horng Chen. “A Li-ion battery charger with smooth control circuit and built-in resistance compensator for achieving stable and fast charging”. In: *IEEE Transactions on Circuits and Systems I: Regular Papers* 57.2 (2009), pp. 506–517.
- [63] Xianke Lin et al. “A comprehensive capacity fade model and analysis for Li-ion batteries”. In: *Journal of The Electrochemical Society* 160.10 (2013), A1701.
- [64] Xinfan Lin et al. “A lumped-parameter electro-thermal model for cylindrical batteries”. In: *Journal of Power Sources* 257 (2014), pp. 1–11.
- [65] Ji Liu, Guang Li, and Hosam K Fathy. “An extended differential flatness approach for the health-conscious nonlinear model predictive control of lithium-ion batteries”. In: *IEEE Transactions on Control Systems Technology* 25.5 (2017), pp. 1882–1889.
- [66] Jie Liu et al. *An adaptive recurrent neural network for remaining useful life prediction of lithium-ion batteries*. Tech. rep. DTIC Document, 2010.
- [67] Ji Liu et al. “Can an Identifiability-Optimizing Test Protocol Improve the Robustness of Subsequent Health-Conscious Lithium-ion Battery Control? An Illustrative Case Study”. In: *2016 American Control Conference* ii (2016), pp. 6320–6325.
- [68] N Lotfi et al. “Electrochemical Model-Based Adaptive Estimation of Li-Ion Battery State of Charge”. In: *ASME 2015 Dynamic Systems and Control Conference*. American Society of Mechanical Engineers. 2015, V001T13A008–V001T13A008.

- [69] Berit Floor Lund and Bjarne A Foss. “Parameter ranking by orthogonalization—applied to nonlinear mechanistic models”. In: *Automatica* 44.1 (2008), pp. 278–281.
- [70] Kaj Madsen, Hans Bruun Nielsen, and Ole Tingleff. “Methods for non-linear least squares problems”. In: (2004).
- [71] Olvi L Mangasarian. “Sufficient conditions for the optimal control of nonlinear systems”. In: *SIAM Journal on control* 4.1 (1966), pp. 139–152.
- [72] James Marcicki et al. “Design and parametrization analysis of a reduced-order electrochemical model of graphite/LiFePO₄ cells for SOC/SOH estimation”. In: *Journal of Power Sources* 237 (2013), pp. 310–324.
- [73] Christopher Mayhew et al. “Investigation of projection-based model-reduction techniques for solid-phase diffusion in li-ion batteries”. In: *2014 American Control Conference*. IEEE. 2014, pp. 123–128.
- [74] Volodymyr Mnih et al. “Human-level control through deep reinforcement learning”. In: *Nature* 518.7540 (2015), p. 529.
- [75] S. J. Moura et al. “Battery State Estimation for a Single Particle Model With Electrolyte Dynamics”. In: *IEEE Transactions on Control Systems Technology* PP.99 (2016), pp. 1–16.
- [76] Scott J Moura. “Estimation and control of battery electrochemistry models: A tutorial”. In: *2015 54th IEEE Conference on Decision and Control (CDC)*. IEEE. 2015, pp. 3906–3912.
- [77] Scott J Moura, Nalin A Chaturvedi, and Miroslav Krstić. “Adaptive partial differential equation observer for battery state-of-charge/state-of-health estimation via an electrochemical model”. In: *Journal of Dynamic Systems, Measurement, and Control* 136.1 (2014), p. 011015.
- [78] Scott J Moura, Miroslav Krstic, and Nalin A Chaturvedi. “Adaptive PDE observer for battery SOC/SOH estimation”. In: *ASME 2012 5th Annual Dynamic Systems and Control Conference joint with the JSME 2012 11th Motion and Vibration Conference*. American Society of Mechanical Engineers. 2012, pp. 101–110.
- [79] Scott J Moura et al. “Battery state estimation for a single particle model with electrolyte dynamics”. In: *IEEE Transactions on Control Systems Technology* 25.2 (2016), pp. 453–468.
- [80] Kumpati S Narendra and Kannan Parthasarathy. “Identification and control of dynamical systems using neural networks”. In: *IEEE Transactions on neural networks* 1.1 (1990), pp. 4–27.
- [81] J Newman. “Fortran programs for the simulation of electrochemical systems: dualfoil5. 2. f (2014)”. In: URL <http://www.cchem.berkeley.edu/jsngrp> ().
- [82] John Newman. *FORTTRAN programs for simulation of electrochemical systems*. 2008.

- [83] Peter HL Notten, JHG Op het Veld, and JRG Van Beek. “Boostcharging Li-ion batteries: A challenging new charging concept”. In: *Journal of Power Sources* 145.1 (2005), pp. 89–94.
- [84] Stanley Osher and Ronald Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces*. Vol. 153. Springer-Verlag, 2003. DOI: 10.1007/b98879.
- [85] P Paoletti and L Mahadevan. “Planar controlled gliding, tumbling and descent”. In: *Journal of fluid mechanics* 689 (2011), pp. 489–516.
- [86] Saehong Park et al. “Optimal experimental design for parameterization of an electrochemical lithium-ion battery model”. In: *Journal of The Electrochemical Society* 165.7 (2018), A1309.
- [87] Saehong Park et al. “Optimal Input Design for Parameter Identification in an Electrochemical Li-ion Battery Model”. In: *Submitted to American Control Conference (ACC), 2018*. IEEE. 2018.
- [88] Saehong Park et al. “Reinforcement Learning-based Fast Charging Control Strategy for Li-ion Batteries”. In: *arXiv preprint arXiv:2002.02060* (2020).
- [89] A Paterson. “Our Guide to Batteries”. In: *Axeon: Aberdeen, UK* (2012), pp. 1–22.
- [90] Lalit Patnaik, AVJS Praneeth, and Sheldon S Williamson. “A closed-loop constant-temperature constant-voltage charging technique to reduce charge time of lithium-ion batteries”. In: *IEEE Transactions on Industrial Electronics* 66.2 (2018), pp. 1059–1067.
- [91] Hector Perez, Niloofar Shahmohammadhamedani, and Scott Moura. “Enhanced performance of li-ion batteries via modified reference governors and electrochemical models”. In: *IEEE/ASME Transactions on Mechatronics* 20.4 (2015), pp. 1511–1520.
- [92] HE Perez et al. “Optimal Charging of Li-Ion Batteries via a Single Particle Model with Electrolyte and Thermal Dynamics”. In: *Journal of The Electrochemical Society* 164.7 (2017), A1679–A1687.
- [93] Lev Semenovich Pontryagin. *Mathematical theory of optimal processes*. Routledge, 2018.
- [94] Andrea Pozzi, Marcello Torchio, and Davide M Raimondo. “Film growth minimization in a Li-ion cell: a Pseudo Two Dimensional model-based optimal charging approach”. In: *2018 European Control Conference (ECC)*. IEEE. 2018, pp. 1753–1758.
- [95] Andrea Pozzi et al. “Optimal design of experiments for a lithium-ion cell: parameters identification of an isothermal single particle model with electrolyte dynamics”. In: *Industrial & Engineering Chemistry Research* 58.3 (2018), pp. 1286–1299.
- [96] Saeed Khaleghi Rahimian, Sean Rayman, and Ralph E White. “Extension of physics-based single particle model for higher charge-discharge rates”. In: *Journal of Power Sources* 224.0 (2013), pp. 180–194. ISSN: 0378-7753.
- [97] Christopher D Rahn and Chao-Yang Wang. *Battery systems engineering*. John Wiley & Sons, 2013.

- [98] John A Rice. *Mathematical statistics and data analysis*. Cengage Learning, 2006.
- [99] Raffaele Romagnoli et al. “A feedback charge strategy for Li-ion battery cells based on Reference Governor”. In: *Journal of process control* 83 (2019), pp. 164–176.
- [100] Michael J Rothenberger et al. “Maximizing Parameter Identifiability of an Equivalent-Circuit Battery Model Using Optimal Periodic Input Shaping”. In: *2014 ASME Dynamic Systems and Control Conference*. 2014, V001T19A004.
- [101] Shriram Santhanagopalan and Ralph E White. “Online estimation of the state of charge of a lithium ion cell”. In: *Journal of power sources* 161.2 (2006), pp. 1346–1355.
- [102] Shriram Santhanagopalan et al. “Parameter estimation and life modeling of lithium-ion cells”. In: *Journal of The Electrochemical Society* 155.4 (2008), A345.
- [103] Shriram Santhanagopalan et al. “Review of models for predicting the cycling performance of lithium ion batteries”. In: *Journal of Power Sources* 156.2 (2006), pp. 620–628. ISSN: 0378-7753.
- [104] Shriram Santhanagopalan et al. “Review of models for predicting the cycling performance of lithium ion batteries”. In: *Journal of power sources* 156.2 (2006), pp. 620–628.
- [105] François Savoye et al. “Impact of periodic current pulses on Li-ion battery performance”. In: *IEEE Transactions on Industrial Electronics* 59.9 (2011), pp. 3481–3488.
- [106] Alexander P. Schmidt et al. “Experiment-driven electrochemical modeling and systematic parameterization for a lithium-ion battery cell”. In: *Journal of Power Sources* 195.15 (2010), pp. 5071–5080.
- [107] John Schulman et al. “Trust region policy optimization”. In: *International Conference on Machine Learning*. 2015, pp. 1889–1897.
- [108] David Silver et al. “Deterministic policy gradient algorithms”. In: 2014.
- [109] Alice E Smith et al. “Penalty functions”. In: *Handbook of evolutionary computation* 97.1 (1995), p. C5.
- [110] Qing Song. “On the weight convergence of Elman networks”. In: *IEEE Transactions on Neural networks* 21.3 (2010), pp. 463–480.
- [111] Venkat R Subramanian et al. “Mathematical model reformulation for lithium-ion battery simulations: Galvanostatic boundary conditions”. In: *Journal of the Electrochemical Society* 156.4 (2009), A260–A271. DOI: 10.1149/1.3065083.
- [112] Peiyi Sun et al. “A review of battery fires in electric vehicles”. In: *Fire technology* (2020), pp. 1–50.
- [113] Bharatkumar Suthar et al. “Optimal charging profiles with minimal intercalation-induced stresses for lithium-ion batteries using reformulated pseudo 2-dimensional models”. In: *Journal of The Electrochemical Society* 161.11 (2014), F3144–F3155.

- [114] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.
- [115] Richard S Sutton et al. “Policy gradient methods for reinforcement learning with function approximation”. In: *Advances in neural information processing systems*. 2000, pp. 1057–1063.
- [116] Shu-Xia Tang et al. “State-of-charge estimation from a thermal–electrochemical model of lithium-ion batteries”. In: *Automatica* 83 (2017), pp. 206–219.
- [117] Tanvir R Tanim, Christopher D Rahn, and Chao-Yang Wang. “A temperature dependent, single particle, lithium ion cell model including electrolyte diffusion”. In: *Journal of Dynamic Systems, Measurement, and Control* 137.1 (2015).
- [118] Tanvir R Tanim, Christopher D Rahn, and Chao-Yang Wang. “State of charge estimation of a lithium ion cell based on a temperature dependent and electrolyte enhanced single particle model”. In: *Energy* 80 (2015), pp. 731–739.
- [119] Karen E Thomas, John Newman, and Robert M Darling. “Mathematical modeling of lithium batteries”. In: *Advances in lithium-ion batteries*. Springer, 2002, pp. 345–392.
- [120] Anna Tomaszewska et al. “Lithium-ion battery fast charging: A review”. In: *ETransportation* 1 (2019), p. 100011.
- [121] Marcello Torchio et al. “Real-time model predictive control for the optimal charging of a lithium-ion battery”. In: *2015 American Control Conference (ACC)*. IEEE. 2015, pp. 4536–4541.
- [122] Kotub Uddin et al. “Characterising lithium-ion battery degradation through the identification and tracking of electrochemical battery model parameters”. In: *Batteries* 2.2 (2016), p. 13.
- [123] Lars Ole Valøen and Jan N Reimers. “Transport properties of LiPF₆-based Li-ion battery electrolytes”. In: *Journal of The Electrochemical Society* 152.5 (2005), A882.
- [124] Hado Van Hasselt, Arthur Guez, and David Silver. “Deep reinforcement learning with double q-learning”. In: *Thirtieth AAAI conference on artificial intelligence*. 2016.
- [125] Jorge Vazquez-Arenas et al. “A rapid estimation and sensitivity analysis of parameters describing the behavior of commercial Li-ion batteries including thermal analysis”. In: *Energy Conversion and Management* (2014), pp. 472–482.
- [126] William N Venables and Brian D Ripley. *Modern applied statistics with S-PLUS*. Springer Science & Business Media, 2013.
- [127] Wladislaw Waag, Christian Fleischer, and Dirk Uwe Sauer. “Critical review of the methods for monitoring of lithium-ion batteries in electric and hybrid vehicles”. In: *Journal of Power Sources* 258 (2014), pp. 321–339.
- [128] Thomas Waldmann et al. “Temperature dependent ageing mechanisms in Lithium-ion batteries—A Post-Mortem study”. In: *Journal of Power Sources* 262 (2014), pp. 129–135.

- [129] Eric Walter and Luc Pronzato. *Identification of parametric models from experimental data*. Springer Verlag, 1997.
- [130] Yan-Wu Wang et al. “Optimal persistent monitoring using second-order agents with physical constraints”. In: *IEEE Transactions on Automatic Control* 64.8 (2018), pp. 3239–3252.
- [131] Yilin Yin et al. “New fast charging method of lithium-ion batteries based on a reduced order electrochemical model considering side reaction”. In: *Journal of Power Sources* 423 (2019), pp. 367–379.
- [132] Hong Yue et al. “Sensitivity analysis and robust experimental design of a signal transduction pathway system”. In: *International Journal of Chemical Kinetics* 40.11 (2008), pp. 730–741.
- [133] Dong Zhang et al. “Real-time capacity estimation of lithium-ion batteries utilizing thermal dynamics”. In: *IEEE Transactions on Control Systems Technology* 28.3 (2019), pp. 992–1000.
- [134] Liqiang Zhang et al. “Multi-objective optimization of lithium-ion battery model using genetic algorithm approach”. In: *Journal of Power Sources* (2014), pp. 367–378.
- [135] Sheng S Zhang, Kang Xu, and TR Jow. “Study of the charging process of a LiCoO₂-based Li-ion battery”. In: *Journal of power sources* 160.2 (2006), pp. 1349–1354.
- [136] Xin Zhou, Jeffrey L Stein, and Tulga Ersal. “Battery state of health monitoring by estimation of the number of cyclable Li-ions”. In: *Control Engineering Practice* 66 (2017), pp. 51–63.
- [137] Changfu Zou et al. “Electrochemical estimation and control for lithium-ion battery health-aware fast charging”. In: *IEEE Transactions on Industrial Electronics* 65.8 (2017), pp. 6635–6645.
- [138] Changfu Zou et al. “Electrochemical estimation and control for lithium-ion battery health-aware fast charging”. In: *IEEE Transactions on Industrial Electronics* 65.8 (2018), pp. 6635–6645.
- [139] Hui Zou, Trevor Hastie, and Robert Tibshirani. “Sparse principal component analysis”. In: *Journal of computational and graphical statistics* 15.2 (2006), pp. 265–286.
- [140] Matthias Zschornak et al. “Fundamental principles of battery design”. In: *Physical Sciences Reviews* 3.11 (2018).