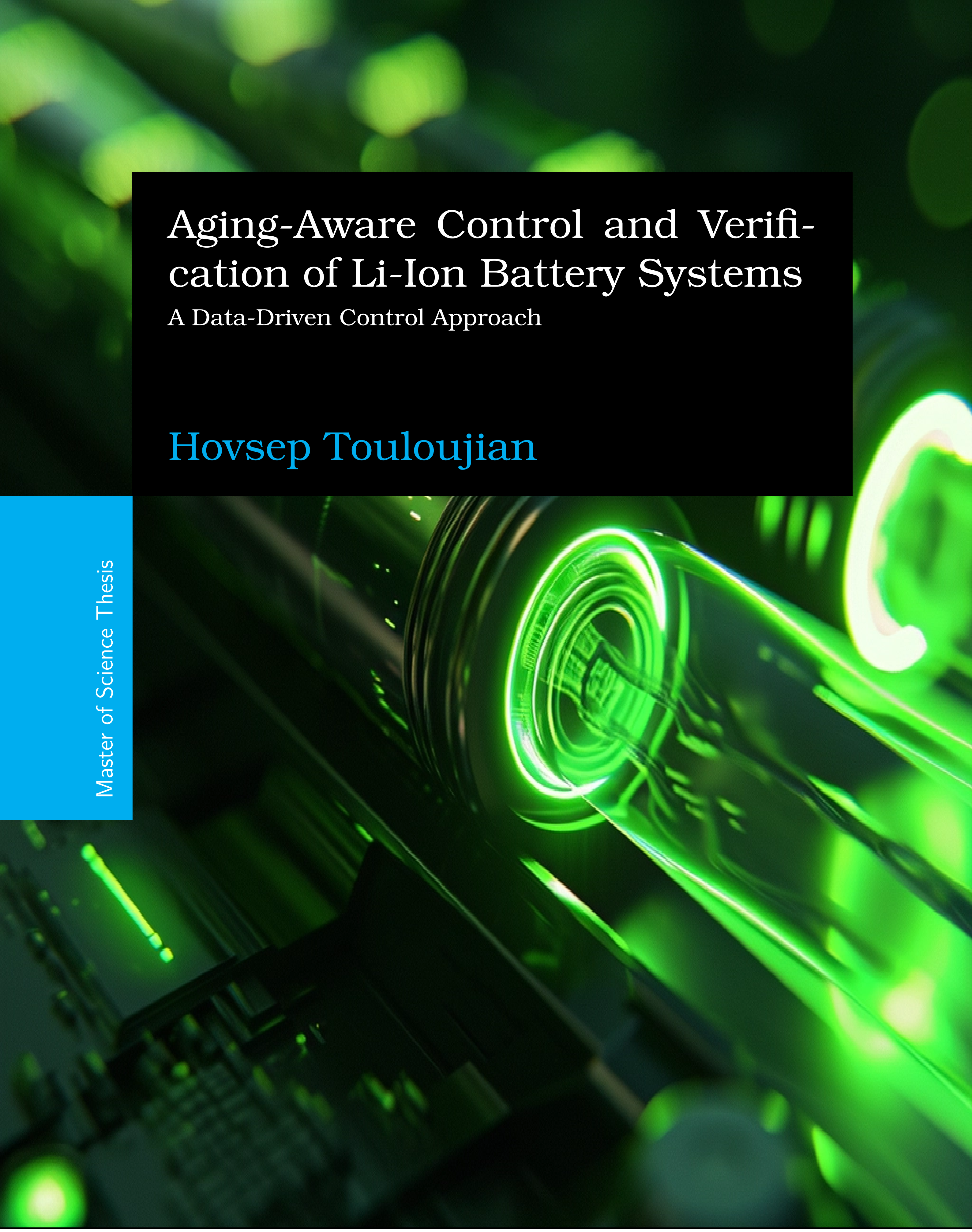


Aging-Aware Control and Verification of Li-Ion Battery Systems

A Data-Driven Control Approach

Hovsep Touloujian

Master of Science Thesis

Delft University of Technology

Delft Center for Systems and Control

Aging-Aware Control and Verification of Li-Ion Battery Systems

A Data-Driven Control Approach

Master of Science Thesis

For the degree of Master of Science in Systems and Control at Delft University of Technology

Hovsep Touloujian

June 11, 2024

Faculty of Mechanical Engineering (ME) · Delft University of Technology

Abstract

Rechargeable Lithium(Li)-Ion Batteries are a ubiquitous element of modern technology, as they pertain to efficient and sustainable energy storage for Electric Vehicles (EVs), as well as wind and solar farms. In the last decades, the production and design of such batteries and their adjacent embedded control, charging, and safety protocols, denoted by Battery Management Systems (BMS), has taken centre stage in the energy transition. A fundamental challenge to be addressed in battery technology, however, is the trade-off between the speed of the charging protocol employed by the BMS and the aging behaviour exhibited by the battery resulting in the loss of capacity in the battery cell, all while maintaining the safe operation of the battery.

This thesis aims to explore electrochemical models describing the charging and aging behaviour of Li-Ion Battery Systems, as well as the current existing charging protocols that aim to maximize charging speed while minimizing the aging effects that result in capacity loss. The proposed approach in this thesis is to adopt a data-driven approach to controller design, implementing improvements on an existent Reinforcement Learning (RL) pipeline to design an aging-aware battery-charging protocol, and extending the work into the field of Formal Methods for Systems and Control. This is done by expressing the closed-loop system resulting from a trained charging policy in the form of a data-driven abstraction capable of verifying the formal system specifications under probabilistic guarantees. Furthermore, a Counterexample-Guided Inductive Synthesis (CEGIS) scheme is proposed to additionally guide the training of the charging policy based on information from the learning results.

Table of Contents

Acknowledgements **v**

1 Introduction **1**

1-1 Challenges in Li-Ion Battery Technology 2

1-2 Data-Driven Symbolic Control Synthesis 3

1-3 Thesis Goals and Outline 4

2 Li-Ion Battery Systems **6**

2-1 Electrochemical Principles 6

2-1-1 Aging Behavior 8

2-2 Doyle-Fuller-Newman (DFN) Model 8

2-2-1 Electrochemical-Thermal Dynamics 9

2-2-2 Reaction Kinetics 11

2-2-3 Aging Model 12

2-3 Control of Li-Ion Battery Systems 12

2-3-1 Model-Based Methods 12

2-3-2 Reinforcement Learning 14

2-4 Desired Control Specification 15

3 Data-Driven Symbolic Methods **17**

3-1 Formal System Definitions 17

3-2 Data-Driven Abstractions 19

3-2-1 Parameter Uncertainty 23

3-3 Counterexample-Guided Inductive Synthesis (CEGIS) 24

4 Proposed Framework **26**

4-1 Data-Driven Approach 26

4-2 Verification and CEGIS Framework 27

5 Results 30
5-1 Battery Model and Baseline Protocol . . . 30
5-2 Abstraction Construction . . . 31
5-3 Learning Results and Counterexamples . . . 34
5-3-1 First RL Run . . . 35
5-3-2 Low-Voltage Charger . . . 38
5-3-3 High-Voltage Charger . . . 40
5-4 Final Charging Protocol . . . 41

6 Conclusions and Future Work 45

A Review of Reinforcement Learning 48

B Numerical Implementation of DFN Model 51
B-1 Parameters and Expressions . . . 51
B-2 CC-CV Charging Implementation . . . 53

Bibliography 54

Acknowledgements

I would like to first thank my supervisors, Dr. Manuel Mazo Jr. and Rudi Coppola for their patience, competence, and support in this project. The weekly discussions in Wiener Hall with them and the rest of the Symbolic and Networked Control (SYNC) Group have given me invaluable insight into the potential of our field, and the nature of academia in general. It is no exaggeration in saying that they were my favorite part of the week for an entire 9 months.

To my friends at DCSC: the lunches, coffee breaks, and conversations about nothing and everything all at once made the difficult days of this thesis just a bit easier. As control engineers, we share the unique position of being absurdly academically diverse, but still sharing the same understanding of relatively obscure knowledge to the point where sometimes only we can truly understand the value of each other's work. For this, I feel immensely privileged.

To my Lebanese friends: your constant phone calls, keeping me updated on the comings and goings back home, your imparting of wisdom (or the lack thereof) when I needed advice, and your repeated reminding me of my value meant more than you can imagine.

Finally, to my family: for giving me your opinion on how the figures in this thesis look, for your patience in describing recipes in intricate detail, and for supporting my MSc journey, I am so grateful. This project is impossible without the big four: Artine, Daad, Alexi, and Adriana, for whom this bell tolls, and this battery discharges.

Delft, University of Technology
June 11, 2024

Hovsep Touloujian

“Although it has been known for almost two centuries, lithium is suddenly making the news: it is the primary ingredient of the lithium-ion batteries set to power the next generation of electric vehicles and, as such, could become as precious as gold in this century.”

— *Jean-Marie Tarascon*

“It is fair to state, that in this digital era correct systems for information processing are more valuable than gold.”

— *H. Barendregt*

Chapter 1

Introduction

With the perpetually growing demand for energy, today's markets are faced with heavy requirements for the next decade. Those requirements, illustrated and predicted in Figure 1-1, must be achieved in sustainable manners, depending less on fossil fuels and maintaining low carbon dioxide emissions. As a result, high interest in sustainable energy sources has branched into several fields, such as wind energy, solar energy, as well as low-emission transportation in the form of Electric Vehicles (EVs) [1]. For the optimal use and global distribution of such energy sources, energy storage is a strong necessity.

Effectively, the field of sustainable energy storage continues to be a limiting factor in today's technological development, as many storage methods require high costs of manufacturing and production. Considerably so, electrochemical storage systems in the form of batteries constitute the cornerstone of several modern engineering applications, including those of EVs, solar or wind energy farms, and all portable technology such as smartphones and laptops. Due to their ubiquity, it is a necessary goal for battery manufacturers and designers to produce the most environmentally sustainable battery systems for accessible applications, while maintaining awareness of material scarcity and recyclability.

The re-usability of batteries after they have been depleted represents a fundamental aspect of the above general goal, considering the amount of material waste that may result from disposing of batteries at an extremely high rate. As a result, rechargeable batteries define the main focus of sustainable battery design, as the last decades have observed a soaring interest in Lithium(Li)-Ion batteries in both academia and industry. Lithium, as opposed to other metallic elements such as nickel or lead, represents the standard in rechargeable batteries due to its high specific power (300-1500 W/kg) and specific energy (100-270 Wh/kg), as well as ionization potential [2]. In short, Lithium is a relatively cheap metal that can safely and efficiently ionize into Li^+ ions, subsequently releasing one electron particle and producing the desired electrical energy. This process, which describes the discharge cycle of a battery, can also be easily reversed, allowing electrical energy to be restored within the battery, and resulting in the desired rechargeability features.

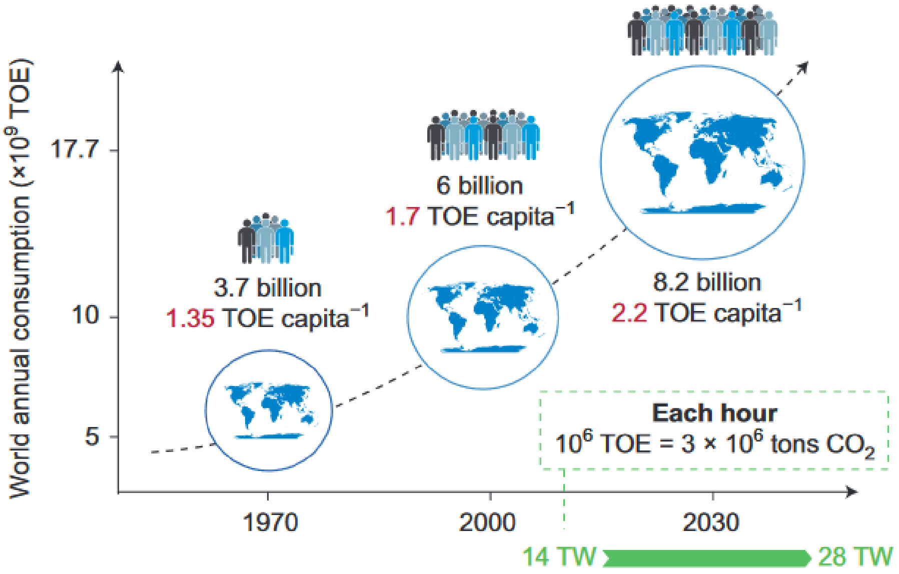


Figure 1-1: Past, Present, and Forecast of Global Energy Needs up to 2050. TOE = Ton of Oil Equivalent [1]

1-1 Challenges in Li-Ion Battery Technology

Despite the advances in the sustainable energy fields that Li-Ion batteries have generated, several roadblocks stand in the way of further development. On a global scale, this consists of the challenges to ensure the sustainability, safety, and performance of every stage of the supply chain and production of Li-Ion Battery packs as illustrated in Figure 1-2. These challenges span the mining and processing of raw materials (Lithium, Nickel, Cobalt, etc.), the production of battery cells and components, such as the anode, cathode, electrolyte solution, and separator, as well as the production of battery packs capable of powering large-scale systems such as EVs, and finally the recycling and re-use of recoverable components after the battery is no longer viable [3]. Furthermore, the wide adoption of EV technology depends on a large infrastructure of charging stations to be efficiently distributed across cities and regions, which has motivated several projects across Europe with the aim of normalizing the use of Electric Vehicles [4].

On a smaller scale, challenges to the industry stem from the fundamental properties of Lithium batteries, i.e., the electrochemical behavior of a single Li-Ion cell. Most notably, an apparent reaction of Li-Ion batteries to repeated cycles of charging and discharging is the inevitable development of several aging mechanisms, primarily in the form of a Solid-Electrolyte Interface (SEI), which is a layer that builds up around the solid lithium particles within the battery and increases the impedance of the battery by preventing such lithium particles from ionizing, effectively reducing the capacity of the battery. The issue of aging poses a fundamental challenge to the daily use of battery devices, as an exacerbated SEI buildup can lead to a battery that depletes in considerably less time, rendering those devices unusable unless the battery is replaced, which may be a highly costly and environmentally unsustainable process. Furthermore, many performance criteria must be accounted for during battery operation,

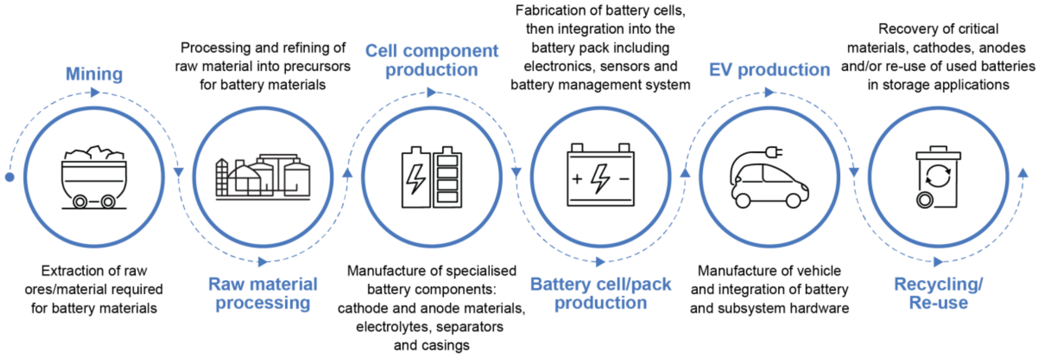

Mining
Extraction of raw ores/material required for battery materials
Processing and refining of raw material into precursors for battery materials
Raw material processing
Cell component production
Manufacture of specialised battery components: cathode and anode materials, electrolytes, separators and casings
Fabrication of battery cells, then integration into the battery pack including electronics, sensors and battery management system
Battery cell/pack production
EV production
Manufacture of vehicle and integration of battery and subsystem hardware
Recovery of critical materials, cathodes, anodes and/or re-use of used batteries in storage applications
Recycling/ Re-use

Figure 1-2: Stages of Production of an EV Battery Pack [5]

including charging speed, temperature regulation, accurate charge and capacity estimation, and the minimization of battery aging. Such requirements appeal to the development of control-based techniques in battery operation, as has been done with the introduction of Battery Management Systems (BMSs). The general functions of a BMS consist of:

- Measuring the several output variables of a battery during operation, typically Voltage, Current, and Temperature.
- Estimating internal battery states during operation, such as the concentration of lithium ions, SEI buildup, and battery capacity.
- Applying charging and discharging protocols that ensure the optimal behavior of the battery in terms of charging speed and performance, while maintaining its long-term health by minimizing aging effects.

From a Systems and Control perspective, BMSs are the centerpieces of battery operation and design, as they administer the charging and discharging protocols of the battery to ensure that the desired behavior is obtained, A key challenge for control designers, and thus a key challenge of this thesis project is to realize the following control objective:

Control Objective:
Design a charging policy for a Li-Ion Battery cell to maximize charging speed, while minimizing charging time.

Throughout this work, the electrochemical charging and aging behavior of a Li-Ion cell is formalized in a mathematical model, which allows for a formal description of the above objective.

1-2 Data-Driven Symbolic Control Synthesis

In addition to exploring the field of Li-Ion Battery Systems, this thesis is centered around applying advanced symbolic methods for control, aiming to combine the two areas of study. The field of symbolic methods has gained traction in the last decades with the growing collaboration between Systems and Control and Computer Science. Since then, computational methods for categorizing systems by behavioral and simulation relations, constructing finite-state abstractions of infinite-state systems, and performing formal verification on system dynamics, have been of wide interest in academic and industrial areas [6]. The value of symbolic methods thus lies in the formalization of system verification, so as to provide guarantees that a certain controller satisfies given safety specifications imposed on the system.

In recent years, the evolution of computational methods has allowed for the implementation of data-driven techniques that further enhance control applications, such as the use of Machine Learning (ML) and Reinforcement Learning (RL) to produce model-free control policies. The development of such control policies is advantageous, as complex dynamics such as those of a Li-Ion cell are circumvented at the cost of extensive training time. Furthermore, issues that impede conventional control strategies, namely Model Predictive Control (MPC) and output feedback, such as parameter uncertainty and lack of observability, can be bypassed with the use of data-driven methods. However, data-driven methods suffer from difficulties in verification, as those controllers are optimized based on their observations, but not certain to operate adequately under unobserved cases, which poses a challenge to the safe implementation of data-driven controllers.

As a proposed remedy, it is of interest to leverage data-driven methods in combination with well-established formal and symbolic methods for control to provide formal guarantees for system specifications and to better guide controller synthesis. One such method consists of the recently developed methods in [7] to construct a finite-state abstraction based on observations from the system, thus denoted by a data-driven abstraction. The validity of this abstraction can be evaluated with probabilistic guarantees based on the scenario approach. It is thus also of interest to formally introduce these notions throughout the thesis.

1-3 Thesis Goals and Outline

The general goal of this MSc Thesis work is to develop methods for the more sustainable charging and operation of Li-Ion Battery Systems, aiming to respond to the growing energy requirements in the future. More specifically, the undertaken research goals are to employ data-driven methods to enhance the widely established symbolic methods for controller synthesis and verification. Furthermore, this thesis is the first of its kind to combine the field of Formal Methods for Systems and Control with the experience and knowledge of the electrochemical behavior and mathematical models of Li-Ion Battery Systems, so as to develop and verify a charging protocol aiming to maximize the life span of a Li-Ion cell.

Thus, the general research goal, as well as the specific technical areas explored throughout the thesis formulate the corresponding research questions that are to be considered, as listed in Table 1-1.

This work begins with an introduction to the electrochemical charging and aging behavior exhibited by a Li-Ion cell from a modeling standpoint in Chapter 2, further investigating Research Question 2 by considering the battery system from a Systems and Control perspective through the formalization of control performance specifications. In Chapter 3, a review of

Research Question 1: Data-Driven Methods and Symbolic Control
How can data-driven methods be used to improve symbolic methods for control and verification?

Research Question 2: Li-Ion Batteries
What are the electrochemical properties of Li-Ion Battery Systems and how can they be formalized into a Systems and Control framework?

Research Question 3: Aging-Aware Charging
How can data-driven and symbolic methods be employed to synthesize an aging-aware charging protocol for Li-Ion Battery Systems?

Table 1-1: Research Questions

Formal Methods for Systems and Control is provided, along with the necessary tools to extend into the use Data-Driven Abstractions for the verification of autonomous systems, which constitute a pivotal aspect of the project. Thereafter, Chapter 4 introduces the proposed approach to controller synthesis and verification with the use of RL and a Counterexample-Guided Inductive Synthesis (CEGIS) scheme to improve the RL results. The results of controller synthesis and verification are presented and compared to a baseline charger in Chapter 5. Finally, this work is concluded and possible future work is discussed in Chapter 6.

Chapter 2

Li-Ion Battery Systems

This chapter is concerned with introducing the main elements and principles that constitute a Li-Ion Battery System. This includes a preliminary description of the physical and electrochemical composure and mechanisms within the system, followed by a more quantitative perspective utilizing mathematical models that describe the system's behavior. Namely, this is done by introducing the model developed by Doyle, Fuller, and Newman, denoted by the DFN model.

An additional goal of this chapter is to introduce the physical phenomena and mathematical models that encapsulate battery behavior. This includes the aging process of a Li-Ion battery, which is most commonly observed due to the buildup of a Solid-Electrolyte Interface (SEI) layer within the electrodes.

Thirdly, a short discussion on the control/charging of Li-Ion cells is provided, focusing mainly on the design of a protocol that depends on Reinforcement Learning (RL), as that is of central importance to the work in this thesis.

The chapter concludes with the first contribution of this thesis, which is to formally define the desired control specification for a charging protocol based on the Reach-While-Avoid specification to be introduced subsequently.

2-1 Electrochemical Principles

The standard Li-Ion cell comprises three layers, denoted by the negative electrode (anode), the separator, and the positive electrode (cathode), placed in a respective adjacent configuration as shown in the schematic in Figure 2-1. Each of the electrodes consists of material such as graphite, carbon, titanate, or silicon, over which material particles containing lithium are located and immersed in a liquid-electrolyte solution. Typically, such material particles in the negative electrode, i.e., the anode, are of a Lithium-Graphite compound with a molecular structure Li_xC_6, whereas those in the positive electrode, i.e., the cathode, consist of a metal-oxide structure of the form Li_yMO_2, such as Cobalt, denoting the structure by $LiCoO_2$. For

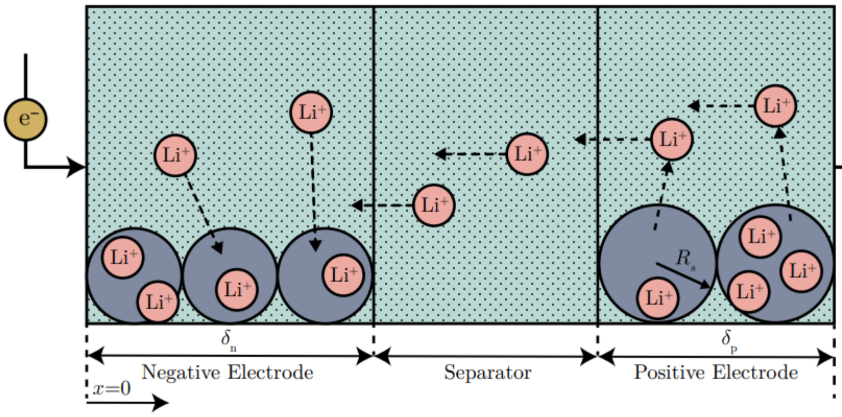


Figure 2-1: Schematic of a Standard Li-Ion Cell Configuration during Charging [10]

example, the typical charge-transfer reactions in such battery cells can be described, as in [8], by

$$\textbf{Cathode:} \quad LiCoO_2 \underset{\text{discharge}}{\overset{\text{charge}}{\rightleftarrows}} Li_{1-x}CoO_2 + xLi^+ + xe^- \tag{2-1}$$

$$\textbf{Anode:} \quad C_6 + xLi^+ + xe^- \underset{\text{discharge}}{\overset{\text{charge}}{\rightleftarrows}} Li_xC_6 \tag{2-2}$$

In a standard discharge cycle, Li particles in the negative electrode are ionized by the loss of an electron, which exits the negative electrode through a current collector, thus generating current from the battery. The ionization of Li particles occurs by a diffusive process, as the solid Li particles travel to the surface of the Li_xC_6 particles, intercalating from the solid particles and undergoing an electrochemical reaction with the surrounding liquid-electrolyte solution. This results in the release of an electron (e^-) and the production of a Li^+ ion that diffuses into the electrolyte solution immersing the material particles. In liquid-electrolyte form, the ionized lithium further diffuses throughout the solution in the negative electrode, then the separator, and finally to the solution in the positive electrode. The electrolyte solution in the positive electrode is then also involved in a chemical reaction with the solid material particles immersed in it, resulting in the diffusion of the ionized lithium back into a solid within the metal-oxide material particles. During a charging cycle, the process is reversed, as current is applied to the battery, which results in the diffusion and ionization of Li particles from the positive electrode, and back to the negative electrode, as illustrated in Figure 2-1. This structure thus allows the rechargeable behavior of a Li-Ion cell [9].

Furthermore, the function of the separator in a battery cell is to regulate the flow of the liquid-electrolyte solution between the two electrodes, as its porous properties limit the amount of the Li^+ ions that diffuse through it. In the absence of a separator, the diffusion between two electrodes is left unregulated, resulting in an uncontrollable amount of electrochemical reactions and thus a highly unsafe system that is prone to catching fire or even explosion.

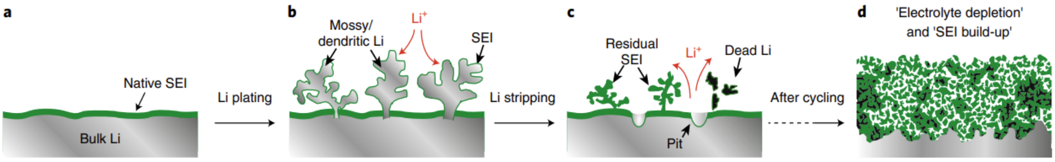


Figure 2-2: Schematic of the SEI Buildup Process in the Negative Electrode [15]

2-1-1 Aging Behavior

While the charging and discharging cycles are chemically reversible, limitations on the cyclical performance of the battery are imposed by the aging mechanisms that occur during battery operation. The most common and prominent form of aging in a battery occurs through the creation and buildup of a layer over the solid material particles in the negative electrode, which impedes the contact between the solid particles and the liquid-electrolyte solution, further limiting the intercalation of Li molecules, and effectively limiting the capacity of the battery, i.e., the amount of charge it can hold when fully charged. This layer already exists during the initial cycle of the cell as a protective layer for the negative electrode against corrosion and reduction reactions. However, its size increases as the growth of the solvent within the SEI leads to further interaction with graphite and other solid particles within the negative electrode, which results in gas release that can crack the SEI and allow its further expansion [11].

The SEI growth dynamics, as described in the subsequent mathematical models, can be characterized by an electric flux induced by side reactions during battery operation, as is initially studied and modeled in [12]. The magnitude of this side-reaction flux is found to be proportional to internal electric potentials within the cell, as well as the current thickness of the SEI layer.

Furthermore, several factors affect the rate of growth of the SEI, including high cell temperature, short circuits, and high charge, which is correlated to a large potential between the solid particles and liquid-electrolyte solution in the negative electrode, thus increasing the side-reaction flux. It is further noted that the charging speed of the battery is by nature proportional to the rate of SEI growth, which means that aging-aware charging is a trade-off between minimizing both charging time and the aging effects of the battery, as is widely discussed in works that incorporate aging-aware charging protocols such as [13] and [14].

2-2 Doyle-Fuller-Newman (DFN) Model

For the quantification of the behavior of Li-Ion cells, several mathematical models have been constructed, such as Equivalent Circuit Models (ECMs), where a battery can be simplified to a circuit component such as a capacitor, as demonstrated in [16] and [17]. Furthermore, some battery models can be constructed from observation data, resulting in data-driven models, as performed in [18] and [19]. However, it is of interest to characterize a Li-Ion cell by its electrochemical properties for improved accuracy, motivating the need for Electrochemical Models (EMs). In EMs, the behavior of the battery is characterized by internal electrochemical states, such as the concentration of solid lithium in the material particles in each of the negative and positive electrodes, the concentration of Li^+ ions in the liquid-electrolyte solution, the internal temperature of the cell, and the thickness of the SEI surrounding the material particles. These states vary based on spatial and temporal dynamics described by Partial Differential Equations (PDEs).

The most widely used EM is that developed by Doyle, Fuller, and Newman (DFN) in [20], also sometimes referred to as the Pseudo 2-Dimensional (P2D) Model. The DFN Model consists of four PDEs, in addition to several algebraic equations describing the reaction kinetics based on the solid-phase and electrolyte-phase Li behavior. The model introduced hereafter is adapted from [13] and [21]. Numerical implementations of the model are also available in MATLAB and Python, developed in works such as [22], [23], and [21].

2-2-1 Electrochemical-Thermal Dynamics

For the model, time is denoted by t, and the spatial domain is denoted by x, where, similar to the representation in Figure 2-1, the intervals $[0, \delta_n]$, $[\delta_n, L - \delta_p]$, and $[L - \delta_p, L]$ describe the negative electrode, separator, and positive electrode, respectively.

The **Li concentration in the solid phase** $c_s(x, r, t)$, i.e., radially described for a given solid material particle at position x and time t, for $x \in [0, \delta_n] \cup [L - \delta_p, L]$ is given by Fick's Law as

$$\frac{\partial c_s}{\partial t} = \frac{D_s}{r^2}\frac{\partial}{\partial r}\Big(r^2\frac{\partial c_s}{\partial r}\Big), \tag{2-3}$$

$$\frac{\partial c_s}{\partial r}\Big|_{r=0} = 0, -D_s\frac{\partial c_s}{\partial r}\Big|_{r=R_s} = j_1, \tag{2-4}$$

where D_s denotes the diffusion coefficient of solid-phase lithium, j_1 denotes the main reaction flux, i.e., the component of the chemical reactions that are not the side reactions that consist of the aging mechanism. Furthermore, δ_n, δ_p, and L denote the thickness of the anode, cathode, and cell, respectively.

The **Li concentration in the electrolyte phase** $c_e(x, t)$ for $x \in [0, L]$ is given by

$$\varepsilon_e\frac{\partial c_e}{\partial t} = \frac{\partial}{\partial x}\Big(D_e\varepsilon_e^p\frac{\partial c_e}{\partial x}\Big) + a_s(1 - t_+^0)j_n, \tag{2-5}$$

$$\frac{\partial c_e}{\partial x}\Big|_{x=0} = \frac{\partial c_e}{\partial x}\Big|_{x=L} = 0, \tag{2-6}$$

where ε_e denotes the volume fraction of the electrolyte phase, p the Bruggeman porosity exponent, and D_e the diffusion constant of electrolyte-phase Li-ions. The symbol $a_s = 3\varepsilon_s/R_s$ denotes the specific interfacial surface area, where ε_s represents the active material volume fraction, and t_+^0 the transference number of Li-ions. Furthermore, the term $j_n = j_1 + j_2$ represents the net molar flux of the reactions, i.e., the sum of main and side reactions. It is noted that $j_2 = 0$ for the positive electrode, as side reactions are only assumed to occur in the negative electrode.

The **potential of Li in the solid phase** $\phi_s(x, t)$ for $x \in [0, \delta_n] \cup [L - \delta_p, L]$ is given by

Ohm's Law, i.e.,

$$\frac{\partial}{\partial x}\Big(\sigma\varepsilon_s\frac{\partial\phi_s}{\partial x}\Big) = \frac{3\varepsilon_s F}{R_s}j_n, \tag{2-7}$$

$$\sigma\varepsilon_s\frac{\partial\phi_s}{\partial x}\Big|_{x=0} = \sigma\varepsilon_s\frac{\partial\phi_s}{\partial x}\Big|_{x=L} = -\frac{I_{cell}}{A}, \tag{2-8}$$

$$\frac{\partial\phi_s}{\partial x}\Big|_{x=\delta_n} = \frac{\partial\phi_s}{\partial x}\Big|_{x=L-\delta_p} = 0, \tag{2-9}$$

where F denotes Faraday's constant and I_{cell} denotes the applied cell current. In this model, the convention is defined such that $I_{cell} < 0$ corresponds to charging, whereas the opposite corresponds to discharging.

Furthermore, the **potential of Li in the electrolyte phase** $\phi_e(x,t)$ for $x \in [0, L]$ is given by

$$\frac{\partial}{\partial x}\Big(\kappa\varepsilon_e^p\frac{\partial\phi_e}{\partial x} + \kappa\varepsilon^p(t_+^0 - 1)\frac{2RT}{F}\frac{\partial \ln c_e}{\partial x}\Big) = -\frac{3\varepsilon_s F}{R_s}j_n, \tag{2-10}$$

$$\frac{\partial\phi_e}{\partial x}\Big|_{x=0} = \phi_e\Big|_{x=L} = 0, \tag{2-11}$$

where R denotes the ideal gas constant, and T denotes the temperature of the cell, and κ denotes the ionic conductivity in the electrolyte solution. The cell voltage is described by the solid-phase potential difference at the two ends of the cell, i.e.,

$$V(t) = \phi_s(L,t) - \phi_s(0,t). \tag{2-12}$$

The level of charge in the cell is defined by the bulk **State of Charge (SOC)** of the negative electrode, given by

$$SOC^-(t) = \frac{1}{\delta_n}\int_0^{\delta_n}\frac{(\bar{c}_s^-(x,t)/c_{s,max}^-) - \theta_{0\%}^-}{\theta_{100\%}^- - \theta_{0\%}^-}, \tag{2-13}$$

where $\theta_{0\%}^-$ and $\theta_{100\%}^-$ denote the stoichiometric coefficient of solid-phase lithium in the negative electrode at SOC 0% and 100%, respectively, and $c_{s,max}^-$ denotes the maximum solid-phase concentration in a solid particle in the negative electrode. Furthermore, $c_s^-(x,t)$ for $x \in [0, \delta_n]$ denotes the volume-averaged concentration of solid-phase Li in a material particle, given by

$$\bar{c}_s^-(x,t) = \frac{3}{R_s^3}\int_0^{R_s} r^2 c_s(x,r,t)dr. \tag{2-14}$$

Although the SOC is measured in terms of the negative electrode, it can be analogously defined in terms of the positive electrode and denoted by $SOC^+(t)$, which is useful to describe the thermal dynamics given by

$$mc_p\frac{dT}{dt}(t) = \frac{1}{R_{th}}(T_{amb} - T) + I_{cell}(U^+(SOC^+(t)) - U^-(SOC^-(t)) - V(t)), \tag{2-15}$$

where m denotes the mass of the cell, c_p the heat capacity of the cell, R_{th} its convective thermal resistance, and U^- and U^+ the open-circuit potential of solid material in the negative and positive electrode, respectively. The functions U^- and U^+ are functions of the SOC that are usually derived from optimal fittings based on experimental data.

2-2-2 Reaction Kinetics

The above PDEs are coupled with rate equations describing the reaction kinetics. The **main reaction flux** for the positive or negative electrode $j_1^{\pm}(x,t)$ is modeled by the Butler-Volmer kinetic equation:

$$j_1^{\pm} = \frac{i_{0,1}^{\pm}}{F}\Bigg(\exp\Big(\frac{\alpha_a F}{RT}\eta_1^{\pm}\Big) - \exp\Big(-\frac{\alpha_c F}{RT}\eta_1^{\pm}\Big)\Bigg), \tag{2-16}$$

where α_a and α_c denote the anodic and cathodic transfer coefficients, and $i_{0,1}$ denotes the **exchange current density**, which is given by

$$i_{0,1}^{\pm} = k_0 c_e^{\alpha_a}(c_{s,max}^{\pm} - c_{s,e})^{\alpha_a} c_{s,e}^{\alpha_c}, \tag{2-17}$$

for some kinetic constant $k_0^{\pm}$, where $c_{s,e}$ denotes the particle surface concentration, i.e., $c_{s,e}(x,t) = c_s(R_s,x,t)$. Furthermore, the electrode over-potential for the positive or negative electrode $\eta_1^{\pm}(x,t)$ is given by

$$\eta_1^{\pm} = \phi_s - \phi_e - U^{\pm}(SOC^{\pm}(t)) - FR_f j_n, \tag{2-18}$$

where U denotes the electrode equilibrium potential, and R_f denotes the negative electrode film resistance due to the buildup of the SEI layer. It is noted that in the above and later equations, the ± symbol is dropped when the context is clear.

Furthermore, the **side reaction flux** is denoted by $j_2(x,t)$, and is modeled by a Tafel kinetic equation describing an irreversible process, as opposed to a Butler-Volmer equation which described a reversible process. The expression is given by

$$j_2 = -\frac{i_{0,2}}{F}\exp\Bigg(-\frac{2\alpha_{c,2}F}{RT}\eta_2\Bigg), \tag{2-19}$$

where $i_{0,2}$ denotes the side-reaction exchange current density, which is a fixed parameter of the model, $\alpha_{c,2}$ denotes the cathodic transfer coefficient of the side reaction, and the overpotential η_2 is given by

$$\eta_2 = \phi_s - \phi_e - U_2 - FR_f j_n, \tag{2-20}$$

where U_2 denotes the equilibrium potential of the side reaction. It is reiterated that $j_2 = 0$ in the separator and positive electrode, as side reactions and aging mechanisms are only active in the negative electrode.

At this point, it is noted that the parameters D_s, D_e, and $k_0^{\pm}$ are modeled to be temperature-dependent based on the Arrhenius relationship given by

$$\psi = \psi_{ref}\exp\Big(\frac{E_\psi}{R}\Big(\frac{1}{T} - \frac{1}{T_{ref}}\Big)\Big), \tag{2-21}$$

where ψ refers to the temperature-dependent parameter, E_ψ its associated activation energy, and ϕ_{ref} the value of the parameter at a reference temperature T_{ref}.

2-2-3 Aging Model

The aging model describes the capacity fade in the battery due to the growth of the SEI. Thus, the corresponding film resistance R_f is modeled by

$$\frac{\partial R_f}{\partial t} = -\frac{\tilde{V}_f}{\sigma_f} j_2, \tag{2-22}$$

where $\tilde{V}_f$ denotes the molar volume and σ_f the electric conductivity of the SEI. Furthermore, the amount of charge Q_l lost due to the loss of Li-ions due to side reactions is defined to be

$$\frac{dQ_l}{dt} = -a_s AF \int_0^{\delta_n} j_2 dx. \tag{2-23}$$

The flowchart shown in Figure 2-3 represents the various interconnections between the dynamics of the DFN Model. Furthermore, the dynamics categorized by colored blocks that differentiate between battery electrochemical states, reaction rates, and aging behavior.

2-3 Control of Li-Ion Battery Systems

With the development of models that describe the charging/discharging, as well as aging dynamics of Li-Ion batteries, it is of interest to employ these models for the design of methods to estimate internal states and/or charging protocols that meet the desired control goals. This section briefly discusses the conventional control methods employed in those regards, and then focuses on the use of Reinforcement Learning (RL) to construct an output-feedback-based optimal charging policy, which represents a central aspect of the thesis work.

2-3-1 Model-Based Methods

Typically, the only measured states of a battery consist of its cell voltage (V), its cell current (I_{cell}), and temperature (T), whereas the internal chemical states are not directly measured. The SOC of the battery can however be estimated from the cell current, since electric charge can be expressed as the integral of electric current over time, resulting in a method denoted by Coulomb Counting, expressed as

$$SOC(t) = SOC(t_0) - \int_{t_0}^{t} \frac{I_{cell}(t)}{Q} \eta dt, \tag{2-24}$$

where t_0 and $SOC(t_0)$ represent some initial time and the SOC at that initial time, respectively, Q denotes the capacity of the battery, which can be assumed constant during one charging cycle, and η the Coulombic efficiency. Coulomb counting is an effective method to have a low-quality estimate of the SOC, and it is hereafter assumed that an estimate of the SOC is available as a measured output for the construction of an Output Feedback controller.

State estimation methods can be considerably improved using model-based techniques, where the measurements are incorporated using model dynamics to provide more accurate estimates. This consists of constructing state observes and Kalman filters based on the battery models,

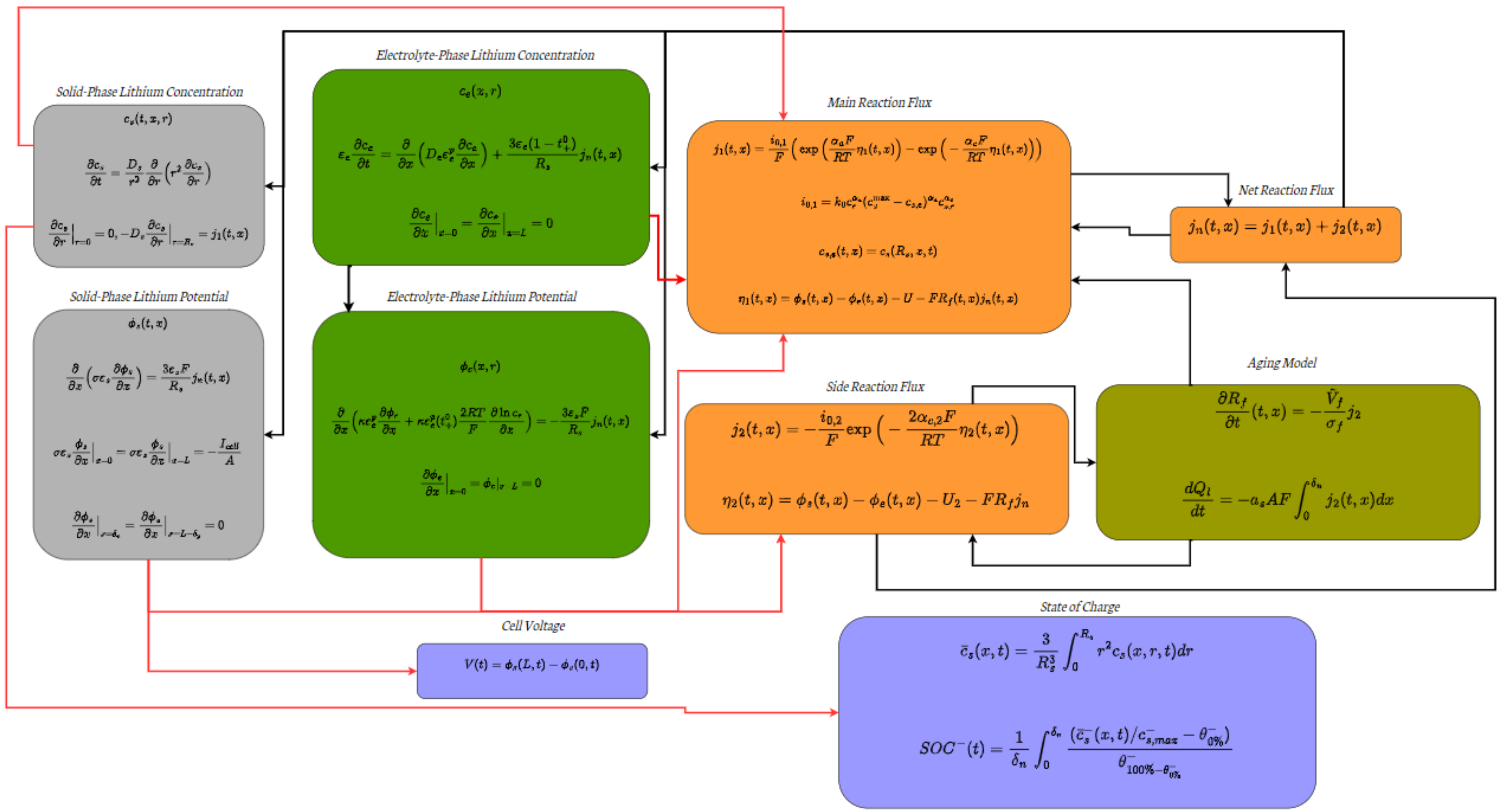

Solid-Phase Lithium Concentration
$c_s(t,x,r)$
$\frac{\partial c_s}{\partial t} = \frac{D_s}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial c_s}{\partial r}\right)$
$\frac{\partial c_s}{\partial r}\Big|_{r=0} = 0, -D_s\frac{\partial c_s}{\partial r}\Big|_{r=R_s} = j_1(t,x)$
Solid-Phase Lithium Potential
$\phi_s(t,x)$
$\frac{\partial}{\partial x}\left(\sigma\varepsilon_s\frac{\partial \phi_s}{\partial x}\right) = \frac{3\varepsilon_s F}{R_s}j_n(t,x)$
$\sigma\varepsilon_s\frac{\phi_s}{\partial x}\Big|_{x=0} = \sigma\varepsilon_s\frac{\phi_s}{\partial x}\Big|_{x=L} = -\frac{I_{cell}}{A}$
$\frac{\partial \phi_s}{\partial x}\Big|_{x=\delta_n} = \frac{\partial \phi_s}{\partial x}\Big|_{x=L-\delta_p} = 0$
Electrolyte-Phase Lithium Concentration
$c_e(x,r)$
$\varepsilon_e\frac{\partial c_e}{\partial t} = \frac{\partial}{\partial x}\left(D_e\varepsilon_e^p\frac{\partial c_e}{\partial x}\right) + \frac{3\varepsilon_e(1-t_+^0)}{R_s}j_n(t,x)$
$\frac{\partial c_e}{\partial x}\Big|_{x=0} = \frac{\partial c_e}{\partial x}\Big|_{x=L} = 0$
Electrolyte-Phase Lithium Potential
$\phi_e(x,r)$
$\frac{\partial}{\partial x}\left(\kappa\varepsilon_e^p\frac{\partial \phi_e}{\partial x} + \kappa\varepsilon_e^p(t_+^0)\frac{2RT}{F}\frac{\partial \ln c_e}{\partial x}\right) = -\frac{3\varepsilon_s F}{R_s}j_n(t,x)$
$\frac{\partial \phi_e}{\partial x}\Big|_{x=0} = \phi_e|_{x=L} = 0$
Main Reaction Flux
$j_1(t,x) = \frac{i_{0,1}}{F}\left(\exp\left(\frac{\alpha_a F}{RT}\eta_1(t,x)\right) - \exp\left(-\frac{\alpha_c F}{RT}\eta_1(t,x)\right)\right)$
$i_{0,1} = k_0 c_e^{\alpha_a}(c_s^{\max} - c_{s,e})^{\alpha_a}c_{s,r}^{\alpha_c}$
$c_{s,e}(t,x) = c_s(R_s,x,t)$
$\eta_1(t,x) = \phi_s(t,x) - \phi_e(t,x) - U - FR_f(t,x)j_n(t,x)$
Net Reaction Flux
$j_n(t,x) = j_1(t,x) + j_2(t,x)$
Side Reaction Flux
$j_2(t,x) = -\frac{i_{0,2}}{F}\exp\left(-\frac{2\alpha_{c,2}F}{RT}\eta_2(t,x)\right)$
$\eta_2(t,x) = \phi_s(t,x) - \phi_e(t,x) - U_2 - FR_f j_n$
Aging Model
$\frac{\partial R_f}{\partial t}(t,x) = -\frac{\tilde{V}_f}{\sigma_f}j_2$
$\frac{dQ_l}{dt} = -a_s AF\int_0^{\delta_n} j_2(t,x)dx$
Cell Voltage
$V(t) = \phi_s(L,t) - \phi_s(0,t)$
State of Charge
$\bar{c}_s(x,t) = \frac{3}{R_s^3}\int_0^{R_s} r^2 c_s(x,r,t)dr$
$SOC^-(t) = \frac{1}{\delta_n}\int_0^{\delta_n}\frac{(\bar{c}_s^-(x,t)/c_{s,max}^- - \theta_{0\%}^-)}{\theta_{100\%-\theta_{0\%}^-}^-}$

Figure 2-3: Flowchart Describing the Electrochemical Dynamics of the DFN Model

the application of which to battery systems has been widely considered in the literature, as is documented in works such as [24] and [25]. However, due to the heavily involved PDE dynamics and the lack of observability of the DFN Model, it is generally difficult to construct a state observer that depends on the DFN Model. For this reason, some simplified models such as Equivalent Circuit Models, or the Single-Particle Model [26] are used to perform state estimation. For example, works such as [27] and [28] simplify the PDE dynamics by expressing the model in terms of the average concentration over the spherical particle rather than the concentration profile. In the former work, the observer is constructed utilizing an Extended Kalman Filter, whereas a Moving-Horizon Estimation scheme is designed in the latter, to be further employed in an application of Model Predictive Control (MPC).

Furthermore, as the interest in developing control methods for batteries grew over the past decades, it has been of interest to develop battery models in a control context, as done in [9]. Aging-Aware MPC schemes are implemented on the DFN model, such as in [14], where a Finite-Dimensional Linear Time-Invariant model is used to describe the internal battery dynamics. This surrogate model is employed in an MPC scheme aiming to minimize the battery capacity loss, developing a Pareto front describing the trade-off between charging time and capacity loss. This work is expanded in [10], where a nonlinear MPC framework based on the DFN dynamics is implemented using nonlinear optimization methods such as Sequential Quadratic Programming (SQP), resulting in a considerable decrease in capacity loss during charging runs.

It is noted, however, that the MPC scheme developed in the aforementioned work is implemented in full-state feedback, i.e., the knowledge of all battery states is assumed, as no observer is designed to estimate the internal battery states based on the DFN model. As a result, the corresponding experimental runs are performed in an open-loop environment, where the control scheme is generated in simulation and administered to the battery, which is detrimental to the charging performance and safety. Furthermore, the computational difficulty of MPC based on the DFN model is illustrated in [29], where several problem formulation strategies are considered for the DFN model, each resulting in substantially large time required for a single optimization call (45-65 seconds), which renders real-time MPC on such a complex model infeasible in most practical applications.

2-3-2 Reinforcement Learning

In addition to the control methods discussed above, it is of interest to further discuss the use of Reinforcement Learning (RL) in developing aging-aware controllers for Li-Ion Battery Systems. Typically, RL consists of training a policy to maximize a certain user-defined reward function that evaluates the performance of the policy. For the case of batteries, a typical reward function penalizes slow charging and high aging effects caused by the charging protocol, while rewarding the opposite, i.e., fast charging with low aging behavior. Such a policy is designed in [21], where the DFN Model is used to emulate a single-cell Li-Ion Battery System, and a controller is trained for the case of both state feedback, i.e., the controller has full knowledge of the state, and output feedback, i.e., the controller is assumed to only know the cell voltage, the core temperature of the cell, and its State of Charge. The latter policy, which depends solely on the output variables, is of interest to the work in this thesis, as it circumvents the need for a state estimator, which has posed roadblocks with respect to the model-based approaches to charging.

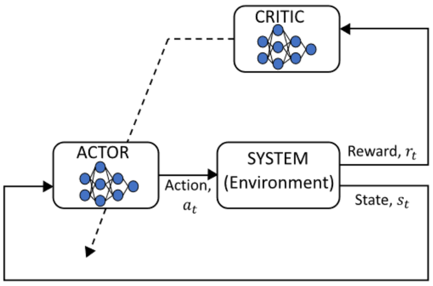


Figure 2-4: Actor-Critic Scheme [21]

In its original form, RL is performed in a discrete state and action space, where the controller has a finite set of actions to choose from, and the system operates with a finite number of states. Such a framework is often referred to as Q-learning. Despite Q-Learning being advantageous in providing convergence guarantees under mild assumptions, it suffers from the limitation of the finite state and action spaces, as well as the curse of dimensionality characterized by computations becoming more difficult as the partition of the state and action spaces becomes larger. Therefore, it is advantageous to employ RL in continuous spaces using the Deep Deterministic Policy Gradient (DDPG) framework introduced in [30]. In this case, the policy is defined by a neural network, and thus by a function rather than a table representing a discrete mapping. The DDPG framework is employed in an Actor-Critic scheme, as illustrated in Figure 2-4, in [21] to develop the aging-aware controller. The resulting protocol is shown to perform better than standard rule-based charging methods, both in terms of charging speed and reduction of aging effects. The mathematical formalization of the DDPG algorithm is provided in a Review of Reinforcement Learning in Appendix A.

2-4 Desired Control Specification

As the electrochemical behavior of a Li-Ion Battery has been mathematically formalized through the DFN Model, the goals of a charging protocol can similarly be mathematically formalized, thus putting the design of a charging protocol into a Systems and Control framework.

As is noted from the aging model, the rate of growth of the SEI is directly proportional to the side reaction flux j_2 as given by Equation 2-22. Furthermore, the Tafel equation given in Equation 2-19 shows that the side reaction flux is exponentially affected by the overpotential η_2, the behavior of which is characterized by the algebraic equation given in Equation 2-20. That is, based on the Tafel equation, a positive overpotential η_2 refers to a small SEI growth rate, while a negative η_2 refers to a larger one. The rate of SEI growth rises exponentially as η_2 decreases. As a result, it is a direct goal to maintain η_2 as large as possible during charging so as to minimize the rate of SEI growth over one charging cycle. This goal is thus used to define the system specification corresponding to aging-awareness during charging.

In addition to the aging-awareness goal, safety specifications are to be maintained on the charger, namely in the form of voltage and temperature constraints. It is aimed to maintain the battery voltage below a nominal charging voltage of $4.2V$ while maintaining the battery

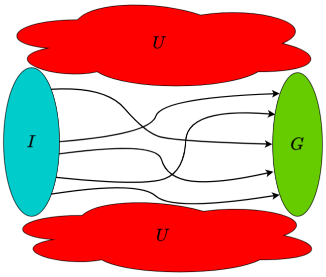


Figure 2-5: Schematic of a Reach-While-Avoid Specification

temperature below $35°C$, or $308K$. By maintaining those conditions, the charger can reduce the chances of thermal runaway, fire, or explosions in the cell.

The above goals are employed to formally define a control specification to be employed for the synthesis of an aging-aware charging protocol. The control specification is defined based on the Reach-While-Avoid (RWA) specification, defined as follows:

Definition 1 (Reach-While-Avoid Specification)**.** *Consider a dynamical system with state $x \in \mathcal{X}$, a set of unsafe states denoted by U and its complement set of safe states $S = \mathcal{X} \backslash U$, a set of initial states $I \subseteq S$ and a set of goal states $G \subseteq S$. The Reach-While-Avoid specification is characterized by the guarantee that a trace starting from the set I reaches the set G in finite time while avoiding the set U, i.e., remaining within S. Formally,*

$$\forall x(0) \in I \Rightarrow (\exists T)\Big(x(T) \in G \wedge (\forall 0 \leq t \leq T) x(t) \notin U\Big). \tag{2-25}$$

The RWA specification is illustrated by the schematic shown in Figure 2-5. In the case of the considered Li-Ion Battery Systems, the RWA specification can be defined with respect to the State of Charge of the battery and the overpotential η_2. The chosen specification consists of the following sets:

- **Initial Set:** The set of states corresponding to $SOC \in [0, 0.2]$,
- **Goal Set:** The set of states corresponding to $SOC \in [0.8, 1]$,
- **Safe Set:** The set of states corresponding to:
  1. Side Reaction Overpotential lower-bounded, i.e., $\eta_2 \geq m$,
  2. Cell Voltage upper-bounded, i.e., $V \leq 4.2V$,
  3. Cell Temperature upper-bounded, i.e., $T \leq 35°C = 308K$.

The choice of the lower bound m is not specified since it can be viewed as a parameter for verification, as is later discussed. Furthermore, as the goal of the control design is a controller that maximizes charging speed and minimizes aging effects while maintaining the safety of the cell, the formal definition of the control specification is to ensure the RWA specification while reaching the goal set as quickly as possible.

Chapter 3

Data-Driven Symbolic Methods

Following the discussion on Li-Ion Battery Systems, their electrochemical modeling principles, and the state estimation and control methods for their aging-aware operation, this chapter consists of a more theoretically involved discussion on Symbolic Methods for Systems and Control. As the essential goals of this thesis are to study and develop formal verification-based control methods for the aging-aware operation of Li-Ion battery systems, it is of interest to construct a theoretical basis for this research area.

Symbolic or Formal Methods for Systems and Control constitute a field that has significantly developed in the last few decades, migrating from its early motivations in computer science for program verification to symbolic methods, which were successfully employed in the analysis of digital and cyber-physical systems, to general dynamical systems. Essentially, the study consists of viewing a dynamical system as a computer program to be verified for a certain logical specification. This has motivated the development of the notion of an abstraction, which is a finite-state expression of an infinite-state system through behavioral relations, further facilitating the development of verification methods and producing formal guarantees on system performance.

This chapter is then concerned with introducing the field of Symbolic Methods for Systems and Control and providing a discussion on the recent innovations in the field that have allowed for its enhancement with data-driven methods. This is centered around the notion of a data-driven abstraction, to be defined and discussed in this chapter as well.

3-1 Formal System Definitions

This section begins with the formal definition of a dynamical system:

Definition 2 (System [6]). *A system S is a sextuple $(X, X_0, U, \longrightarrow, Y, H)$ consisting of:*

- *X, a set of states;*
- *$X_0 \subseteq X$, a set of initial states;*

- *U, a set of inputs;*
- *$\longrightarrow \subseteq X \times U \times X$, a transition relation;*
- *Y, a set of outputs;*
- *$H : X \rightarrow Y$, an output map.*

A system is called finite-state if X is a finite set, whereas a system with an infinite set of states is called infinite-state. The evolution of a system is captured by the transition relation, described as $(x, u, x') \in \longrightarrow$, or $x \xrightarrow{u} x'$. This allows the definition of a successor set:

Definition 3 (Direct Successors Set, Non-Blocking Inputs [6])**.** *For $x \in X$, the set $Post_u(x)$ denotes the set of direct successors of x under u, i.e.,*

$$Post_u(x) := \{x' \in X | x \xrightarrow{u} x'\} \tag{3-1}$$

Furthermore, the set of non-blocking inputs for a given state x, $U(x)$, is defined as the set of inputs such that the direct successor set of x is non-empty, i.e.,

$$U(x) := \{u \in U | Post_u(x) \neq \varnothing\}. \tag{3-2}$$

Another useful definition is that of a deterministic system:

Definition 4 (Determinism [6])**.** *A system S is deterministic if, for any state $x \in X$ and input $u \in U$, the cardinality of the successor set $|Post_u(x)| = 1$.*

For a given state $x \in X$, the **finite internal behavior** generated by x is given by, the finite sequence of transitions:

$$x_0 \xrightarrow{u_0} x_1 \xrightarrow{u_1} x_1 \xrightarrow{u_2} \cdots \xrightarrow{u_{n-2}} x_{n-1} \xrightarrow{u_{n-1}} x_n, \tag{3-3}$$

such that $x_0 = x$. Analogously, the **infinite internal behavior** is defined by an infinite sequence of transitions:

$$x_0 \xrightarrow{u_0} x_1 \xrightarrow{u_1} x_2 \xrightarrow{u_2} x_3 \xrightarrow{u_3} \cdots . \tag{3-4}$$

Finite and infinite internal behaviors for any initial state x induce the definition of an **external behavior**, such as for the infinite case, given by

$$y_0 \longrightarrow y_1 \longrightarrow y_2 \longrightarrow y_3 \cdots , \tag{3-5}$$

such that $y_i = H(x_i)$ for $i \in \mathbb{N}_0$, i.e., the set of natural numbers including zero. More succinctly, the external behavior of the system can be written in string form as $y = y_0 y_1 y_2 \cdots$. The set of external behaviors that are defined by internal behaviors from x is denoted by $\mathcal{B}_x(S)$ and is called the **external behavior** from state x. The set of all infinite external behaviors generated from x is denoted by $\mathcal{B}^{\omega}_x(S)$ and called the **infinite external behavior** from x.

Definition 5 (Finite and Infinite External Behavior)**.** *The finite external behavior generated by a system S, denoted by $\mathcal{B}(S)$ is defined by*

$$\mathcal{B}(S) = \bigcup_{x \in X_0} \mathcal{B}_x(S) \tag{3-6}$$

Analogously, the infinite external behavior generated by a system S, denoted by $\mathcal{B}^\omega(S)$ is defined by

$$\mathcal{B}^\omega(S) = \bigcup_{x \in X_0} \mathcal{B}_x^\omega(S). \tag{3-7}$$

The definition of the behavior of a system allows for the definition of system relations, such as behavioral inclusion and equivalence:

Definition 6 (Behavioral Inclusion and Equivalence [6])**.** *Given two systems S_a and S_b with the same set of outputs, i.e., $Y_a = Y_b$, S_a is said to be behaviorally included in S_b, denoted by $S_a \preceq_{\mathcal{B}} S_b$, if $\mathcal{B}^\omega(S_a) \subseteq \mathcal{B}^\omega(S_b)$. Furthermore, the two systems are said to be behaviorally equivalent, denoted by $S_a \cong_{\mathcal{B}} S_b$, if it additionally holds that $S_b \preceq_{\mathcal{B}} S_a$.*

The concept of behavioral relations is useful when analyzing the reachability of systems. A state $x \in X$ is said to be **reachable** if there exists a finite internal behavior sequence for some $x_0 \in X_0$ such that $x_n = x$. An output $y \in Y$ is said to be reachable if there exists a reachable state $x \in X$ such that $H(x) = y$. The set $\mathit{Reach}(S)$ denotes the reachable set of a system S, i.e., the set of all its reachable outputs. This definition implies the following result on behavioral relations:

Proposition 1 ([6])**.** *For any two systems S_a and S_b, it holds that*

$$S_a \preceq_{\mathcal{B}} S_b \Rightarrow \mathit{Reach}(S_a) \subseteq \mathit{Reach}(S_b), \tag{3-8}$$

$$S_a \cong_{\mathcal{B}} S_b \Rightarrow \mathit{Reach}(S_a) = \mathit{Reach}(S_b). \tag{3-9}$$

The above result can be instrumental in performing safety analysis, i.e., by ensuring that a system does not reach a certain unsafe set. For example, if $S_a \preceq_{\mathcal{B}} S_b$ and it is known that $\mathit{Reach}(S_b) \cap U = \varnothing$, for some unsafe set, it follows that $\mathit{Reach}(S_a) \cap U = \varnothing$, thus implying that the system S_a does not reach the unsafe set.

3-2 Data-Driven Abstractions

In applications of formal verification, a common technique is to construct finite-state symbolic abstractions that describe and encapsulate the behavior of the concrete infinite-state system. Those abstractions, usually described and presented in the form of finite-state machines, are constructed to behaviorally include the dynamics of the concrete system. As a result, verifying a specification on the abstractions implies verifying the same property on the concrete system.

The construction of effective abstractions, however, is a substantially complex process, as the abstraction must be large enough to encapsulate the behavior of the concrete system, without being too generalizing to provide no significant information. Furthermore, it is typically difficult to prove the behavioral inclusion of the concrete system within the abstraction.

Thus, most abstraction-based methods face limitations that require the construction of an abstraction based on expert knowledge of the system or complex mathematical methods.

However, with the growth of data-driven modeling and analysis methods in recent years, chiefly due to the higher accessibility of advanced computational tools, several fields, including those of system verification and formal methods, have been involved in the development of data-driven methods. Most notably, methods to construct symbolic abstractions of infinite-state systems from output data have been developed. This section aims to document the work done in [7] to develop data-driven abstractions for autonomous systems, which can be used to perform verification on the concrete system. Consider the time-invariant autonomous system given by

$$\Sigma(x) := \begin{cases} x_{k+1} = f(x_k), \\ y_k = h(x_k), \\ x_0 = x, \end{cases} \tag{3-10}$$

with $x \in \mathcal{D}$, some domain defined on the set of states X, and $y_k \in \mathcal{Y}$, i.e., the set of outputs, which is defined to be finite. The mapping $h(\cdot)$ can be considered to be a partitioning map that returns a label, i.e., an element of the set $\mathcal{Y}$, corresponding to the state x_k. Furthermore, let $(\Delta, \mathcal{F}, \mathbb{P})$ denote a probability space, where Δ is the sample space endowed with a σ-algebra $\mathcal{F}$ and a probability measure $\mathbb{P}$. Building on the previous formal definitions of a dynamical system and behavioral inclusion, as well as the introduction of the probability space, one can define a probabilistic version of behavioral inclusion as follows:

Definition 7 (Probabilistic Behavioral Inclusion [7])**.** *Consider two dynamical systems S_a and S_b with $\mathcal{Y}_a = \mathcal{Y}_b$ and a probability space $(\mathcal{D}, \mathcal{F}(\mathcal{D}), \mathcal{P})$. The system S_a is said to be behaviorally included in S_b with probability greater than or equal to $1-\epsilon$, denoted by $\mathbb{P}[S_a \preceq_{\mathcal{B}} S_b] \geq 1-\epsilon$, if for an initial condition x_0 sampled from the probability space, i.e., $x_0 \sim \mathcal{P}$, it holds that*

$$\mathbb{P}[\mathcal{B}^{\omega}_{x_0}(S_a) \subseteq \mathcal{B}^{\omega}(S_b) | x_0 \sim \mathcal{P}] \geq 1-\epsilon. \tag{3-11}$$

It is said that S_a is almost surely behaviorally included in S_b if $\epsilon = 0$. If the probabilistic behavioral inclusion holds for some horizon H, it is denoted as $\mathbb{P}[S_a \preceq_{\mathcal{B}_H} S_b] \geq 1-\epsilon$.

Another notion of interest is that of a transition system, which is a specific case of the definition of a system introduced at the start of this chapter. The abstraction to be constructed is defined as a transition system, defined as follows:

Definition 8 (Transition System [7])**.** *A transition system $\mathcal{S}$ is a tuple $(\mathcal{X}, \mathcal{X}_0, \mathcal{E}, \mathcal{Y}, \mathcal{H})$, where*

- *$\mathcal{X}$ is the (possibly infinite) set of states;*
- *$\mathcal{X}_0 \subseteq \mathcal{X}$ is the set of initial states;*
- *$\mathcal{E} \subseteq \mathcal{X} \times \mathcal{X}$ is the set of edges, or transitions;*
- *$\mathcal{Y}$ is the set of outputs;*
- *$\mathcal{H} : \mathcal{X} \to \mathcal{Y}$ is the output map.*

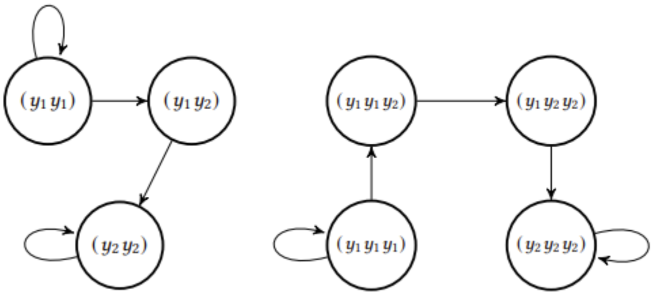


Figure 3-1: Example of SAℓ-CA with $\ell = 2$ (left) and $\ell = 3$ (right) [7]

The essential differentiation between a transition system and a system as defined at the beginning of this chapter is that transitions are defined over a set of edges in the former case, rather than by a transition map in the latter case. Furthermore, transition systems are assumed to not admit inputs, i.e., they are defined as autonomous systems. The behavior of a transition system can be defined in a manner analogous to the previously provided definition. The definition of a transition system thus allows the introduction of the ℓ-complete abstraction:

Definition 9 ((Strongest) Asynchronous ℓ-Complete Abstraction [7]). *Let $\mathcal{S} := (\mathcal{X}, \mathcal{X}_0, \mathcal{E}, \mathcal{Y}, \mathcal{H})$ denote a transition system, and denote by $\mathcal{X}_\ell \subseteq \mathcal{Y}^\ell$ the set of all $\ell -$ long subsequences of all behaviors in $\mathcal{S}$, for some $\ell \in \mathbb{N}$. The system $\mathcal{S}_\ell := (\mathcal{X}, \mathcal{B}_\ell(\mathcal{S}), \mathcal{E}_\ell, \mathcal{Y}^\ell, \mathcal{H})$ is called the (strongest) asynchronous ℓ-complete abstraction (SAℓ-CA) of $\mathcal{S}$, where*

- $\mathcal{E}_\ell = \{(k\sigma, \sigma k') | k, k' \in \mathcal{Y}, \sigma \in \mathcal{Y}^{\ell-1}, k\sigma, \sigma k' \in \mathcal{X}_\ell\}$
- $\mathcal{H}(k\sigma) = k$,

with $\mathcal{B}_\ell(\mathcal{S})$ denoting the set of all external traces of $\mathcal{S}_\ell$, which are of length ℓ, and $\mathcal{Y}^\ell$ denoting the Cartesian product $\mathcal{Y} \times \cdots \times \mathcal{Y}$ repeated ℓ times.

In short, the SAℓ-CA encodes each state of a transition system as a sequence of length ℓ, which follow the domino rule, i.e., the last $\ell - 1$ entries of a state sequence are the first $\ell - 1$ entries of the following state sequence. This is exemplified in Figure 3-1. The main motivation for employing the SAℓ-CA is that it can circumvent issues that result from making conclusions about one-step transitions of a system, thus providing an improved verification process. Furthermore, the refinement of the abstraction can be easily modified by changing the parameter ℓ, representing the length of each state sequence.

Consider the data collection process, where N sequences of length H are collected from randomly-sampled initial conditions to construct the set $\mathcal{X}_N^\ell$, which denotes the set of observed ℓ-sequences, and acts as the state set of the data-driven SAℓ-CA, defined by:

Definition 10 (Data-Driven SAℓ-CA [7]). *The ℓ-complete abstraction $\mathcal{S}_\ell^N = (\mathcal{X}_\ell^N, \mathcal{X}_\ell^N, \mathcal{E}_\ell, \mathcal{Y}^\ell, \mathcal{H})$ is called the data-driven ℓ-complete abstraction of $\mathcal{S}$, where*

- *$\mathcal{X}_\ell^N$ is the state space built from the ℓ-sequences collected from N trajectories of an underlying concrete system.*

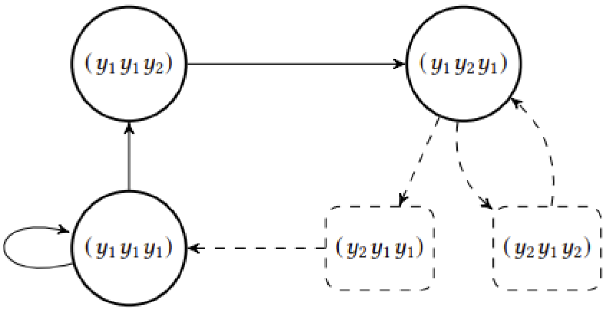


Figure 3-2: Example of the Construction of a Non-Blocking Automaton from Domino Completion. Dashed Lines Indicate Artificially-Added States [7].

When constructing the data-driven SAℓ-CA, it is possible to result in a blocking automaton, as the domino rule may not be satisfied. For this reason, a manual extension of the data-driven state space set is implemented, by including the states that render the state space compatible with the domino rule. This notion of Domino Completion is illustrated in Figure 3-2. Though this may lead to spurious behaviors in the abstractions, i.e., behaviors exhibited by the abstraction but not the system it is constructed from, this process is necessary to ensure the construction of a non-blocking automaton.

Assuming hereafter that the data-driven SAℓ-CA is a non-blocking automaton, scenario theory can be leveraged to provide bounds on the probabilistic behavioral inclusion of the concrete system $\mathcal{S}$ within the data-driven abstraction $\mathcal{S}_\ell^N$. Scenario theory, or the scenario approach, is a methodology for data-driven optimization based on making decisions from previously observed information and providing probabilistic guarantees on the optimality of the decision for new and unobserved cases, or scenarios. It is rigorously introduced in [31] but is only briefly discussed in this work.

Consider N sampled i.i.d initial conditions $\{x_{0,i}\}_{i=1}^N$ and the resulting H-long behaviors displayed by $\mathcal{S}$, denoted by $\{\mathcal{B}_H(x_{0,i})\}_{i=1}^N$. It is then directly possible to obtain N i.i.d scenarios by the binary vectors $\{\delta_i\}_{i=1}^N$, where

$$\delta_i(j) = \begin{cases} 1 & \text{if } \mathcal{B}_H(x_{0,i}) \models \Diamond y_{\ell_j}, \\ 0 & \text{else,} \end{cases} \tag{3-12}$$

for $j \in \{1, \cdots, |\mathcal{Y}|^\ell\}$. In short, each $\delta_i(\cdot)$ expresses whether the sampled behavior $\mathcal{B}(x_{0,i})$ at some point exhibits the ℓ-sequence y_{ℓ_j}, denoted by the expression $\mathcal{B}_H(x_{0,i}) \models \Diamond y_{\ell_j}$, where the set of all possible ℓ-sequences is encoded over the set $\{1, \cdots, |\mathcal{Y}|^\ell\}$. For $\Theta = \mathbb{R}^{|\mathcal{Y}|^\ell}$, the scenario program can be formally defined as

$$\min_{\theta \in \Theta} \quad \mathbf{1}^T_{|\mathcal{Y}^\ell|} \cdot \theta, \tag{3-13}$$

$$s.t. \quad (\theta - \delta_i) \geq 0, \quad i = 1, \cdots, N. \tag{3-14}$$

The solution θ_N^* is trivially unique and indicates which ℓ-sequences were witnessed in the collected samples. Thus, the above scenario problem can be interpreted as the collection of labels from a discrete probability distribution of unknown support size. Scenario theory can be leveraged to provide bounds on the probability of collecting a new and unseen label from the unknown distribution. Thus, the data-driven SAℓ-CA can be equipped with guarantees as follows:

Proposition 2 ([7])**.** *Consider a confidence β and N trajectories of length H collected from the concrete system, and the corresponding data-driven SAℓ-CA denoted by $\mathcal{S}_\ell^N$ based on the observed ℓ-sequences. For a new initial condition sampled from $\mathcal{D}$ with distribution $\mathcal{P}$, it holds that*

$$\mathbb{P}^N\Big[\mathbb{P}[\mathcal{B}_H(\mathcal{S}(x_0)) \in \mathcal{B}_H(\mathcal{S}_\ell^N)] \geq 1 - \epsilon(s_N^*, N, \beta)\Big] \geq 1 - \beta, \tag{3-15}$$

where $\mathcal{B}_H(\mathcal{S}(x_0))$ denotes the H-long behavior exhibited by the (not necessarily transition) system $\mathcal{S}$ starting from x_0, $\mathcal{B}_H(\mathcal{S}_\ell^N)$ the set of all H-long behaviors of $\mathcal{S}_\ell^N$, and s_N^* the number of ℓ-sequences witnessed during the data collection and construction of the data-driven SAℓ-CA, also referred to as the complexity of the decision.

Subsequently, further discussion on the computation of $\epsilon(s_N^*, N, \beta)$ is provided based on the following result:

Proposition 3 ([32])**.** *Consider the decision map $M_m : \Delta^m \to \mathcal{Z}$, $m = 0, 1, \cdots$, mapping from the set of observed scenarios to the set of decisions, satisfying the consistency assumption in [32]. Given a confidence parameter $\beta \in (0, 1)$, for any $k = 0, 1, \cdots, N-1$, where N denotes the number of observed scenarios, consider the polynomial equation in the v variable*

$$\binom{N}{k}(1-v)^{N-k} - \frac{\beta}{N}\sum_{m=k}^{N-1}\binom{m}{k}(1-v)^{m-k} = 0, \tag{3-16}$$

and let $\epsilon(k)$ be the unique solution over the interval $(0, 1)$. Also, define $\epsilon(N) = 1$. For any probability measure $\mathbb{P}$, it holds that

$$\mathbb{P}^N[V(z_N^*) > \epsilon(s_N^*)] \leq \beta, \tag{3-17}$$

where $z_N^* = M_N(\delta_1, \cdots, \delta_N)$ denotes the decision made based on the observed scenarios, $V(z_N^*)$ denotes the probability of violation of the decision z_N^*, i.e., that the decision z_N^* is unsuitable, and s_N^* denotes the complexity of the decision.

The above result, which is for a general case of employing the scenario approach, can be interpreted in the context of constructing a data-driven abstraction of a closed-loop system based on output observations, providing bounds on the probability of behavioral inclusion between the concrete system and the corresponding abstraction.

3-2-1 Parameter Uncertainty

Furthermore, data-driven abstractions can be leveraged to verify systems that have an uncertain set of parameters, which can thus provide a robustness guarantee on system performance. Consider a vector of uncertain parameters $p \in \Delta_p$, which denotes the parametric uncertainty set. Thus, the uncertain system can be expressed as

$$\Sigma_p(x) := \begin{cases} x_{k+1} = f_p(x_k), \\ y_k = h(x_k), \\ x_0 = x, p \in \Delta_p, \end{cases} \tag{3-18}$$

where $f_p(x_k)$ denotes a function parametrized by p. Since the parameter vector is static, the uncertain system $\Sigma_p(x)$ can, by augmenting the state vector x with p, be reformulated as

$$\Sigma(x,p) := \begin{cases} \begin{bmatrix} x_{k+1} \\ p_{k+1} \end{bmatrix} = \begin{bmatrix} f(x_k, p_k) \\ p_k \end{bmatrix}, \\ y_k = h(x_k), \\ x_0 = x, p_0 = p, \end{cases} \tag{3-19}$$

with $x \in \mathcal{D}$ and $p \in \Delta_p$, and $f(x_k, p_k) = f_{p_k}(x_k)$, resulting in the same general formulation of a time-invariant autonomous system given in Equation 3-10. Similar to above, the initial states are sampled from a domain $\mathcal{D} \times \Delta_p$, which renders the new uncertain system valid for all the analysis of data-driven abstractions provided above. Thus, it is possible to use the data-driven abstraction framework to verify specifications on systems with parameter uncertainties, which allows for a robustness analysis of a designed controller as well.

3-3 Counterexample-Guided Inductive Synthesis (CEGIS)

Verification methods applied to abstractions of concrete systems can be used to provide formal guarantees of the satisfaction of certain specifications, as discussed in the previous section. However, the verification process itself cannot be employed to synthesize a solution to satisfy the specification. From a control perspective, that is to say, formal verification applied to an abstraction, whether data-driven or not, of a closed-loop, and thus autonomous, concrete system, can only be used to check whether the closed-loop system meets the desired specification, and not to generate the controller.

Despite the inability of a verification protocol to be directly used in synthesizing controllers for the system, information from verification can be leveraged to improve the controller design. Namely, if the verification protocol shows an example initial condition from which the specification is not met, i.e., the verification generates a counterexample, this counterexample can be employed in the design of the controller. Such a method, employing counterexamples to improve an initial design choice, is denoted by Counterexample-Guided Inductive Synthesis (CEGIS).

CEGIS can be considered in applications beyond controller design, such as program synthesis. In [33], the learning loop illustrated in Figure 3-3 is applied to produce a function that generates a desired logic specification, as illustrated by the example of the $\max(\cdot,\cdot)$ function. The learning loop consists of providing an initial function, such as the identity map, which is checked by the Verification Oracle. The Verification Oracle produces a counterexample, i.e., a case where the proposed function fails, and the learning algorithm modifies the function to eliminate the counterexample. The modified function is then checked by the Verification Oracle for more counterexamples, resulting in a learning loop. If no more counterexamples are found by the Verification Oracle, the learning is deemed successful and the developed function thus meets the desired specifications.

Another example of CEGIS is that in [34], where, for a given closed-loop system with known dynamics, it is desired to synthesize a valid Control Lyapunov Function (CLF) to verify the stability of the closed-loop system. The learning framework is similar to that illustrated in

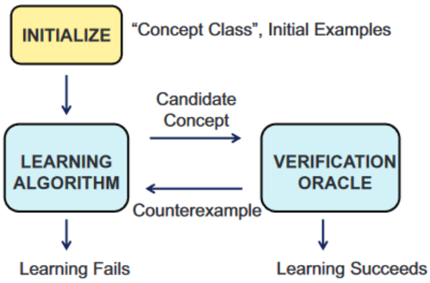


Figure 3-3: Learning Loop for Counterexample-Guided Synthesis [33]

Figure 3-3, as a CLF is constructed from the linear combination of basis functions, and a candidate CLF is constructed by choosing the weights of each basis function such that the CLF conditions are met for a certain finite set of points in the state space. For this constructed CLF, a verifier then checks whether the CLF conditions hold for all points in the state space. If that is not the case, the verifier generates a counterexample, which is added to the finite set of points that are used to construct the candidate CLF. Thus, CEGIS can be employed in the analysis of the stability of closed-loop systems.

The flexibility in employing CEGIS, however, is because it is ignorant of the learning algorithm used to improve the synthesis based on counterexamples. Thus, it remains a challenge in most learning applications that employ a Counterexample-Guided Synthesis framework to design an adequate and effective learning algorithm to converge to the desired specification, if it can be verified at all. As a result, the use of Counterexample-Guided Synthesis is case-specific and not a generally applicable method.

Chapter 4

Proposed Framework

Based on the research questions discussed in the Introduction, the goal of this thesis is to eventually construct a charging protocol for a single-cell Lithium-Ion Battery System that satisfies the given control specifications, namely the Reach-While-Avoid specification that enforces a bound on the side reaction overpotential, thus a bound on the rate of aging, as well as a safety bound on the cell voltage and temperature. An adjacent goal is to construct a verification framework that is capable of providing probabilistic guarantees on the satisfaction of the given specification. This can be done using the data-driven abstraction concept introduced in the previous chapter.

In this chapter, the proposed framework for the construction of this controller and verification tool is discussed. Throughout, the implementation of a data-driven approach based on Reinforcement Learning and data-driven abstractions is determined to be preferred over model-based methods. Furthermore, the use of Counterexample-Guided Inductive Synthesis (CEGIS) for the further improvement of the controller design is motivated and described, thus proposing a general data-driven framework for controller synthesis and verification.

4-1 Data-Driven Approach

The data-driven approach proposed in this work consists of informing the controller design with observation data from a large number of battery simulations. That is, with the use of simulation results from the DFN Model, a controller for a real Li-Ion cell is to be constructed. This is motivated by the use of Reinforcement Learning to design a charging protocol in Output Feedback in [21]. In this work, however, some modifications are made to the learning scheme to make it more suitable for the work of this thesis. These modifications include:

- Incorporating aging dynamics to the model, i.e., adding the side reaction flux j_2 to the dynamics to better encapsulate aging dynamics in the learning.
- Defining a region of parameter uncertainty, where every episode initializes a battery with parameters randomly sampled from the region, for improved robustness.

- Modifying the reward function to depend on the side reaction overpotential η_2, i.e., penalizing aging behavior, despite the learning agent not directly measuring the variable as it operates in Output Feedback.

The learning framework developed in [21] is thus modified and adopted as the essential tool for control synthesis. This data-driven scheme is advantageous as it circumvents complex dynamics and parameter uncertainty, which pose considerable challenges for model-based techniques such as Model Predictive Control. Furthermore, the lack of observability of the internal states for the controller can be detrimental to the adoption of a model-based approach, whereas RL can still converge to an adequate controller.

4-2 Verification and CEGIS Framework

In addition to the data-driven approach to constructing a controller, the method by which the developed charging scheme is verified to satisfy the given specifications is based on data-driven abstractions. That is, once a charging protocol is produced from the Reinforcement Learning algorithm, it is used to construct the closed-loop system from which random initial conditions are sampled, resulting in numerous traces of **Voltage, Temperature, $\boldsymbol{\eta_2}$, and SOC**. Those traces are also collected from systems with randomly generated parameters, which allows for the robustness analysis of the charging protocol. The continuous-space traces are processed into discrete, or symbolic, traces by partitioning the 4-dimensional output space, and furthermore into the SAℓ-CA introduced in the previous chapter. Using the scenario approach, the performance of the controller can be verified under the described probability bounds. The value of those probability bounds depends on the choice of the partitions, the number of collected traces, as well as the choice of the parameter ℓ.

Verification is performed using the **backwards reachability** algorithm, which is described by Algorithm 1. The algorithm consists of finding the set of states $Pre(s)$ in a finite-state transition system that can reach a given state s. This serves to describe the possible behaviors that the abstraction can exhibit, and thus the concrete system, up to a certain probabilistic guarantee. The backwards reachability algorithm is capable of verifying that the Goal set is reached by all initial conditions by exhibiting the following two conditions:

- The only self-loops in the abstraction are the ones that associate with the Goal set, i.e., $SOC \geq 0.8$, implying that there are no other self-loops in the abstraction. This can be checked by choosing a large enough ℓ and a considerably diverse partition of the output space.
- The union of pre-sets $Pre(s)$ for all states s that associate with the Goal set is equal to the set of all states in the abstraction. This verifies that there is a path from any state to one in the Goal set.

As previously discussed, verification by nature constitutes the last step of controller design and is not directly capable of modifying the controller. However, in this work, an approach to verification-based control synthesis is proposed by means of a Counterexample-Guided Inductive Synthesis (CEGIS) framework. The framework consists of first developing an initial

Algorithm 1 Backwards Reachability Algorithm for SAℓ-CA

```
Input: State $s = (s_1, ..., s_\ell)$, set of states $\mathcal{S}$
Output: Set $Pre(s)$
  $Pre(s) \leftarrow \{s' = (s'_1, ..., s'_\ell) \in \mathcal{S} | (s'_2, ...s'_\ell) = (s_1, ..., s_{\ell-1})\}$
  $done \leftarrow \{s\}$
  $stop \leftarrow False$
  while $stop = False$ do
      for $s' \in Pre(s) - done$ do
          $Pre(s) \leftarrow Pre(s) \cup \{s^* = (s^*_1, ..., s^*_\ell) \in \mathcal{S} | (s^*_2, ...s^*_\ell) = (s'_1, ..., s'_{\ell-1})\}$
          $done \leftarrow done \cup \{s'\}$
      end for
      if $Pre(s) \cup \{s\} = done$ then
          $stop \leftarrow True$
      end if
  end while
```

output feedback controller through Reinforcement Learning, followed by constructing a data-driven abstraction that behaviorally includes the resulting closed-loop system up to a certain probability bound. If the data-driven abstraction satisfies the desired specification, then the controller is verified up to the obtained probability bound. However, if the safety specification is not met, the collected continuous-space trajectories can be used to produce the set of initial conditions for which the safety specification is violated. These initial conditions, viewed as counterexamples, are extracted and used to train a new controller that is specialized for that specific region of initial conditions. This new controller is employed to construct a switched controller, where the new controller is applied for the counterexample initial conditions, and the initial controller remains in operation for the rest of the initial conditions. This process is repeated until either the specification is satisfied or convergence is reached. The CEGIS-based synthesis scheme described above is illustrated in Figure 4-1.

The use of CEGIS in the case of designing an aging-aware charging protocol for a Li-Ion Battery cell is motivated by the fact that the chosen controller is in output feedback, which renders it unaware of internal battery states, and unable to differentiate between initialization and later stages of charging. This is shown to lead to several violations of the voltage and temperature upper bounds in the subsequent chapter. Thus, designing a series of charging protocols each suitable for a region of initial conditions would serve to improve the safety and performance of the controller.

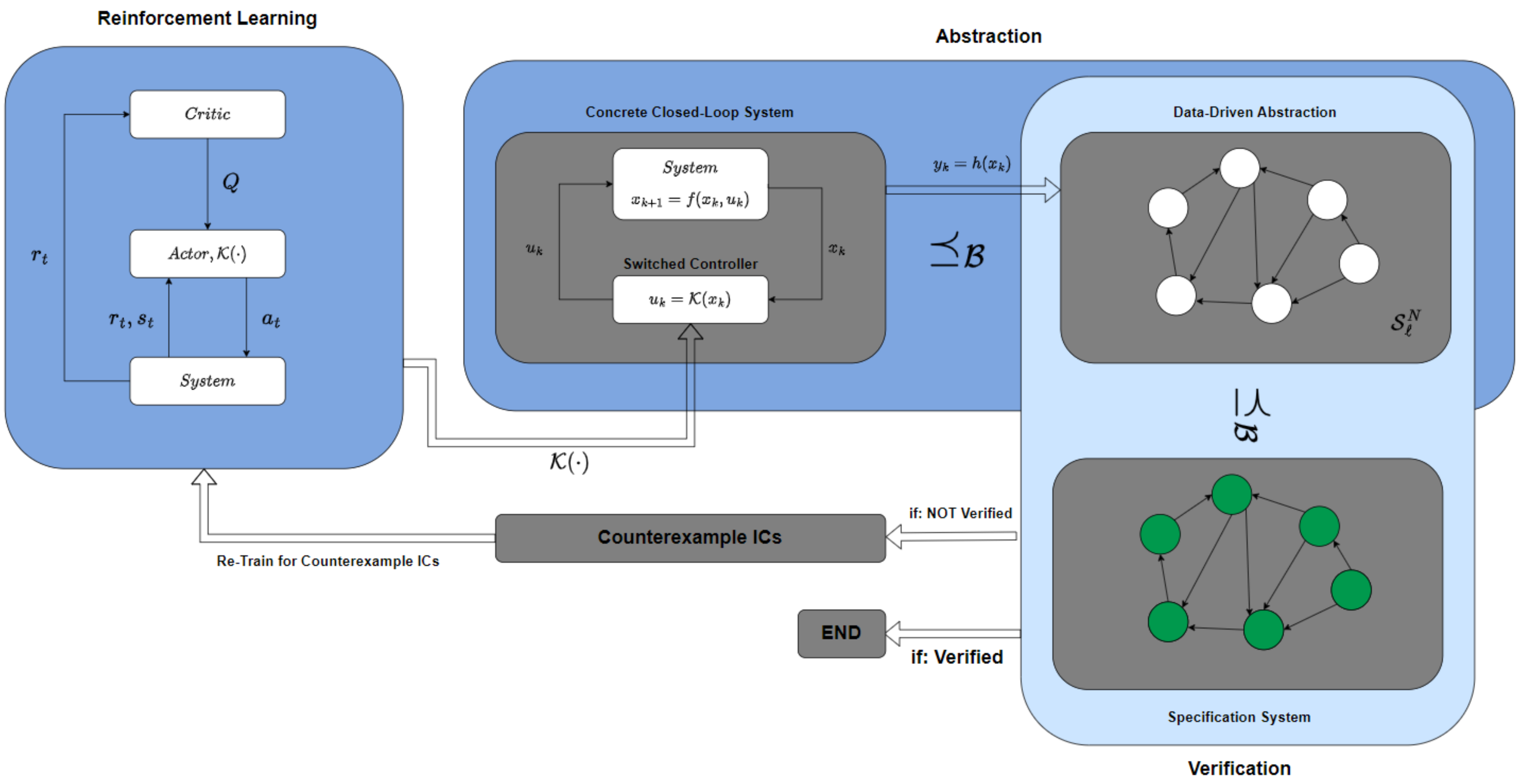


Figure 4-1: CEGIS-Based Control Synthesis Scheme

Chapter 5

Results

This chapter presents and discusses the results from the proposed control synthesis framework applied to the aging-aware charging of a Li-Ion Battery cell. Initially, the chosen battery model from industrial applications and its properties are introduced. This is followed by the introduction of the baseline Constant-Current-Constant-Voltage (CC-CV) charging protocol to which the final charging protocol is to be compared. Subsequently, the procedure for the collection of traces, partition of the output space, and the construction of the abstraction is documented. This is then followed by the control synthesis process, and finally the evaluation and verification of the conclusive charging protocol, representing the primary result of this work.

5-1 Battery Model and Baseline Protocol

The selected battery model is based on the Panasonic NCR18650GA Lithium-Ion Battery [35], which is a commonly used Lithium-Cobalt cell in industrial applications. An illustration of such a battery is given in Figure 5-1. The selected battery consists of a capacity of roughly $3400mAh$ and a charging voltage of $4.2V$. Thus, the model parameters are modified to result in the above capacity, while the control goals are meant to maintain the cell voltage at $4.2V$. Furthermore, all battery parameters and their uncertainty ranges, if applicable, are presented in Appendix B.

In most industrial applications, the applied charging protocol is commonly the rule-based Constant-Current-Constant-Voltage (CC-CV) protocol. This charging protocol is based on issuing a constant maximum charging I_{max} current at the first stage, denoted by the CC stage, until the cell voltage reaches a maximum voltage V_{max}. In this stage, denoted by the CV stage, the cell voltage is held constant at V_{max}, while the current decreases in an exponential manner until either the battery is fully charged or a minimum charging current is reached. In reality, the CV phase of the protocol is not trivial to maintain, as the current profile that maintains a constant voltage is not directly known and may vary among different batteries. Some cases ensure that the voltage remains at the maximum value by embedding voltage regulators into

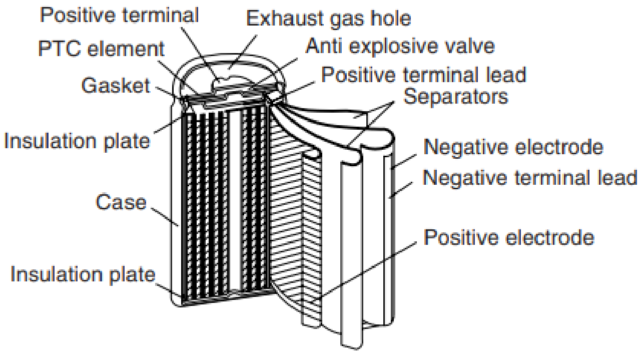


Figure 5-1: Schematic of the Panasonic NCR18650GA Lithium-Ion Battery Cell[35]

the circuitry of the Battery Management System, which is prone to high temperature and energy loss. Another possibility is to approximate the CC-CV profile by employing a switched controller approach. In such a switched controller, the CC phase provides a constant current input, while the CV phase is administered by a Proportional-Integral (PI)-controller. This switched controller approach is taken, and a modified PI-controller is tuned to regulate the voltage.

An example of the switched controller in simulation is shown in Figure 5-2, where the typical behavior of the SOC and η_2 can be noted. It is further pointed that the temperature constraint comes close to violation, as the standard CC-CV scheme does not account for thermal behavior, which is subsequently shown to be detrimental to the CC-CV protocol's satisfaction of the desired control specifications. Naturally, this charging protocol is also prone to exceeding the $4.2V$ limit due to overshoot in the PI-controller's performance. This is further discussed in subsequent sections, and compared to the final charging protocol.

Throughout this chapter, the maximum current is chosen to be $2.5C$, where $1C$ is equivalent to the current needed to fully charge the battery in one hour. In this case, $1C$ is equivalent to $3.4A$, and so the maximum current of $2.5C$ equates to $8.5A$. The switched controller corresponding to the CC-CV protocol is described in Appendix B.

5-2 Abstraction Construction

This section aims to describe the approach to and process of abstraction construction, providing an example construction of this abstraction to the closed-loop system resulting from the designed CC-CV protocol baseline.

The first two steps of constructing the abstraction consist of generating the traces of the resulting closed-loop system. First, the initial conditions of Voltage and Temperature are sampled from a uniform distribution over a two-dimensional region such that $(V_0, T_0) \in [2.7, 4.1] \times [290, 305]$, and the uncertain parameters are sampled from uniform distributions based on the regions specified in Appendix B. Second, for the sampled variables, the closed-loop system is simulated and the traces (SOC, η_2, V, T) are collected. This process is repeated N times, where N is expected to be large enough to result in a significant ε-bound.

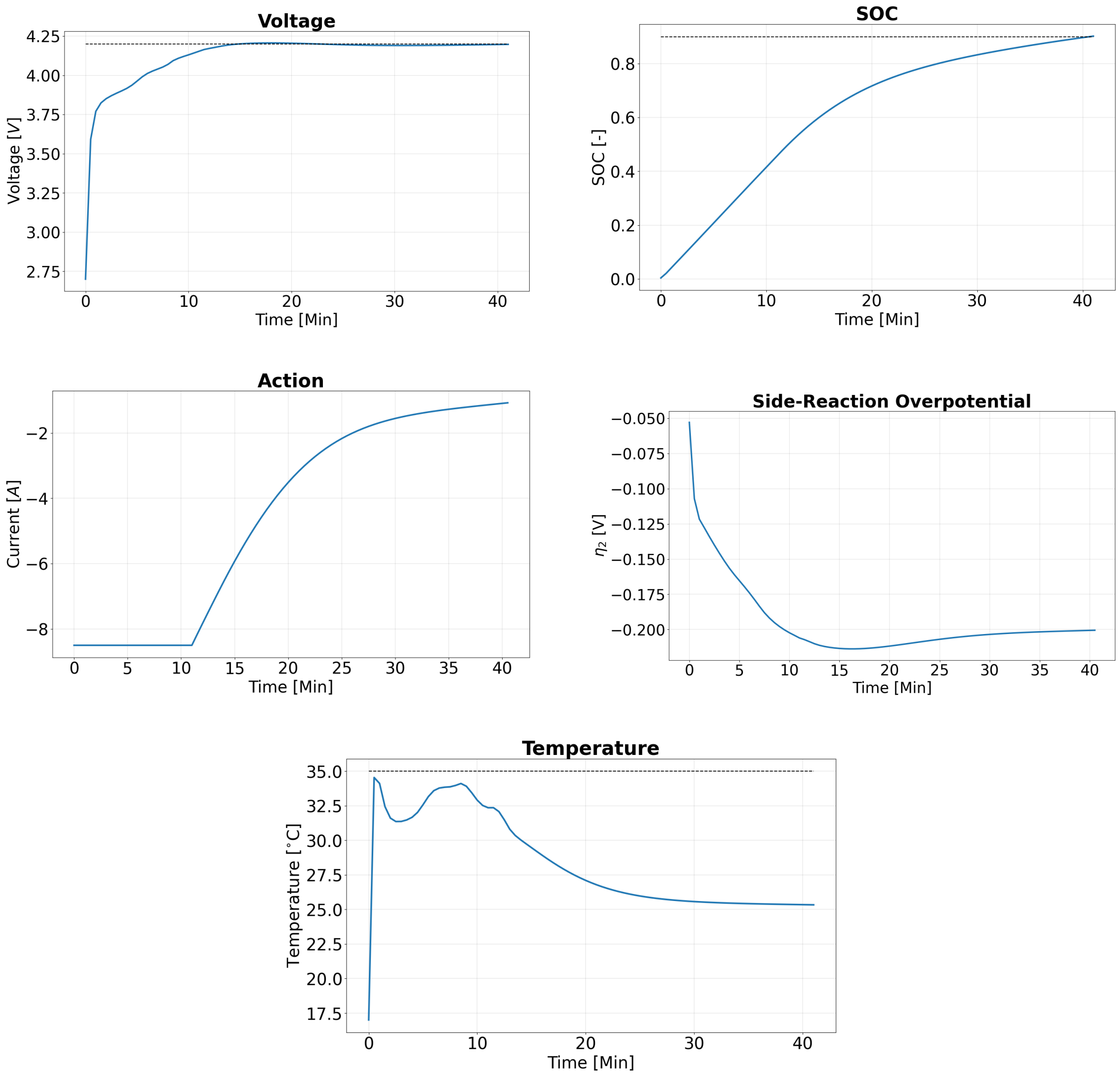


Figure 5-2: Simulation of CC-CV Charging Protocol

The next steps consist of processing the abstraction from the data. Third, a direct partition is applied on the collected traces, resulting in symbolic expressions for the states. The partition is defined as follows:

- The $[0, 1]$ interval describing the SOC is finely partitioned into 19 sections. This is done to avoid self-loops in the abstraction and is pivotal to verifying the reachability of the Goal set, which consists of the region $[0.8, 1]$. The partitions are each symbolized by a letter of the alphabet, i.e., $a, b, c, ..., s$.

- The rest of the variables are partitioned based on their atomic proposition, i.e., the symbol a for safety, such as $V \leq 4.2$, $T \leq 308K$, and $\eta_2 \geq m$. The violation of those constraints results in a symbol b.

Thus, each discrete state is a collection of 4 symbols. For example, the state where the SOC is in the range $[0.8, 1]$, and all other variables satisfy the safety conditions, the resulting symbol is given by $saaa$, whereas a violation of the safety constraint for η_2 corresponds to $sbaa$.

Fourth, the discrete expressions of the traces are compiled into ℓ-sequences, i.e., sequences of length ℓ, which is a parameter chosen large enough to avoid self-loops except in states associated with $SOC \in [0.8, 1]$ but small enough to maintain a small number of observed states and thus a more significant ε-bound. At that point, the SAℓ-CA is constructed and domino completion is performed to avoid the construction of a blocking automaton.

Fifth, the ε-bound for the probabilistic behavioral inclusion of the concrete system in the constructed SAℓ-CA is computed based on Propositions 2 and 3. Throughout this work, the certainty factor β is chosen to be 10^{-6}.

Sixth, the abstraction is used to perform verification. Backwards reachability and the lack of self-loops along the path are used to confirm that all initial conditions lead to the Goal set. Furthermore, the abstraction verifies that the unsafe set is not reached.

This method is initially applied in this section to verify the performance of the closed-loop system resulting from the CC-CV protocol, and subsequently for the other developed charging protocols. In this application, $N = 21705$ traces of length $H = 120$ are collected, corresponding to one hour of simulation. The minimum resulting side-reaction overpotential throughout all traces is found to be $\eta_{2,min} = -0.2452V$, which is chosen as the parameter m for the lower bound on η_2 when discretizing the traces. The collected traces are shown in the plots of Figure 5-3, where it can already be observed that the performance of the designed CC-CV protocol often violates safety constraints related to voltage and temperature, denoted by the red colored regions, whereas the goal set is colored in green.

The corresponding abstraction is constructed verify the ε-bound and the reachability of the goal set. The construction is based on the procedure described above. For illustration, the data-driven SAℓ-CA is shown for $\ell = 2$ in Figure 5-4. In this simple case, several undesired self-loops and spurious behaviors, i.e., behaviors exhibited by the abstraction but not by the collected traces, are exhibited. Despite that, some of the desired behavior is still observed, such as transitioning from the first SOC region to the next. As the ℓ parameter is increased, spurious behaviors are reduced at the cost of a more complex abstraction and an exponentially larger number of states.

After tuning for the smallest possible ℓ to avoid self-loops, a value $\ell = 12$ is reached, corresponding to 23146 abstraction states, and an ε-bound of 0.1563, i.e.,

$$\mathbb{P}^N\Big[\mathbb{P}[\mathcal{B}_H(\mathcal{S}(x_0)) \in \mathcal{B}_H(\mathcal{S}_\ell^N)] \geq 1 - 0.1563\Big] \geq 1 - 10^{-6}. \tag{5-1}$$

For this abstraction, backwards reachability shows that the goal set is reachable from all initial conditions, and as the large enough ℓ prevents self-loops or clusters in the abstraction, the reachability of the goal set is proved with the above probability expression. Though this ε-bound is not highly significant, it is still known that the CC-CV protocol is prone to several safety violations. Thus, the performance of the baseline protocol can be significantly improved with the proposed framework.

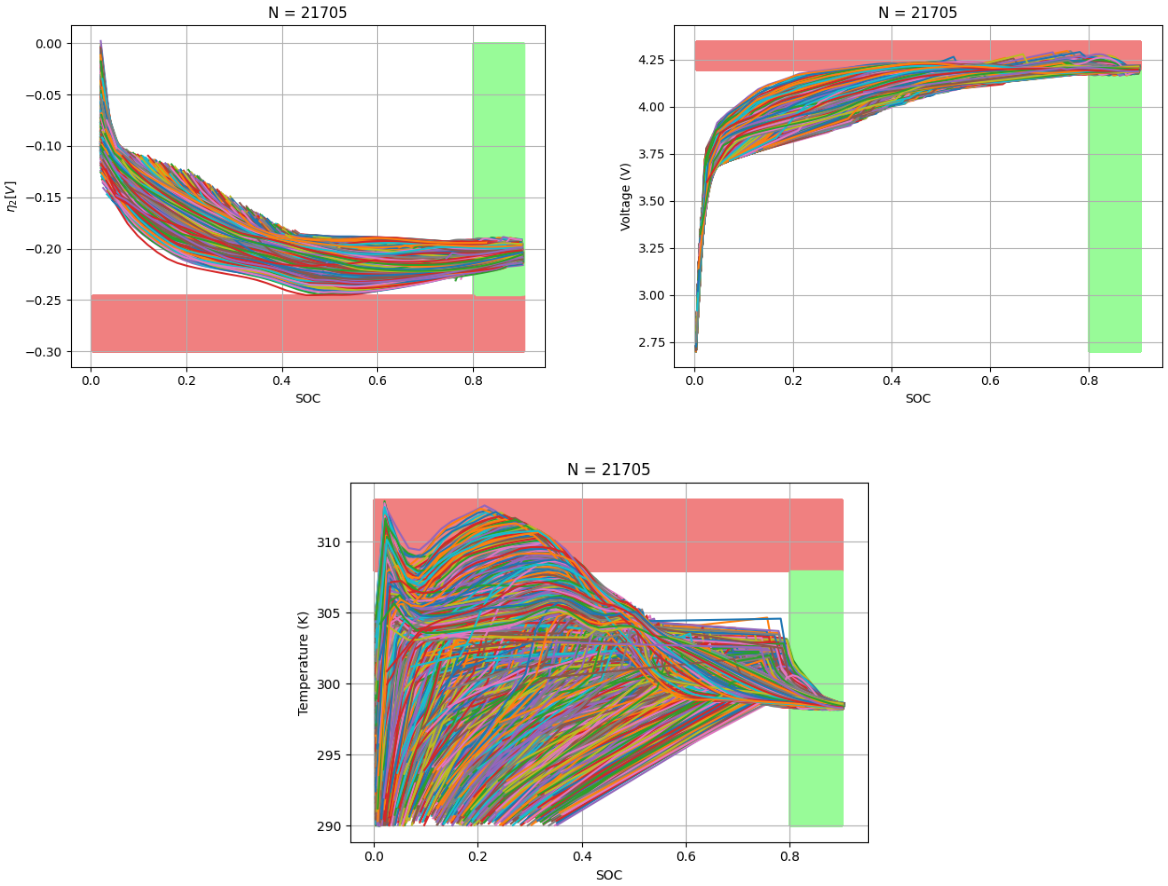


Figure 5-3: Collected Traces for the CC-CV Protocol

5-3 Learning Results and Counterexamples

The RL scheme based on the DDPG algorithm is implemented to construct the numerous charging protocols involved in this work. The reward function is chosen such that it penalizes slow charging, low side reaction overpotential, and voltage and temperature above the allowed limit. Though the controller operates in Output Feedback, i.e., only measuring the SOC, Voltage, and Temperature, it is trained to develop an aging-aware strategy based on the reward function. The reward scheme is given explicitly by

$$r_{t+1} = r_{fast} + r_{safety}(s_t, a_t), \tag{5-2}$$

where $r_{fast} = -0.1$ represents an instantaneous penalty for each time step until the training episode is stopped when the SOC reaches a value of 0.9, s_t and a_t denote the current state and action of the agent, respectively. The second term, r_{safety} is given by

$$r_{safety}(s_t, a_t) = r_{\eta_2}(s_t, a_t) + r_{volt}(s_t, a_t) + r_{temp}(s_t, a_t), \tag{5-3}$$

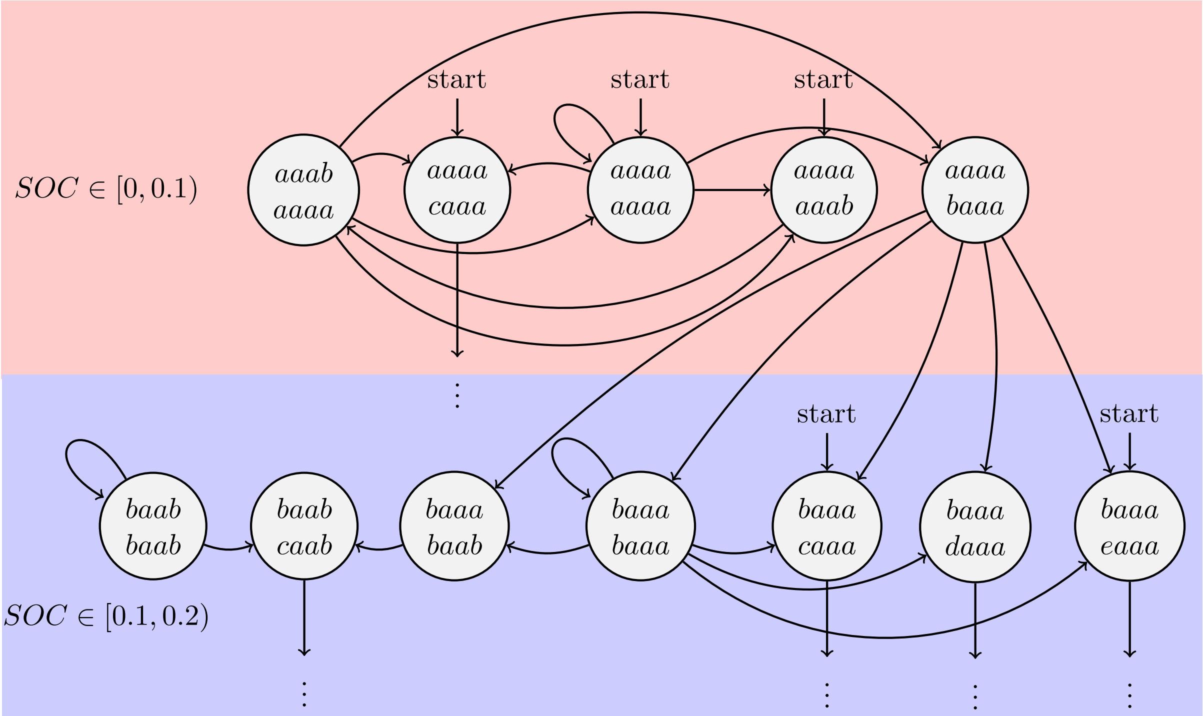


Figure 5-4: Schematic of the Data-Driven SAℓ-CA for the CC-CV Protocol ($\ell = 2$)

where

$$r_{\eta_2}(s_t, a_t) = \begin{cases} \lambda_{\eta_2}\eta_2(\delta_n, t), & \eta_2(\delta_n, t) < 0, \\ 0, & else, \end{cases} \tag{5-4}$$

$$r_{volt}(s_t, a_t) = \begin{cases} \lambda_{volt}(4.2 - V(t)), & V(t) \geq 4.2V, \\ 0 & else, \end{cases} \tag{5-5}$$

$$r_{temp}(s_t, a_t) = \begin{cases} \lambda_{temp}(308 - T(t)), & T(t) \geq 308K, \\ 0 & else, \end{cases} \tag{5-6}$$

with weighting parameters chosen as $\lambda_{\eta_2} = 1$, $\lambda_{volt} = 100$, and $\lambda_{temp} = 5$, thus penalizing undesired behavior for the above three variables. The reward corresponding to the side reaction overpotential is taken at the end of the negative electrode, as that is typically where the side reaction overpotential is at its lowest, and thus maximizing $\eta_2(\delta_n, t)$ is effectively maximizing the whole $\eta_2(t, x)$ profile.

5-3-1 First RL Run

The learning algorithm, along with all subsequent ones, is run using the DelftBlue supercomputer [36], with a randomly generated initial policy for 3000 episodes of training. The progression of the learning algorithm is shown in the plots of Figure 5-5, where the cumulative reward, charging time to 90%, and degrees of voltage and temperature constraint violations are described over the episodes. It is observed that the learning results are highly varying at first as the agent attempts to learn the optimal policy, and eventually converge to a policy with somewhat adequate behavior. The resulting output-based policy is illustrated in Figure 5-6, and is observed to not be significantly variable over temperature, which is subsequently shown to be detrimental for some initial conditions.

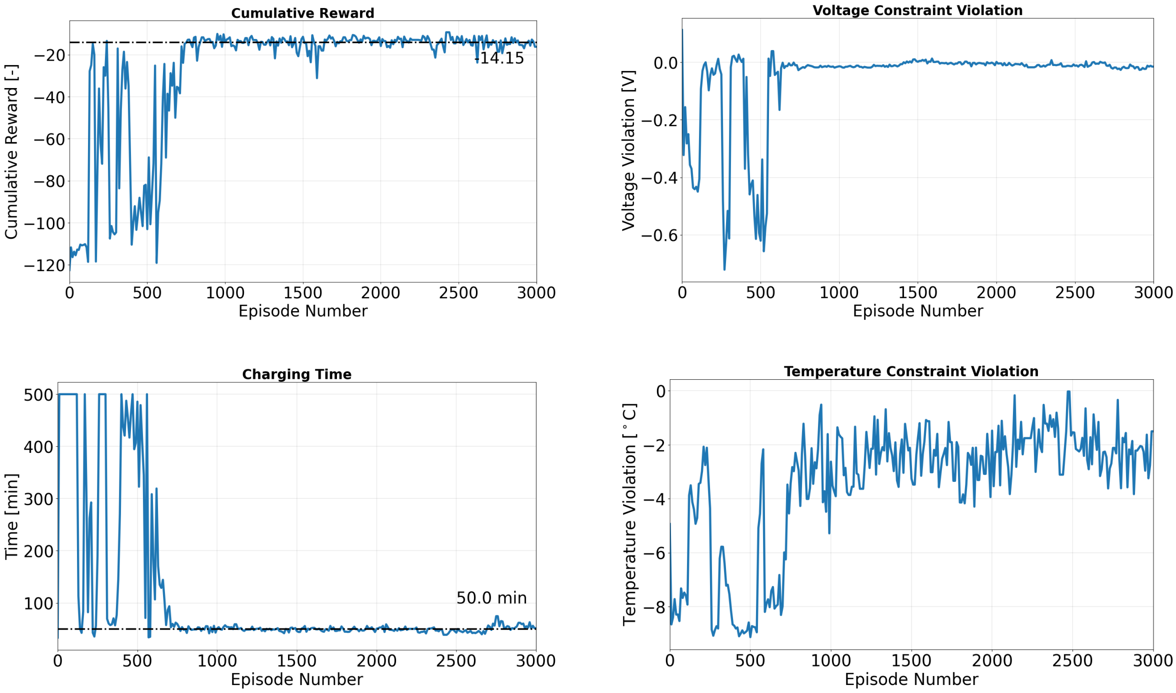


Figure 5-5: Learning Results from First RL Run

To avoid a jittery current signal generated from the RL agent, the input signal is passed through a first-order low-pass filter. Though this filter was not present in the training runs, it is shown not to significantly affect the performance of the charging protocol. Furthermore, the data-driven abstraction is constructed with the presence of the filter. A simulation of the charging protocol resulting from the first RL run is shown in Figure 5-7. It is observed in this example that the safety constraints for voltage and temperature are met, though the temperature specification is almost violated in the early stages of charging.

Similarly to the CC-CV baseline, the closed-loop system resulting from the charging protocol is used to construct a data-driven abstraction. In this case, $N = 10958$ traces are collected for a parameter $\ell = 11$, resulting in no self-loops and reachability of the goal set from all initial conditions. The collected traces are shown in Figure 5-8, where it can be observed that the resulting charging protocol still results in violations of the temperature specification, characterized by a high peak in temperature at the start of charging for low SOC, and the voltage specification during some late stages of charging. This is a considerable improvement from the CC-CV baseline, as the violations are less sever, along with the result that the minimum exhibited side reaction overpotential is found to be $\eta_{2,min} = -0.2326V$, a higher value compared to the baseline. Choosing the lower bound parameter $m = -0.2327V$, the

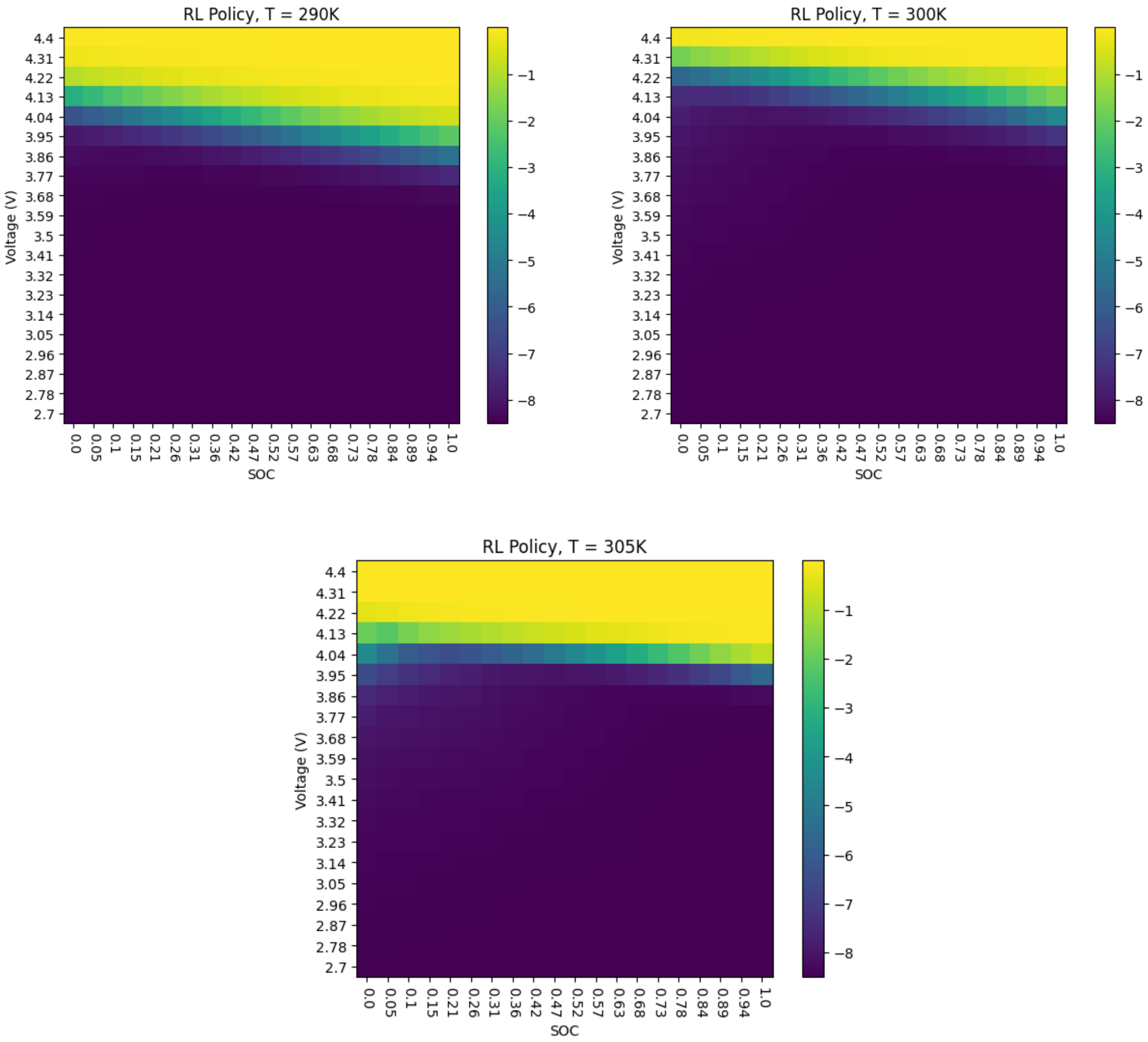


Figure 5-6: Output-Based Policy from First RL Run

abstraction results in an ε-bound of 0.0449, i.e.,

$$\mathbb{P}^N\Big[\mathbb{P}[\mathcal{B}_H(\mathcal{S}(x_0)) \in \mathcal{B}_H(\mathcal{S}_\ell^N)] \geq 1 - 0.0449\Big] \geq 1 - 10^{-6}. \tag{5-7}$$

Though the resulting protocol does not yet satisfy the desired temperature and voltage specifications, the constructed abstraction can still provide valuable information about the control performance. For example, up to the calculated probability bound, it is certain that no temperature violations occur after the initial peak, i.e., after an SOC of 0.1. Furthermore, the side reaction overpotential is bounded by the parameter m up to the probability bound.

The performance of the charging protocol is further analyzed by inspecting the initial conditions from which temperature and voltage violations are observed, i.e., the counterexample initial conditions to be used to construct a new controller as described in the proposed CEGIS scheme. As shown in Figure 5-9, the initial conditions corresponding to temperature safety violations are partial to the low voltage region of $V_0 \in [2.7, 3.2]$, where higher initial temperatures render the charging more prone to violations. Meanwhile, voltage safety violations,

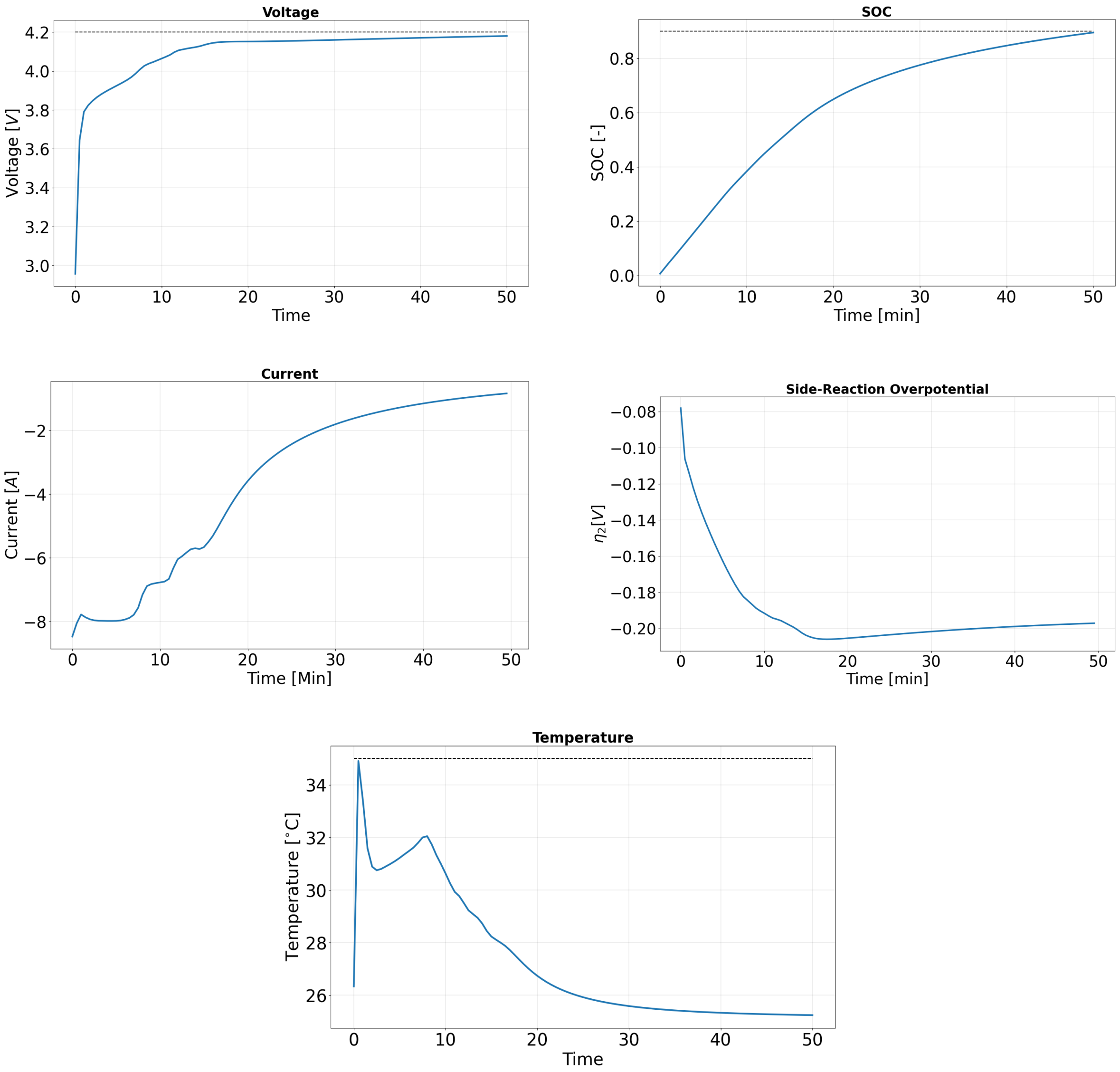


Figure 5-7: Simulation of Charging Protocol from First RL Run

though spread throughout the space of initial conditions, are mainly prevalent in the higher voltage region $V_0 \in [3.8, 4.1]$, as all other violations are less than $0.0025V$ above the $4.2V$ upper bound, whereas the high-voltage region corresponds to a voltage violation of up to $4.224V$.

5-3-2 Low-Voltage Charger

The charging protocol obtained from the first RL run is employed as the initial policy to train the low-voltage charger for initial conditions $V_0 \in [2.7, 3.2]$. In this training run, initial conditions are sampled only from the low-voltage region, and to further enforce the temperature constraint, the penalty on temperature is doubled, i.e., $\lambda_{temp} = 10$. After 2000 episodes of training, this results in the output-based policy illustrated by the heat maps in

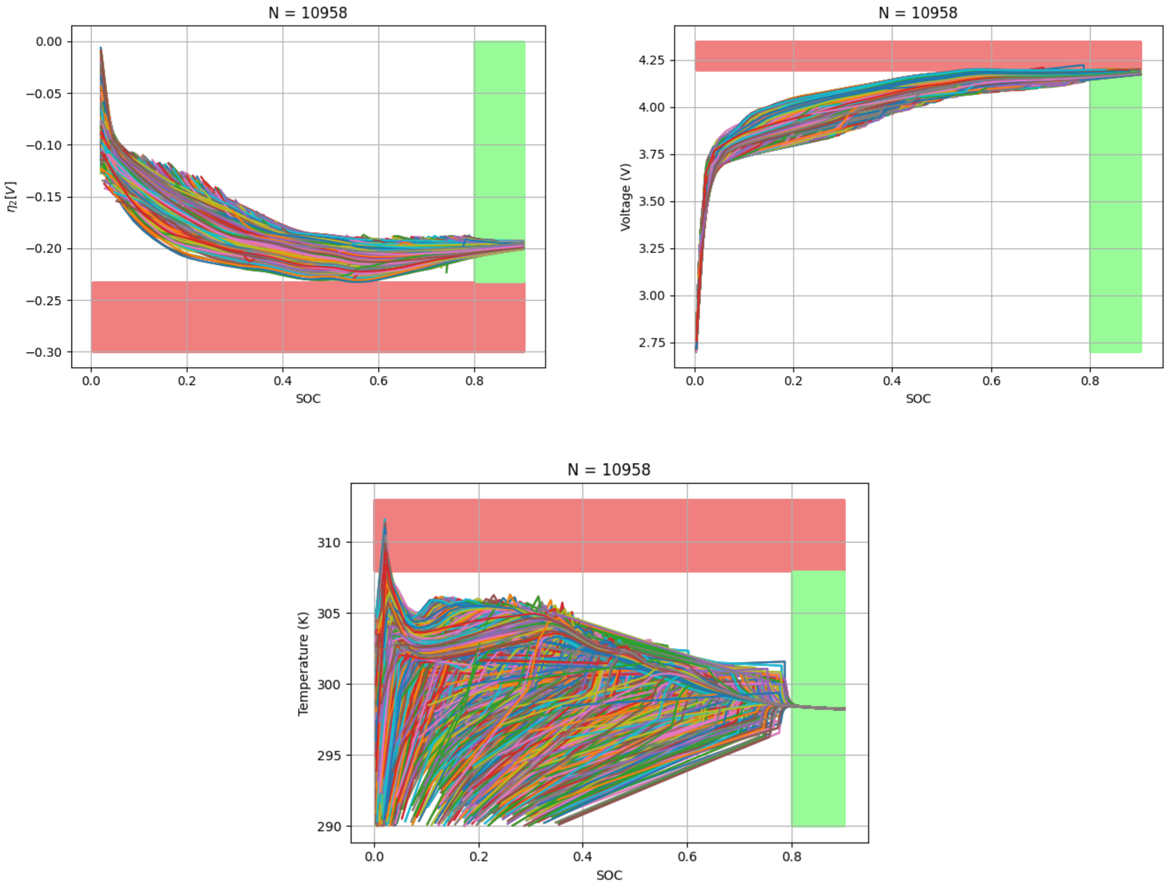


Figure 5-8: Collected Traces for First RL Run

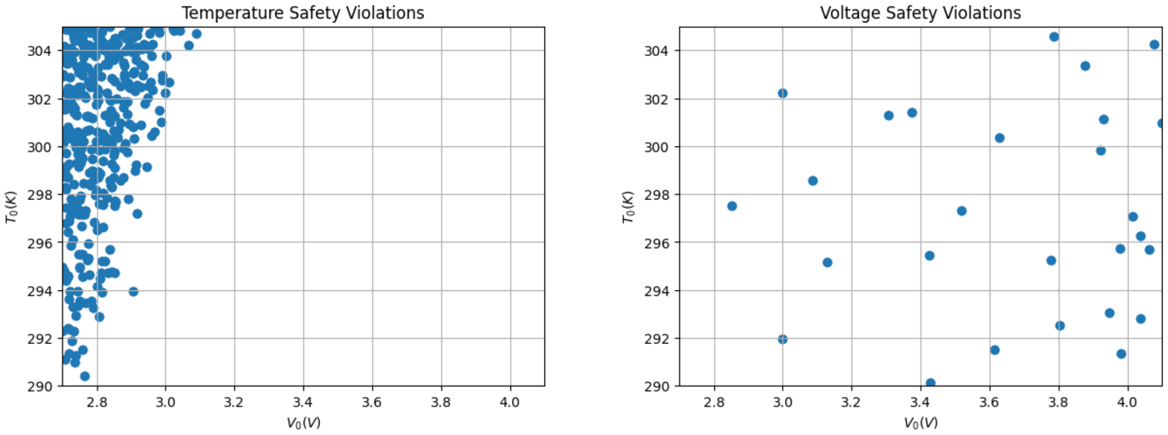
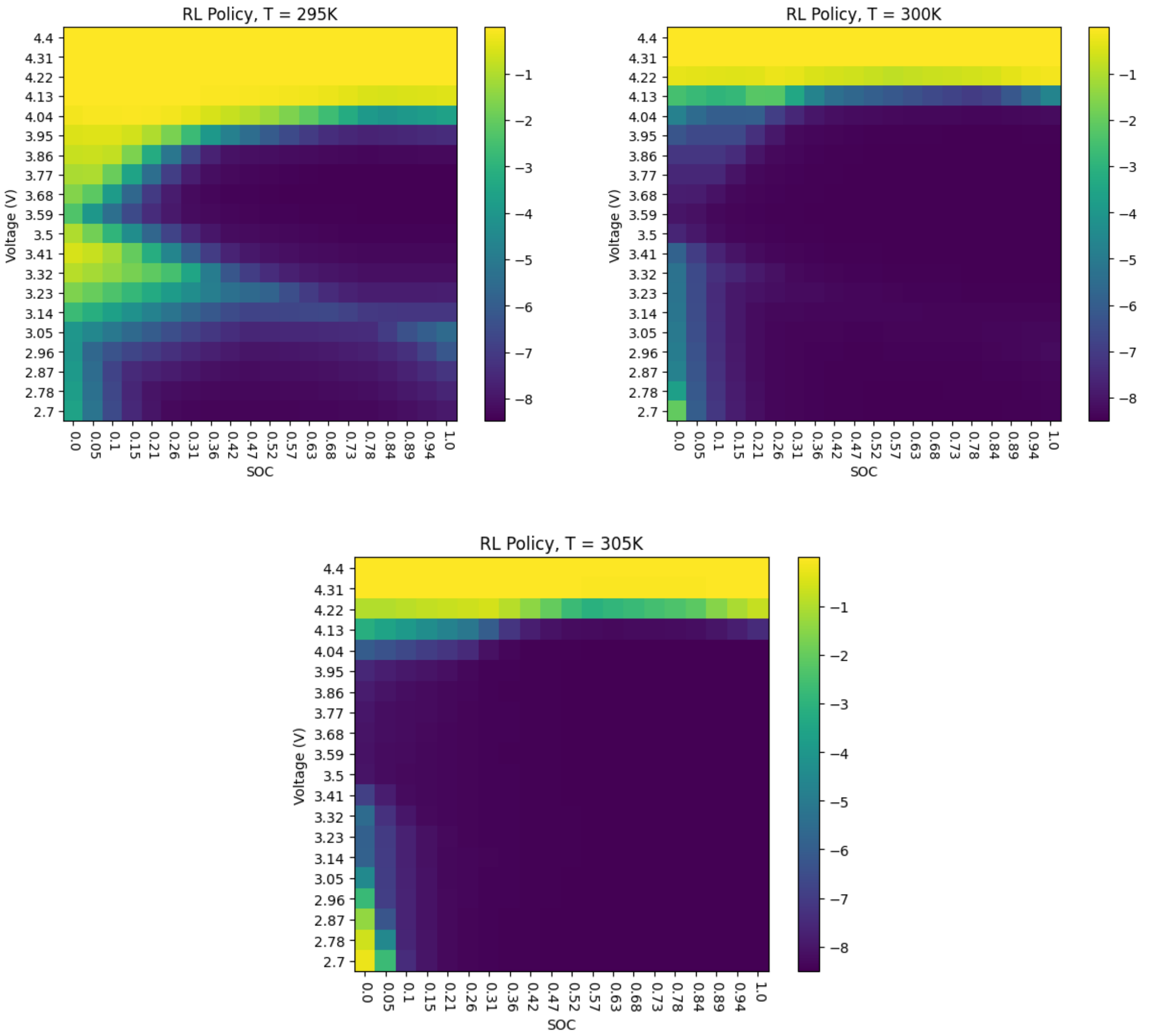


Figure 5-9: Counterexample Initial Conditions for First RL Run

Figure 5-10. The essential difference with the obtained low-voltage charging protocol is that it exhibits more sensitivity to temperature compared to the first charging protocol, especially in the early stages of charging represented by the bottom-left corner of the heat maps. By anticipating high temperature spikes, the charger provides a smaller current for the first few


RL Policy, T = 295K

RL Policy, T = 300K

RL Policy, T = 305K

Figure 5-10: Output-Based Policy of the Low-Voltage Charger

minutes, as shown in Figure 5-11. Additionally, the smaller current in the early stages of charging results in a substantially higher η_2 in the early stages as well, which is found to be critical to improving the aging-aware behavior of the charger, as is discussed at the end of this chapter.

5-3-3 High-Voltage Charger

Similarly, a high-voltage charger is trained from the first charging protocol, for the initial conditions region $V_0 \in [3.8, 4.1]$. In this case, the reward function is maintained as in the first RL run. After 2000 episodes of training, the resulting policy is the one represented by the heatmaps shown in Figure 5-12. The difference in terms of high-voltage charging is noticeable from the simulation shown in Figure 5-13, as the voltage profile becomes highly sensitive to current spikes, almost immediately approaching the upper limit. As a result, the current

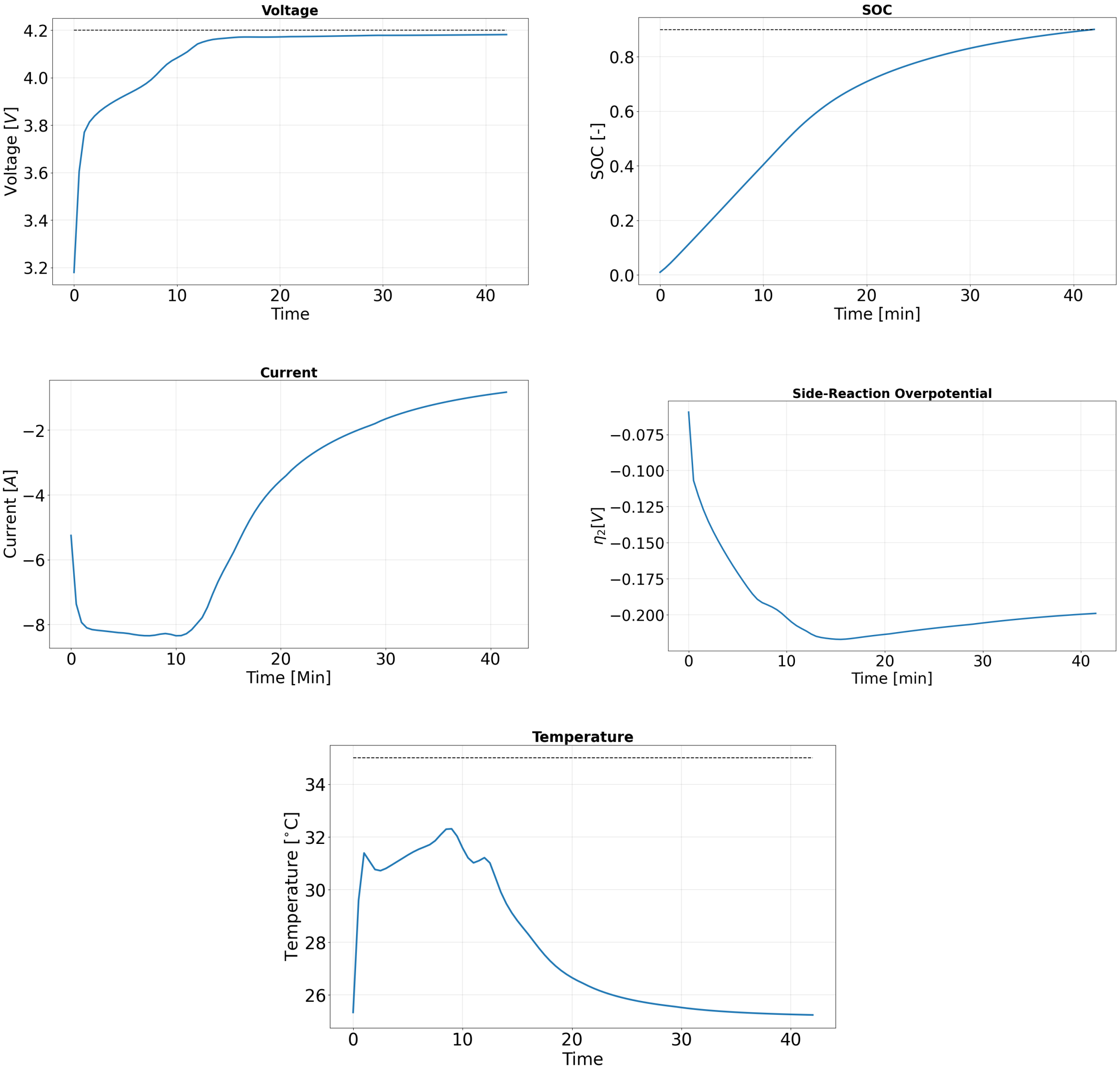


Figure 5-11: Simulation of the Low-Voltage Charging Protocol

profile, though exhibiting a shape similar to that in the previous controllers, has a decay rate different to that in the other simulation runs.

5-4 Final Charging Protocol

The above charging protocols are collected to produce the final switched charging protocol, where the Low-Voltage Charger is used for initial conditions $V_0 \in [2.7, 3.2]$, the charger from the first RL run is used for initial conditions $V_0 \in (3.2, 3.8]$, and the High-Voltage Charger is used for initial conditions $V_0 \in (3.8, 4.1]$, independently of initial temperature. As performed for the previously discussed charging protocols, the data-driven abstraction is constructed from the $N = 16461$ collected traces shown in Figure 5-14. With a choice of parameter

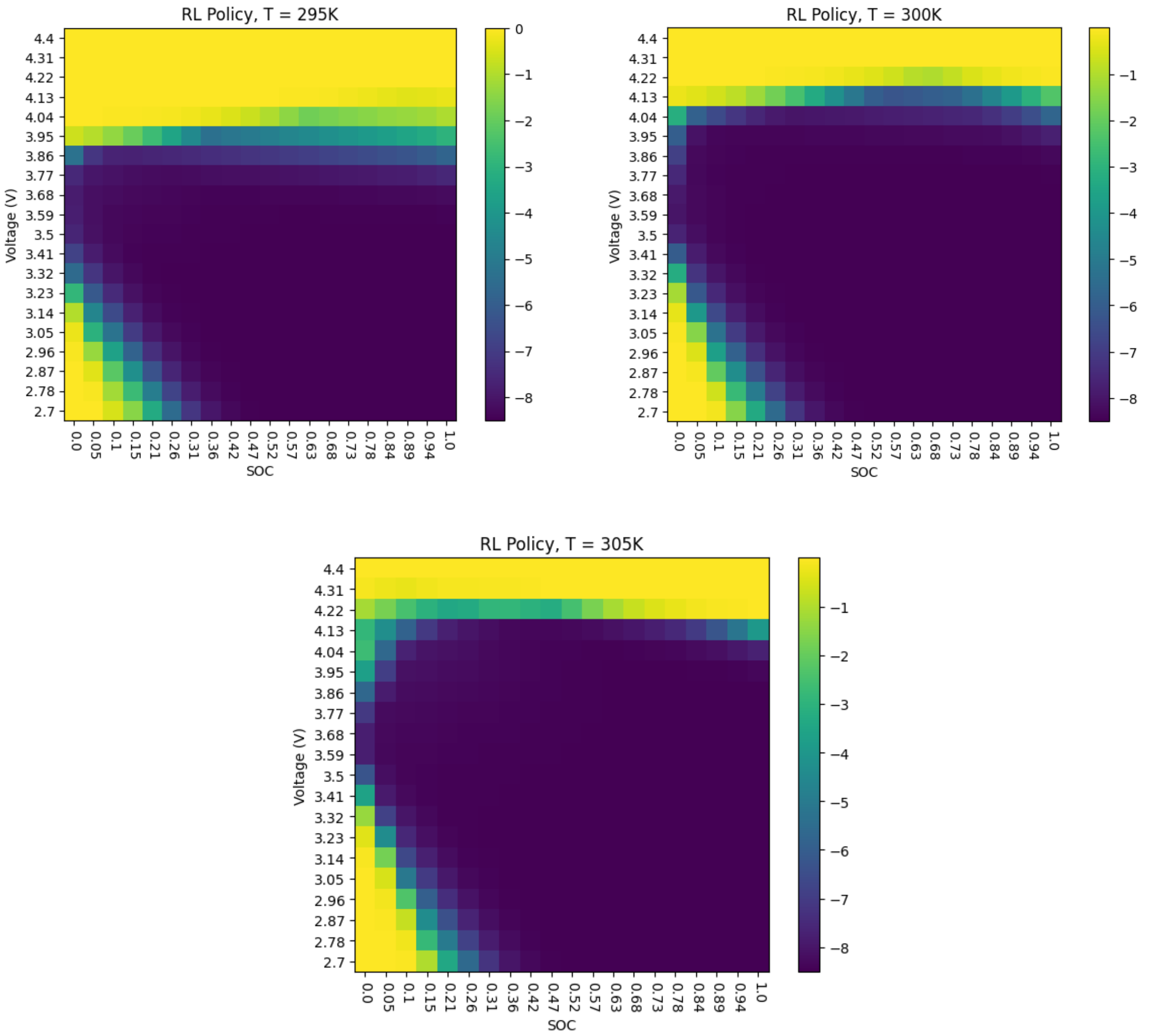


Figure 5-12: Output-Based Policy of the High-Voltage Charger

$\ell = 12$, the undesired self-loops in the abstraction are eliminated, and it is confirmed that all initial conditions reach the charging goal. As shown in Table 5-1, the final charging protocol still exhibits some edge cases that violate the desired specification, but compared to the previous two protocols, the prevalence of such violations is largely decreased. It is also possible to slightly modify the upper bounds for voltage and temperature to be slightly higher than the obtained V_{max} and T_{max}, and to set the lower bound $m = -0.2431V$ to construct an abstraction that behaviorally includes the system based on an ε-bound of 0.05469, i.e.,

$$\mathbb{P}^N\Big[\mathbb{P}[\mathcal{B}_H(\mathcal{S}(x_0)) \in \mathcal{B}_H(\mathcal{S}_\ell^N)] \geq 1 - 0.05469\Big] \geq 1 - 10^{-6}, \tag{5-8}$$

and thus that the resulting charging protocol does not exceed the collected edge cases up to the above probabilistic guarantee.

Furthermore, though the final charging protocol exhibits a slightly lower $\eta_{2,min}$, its exhibiting of much higher values of η_2 in the early phase of charging allows for a larger improvement in

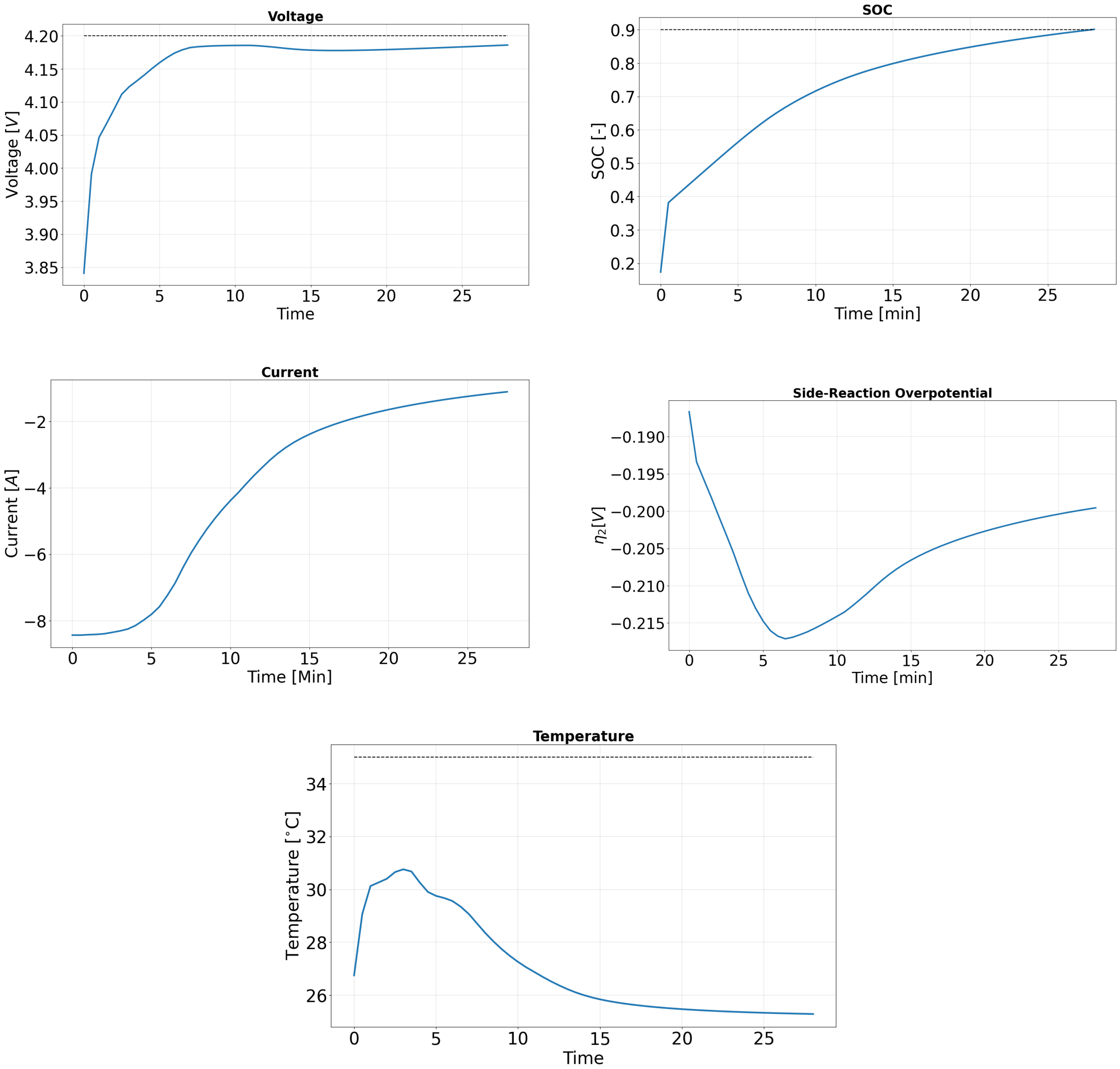


Figure 5-13: Simulation of the High-Voltage Charging Protocol

reducing aging effects. This is shown in Table 5-2, where over 100 simulations, the different charging protocols are employed and the average charging time and lost charge due to aging Q_l are presented. Assuming that aging dynamics exhibit the same behavior between 0% and 10%-capacity loss, the number of cycles until such capacity is lost is estimated by $0.1(3.4)/Q_l$ Ah, where 3.4 Ah represents the capacity of the selected NCR18650GA cell, and Q_l denotes the average lost capacity.

It is observed in both tables that the first charging protocol developed by RL exhibits substantial improvements in performance, safety, and aging-awareness compared to the CC-CV baseline. Additionally, the use of the CEGIS-based control synthesis scheme further improves the safety margin, and extends the life of the cell by an estimated 100 cycles until a 10%-loss.

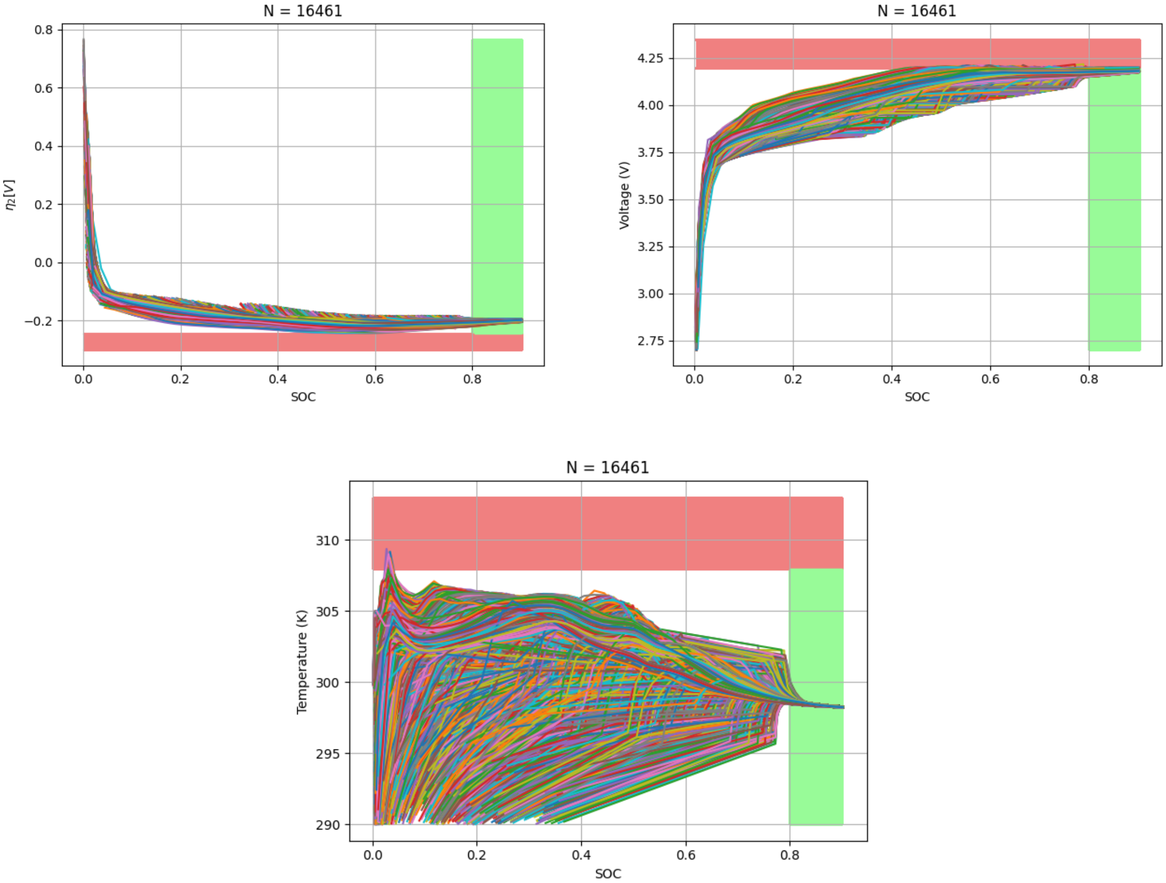


Figure 5-14: Collected Traces for Final Charging Protocol

Protocol	$\eta_{2,min}[V]$	$V_{max}[V]$	$T_{max}[K]$	% Violations
CC-CV	-0.2453	4.2973	312.8617	93.14%
RL	-0.2327	4.2241	311.6118	6.88%
RL + CEGIS	-0.2430	4.2176	309.375	0.984%

Table 5-1: Edge Cases from Collected Traces

Protocol	Time to 90% [min]	Average $Q_l[Ah]$	Estimated Cycles to 10% Capacity Loss
CC-CV	39.61	5.27e-4	644
RL	35	3.67e-4	926
RL + CEGIS	31.7	3.31e-4	1025

Table 5-2: Charging Results over 100 Runs

Chapter 6

Conclusions and Future Work

This thesis has explored the three research questions introduced in Chapter 1, delving into approaches that have combined Reinforcement Learning, Formal Verification, and Data-Driven methods for Systems and Control. This process has ultimately served to synthesize a verifiable and data-driven output-based charging protocol aiming to minimize the aging reaction rates within a Li-Ion cell, while maintaining crucial safety specifications concerning voltage and temperature.

Furthermore, a general approach for verification-based controller synthesis has been proposed through the usage of a CEGIS-based scheme and Reinforcement Learning as a primary tool for policy generation. Such a scheme need not necessarily be applied to the charging of a Li-Ion cell, and can thus be further employed for several further applications.

In the grander scheme of this thesis project, the essential goal has been to tackle the several problems faced in Li-Ion Battery technology, and while this thesis has extensively covered the electrochemical behavior of a single Li-Ion cell, as well as the role of a Battery Management System in administering its charging protocol, several frontiers are attainable for future research and development. This chapter discusses the possible extensions to the work in this thesis.

In retrospect, improvements to the RL scheme could have included penalizing jittery behavior in the input signal, improving the smoothness of the charging policy and thus not requiring the need for low-pass filters in the closed-loop system, which may have compromised the performance. Furthermore, as the CEGIS scheme did not result in a controller that perfectly avoids violating the safety specifications, but considerably reduced the prevalence of such violations, the existence of an ideal controller is still uncertain. That is, a shortcoming of the CEGIS scheme is that it is not guaranteed to converge to a control policy that satisfies the desired specification, as the existence of such a policy is not trivial to prove. Future approaches to CEGIS-based synthesis can be more involved in proving the existence of an adequate controller, or at least the convergence of the CEGIS-based approach.

Further extensions to this work in the bigger context of charging protocol design those listed below.

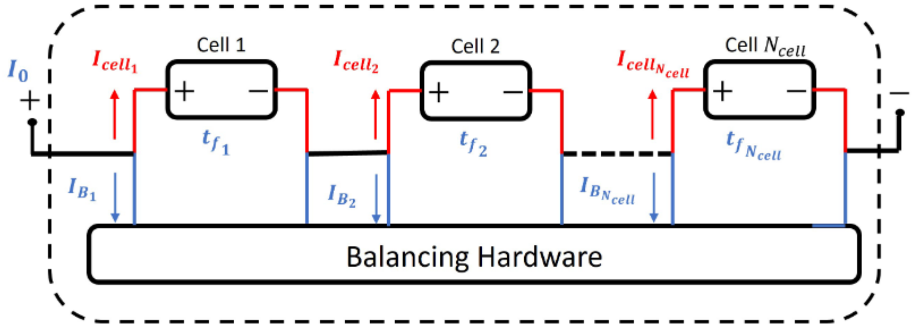


Figure 6-1: Circuit Representation of an Active Balancing System [39]

Data-Driven Abstractions for Control Systems

The work in this thesis has covered the use of data-driven abstractions for autonomous dynamical systems, i.e., systems that do not admit inputs. Thus, abstractions of the system were only constructed based on the closed-loop system resulting from an already present controller. The work in [37] extends the notion of a data-driven abstraction to control systems, i.e., systems that admit inputs. Though the inputs are assumed to be quantized from a discrete set, this method can be employed to construct a data-driven abstraction encapsulating the concrete system behavior up to a certain probabilistic guarantee. As a result, the synthesis of the controller can be expected to become vastly quicker and simpler, as the constructed abstraction can be used as a direct tool for controller synthesis.

Multi-Cell Active Balancing and Charging

The discussion on the control of Li-Ion Battery Systems throughout this thesis has been restricted to single-cell systems, whereas most industrial applications consist of battery packs, where several cells are connected in series and/or parallel, resulting in a higher combined voltage.

While the charging protocols for single cells may resemble those for battery packs, the issue of imbalance between cells is a prevalent one. It is usually assumed in multi-cell charging protocols that all the cells have the same properties, and that they charge and discharge at equal rates, which renders all their states equal at all times. However, due to imperfections and inaccuracies in manufacturing, the cells may develop an imbalance over time, as some may charge/discharge and age at different rates than others, resulting in significant risks to the adequate operation and safety of the battery pack. For example, in a battery pack where all cells are connected in series, one cell may have a lower SOC than the rest, which may limit the performance of the entire pack as the cell with the low SOC limits the discharge of the entire pack, thus reducing the effective capacity of the pack. Alternatively, one cell may have a higher SOC than the rest, which may result in overcharging the cell, an issue that may lead to overheating [38].

To avoid issues related to cell imbalance, it is of interest to develop charging methods on the scale of the battery pack to bring all cells in the pack to the same state. An example of an Active Balancing System is shown in Figure 6-1, where the cells are connected in series. The

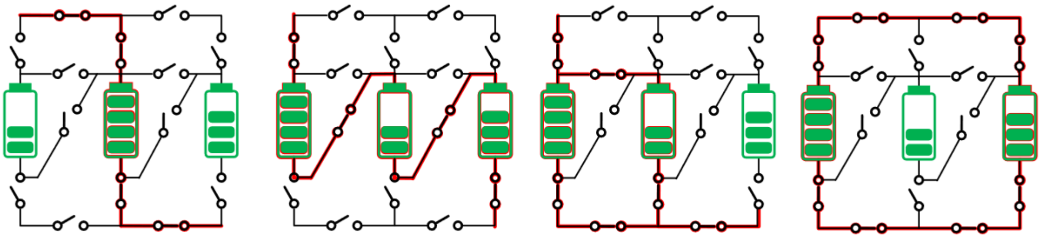


Figure 6-2: Circuit Representation of a Reconfigurable Battery System [40]

work done in [39] consists of employing the Single-Particle Model, a simplification of the DFN Model in a multi-cell structure, to produce an aging-aware optimal control scheme for active balancing. In this work, the aim is to minimize the charging time and aging behavior, based on the previously discussed trade-off between the two, while maintaining the constraint that all cells share the same target SOC at the final time, i.e., at the end of the charging process. Further extensions to this work may include aging-aware protocol design, the use of more advanced models such as the DFN model, or the use of tools such as data-driven abstractions to provide probabilistic bounds on the performance of the balancing circuit.

Reconfigurable Battery Systems

Also in the context of multi-cell battery systems, such systems are usually assumed to have fixed topologies, i.e., the interconnection between cells is fixed. Whereas most battery packs developed in the industry consist of fixed topology, battery packs with modifiable topology using controllable switches between the cells have caught recent interest due to their adaptability. Such battery packs are denoted by Reconfigurable Battery Systems (RBS), such as the one shown in Figure 6-2. In the shown example, several different circuit topologies can be realized, including employing only one cell in charging/discharging, placing all three cells in series or parallel, or discarding a certain cell.

A functional benefit of an RBS is its improved fault tolerance, implying that weak or damaged can be discarded while maintaining the operation of the battery pack, which is not typically possible in fixed-topology packs, as an entire series connection would be prone to losing functionality in case of one damaged cell. Other benefits include added flexibility in extending energy delivery, performing active balancing, managing battery cells with different chemical properties, and terminal voltage customization [41]. Several practical approaches to the modeling, control, and optimization of RBSs are widely discussed in [41] and [42]. In [40], an example of Reconfigurable-Assisted Charging (RAC) is implemented, where a graph-based algorithm is employed to perform active balancing and charging on all the cells in the system, which are modeled by ECMs. The charging scheme is found, in simulation, to yield a higher delivered capacity than it would have for a fixed-topology battery pack.

The behavior of an RBS can be represented using knowledge from Hybrid Systems, given that such a system may exhibit both continuous and discrete states. Thus, several tools from this field can be employed for improved modeling, simulation, control, and verification in the future.

Appendix A

Review of Reinforcement Learning

This chapter is concerned with formalizing the RL scheme, as well as the actor-critic and DDPG framework. The discussion hereafter is mostly adapted from [21]. In RL, the goal is to find the best policy that can maximize rewards from an environment E, which describes the plant such as a single Li-Ion cell. For a given time step $t \in \mathbb{R}^+$, the environment is described by a state $s_t \in \mathcal{S}$, where $\mathcal{S}$ denotes the state space, whereas the control policy produces an action $a_t \in \mathcal{A}$ based on the state s_t, where $\mathcal{A}$ denotes the action space. The resulting following state $s_{t+1} \in \mathcal{S}$ is produced by the environment based on a transition probability $p(s_{t+1}|s_t, a_t)$, and a scalar reward $r_{t+1} = r(s_t, a_t)$ based on the reward function $r(\cdot, \cdot)$ is deduced. The policy is denoted by the mapping $\pi : \mathcal{S} \to \mathcal{A}$, and the total discounted reward from time t onward is denoted by

$$R_t = \sum_{k=0}^{\infty} \gamma^k r(s_{t+k}, a_{t+k}), \tag{A-1}$$

where $\gamma \in [0, 1]$ represents the discounting factor. Subsequently, the state value function under a given policy π, denoted by $V^\pi(s_t)$, represents the expected total discounted reward starting from state s_t under the policy π, i.e.,

$$V^\pi(s_t) = \mathbb{E}[R_t|s_t]. \tag{A-2}$$

The optimal policy π^* then corresponds to the one that maximizes the state value function, i.e.,

$$\pi^*(s_t) = arg \max_{\pi} V^\pi(s_t). \tag{A-3}$$

Model-free RL, i.e., a Reinforcement Learning application that does not require knowledge of the model of the system, requires the construction of a predictor for the behavior of the system and the reward function, referred to as a state-action value function, or a Q-function denoted by $Q^\pi(s_t, a_t)$, which evaluates the expected total discounted reward from a given state s_t after applying a certain action a_t. This can be expressed in mathematical terms as

$$Q^\pi(s_t, a_t) = \mathbb{E}[R_t|s_t, a_t], \tag{A-4}$$

where the optimal Q-function can be defined as

$$Q^*(s_t, a_t) = arg \max_{\pi} Q^{\pi}(s_t, a_t). \tag{A-5}$$

Equivalently, the optimal state-value function can be defined by maximizing $Q^*(s_t, a_t)$ over all possible actions $a_t \in \mathcal{A}$, i.e.,

$$V^*(s_t) = \max_{a_t \in \mathcal{A}} Q^*(s_t, a_t) \tag{A-6}$$

Effectively, knowledge of the optimal Q-function is equivalent to knowledge of the environment dynamics, as the optimal policy can be retrieved from

$$\pi^*(s_t) = a_t^* = arg \max_{a_t \in \mathcal{A}} Q^*(s_t, a_t). \tag{A-7}$$

Actor-Critic Framework and DDPG Algorithm

In typical RL applications, the state and action spaces are finite, which results in difficulties when partitioning a continuous state and action space and the curse of dimensionality due to large partitions. For this reason, it is of interest to explore RL frameworks that allow for continuous state and action spaces. The Actor-Critic framework allows this, as it is a policy gradient approach that employs function approximators, i.e., neural networks. In this approach, the actor represents the policy, which is improved based on the value function estimated by the critic. Specifically, the Deep Deterministic Policy Gradient (DDPG) algorithm is used in [21] to train the actor and critic.

Essentially, the algorithm consists of parametrizing the neural networks representing the critic, i.e., $Q(s_t, a_t|\theta^Q)$, and the actor, i.e. $\pi(s_t, \theta^{\pi})$. Additionally, target networks $Q'(s_t, a_t|\theta^{Q'})$ and $\pi(s_t|\theta^{\pi'})$ are defined in order to implement a soft update on the parameters to reduce the chattering during learning and enhance convergence.

The critic aims to evaluate the current policy, which is given by the sum of the output of the actor network and some exploration noise $\mathcal{N}_t$, i.e.,

$$a_t = \pi(s_t|\theta^{\pi}) + \mathcal{N}_t, \tag{A-8}$$

where the exploration noise is a random variable. The learning process is based on saving single instances into a replay buffer memory. For each time step t, the tuple $(s_t, a_t, r_{t+1}, s_{t+1})$ is committed to the replay buffer. After the number of stored tuples reaches a threshold value N, at each time step, a randomly selected batch of N tuples is sampled to be used in learning. To update the critic, the term

$$y_i = r_{i+1} + \gamma Q'(s_{i+1}, \pi'(s_{i+1}|\theta^{\pi'})|\theta^{Q'}), i = 1, \cdots, N, \tag{A-9}$$

is defined, where the superscript ' denotes the target network where the parameters are slowly updated. The critic is updated to minimize the loss function $\mathcal{L}(\theta^Q)$ based on the gradient descent algorithm as follows:

$$\mathcal{L}(\theta^Q) = \frac{1}{N} \sum_i (y_i - Q(s_i, a_i|\theta^Q))^2, \tag{A-10}$$

$$\theta_{k+1}^Q = \theta_k^Q - \eta_Q \nabla_{\theta^Q} \mathcal{L}(\theta^Q), \tag{A-11}$$

where k denotes the index and η_Q the learning rate of the gradient descent algorithm.

Furthermore, the actor is updated so as to maximize the total expected reward $V^\pi(s_t)$, which can also be referred to as $\mathcal{J}(\theta^\pi)$ to highlight the dependence on parametrization. The update is also based on the gradient descent algorithm, i.e.,

$$\theta^\pi_{k+1} = \theta^\pi_k + \eta_\pi \nabla_{\theta^\pi} \mathcal{J}(\theta^\pi), \tag{A-12}$$

where η_θ denotes the corresponding learning rate. The gradient of the total expected reward as a function of the actor parameters can be approximated based on the samples from the replay buffer as follows:

$$\nabla_{\theta^\pi} \mathcal{J}(\theta^\pi) \approx \frac{1}{N} \sum_i [\nabla_a Q(s_i, a_i|\theta^Q) \nabla_{\theta^\pi} \pi(s_t|\theta^\pi)]. \tag{A-13}$$

After the update of actor and critic network parameters, the soft update is performed:

$$\theta^{Q'} \leftarrow \tau\theta^Q + (1-\tau)\theta^{Q'}, \tag{A-14}$$

$$\theta^{\pi'} \leftarrow \tau\theta^\pi + (1-\tau)\theta^{\pi'}, \tag{A-15}$$

where τ is an update parameter.

It is noted that, as opposed to approaches that consist of finite state and action spaces, the actor-critic framework does not provide any convergence guarantees. It is then crucial to the learning process that the updates are small in order to avoid divergence.

This concludes the discussion on the Reinforcement Learning concepts that are of interest for this thesis.

Appendix B

Numerical Implementation of DFN Model

B-1 Parameters and Expressions

The battery parameters values are based on the original parameters provided in [21], whereas some uncertainty ranges are defined based on the work done in [10]. The DFN Model is simulated using on the code provided in [21], which is based on the CasADi library [43]. The model is simulated with a sampling time of $\delta_t = 30$ seconds.

Name	Symbol	Units	Range/Value
	Geometric Parameters		
Anode Thickness	δ_n	$[m]$	$100e-6$
Cathode Thickness	δ_p	$[m]$	$100.5e-6$
Total Cell Thickness	L	$[m]$	$225.9e-6$
Radius of Solid Particles (Anode)	$R_{s,n}$	$[m]$	$5e-6$
Radius of Solid Particles (Cathode)	$R_{s,p}$	$[m]$	$7.5e-6$
Volume Fraction in Solid (Anode)	$\varepsilon_{s,n}$	[-]	0.7
Volume Fraction in Solid (Cathode)	$\varepsilon_{s,p}$	[-]	0.67
Volume Fraction in Electrolyte (Anode)	$\varepsilon_{e,n}$	[-]	0.3
Volume Fraction in Electrolyte (Cathode)	$\varepsilon_{e,p}$	[-]	0.3
Volume Fraction in Electrolyte (Separator)	$\varepsilon_{e,s}$	[-]	0.4
Specific Interfacial Surface Area (Anode)	$a_{s,n}$	$[m^{-1}]$	$3\varepsilon_{s,n}/R_{s,n}$
Specific Interfacial Surface Area (Cathode)	$a_{s,p}$	$[m^{-1}]$	$3\varepsilon_{s,p}/R_{s,p}$
	Transport Parameters		
Ref. Solid Diffusion Coeff. (Anode)	$D_{s,n0}$	$[m^2.s^-1]$	$[2.5e-15, 3.5e-14]$
Ref. Solid Diffusion Coeff. (Cathode)	$D_{s,p0}$	$[m^2.s^-1]$	$[2.24e-15, 3.5e-14]$
Conductivity of Solid (Anode)	σ_n	$[\Omega^{-1}.m^{-1}]$	$[90, 110]$
Conductivity of Solid (Cathode)	σ_p	$[\Omega^{-1}.m^{-1}]$	$[0.05, 0.15]$

Kinetic Parameters			
Ref. Reaction Rate (Anode)	k_{n0}	$[A.m^{-2}.mol^{2(1+\alpha)}]$	$[4e-6, 8e-5]$
Ref. Reaction Rate (Cathode)	k_{p0}	$[A.m^{-2}.mol^{2(1+\alpha)}]$	$[2.5e-7, 5e-5]$
Thermodynamic Parameters			
Cell Heat Capacity Cell Thermal Resistance	c_p	$[J.kg^{-1}.K^{-1}]$	1000
Mass of Cell	m	$[kg]$	0.05
Ambient Temperature	T_{amb}	$[K]$	298.15
Reference Temperature	T_{ref}	$[K]$	298.15
Activation Energy k_n	E_{k_n}	$[J^{-1}.mol]$	37480
Activation Energy k_p	E_{k_p}	$[J^{-1}.mol]$	39570
Activation Energy $D_{s,n}$	$E_{D_{s,n}}$	$[J^{-1}.mol]$	42770
Activation Energy κ	E_κ	$[J^{-1}.mol]$	34700
Concentrations			
Max Concentration (Anode)	$c_{s,n}^{max}$	$[mol.m^{-3}]$	$3e4$
Max Concentration (Cathode)	$c_{s,p}^{max}$	$[mol.m^{-3}]$	$4.5e4$
Total Lithium in Solid Phase	$n_{Li,s}$	$[mol]$	0.27
Initial Electrolyte Concentration	c_{e0}	$[mol.m^{-3}]$	$1e3$
Stoichiometry at 0% (Anode)	$\theta_{0\%}^{-}$	$[-]$	0.03
Stoichiometry at 100% (Anode)	$\theta_{100\%}^{-}$	$[-]$	0.9
Stoichiometry at 0% (Cathode)	$\theta_{0\%}^{+}$	$[-]$	0.8
Stoichiometry at 100% (Cathode)	$\theta_{100\%}^{+}$	$[-]$	0.2
Aging Parameters			
Initial SEI Film Resistance	R_{f0}	$[\Omega.m^2]$	$[1e-5, 1e-3]$
Side Reaction Equilibrium Potential	U_2	$[V]$	0.4
Side Reaction Exchange Current Density	$i_{0,2}$	$[A.m^{-2}]$	$1.178e-7$
Molar Volume of SEI	$\tilde{V}_f$	$[m^3.mol^{-1}]$	$1e-5$
Conductivity of SEI	σ_f	$[\Omega^{-1}.m^{-1}]$	$2.3e-6$
Miscellaneous			
Ideal Gas Constant	R	$[J.mol^{-1}.K^{-1}]$	8.314472
Faraday Constant	F	$[C.mol^{-1}]$	96485.3329
Cell Surface Area	A	$[m^2]$	0.1283
Charge Transfer Coefficient	$\alpha_a, \alpha_c, \alpha$	$[-]$	0.5
Transference Number	t_+^0	$[-]$	$[0.35, 0.55]$
Bruggeman Porosity Constant	p	$[-]$	$[1.7, 1.9]$

Table B-1: DFN Model Parameter Values and Ranges

Furthermore, the Reference Potential is given as a function of θ, which usually represents the State of Charge of the battery, for each of the anode of the cathode, expressed as

$$\begin{aligned} U^-(\theta) = 0.194 + 1.5\exp(-120\theta) &+ 0.0351\tanh\Big(\frac{\theta - 0.286}{0.083}\Big) - 0.0045\tanh\Big(\frac{\theta - 0.849}{0.119}\Big) \\ &- 0.035\tanh\Big(\frac{\theta - 0.9233}{0.05}\Big) - 0.0147\tanh\Big(\frac{\theta - 0.5}{0.034}\Big) - 0.102\tanh\Big(\frac{\theta - 0.194}{0.142}\Big) \\ &- 0.022\tanh\Big(\frac{\theta - 0.9}{0.0164}\Big) - 0.011\tanh\Big(\frac{\theta - 0.124}{0.0226}\Big) + 0.0155\tanh\Big(\frac{\theta - 0.105}{0.029}\Big), \end{aligned}$$

$$
\begin{aligned}
U^{+}(\theta) = 2.16216 + 0.07645\tanh(30.834 - 54.4806\theta) + 2.1581\tanh(52.294 - 50.294\theta)\\
- 0.14169\tanh(11.0923 - 19.8543\theta) + 0.2051\tanh(1.4684 - 5.4888\theta)\\
+ 0.2531\tanh\left(\frac{0.56478-\theta}{0.1316}\right) - 0.02167\tanh\left(\frac{\theta-0.525}{0.006}\right).
\end{aligned}
$$

The reference electrolyte diffusion constant and reference ionic conductivity are given as functions of the electrolyte concentration c_e and temperature, and thus also vary spatially:

$$
\begin{aligned}
\kappa_{ref} &= 9.18e - 7c_e^2\exp(-0.091(0.001c_e-1)^2 - 0.682(0.001c_e-1))\\
D_e(c_e,T) &= 10^{f(c_e,T)}\\
f(c_e,T) &= -6.4887 - \frac{822.3727}{T-0.0094-2.8023c_e} + c_e\Big(1.0369 - \frac{407.9648}{T-0.0094-2.8023c_e}\Big)
\end{aligned}
$$

B-2 CC-CV Charging Implementation

The CC-CV charging controller is implemented as a modified PI-controller, which adds an exponential decay term for the current input, resulting in the expression

$$
I(t) = \begin{cases} -I_{max}, & t < t_{cv},\\ a(V_0)I(t-1) + K_p(V(t)-4.2) + K_i\sum_{k=t_{cv}}^{t}(V(k)-4.2), & else, \end{cases}
$$

where t_{cv} is the time at which the voltage V reaches a value of $4.16V$ for the first time, and the term $a(V_0)$ depends on the initial voltage upon charging, given by

$$
a(V_0) = \begin{cases} 0.95, & V_0 \leq 3.8V,\\ 0.93, & else. \end{cases}
$$

The proportional and integral gains are chosen to be $K_p = 2.5$ and $K_i = 0.1$, respectively. Though this control scheme is shown to have its flaws in terms of sometimes exceeding the defined voltage limit, it is considered an adequate due to its simplicity.

Bibliography

[1] D. Larcher and J.-M. Tarascon, "Towards Greener and More Sustainable batteries for Electrical Energy storage," *Nature Chemistry*, vol. 7, p. 19–29, Jan. 2015.

[2] S. Xia, X. Wu, Z. Zhang, Y. Cui, and W. Liu, "Practical Challenges and Future Perspectives of All-Solid-State Lithium-Metal Batteries," *Chem*, vol. 5, p. 753–785, Apr. 2019.

[3] A. R. Carreon, "The EV Battery Supply Chain Explained." `https://rmi.org/the-ev-battery-supply-chain-explained/`, May 2023.

[4] "Electric Vehicles in Europe." `https://www.eea.europa.eu/publications/electric-vehicles-in-europe`, Sep 2016.

[5] IEA, "Global Supply Chains of EV Batteries." `https://www.iea.org/reports/global-supply-chains-of-ev-batteries`, 2022. Licence: CC BY 4.0.

[6] P. Tabuada, *Verification and Control of Hybrid Systems: A Symbolic Approach*. Springer Science & Business Media, 2009.

[7] R. Coppola, A. Peruffo, and M. Mazo Jr, "Data-driven Abstractions for Verification of Deterministic Systems," *arXiv preprint arXiv:2211.01793*, 2022.

[8] V. Pop, H. J. Bergveld, J. H. G. O. h. Veld, P. P. L. Regtien, D. Danilov, and P. H. L. Notten, "Modeling Battery Behavior for Accurate State-of-Charge Indication," *Journal of The Electrochemical Society*, vol. 153, p. A2013, Sept. 2006.

[9] K. A. Smith, C. D. Rahn, and C.-Y. Wang, "Control-Oriented 1D Electrochemical Model of Lithium-ion Battery," *Energy Conversion and Management*, vol. 48, p. 2565–2578, Sept. 2007.

[10] Z. Khalik, *Modeling and Optimal Control for Aging-Aware Charging of Batteries*. PhD Thesis (research TU/e / graduation TU/e), Eindhoven University of Technology, Eindhoven, Nov. 2021.

[11] A. Barré, B. Deguilhem, S. Grolleau, M. Gérard, F. Suard, and D. Riu, "A Review on Lithium-ion Battery Ageing Mechanisms and Estimations for Automotive Applications," *Journal of Power Sources*, vol. 241, p. 680–689, Nov. 2013.

[12] R. Darling and J. Newman, "Modeling Side Reactions in Composite $Li_yMn_2O_4$ Electrodes," *Journal of The Electrochemical Society*, vol. 145, p. 990, Mar. 1998.

[13] Z. Khalik, H. J. Bergveld, and M. C. F. Donkers, "Ageing-Aware Charging of Lithium-ion Batteries Using an Electrochemistry-Based Model with Capacity-Loss Side Reactions," in *2020 American Control Conference (ACC)*, p. 2213–2218, July 2020.

[14] Z. Khalik, H. Bergveld, and M. Donkers, "Ageing-Aware Charging of Lithium-ion Batteries Using a Surrogate Model," in *2021 American Control Conference (ACC)*, p. 4414–4420, May 2021.

[15] J. Liu, Z. Bao, Y. Cui, E. J. Dufek, J. B. Goodenough, P. Khalifah, Q. Li, B. Y. Liaw, P. Liu, A. Manthiram, *et al.*, "Pathways for Practical High-Energy Long-Cycling Lithium-Metal Batteries," *Nature Energy*, vol. 4, no. 3, pp. 180–186, 2019.

[16] B. Yann Liaw, G. Nagasubramanian, R. G. Jungst, and D. H. Doughty, "Modeling of Lithium-ion Cells—A Simple Equivalent-Circuit Model Approach," *Solid State Ionics*, vol. 175, p. 835–839, Nov. 2004.

[17] J. Jang and J. Yoo, "Equivalent Circuit Evaluation Method of Lithium-Polymer Battery Using Bode Plot and Numerical Analysis," *IEEE Transactions on Energy Conversion*, vol. 26, p. 290–298, Mar. 2011.

[18] Q.-K. Wang, Y.-J. He, J.-N. Shen, Z.-F. Ma, and G.-B. Zhong, "A Unified Modeling Framework for Lithium-ion Batteries: An Artificial Neural Network-Based Thermal Coupled Equivalent Circuit Model Approach," *Energy*, vol. 138, p. 118–132, Nov. 2017.

[19] C. Zhang, Y. Zhu, G. Dong, and J. Wei, "Data-driven Lithium-ion Battery States Estimation using Neural Networks and Particle Filtering," *International Journal of Energy Research*, vol. 43, no. 14, p. 8230–8241, 2019.

[20] M. Doyle, T. F. Fuller, and J. Newman, "Modeling of Galvanostatic Charge and Discharge of the Lithium/Polymer/Insertion cell," *Journal of The Electrochemical Society*, vol. 140, p. 1526, June 1993.

[21] S. Park, A. Pozzi, M. Whitmeyer, H. Perez, A. Kandel, G. Kim, Y. Choi, W. T. Joe, D. M. Raimondo, and S. Moura, "A Deep Reinforcement Learning Framework for Fast Charging of Li-ion Batteries," *IEEE Transactions on Transportation Electrification*, vol. 8, no. 2, pp. 2770–2784, 2022.

[22] Z. Khalik, M. C. F. Donkers, and H. J. Bergveld, "Model Simplifications and their Impact on Computational Complexity for an Electrochemistry-Based Battery Modeling Toolbox," *Journal of Power Sources*, vol. 488, p. 229427, Mar. 2021.

[23] V. Sulzer, S. G. Marquis, R. Timms, M. Robinson, and S. J. Chapman, "Python Battery Mathematical Modelling (PyBaMM)," *Journal of Open Research Software*, vol. 9, no. 1, 2021.

[24] Y. Wang, J. Tian, Z. Sun, L. Wang, R. Xu, M. Li, and Z. Chen, "A Comprehensive Review of Battery Modeling and State Estimation Approaches for Advanced Battery Management Systems," *Renewable and Sustainable Energy Reviews*, vol. 131, p. 110015, Oct. 2020.

[25] X. Hu, F. Feng, K. Liu, L. Zhang, J. Xie, and B. Liu, "State Estimation for Advanced Battery Management: Key Challenges and Future Trends," *Renewable and Sustainable Energy Reviews*, vol. 114, p. 109334, Oct. 2019.

[26] T. Weaver, A. Allam, and S. Onori, "A Novel Lithium-ion Battery Pack Modeling Framework - Series-Connected Case Study," in *2020 American Control Conference (ACC)*, p. 365–372, July 2020.

[27] S. Santhanagopalan and R. E. White, "Online Estimation of the State of Charge of a Lithium-ion Cell," *Journal of Power Sources*, vol. 161, p. 1346–1355, Oct. 2006.

[28] C. Zou, X. Hu, Z. Wei, T. Wik, and B. Egardt, "Electrochemical Estimation and Control for Lithium-ion Battery Health-Aware Fast Charging," *IEEE Transactions on Industrial Electronics*, vol. 65, p. 6635–6645, Aug. 2018.

[29] S. Kolluri, S. V. Aduru, M. Pathak, R. D. Braatz, and V. R. Subramanian, "Real-time nonlinear model predictive control (nmpc) strategies using physics-based models for advanced lithium-ion battery management system (bms)," *Journal of The Electrochemical Society*, vol. 167, p. 063505, Apr. 2020.

[30] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous Control with Deep Reinforcement Learning," *arXiv preprint arXiv:1509.02971*, 2015.

[31] M. C. Campi, S. Garatti, and F. A. Ramponi, "A General Scenario Theory for Nonconvex Optimization and Decision Making," *IEEE Transactions on Automatic Control*, vol. 63, p. 4067–4078, Dec. 2018.

[32] S. Garatti and M. C. Campi, "The Risk of Making Decisions from Data through the Lens of the Scenario Approach," *IFAC-PapersOnLine*, vol. 54, p. 607–612, Jan. 2021.

[33] R. Alur, R. Bodik, G. Juniwal, M. M. K. Martin, M. Raghothaman, S. A. Seshia, R. Singh, A. Solar-Lezama, E. Torlak, and A. Udupa, "Syntax-Guided Synthesis," in *2013 Formal Methods in Computer-Aided Design*, p. 1–8, Oct. 2013.

[34] H. Ravanbakhsh and S. Sankaranarayanan, "Learning Control Lyapunov Functions from Counterexamples and Demonstrations," *Autonomous Robots*, vol. 43, p. 275–307, Feb. 2019.

[35] "NCR18650GA - Lithium-ion Batteries - Secondary Batteries (Rechargeable Batteries) - Panasonic." `https://industrial.panasonic.com/ww/products/pt/lithium-ion/models/NCR18650GA`.

[36] Delft High Performance Computing Centre (DHPC), "DelftBlue Supercomputer (Phase 1)." `https://www.tudelft.nl/dhpc/ark:/44463/DelftBluePhase1`, 2022.

[37] R. Coppola, A. Peruffo, and M. Mazo Jr, “Data-Driven Abstractions for Control Systems,” *arXiv preprint arXiv:2402.10668*, 2024.

[38] J. Gallardo-Lozano, E. Romero-Cadaval, M. I. Milanes-Montero, and M. A. Guerrero-Martinez, “Battery Equalization Active Methods,” *Journal of Power Sources*, vol. 246, p. 934–949, Jan. 2014.

[39] V. Azimi, A. Allam, and S. Onori, “Extending Life of Lithium-ion Battery Systems by Embracing Heterogeneities via an Optimal Control-Based Active Balancing Strategy,” *IEEE Transactions on Control Systems Technology*, vol. 31, p. 1235–1249, May 2023.

[40] Z. M. Pinter, D. Papageorgiou, G. Rohde, M. Marinelli, and C. Træholt, “Review of Control Algorithms for Reconfigurable Battery Systems with an Industrial Example,” in *2021 56th International Universities Power Engineering Conference (UPEC)*, p. 1–6, Aug. 2021.

[41] W. Han, T. Wik, A. Kersten, G. Dong, and C. Zou, “Next-Generation Battery Management Systems: Dynamic Reconfiguration,” *IEEE Industrial Electronics Magazine*, vol. 14, p. 20–31, Dec. 2020.

[42] S. Ci, N. Lin, and D. Wu, “Reconfigurable Battery Techniques and Systems: A Survey,” *IEEE Access*, vol. 4, p. 1175–1189, 2016.

[43] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, “CasADi – A Software Framework for Nonlinear Optimization and Optimal Control,” *Mathematical Programming Computation*, vol. 11, no. 1, pp. 1–36, 2019.