Improvements on physics-informed models for lithium batteries

Brady Planden

School of Engineering, Computing, and Mathematics
Oxford Brookes University

This dissertation is submitted for the degree of
Doctor of Philosophy

January 2023

To my parents for outlining a path for me to succeed.

To every mentor that provided guidance over that always-changing path.

To Katie, for ensuring that flowers were appreciated throughout that path.

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this or any other university. This thesis is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Publications.

Brady Planden
January 2023

Acknowledgements

I would like to thank my director of studies, Professor Denise Morrey, for their guidance, support, and patience throughout my PhD. This academic and personal support has been appreciated, and I am likewise grateful for the additional independence I was given throughout my research.

I would also like to thank my supervisory team, Mrs. Gordana Collier and Dr. Andrew Bradley, for supporting me, providing guidance, and enabling the research directions I investigated throughout my time under their supervision. I would also like to thank Dr. Paul Henshall for providing guidance throughout my academic development and providing feedback on this thesis.

I would like to acknowledge the support and collaborative environment I received from the High Voltage and Energy Storage Group members. The development of this group and its corresponding expansion provided me with support, knowledge, and an outlet to scope my research direction.

I would like to thank Katie Lukow for standing with me throughout this work and giving me affection, love, and a partner to share the journey with. She has been an immense source of strength and admiration throughout the path from undergraduate to doctorate.

Lastly, I would like to thank my parents and family for encouraging me throughout this journey. They have provided me with a constant source of support and guidance, especially enabling my professional career from Canada to the United States and now, finally, to the United Kingdom.

Publications

Partial work in this thesis has been published in the following journal articles,

[1] Brady Planden et al. “A computationally informed realisation algorithm for lithium-ion batteries implemented with LiiBRA.jl”. In: *Journal of Energy Storage* 55 (Nov. 2022), p. 105637. ISSN: 2352152X. DOI: 10.1016/j.est.2022.105637. URL: https://linkinghub.elsevier.com/retrieve/pii/S2352152X22016255 (visited on 09/19/2022)

[2] Taejin Jang et al. “BattPhase—A Convergent, Non-Oscillatory, Efficient Algorithm and Code for Predicting Shape Changes in Lithium Metal Batteries Using Phase-Field Models: Part I. Secondary Current Distribution”. In: *Journal of The Electrochemical Society* (Aug. 3, 2022). ISSN: 0013-4651, 1945-7111. DOI: 10.1149/1945-7111/ac86a7. URL: https://iopscience.iop.org/article/10.1149/1945-7111/ac86a7 (visited on 08/23/2022)

For [1], I contributed to the publication’s theory, software development, and writing. In [2], I contributed to the publication through the creation of an alternative software package enabling performance improvements.

Abstract

The fast adoption of battery electric vehicles (BEV) has resulted in a demand for rapid technological advancements. Strategic areas undergoing this development include lithium-ion energy storage. This is inclusive of electrochemical design improvements and advanced battery management control architectures. Field objectives for these developments include but are not limited to, reductions in cell degradation, improvements in fast charging capabilities, increases in system-level energy densities, and a reduction in energy storage costs. Improvements in online predictive models provide a path for realising these objectives through informed control interactions, reduced degradation effects, and decreased vehicle costs. This thesis contributes to these developments through improvements in fast physics-informed battery models for both lithium-ion and lithium-metal batteries.

The key novelty presented is the improvement of real-time, physics-based electrochemical model generation for lithium-ion batteries. A computationally informed realisation algorithm is developed and expands on the previously published realisation algorithm methods. An open-source Julia-based architecture is presented and provides a high-performance implementation while maintaining dynamic language capabilities for fast code development, and readability. A performance improvement of 21.7% was shown over the previous discrete realisation algorithm, with an additional framework improvement of 3.51 times when compared to the previously published framework. A methodology for the creation and modification of the reduced order models via in-vehicle hardware is presented and validated through an ARM-based model generation investigation. This addition provides a versatile method for cell degradation prediction over the battery life and can provide an interface for improved prediction of cell-to-cell variations. This methodology is applied to intercalation-based NMC/graphite batteries and is both numerically and experimentally validated.

A further element of novelty produced in this thesis includes advancements in lithium-metal phase-field representations through the creation of a Julia-based numerical framework optimised for high-performance predictions. This framework is then utilised as a ground truth model for the development of an autoregressive physics-informed neural solver aimed to predict lithium-metal evolution. Through the implementation of the physics-informed neural solver, a reduction in the numerical prediction time of 40.3% compared to the underlying phase-field representation was achieved. This methodology enables fast lithium-morphology predictions for improved design space explorations, online deployment, and advancements in electrodeposition material discovery for lithium-metal batteries.

Table of contents

List of figures

List of tables

Nomenclature

Constants

F	Faraday's constant
R	Universal gas constant

Superscripts

$\hat{\ }$	Fourier coefficent operator

Subscripts

n	Negative domain
p	Positive domain
s	Separator domain

Other Symbols

a_k	Solid surface area density, $k \in \{n, s, p\}$
β	Jacobsen-West transfer function parameter
$c_{e,k}$	Electrolyte lithium concentration, $k \in \{n, s, p\}$
$c_{s,k}$	Solid lithium concentration, $k \in \{n, s, p\}$
D_e	Electrolyte Diffusivity

D_s	Solid Diffusivity
ϵ_k	Electrolyte volume fraction, $k \in \{n, s, p\}$
F_{tf}	Transfer function sampling frequency
$\mathcal{H}_m$	Number of Hankel columns
$\mathcal{H}_n$	Number of Hankel rows
I_{app}	Applied current density, $[A/m^2]$
$i_{e,k}$	Ionic current density, $k \in \{n, s, p\}$
κ_e	Electrolyte conductivity
L_k	Domain length, $k \in \{n, s, p\}$
M	State-Space System order
m_k	Reaction rate, $k \in \{n, s, p\}$
$N_{e,k}$	Electrolyte molar flux, $k \in \{n, s, p\}$
$\phi_{e,k}$	Electrolyte potential, $k \in \{n, s, p\}$
$\phi_{s,k}$	Solid potential, $k \in \{n, s, p\}$
r	Unitless radial dimension
R_{ct}	Charge Transfer Resistance, $k \in \{n, p\}$
R_{film}	Film Resistance, $k \in \{n, p\}$
R_s	Particle radius
$S_{e,m}$	Number of spatial electrolyte locations
$S_{s,m}$	Number of spatial electrode locations

$\sigma_{e,k}$	Solid conductivity, $k \in \{n, s, p\}$
T_{len}	Transfer function sampling length
T_s	Final system sampling time
t^+	Transference number
T	Cell Temperature
$U_{k,ref}$	Reference open circuit potential, $k \in \{n, s, p\}$
x	Cell coordinate location across cell
z	Unitless dimension across electrode

Chapter 1

Introduction

1.1 Research Motivation

As climate impacts are accelerating, worldwide governing bodies are looking to reduce carbon emissions through expansion in renewable energy, emission taxes, and rebates for low-carbon vehicles, among other mechanisms. One such method is the adoption of battery electric vehicles (BEV), which have gained market share as passenger vehicle emission standards have increased across the globe [3]. As these standards become difficult with internal combustion engine (ICE) vehicles, automotive manufacturers have turned to hybrid-electric (HEV) and pure battery electric production. BEVs store energy from grid energy solutions such as hydroelectric, gas-turbine, photovoltaic, wind turbine, nuclear-fission, and coal combustion in onboard electrochemical batteries. This onboard storage of grid energy results in the life-cycle carbon footprint of BEVs varying dependent on the local grid generation mixture [4]. Due to this coupling, low carbon generation is required to minimise the life-cycle footprint and is projected to increase with renewables meeting 80% of the global electricity demand during the next decade [5]. As the grid-generated carbon footprint decreases, the individual BEV footprint trends towards a minimum governed by the vehicle manufacturing process [6, 7]. Figure 1.1 provides a predictive comparison of life-cycle carbon footprints for internal combustion, hybrid, and battery electric vehicles in the EU market from 2021 to 2038. As low carbon grid generation and electric vehicle manufacturing mature, the carbon footprint is expected to decrease further [8]. Furthermore, with current grid generation data (2021) and manufacturing capabilities, utilisation of BEVs provide a reduction in fleet emissions, satisfying the automotive emissions standards and providing consumers with a lower carbon footprint alternative [8].

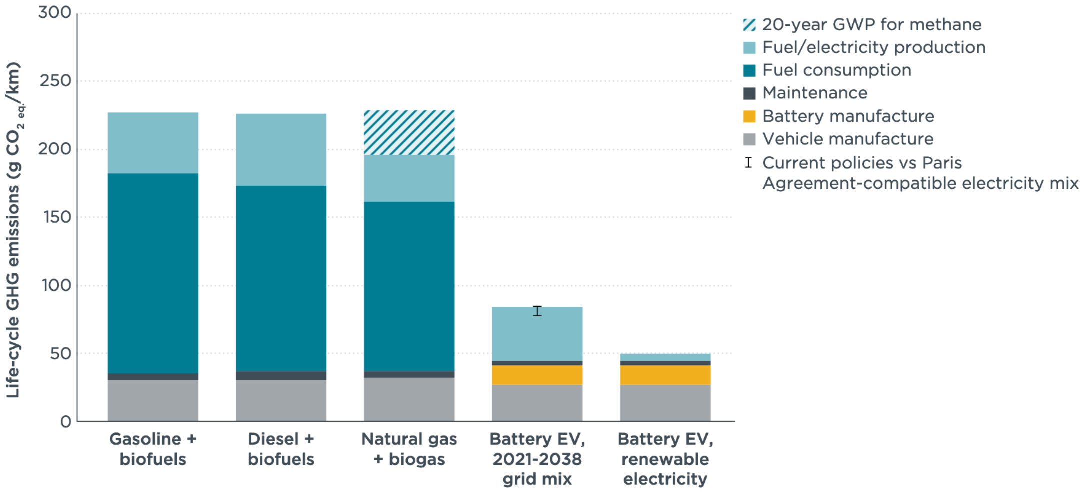


Fig. 1.1 Comparison of predicted life-cycle carbon emissions of conventional combustion powertrain and battery electric small segment vehicles from 2021-2038 in the EU. This figure is reproduced from the ICCT white paper: "A Global Comparison of the Life-Cycle Greenhouse Gas Emissions of Combustion Engine and Electric Passenger Cars 2021" [8]

New challenges present themselves as the automotive industry transitions from internal combustion to battery electric, requiring research attention to ensure consumers experience a seamless transition. One such challenge is the current onboard energy deficiency BEVs have compared to conventional combustion vehicles. This deficiency stems from the large gap between usable energy density in current lithium-based batteries and conventional petrol [9]. This currently results in higher mass vehicles and/or lower vehicle range depending on the design decisions made in the vehicle concept phase. As well, current electrochemical storage methods are charge power limited, resulting in major challenges in reducing the minimum time for charging [10]. Lastly, the electrochemical storage experiences capacity degradation over the vehicle's lifetime, which provides a societal challenge in the transition from internal combustion, as it does not experience this form of degradation. Furthermore, this degradation occurs through multiple electrochemical and mechanical mechanisms, providing a difficult problem to constrain and achieve immediate improvements.

The current mechanism to minimise electrochemical degradation is the battery management system (BMS). This system ensures safe, reliable operation of the onboard energy storage and acts as a supervisory control structure. This control system is conventionally developed to prioritise safe operation over battery performance. The system monitors observable information such as temperature, current, and terminal potential to determine acceptable power limits of the onboard battery. This structure provides a stable, safe interface to the vehicle control unit but does little to optimise the vehicle's performance. Integration of internal state predictions has been previously shown to provide performance and safety improvements to the BMS [11, 12]; however, achieving accurate electrochemical state predictions can be computationally expensive for conventional BMS hardware resulting in increased vehicle manufacturing costs via high-performance computational hardware. This is mainly due to the modelling complexity of the electrochemical system, with highly non-linear dynamics needing to be predicted.

1.1.1 Thesis Aims

The aims of this thesis are separated into two main concepts. The first is to provide a robust method to acquire the internal state information for current chemistries and future developments at a deployable computational cost. Achieving this aim enables the ability to accurately model and predict the electrochemical states of the battery, in real-time, on low-cost hardware, which enables an improved understanding of the safe operating limits of the cells; thus, higher levels of energy can be extracted without exceeding safe operating limits. This improvement in extracted energy would directly enable increased vehicle range, performance and safety. Inside this overarching aim, this work aims to provide an improved methodology advancing the field towards improvements in onboard battery utilisation for future transportation sector electrification.

The second aim of this thesis is to develop methods to enable fast internal state prediction of high-energy next-generation batteries, advancing current challenges in achieving cycle life requirements for automotive applications (>800 cycles [13]). This includes the development of improved numerical frameworks for conventional numerical solvers, as well as the implementation of neural solvers for low-computation electrochemical predictions. By enabling fast electrochemical modelling of lithium-metal batteries, this thesis provides an incremental improvement for future BEVs through a reduction in vehicle mass, extended range, and lower costs which will be assessed from an application perspective via an introduced multi-scale battery design framework.

The remainder of this chapter presents the necessary background and literature assessment of lithium-based batteries needed for later chapters. Finally, an overview of the structure of the thesis is presented.

1.2 Lithium-Based Battery Mechanics

This section presents the underlying operating principles for lithium-based batteries and introduces the reader to the necessary knowledge for the work presented in later chapters. Due to the variety in lithium-based chemistries, this section aims to limit the material introduced to a high level, with a further literature review provided in the following chapter.

1.2.1 Lithium-ion Mechanics

The operation of an intercalation-based lithium-ion battery involves ion transportation between a porous positive electrode (cathode) to a porous negative electrode (anode) through an electrolyte medium as shown in Figure 1.2.

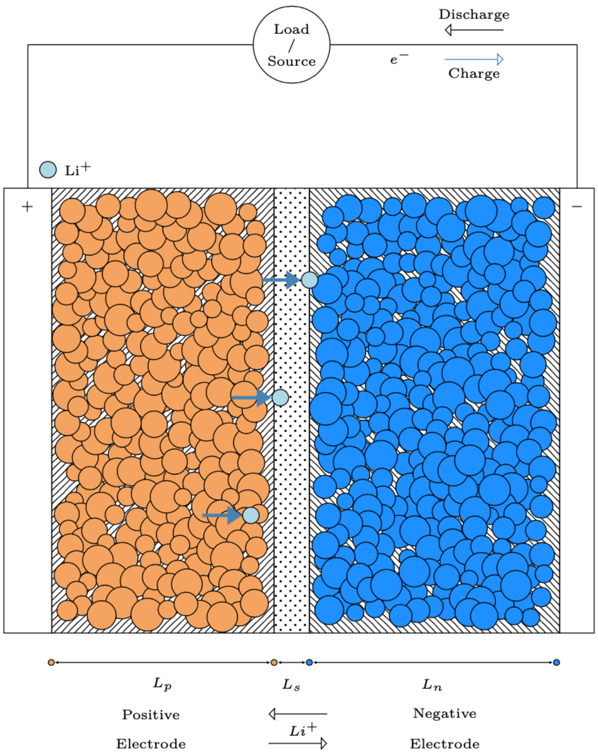


Fig. 1.2 Lithium-ion unit cell with relative thickness scale to a production LG Chem M50 21700 cylindrical battery undergoing a charge operation. Ion transfer (light blue) is shown with the corresponding electron transfer direction (light blue).

A nonconductive porous separator is placed between the two electrodes enabling ionic transport while providing the required electrical insulation. As an external electrical load is applied to the current collectors of the cell, lithium is oxidised at the negative electrode, and an electron and ion pair is produced. The electron travels through an external circuit towards the positive electrode. The lithium ions likewise travel through the porous separator via the electrolyte to the positive electrode, where reduction occurs. For liquid electrolyte cells, the electrodes are immersed in the electrolyte, thereby increasing the ion diffusivity from porous electrode to porous electrode. The process is inverted when the external load is replaced with an electrical source, with Figure 1.2 providing a visual example of the lithium-ion cell under this condition.

Material selection for each component provides a mechanism for tailoring the final cell to specific applications, with common compromises between battery lifetime, capacity, energy, and power. The electrochemical dynamics can be engineered for different applications by modifying the composition and physical geometry of these electrochemically active components. Current common cell geometries include cylindrical, prismatic, coin, and pouch, with the expansion of these common geometries in formats such as blade [14] and structurally integrated [15]. The four common geometries are shown in Figure 1.3.

Each of these formats provides applications specific benefits, such as the high volume densities and low-cost capabilities for the cylindrical format; however, they also have compromises, such as the higher nominal impedance for the same conventional cylindrical format. In addition to cell geometries, electrochemical design is commonly a process of optimising micro-scale parameters for a predetermined operational characteristic. For example, by increasing electrode thickness while maintaining constant porosity, the reversible capacity can be increased [17]; however, this commonly results in higher electrochemical polarisation and a decrease in the capable power output of the cell.

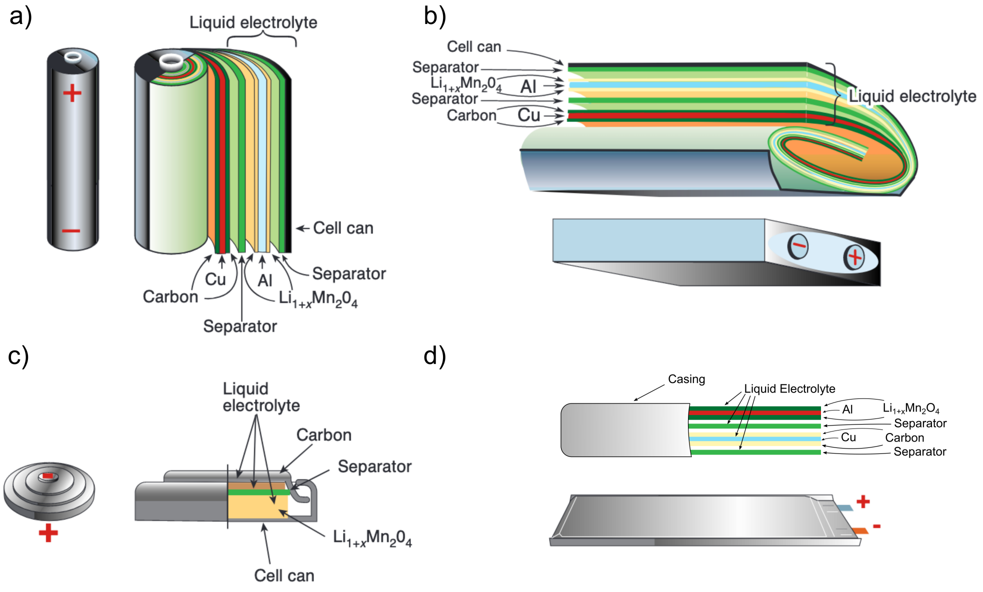


Fig. 1.3 Visual representation of a) Cylindrical, b)Prismatic, c) Coin, and d) Pouch geometries. This figure is an adaptation from "Issues and challenges facing rechargeable lithium batteries" [16].

1.2.2 Lithium-ion Chemistries

With the development of the lithium titanium disulfide ($LiTiS_2$) cathode by Goodenough et al. [18] in 1980, a commercially viable cathode material was available for rechargeable lithium-based batteries that were capable of reversible (de) insertion of lithium ions. The creation of the lithium cobalt oxide, $LiCoO_2$ (LCO) cathode enabled the wide adoption of intercalation lithium-ion batteries [19]. This discovery resulted in approximately twice the energy density compared to the $LiTiS_2$ chemistry due to the increase in nominal potential and the improvements in the reversibility of lithium insertion at low cathode concentrations. From that point forwards, improvements in cathode active material composition have resulted from multiple years of academic and industrial research.

Lithium cobalt oxide provides a very capable solution and is still commonly used in portable electronics [20]. This composition is comprised of a rock-salt layered structure and is capable of a specific capacity of approximately 170 mAh/g with a high nominal potential of 3.8V; however, due to its low thermally stability of 200°C and large environmental and humanitarian impacts from the high percentage of cobalt, alternative compositions are actively researched. Nickel and manganese additions to the cobalt cathode have resulted in hybrid compositions known as nickel-cobalt-manganese (NCM). Depending on the molar ratio between these species, higher specific energy can be achieved, with NCM811 (8:1:1) providing up to 200 mAh/g, with other common compositions being NCM111 (1:1:1) and NCM622 (6:2:2). These hybrid compositions have reduced the required cobalt in lithium-ion batteries; however, increased nickel ratios have shown a decrease in thermal stability, with NCM811 being thermally stable up to 232°C in comparison to NCM111 at 306°C [21].

As these hybrid compositions enable higher specific capacities, with similar nominal potential (3.7V vs 3.8V), an energy density improvement over the conventional LCO composition is commonly experienced. This increased energy density, coupled with the lower cobalt requirement, has resulted in NCM compositions commonly used in transportation applications such as automotive and aerospace. Alternatively, tunnel-like octahedral (olivine) structured compositions such as $LiFePO_4$ (LFP) provide high thermal stability with a specific capacity of approximately 165 mAh/g; however, due to a 3.45V nominal potential, the energy density is reduced compared to the previously discussed layered structure compositions. These olivine structured compositions offer lower environmental impacts than the layered structure compositions due to the lack of cobalt with improved worldwide access to the active materials; as such, they are heavily utilised in grid-storage applications and low-cost automotive BEVs. Table 1.1 below presents a list of common active material compositions currently utilised in various applications. As previously stated, these compositions provide different characteristics, such as safe operating temperatures, specific capacity, and rated cycle life. This table is not meant to be exhaustive but to reference the current state of cathode compositions.

Table 1.1 Common Cathode Material Compositions [22]

Short Name	Composition	Nominal Potential [V]	Specific Capacity Theoretical / Experimental $[mAh \cdot g^{-1}]$	Volum. Capacity $[mAh \cdot cm^{-3}]$
LTS	$LiTiS_2$	1.9	225 / 210	697
LNO	$LiNiO_2$	3.8	275 / 150	1280
LCO	$LiCoO_2$	3.98	274 / 160	1363/550
NMC111	$LiNi_{0.33}Mn_{0.33}Co_{0.33}O_2$	3.84	280 / 160	1333/600
NMC811	$LiNi_{0.8}Mn_{0.1}Co_{0.1}O_2$	3.84	280 / 215 [23]	- / 960 [24]
NCA	$LiNi_{0.8}Co_{0.15}Al_{0.05}O_2$	3.84	279 / 199	1284/700
LFP	$LiFePO_4$	3.45	170 / 165	589

Anode material compositions have equally been developed to improve battery lifetime, specific capacity, and specific energy. On large, graphite composition has been commonly utilised in most commercial applications. This is mainly due to graphite's low nominal potential with reference to Li/Li^+, thermal stability, and strong ionic rate capabilities. Graphite has a theoretical specific capacity of 372 mAh/g with storage as one lithium ion per six carbon atoms as shown in its lithiated atom structure of LiC_6 and provides a low expansion ratio during lithiation of 10% [25]. However, due to its low reference potential to lithium, under high rates, the storage mechanism can change from intercalation to plating when a combination of negative local overpotential and graphite surface saturation [26]. Alternative compositions such as lithium titanate (LTO) and silicon/graphite are also being utilised. Specific capacity improvements can be obtained by doping graphite with a small molecular percentage of silicon. This is realised through the high alloying ratio of silicon to lithium ions, i.e. 1 silicon atom per 4 lithium ions. High volume expansion is commonly experienced when utilising pure silicon as an anode and is commonly minimised through doping graphite to achieve a hybrid composition. Alternatively, LTO offers low volume expansion and high ionic rate capabilities with minimal capacity degradation; however, the full cell energy density is greatly reduced due to the high Li/Li^+ potential. Finally, storage as pure lithium has received research interest in the last decade, specifically due to the high theoretical specific capacity (3860 mAh/g); however, as introduced in the next section, challenges in achieving high cycle life as well as high lithium costs are currently hindering this objective. A list of commonly utilised anode compositions are displayed in Table 1.2 below.

Table 1.2 Anode Material Composition [22, 27]

Short Name	Composition	Specific Capacity $[mAh \cdot g^{-1}]$	Lithiation potential [V]	Delithiation potential [V]	Diffusion Const (cm^2s^{-1})	Volume Delta
Graphite	LiC_6	372 [28]	0.07,0.10, 0.19	0.1, 0.14, 0.23	$10^{-11} - 10^{-7}$	10%
Silicon	c-Si	2800 [29]	0.05, 0.21	0.32, 0.47	$10^{-13} - 10^{-11}$	270%
CuSn	$Cu_6Sn_5 - Sn$	1020 [30]	0.4, 0.57, 0.69	0.58, 0.7, 0.78	$10^{-16} - 10^{-10}$	255%
LTO	$Li_4Ti_5O_{12}$	175 [31]	1.55	1.58	$10^{-12} - 10^{-11}$	0.20%
NWO	$Nb_{16}W_5O_{55}$	225[32]	1.0	3.0	$2.1 \times 10^{-8} - 1.7 \times 10^{-9}$	5.5%

Furthermore, the requirements of the separator are to prevent the electrodes from forming short circuits while providing high ionic conductivity for ion transportation. Commonly, polypropylene and polyethene thin film compositions are utilised. Next, lithium salt compositions dissolved in organic carbonate solvents are commonly utilised for liquid electrolyte batteries. As the electrolyte is required to maintain a robust, stable medium for ionic transport, alternatives are consistently being investigated to improve ionic conductivity, dielectric constants, and transference number while firstly maintaining a feasible manufacturing cost and achieving viscosity, volatility, and operational temperature constraints. A common electrolyte composition is 1M of lithium hexafluorophosphate ($LiPF_6$), combined with ethylene carbonate (EC) and diethyl carbonate (DEC) at a volumetric ratio of 1:2; however, electrolyte development is an active field with improved compositions currently being discovered.

1.2.3 Lithium-Metal

Lithium-metal batteries provide an attractive solution for advancements in battery energy density while reducing potential manufacturing costs. These batteries utilise pure electrodeposited lithium as storage in place of the porous negative electrode conventionally used in the insertion-based lithium-ion batteries described above. The development of lithium-metal batteries started in the 20th century; however, it was replaced by intercalation-based lithium-ion due to thermal safety issues and poor cycle life. Research interest in lithium-metal batteries has recently expanded due to the trend towards higher energy requirements for electrified transportation. Lithium-metal batteries can be split into two sub-groups: anode-free and metal anode. An active material anode is omitted during manufacturing for anode-free batteries, with pure lithium-metal depositing on the current collector during the formation cycle. Anode-free solutions offer the highest available energy density and specific capacity (3680 mAh/g); however, due to the lack of lithium reservoir, they are commonly susceptible to failure mechanisms and have short cycle lifetimes [33, 34, 35]. The metal-anode alternative utilises a lithium metal foil alternative as a lithium reservoir from the first cycle. It can provide higher cycle lifetimes if the loss of lithium is the predominant failure mechanism. Due to the high cost of lithium, minimising the lithium reservoir is widely beneficial, as it also improves the energy density of the battery. Figure 1.4 presents a visualisation of a lithium-metal unit cell during charging operation, with ionic transport in the liquid electrolyte enabling lithium anode electrodeposition.

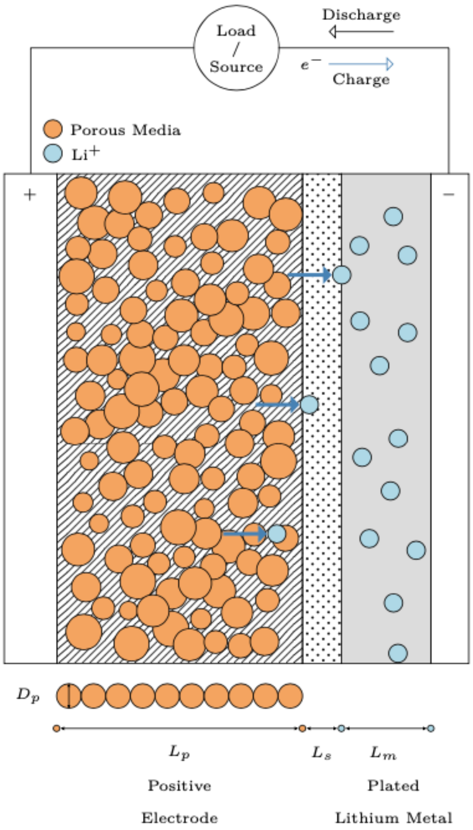


Fig. 1.4 Lithium-metal unit cell with scale representative LG Chem. M50 cathode and separator, and 27.5µm thick electrodeposited lithium anode.

Further challenges with lithium-metal batteries stem from the plated storage mechanism, specifically the potential for the lithium morphology to tend toward mossy dendritic growth. Previous investigations have found that this dendritic, tree-like growth can result in lithium punctures in the separator, causing internal short circuits [36, 37]. Additional to this short circuit failure mode, the cycling efficiency can be greatly impacted by this dendritic lithium [38, 39, 40]. These tree-like structures can also become electronically isolated during discharge cycling through disconnection from the current collector. Recent studies have investigated methods to reconnect this isolated lithium through increases in stripping current [39]; however, a definite path to reverse this failure mechanism is unclear. It appears that the most successful method is to avoid the creation of mossy lithium formation, of which success has been found in increased stack pressures [40, 41], optimising the charge/discharge rate ratio [35], and dual-salt electrolyte compositions [42].

1.2.4 Battery Chemistry: A Multi-Scale Optimisation Problem

Many fields benefit from fast, robust electrochemical models, as previously discussed. Another field currently heavily reliant on experimental discoveries is battery design and manufacturing. The development of future material compositions can greatly benefit from predictive models aimed at improving system understanding and design-space explorations. Experimental fabrication and electrochemical characterisations actively lead next-generation electrochemical material discovery. This is largely due to the difficulty in creating high-fidelity multi-scale electrochemical models capable of effectively predicting atomistic scale interactions at mesoscale applications. The Faraday Institute multi-scale modelling project [43] is one such consortium aimed at addressing this disconnect. This research aims to discover the underpinning parameters required for connecting differing electrochemical scales in a robust, identifiable method. A visual representation of the different length scales and their connections across the field is shown in Figure 1.5.

While the work in this thesis does not aim to solve the multi-scale challenge, the presented modelling framework in Chapter 4 is aimed toward providing insight into multi-scale interactions. This framework includes a physics-based representation that provides the underlying data requirements for future physics-informed data-driven approaches. Additionally, through the work discussed in Chapter 4, a software package for high-level decision and scoping which utilises multi-scale parameters is presented.

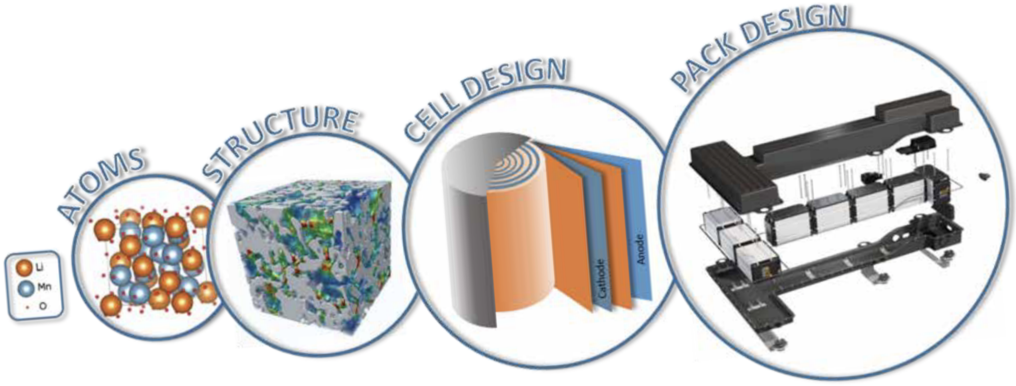


Fig. 1.5 Visual representation of the multiple scales in electrochemical modelling, from atomistic scale defining atom interactions and crystal orientations to full-vehicle or grid attached energy storage pack design. This figure is reproduced from the Faraday Institute Multi-Scale Modelling Project [43].

1.3 Fast, Robust Electrochemical Modelling

Battery management systems are utilised in electrochemical energy storage to control system demands, minimise long-term degradation, and meet pack safety constraints. The BMS will vary in control structure, system sensors, and embedded hardware; however, an overarching similarity of these systems is the gathering of system information, processing and filtering of this information, and then applying a control decision from the processed information. These control systems are developed on real-time embedded hardware ensuring decision-making is consistent and independent of system operating states. Accurately capturing the current state of operation for a lithium-based battery is an ongoing research area, as it is difficult to guarantee predictions across all operational conditions; however, the need for accurate predictions is growing as the demand for longer cycle-life, power performance, and cost reductions in the onboard energy storage increases.

The current state of the art for onboard battery modelling depends on the application field and the requirement of the system; however, with advancements in optimal control strategies, improved online prediction models have expanded.

Three main model frameworks are actively utilised and investigated for online battery management systems, data-driven, physics-based, and a hybrid combination. However, each modelling type provides capabilities to accurately predict the online energy storage state with differing creation requirements.

Data-driven approaches have conventionally utilised electrical component-based representations, such as equivalent circuits [44, 45, 46, 47] representing the cell through common electrical components such as resistors and capacitors. As the name suggests, data-driven or empirical models require observed data of the system in question. This data can be generated from higher order models, often known as digital twins, or through prior experimental measurements of the system, both of which can be in the time domain, frequency domain, or a combination of both [48, 49, 50]. Acquiring this data can be resource expensive, as it often requires specialised equipment and extensive experimental protocols over months and sometimes years.

Alternatively, physics-based models provide insight into the battery's underlying states, providing an attractive solution for optimal control strategies aimed at improving cycle-life and maximising performance [11, 12]. The creation of these models is numerically complex compared to the conventional data-driven approach and requires multiple experimentally captured physical parameters to guarantee fidelity. Additional methods to reduce the experimental data requirements for this approach have been investigated and included system identification through frequentist, and bayesian approaches [51, 52].

Finally, a hybrid approach that includes both underlying physical state information and data-driven approaches has recently become of interest to the field [53, 54, 55]. This framework can provide extrapolation guarantees unavailable for pure data-driven approaches while offering the state information that the physics-based models provide. Challenges with this framework include minimising the data requirements to ensure resource costs are competitive and developing robust methodologies that ensure the final model correctly captures the underlying system dynamics.

System information for data synchronisation and state estimation in models is obtained through multiple system sensors, including terminal potential, current, and temperature sensing. This information is gathered spatially in the energy storage pack at varying frequencies depending on the system dynamics. Due to the cost and system complexity, reducing sensing equipment in the pack is desirable. Additional system challenges, such as reducing management systems' available information, are introduced by removing this equipment. The lack of state knowledge can be mitigated by augmenting this reduction of sensing equipment with high-fidelity battery models. This can also provide system benefits, such as reductions in system faults and overall energy storage mass, while providing additional system information and improved safety decisions.

1.4 Thesis Outline

Chapter 1 (Introduction) introduces the research motivation for this thesis and presents background on lithium-based battery operation, the current state-of-art of online battery modelling for battery management systems and the corresponding challenges. Lastly, the need for fast predictive models to support research and design operations is described. This chapter includes an overview of the originality and publications created in this thesis.

Chapter 2 (Literature Review) summarises the modelling methodologies utilised in this thesis and introduces a methodology for high-level design decisions in lithium-based battery development. This chapter presents the current state-of-the-art in reference to the aims and objectives presented in the previous chapter. Finally, this chapter frames the motivation for fast, order-reduced methods, which are further expanded in the following chapters.

Chapter 3 (Fast, deployable, lithium-ion models) presents the computationally informed discrete realisation algorithm capable of generating physics-informed reduced-order models. This chapter introduces a modelling framework capable of creating offline and online reduced order models, providing a novel degradation-capable, real-time physics-informed solution. This chapter also validates the presented framework for both experimental data as well as the full-order system. Lastly, this chapter deploys the generated models onto an embedded target and verifies real-time capabilities with additional surrogate model coupling for fast linearisation.

Chapter 4 (Multi-scale Battery Modelling) acts as a bridged chapter and introduces the benefits of multi-scale models for system design while providing the reader with the required research motivation for Chapter 5. This chapter investigates differing storage mechanisms for lithium-based batteries and introduces a software package capable of capturing microscale parameter effects on full-scale vehicle drive-cycles. An example is investigated that utilises a high-performance motorsport application to predict system-level characteristics.

Chapter 5 (Message-Passing Graph Neural Networks for Lithium-Metal Modelling) introduces a phase-field model developed to numerically predict lithium-metal anode evolution during stripping and plating for differing initial geometry. This chapter also discusses the completed numerical optimisation to advance the phase-field representation. A message-passing graph neural network framework is presented to capture underlying physical dynamics in the moving boundary lithium-metal anode. This chapter dives further into data-driven modelling methods, intending to retain underlying physical information for global optimisation and design capabilities.

Chapter 6 (Conclusion) summarises the work presented in this thesis and the novelty presented. This chapter also discusses future work and the limitations of the presented work.

Chapter 2

Lithium-based battery models

Modelling lithium-based batteries provide the capability to improve system understanding. Further, by improving real-time capable physics-informed models, this is extended embedded control systems for improved performance. This chapter introduces the battery models, challenges, and assessment of gaps in the literature for this thesis. This chapter will assess the current methods in fast, physics-informed battery modelling while introducing the concepts needed for the work presented in future chapters. Once the gaps in the literature have been discussed, an assessment of the novelty presented in this thesis will be presented.

Section 2.1 overviews lithium-ion cells and introduces the need for reduced-order battery models. It first provides a background on various models before diving further into the derivation of the models utilised in this thesis. Section 2.1.2 introduces the Doyle-Fuller-Newman (DFN) model and its multi-particle representation of the electrochemical dynamics. The DFN is utilised as the basis for order reduction in the work presented in Chapter 3. The model derivation, with a discussion of the computational requirements for the DFN, is presented.

Section 2.2 overviews lithium-metal-based cells and the requirement for new modelling methods as advancements in cell design, manufacturing, and deployment are completed. Section 2.2.2 provides an introduction and review of physics-informed machine learning methods. This section assesses previous implementations in the literature and discusses the current gaps for micro-scale models. This section also provides the reader with the background needed for the work presented in Chapter 5.

Finally, Section 2.3 concludes the assessed work, with a discussion on the gaps with Section 2.3.1 summarising the novelty introduced in the remainder of this thesis.

2.1 Lithium-ion models

With the rapidly increasing adoption of battery electric vehicles (BEVs), expansion of in-vehicle battery modelling and control are required to improve safety and driving performance while ensuring the vehicle battery pack reaches the desired lifetime. Providing a viable method for capturing real-time degradation mechanisms coupled with physics-based electrochemical models is a key achievement required for future electric vehicle advancements [56]. This coupling benefits from meaningful physical states captured during online predictions that correspondingly map to degradation modes that can be experimentally validated. These degradation modes have corresponding mechanisms with recent literature aiming to quantify and identify models that enable high-fidelity predictions. Combining this, reduced-order physics-based models that predict battery degradation will enable a control mechanism to further optimise and mitigate battery degradation during operation. To achieve this target, battery models capable of being deployed onto battery control systems with electrochemical information are required. As previously discussed, battery management systems ensure hardware limits are maintained while providing the requested interaction from the operator. These systems accomplish this by ensuring the pack is in a safe state for operation, protecting the individual cells from abuse, and reducing the battery pack degradation over the lifetime of operation. This is often achieved through online predictive models for hardware limit forecasting, plant-based control structures, and state estimation.

Data-driven models such as an equivalent circuit model (ECM) [47, 57, 58] are commonly utilised for online prediction as they provide reasonable performance and have a well-established path for model creation. These models are numerically deployed onto in-vehicle embedded systems and provide information to the BMS that typically would not be attainable via direct sensing methods. This information is provided at designated non-flexible time intervals to the onboard control strategy with key performance indicators such as state-of-power (SOP), state-of-charge (SOC), and state-of-health (SOH) calculated through onboard measurements. Each of these state variables provides insight into the vehicle's capabilities for future operation. These models can provide a fast, reliable solution; however, the creation requires existing data that encompasses the entire operating range of the cell to ensure a stable response to the predicted operating conditions. Obtaining this data is time-consuming, of the order of multiple months to years of test channel time, and also requires expensive test equipment. These models also lack electrochemical generality due to their nature and the model data requirements needed to achieve acceptable performance. For example, ECMs utilise idealised, theoretical electrical components to represent cell behaviour, whose properties are numerically calibrated so that the model output is consistent with only a few basic measured cell characteristics, such as terminal voltage [59]. As such, generality isn't achievable for cell characteristics across varying chemistries, geometries, and operating conditions. Additionally, without observing internal electrochemical states during data acquisition, insight into these properties of the cell is not available, making predictions of long-term battery pack degradation difficult and inaccurate [56, 60, 61, 11, 62]. For many OEM's that are looking to have a vast array of performance from commuter vehicles to high-performance sports cars, this lengthy process is needed for each separate cell and pack configuration they wish to utilise.

An alternative to data-driven models is physics-based models, such as the Doyle-Fuller-Newman Pseudo-2D (DFN) [63, 64] or the Single Particle Model (SPM) [65]. These models provide internal electrochemical insight and can offer a viable solution for degradation-sensitive next-generation lithium-ion cells such as nickel-manganese-oxide (LNMO) or high-voltage NMC chemistries [66, 67, 68]. This coupling is mathematically complex and requires knowledge of multiple physical parameters, which can be difficult and/or expensive to obtain. The information provided by these models is beneficial for theoretical development, cell design iteration and development, as well as pack design and validation. This work enables advancements at a lab-based level where computational load and time are available; however, it is not currently a reasonable solution for onboard deployment. Furthermore, accurate long-term predictions are within these model's capabilities, providing coupling for cell degradation mechanisms such as intercalation electrode lithium plating, loss of active material (LAM) and loss of lithium inventory (LLI), pore-clogging, and dendrite growth [56, 68]. Due to the beneficial information provided by physics-based models, work has been completed to reduce the numerical complexity and computational performance requirements.

Simplifying the partial differential equations governing the system is one such method and has resulted in the SPM and its electrolyte capable (SPMe) form [65]. Additional methods include, Padé approximations [69], residue grouping [70, 71], and parabolic solid-phase diffusion approximations [72]. Further reduction is required to achieve deployment on battery management systems; however, one such method reduces the partial differential equations to continuous-form transfer functions combined with eigensystem realisation algorithms [73, 58, 74]. Likewise, Jin et al. developed a reduced-order capacity-loss model for graphite anodes that focused on only the most significant degradation mechanisms to improve computational efficiency [75]. Similarly, Han et al. developed a reduced order lumped electrochemical-thermal cell model by applying a state space approach to transform partial differential equations into ordinary differential equations [76]. These reductions aim to deploy capable predictive models to battery management systems and are heavily numerically reduced. The reduction of these models provides a computationally reasonable model that provides electrochemical information capable of providing this additional insight beyond the capability of conventional ECM. However, these methods have not achieved in-vehicle model generation; that is to say, these models are conventionally formed offline and deployed on the embedded targets via standard deployment tools. To achieve a degradation-capable implementation of these reduced order models, this thesis will present improvements that enable model generation in-vehicle, thus showcasing the potential for updated parameterisation over the energy storage lifetime.

Beyond model fidelity and computational cost, the parameterisation of these models is a critical requirement that recent literature has provided improvements [77, 78, 79, 80, 81, 82]. The parameterisation of physics-based models requires further insight than conventional data-driven methods such as the ECM. Due to the micro-scale parameter definitions, standard parameterisation methods include pre/post mortem deconstruction of the cell, half-cell electrochemical experiments, and physical measurements at the micro-scale through scanning electron microscopy. Alternative methods for parameterising the physics-based models have been investigated that minimise the experimental requirements through statistical inference and parameter fitting [51]. These methods offer a viable solution to achieve physical parameterisations; however, due to the nature of this methodology, validation of the corresponding model should be completed to avoid low model fidelity outside of the experimental fitting domain.

As discussed above, there are numerous benefits to deploying these reduced-order electrochemical models onto a BMS, including improved accuracy for predictions in SOC, SOP, and SOH [83, 84]; however, to ensure robust, stable operation of the BMS, the deployed model needs to be real-time capable for the given hardware. This requirement is fulfilled if the online model can be solved before the BMS is required to communicate the solution or provide a control interaction. Depending on the application, this solution rate can have requirements as low as 1 Hz to upwards of 10 Hz in fast dynamic systems. Therefore, the final reduced model must be capable within these ranges to be seen as a viable solution.

2.1.1 Equivalent circuit model

As previously discussed, data-driven battery representations are commonly used in battery management systems due to the ease of parameterisation and low computational requirements. One such model, the equivalent circuit model (ECM), is commonly used throughout the industry to predict the electrical voltage response of the lithium-ion battery. This model represents the electrochemical reactions in the battery as pure electrical components, such as resistors, capacitors, and voltage sources. Each of these components requires parameter estimation for accurate fitting and corresponding predictions. Due to this requirement, experimental testing of the physical battery, with a procedure representing the final prediction application, is commonly completed. Figure 2.1 below shows a visual representation of the ECM.

The above figure introduces the two-branch equivalent circuit model, which utilises two parallel resistors-capacitor pairs, C_1, R_1 and C_2, R_2, to predict the electrical dynamics of the system. A series resistor, R_0, is introduced to capture the ohmic resistance of the cell, with a voltage source term, OCV, commonly

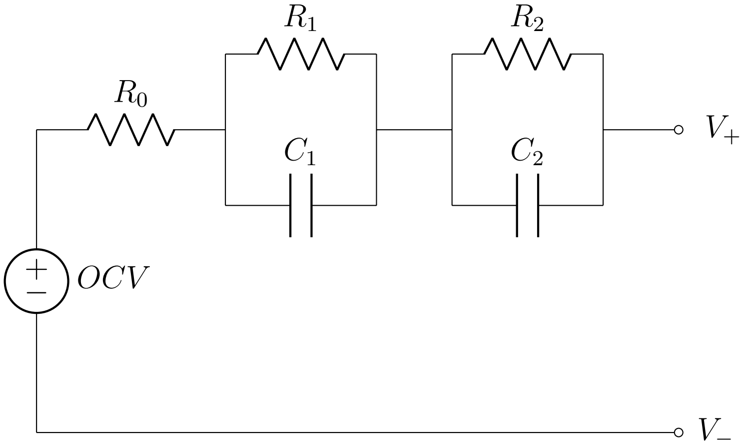


Fig. 2.1 Equivalent circuit model with two resistor-capacitor branches.

modelled via polynomials with a functional input as battery state-of-charge providing the open-circuit potential or through simplistic look-up tables. Finally, each parallel resistor-capacitor branch represents diffusion voltage, with additional branches added to capture higher complexity dynamics. The terminal battery voltage is computed through the following equation,

$$V_t = OCV(\theta) - \sum_{i=1}^{j} R_n i_n - R_0 i_0, \tag{2.1}$$

where θ denotes the battery state of charge, the polarisation due to each resistor-capacitor pair is captured through a summation operator of the corresponding resistor, R_n and loop current, i_n for each resistor-capacitor pair. Finally, the pure ohmic polarisation of the cell is defined by R_0 and i_0. Common methods for parameter fitting of R_0, C_n, R_n for this model include, particle-swarm optimisation [57, 85], least-square [86, 87], and genetic algorithms [86, 88]. As equivalent circuit models are commonly utilised in real-time control applications, they will be utilised as a benchmark for the reduction methods presented in Chapter 3.

2.1.2 Doyle-Fuller-Newman Model

Perhaps the most common continuum-order physics model is the Doyle-Fuller-Newman (DFN) model. This model, first presented in two main publications [63, 64], has become a popular choice for electrochemical battery modelling due to its ability to capture multi-scale electrochemical processes within a lithium-ion cell utilising multi-particle interactions. The DFN describes electrochemical electrodes of scale $\sim$ 100µm and active material particle size of scale $\sim$ 1µm. These length scales are modelled one-dimensionally and coupled to produce a pseudo-two-dimensional model space, often alternatively known as the "P2D" model. The geometry captured includes three domains: the positive electrode, the negative electrode, and the separator, with electrolyte spanning all three domains. As previously discussed in Section 1.2.1, under discharge, the negative electrode oxides lithium ions into the liquid electrolyte, with the positive electrode reducing those ions into the porous media. This process is reversed for charging and is the basis behind a reversible electrochemical battery.

This model introduces charge transfer reactions distributed throughout the unit cell thickness $x \in [0, L]$ with intercalated lithium diffused through the spherical domain $r \in [0, R_k]$ where subscript k defines the corresponding component domain, i.e. $k \in \{n, s, p\}$ where n represents the negative electrode, s the separator, and p the positive electrode domain. Mathematically solving this system across the domains above results in most of the differential equations spanning the unit cell thickness, except lithium migration in the solid phase active material being defined via diffusion and solved at a microscale length dependent on the average pore size for each electrode. Figure 2.2 below presents a diagram describing the unit cell. The multi-particle representation is shown with the corresponding particle diameter relating to the average pore size in the porous electrodes denoted.

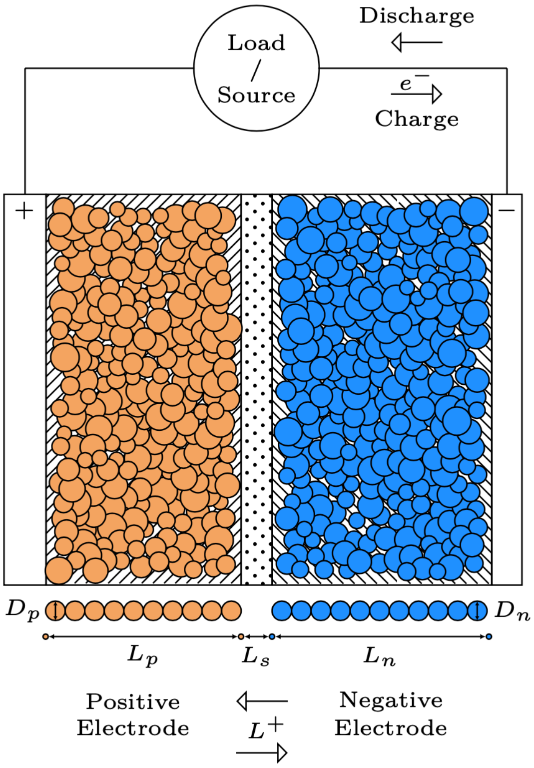


Fig. 2.2 Unit cell with Doyle-Fuller-Newman representation and corresponding electrical circuit. Stack thickness is scaled from an LG M50 lithium-ion battery[1]

2.1.2.1 Solid-phase lithium transport

Lithium transport occurs through (de)intercalation in the active material and migration in the electrolyte governed by concentration gradients. For the active material, porous electrode theory is utilised to capture the transport, as well as charge balance at the solid electrode-electrolyte interface (SEI) and charge transfer kinetics. This theory defines the porous electrodes as a superposition of three states: electrolyte, electrochemically active material and non-active material such as binders [89]. This solid state lithium (de)intercalation is governed through

Fickian diffusion and is presented in equation (2.2) below.

$$\frac{\partial c_{s,k}}{\partial t} = \frac{1}{(r)^2}\frac{\partial}{\partial r}\left(D_{s,k}(r)^2\frac{\partial c_{s,k}}{\partial r}\right), \tag{2.2}$$

where the solid-state diffusion coefficient is $D_{s,k}$ and is conventionally acquired through experimental testing methods such as galvanic intermittent titration testing (GITT). The mean active particle radius is introduced as r, representing the corresponding mean pore size for the given domain. Through the spherical particle symmetry, the boundary condition at $r = 0$ for the above diffusion equation is defined as,

$$\left.\frac{\partial c_{s,k}}{\partial r}\right|_{r=0} = 0. \tag{2.3}$$

Additionally, the boundary condition at the outer particle surface $r = R_s$ is defined through the inhomogeneous Neumann condition,

$$-D_{s,k}\frac{\partial c_{s,k}}{\partial r}\bigg| r = R_s = \frac{-j_k}{F}, \tag{2.4}$$

where j_k represents the lithium flux between the solid phase active material and the electrolyte, and F is the Faraday constant.

2.1.2.2 Lithium transport in the electrolyte

The domains for the DFN are separated into solid-phase active-material and electrolyte. Concentration solution theory describes the ionic species migration for lithium transportation in the electrolyte domain. This is completed through the relation of the electrochemical potential gradient to the mass flux [90]. The above, combined with the differentiated applied current density source term, gives the electrolyte mass balance equation (2.5), capturing the ionic concentration evolution.

$$\frac{\partial(\epsilon_k c_{e,k})}{\partial t} = \frac{\partial N_{e,k}}{\partial x} + \frac{1}{F}\frac{\partial i_{e,k}}{\partial x}, \tag{2.5}$$

where ϵ_k defines the volume fraction of active material in the porous electrode. $N_{e,k}$ introduces the molar flux of the electrolyte species, and $i_{e,k}$ defines the ionic current density in the electrolyte. Additionally, the boundary conditions defining the electrolyte concentration between the separator and the active material domains are,

$$c_{e,n}|_{x=L_n} = c_{e,s}|_{x=L_n}, \tag{2.6}$$

$$c_{e,p}|_{x=L_n+L_s} = c_{e,s}|_{x=L_n+L_s}. \tag{2.7}$$

The electrolyte molar flux, $N_{e,k}$, can then be represented through ionic diffusion at the interface between solid-phase active material and electrolyte as well as the ionic current density as shown in equation 2.8 below,

$$N_{e,k} = \epsilon_k^b D_e(c_{e,k}) \frac{\partial c_{e,k}}{\partial x} + \frac{t^+ i_{e,k}}{F}, \tag{2.8}$$

and is subject to homogenous Neumann boundary conditions describing the molar flux at each current collector as well as a symmetric constraint at the solid-phase interface,

$$N_{e,n}|_{x=0} = 0, \tag{2.9}$$

$$N_{e,p}|_{x=L} = 0, \tag{2.10}$$

$$N_{e,n}|_{x=L_n} = N_{e,s}|_{x=L_n}, \tag{2.11}$$

$$N_{e,p}|_{x=L_n+L_s} = N_{e,s}|_{x=L_n+L_s}. \tag{2.12}$$

2.1.2.3 Solid and electrolyte phase potential

For the solid-phase potential, the interfacial charge balance describes the applied current density in the porous electrode via Ohm's law and is presented as,

$$I - i_{e,k} = \sigma_k \frac{\partial \phi_{s,k}}{\partial x}, \tag{2.13}$$

with σ_k as the solid-state conductivity, and $\phi_{s,k}$ denotes the solid state potential, and I is the applied current density. Similarly, in the electrolyte, a modified ohm's relation governs the ionic transport created from local concentration variations and is commonly known as the diffusion overpotential,

$$i_{e,k} = \epsilon_k^b \kappa_e(c_{e,k}) \left(-\frac{\partial \phi_{e,k}}{\partial x} + 2(1 - t^+) \frac{RT}{F} \frac{\partial}{\partial x} (\log(c_{e,k})) \right). \tag{2.14}$$

The above equation scales linearly with the conductivity in the electrolyte, κ_e, solid state active volume fraction ϵ_k, and logarithmically with the electrolyte ionic concentration, $c_{e,k}$. Additional dependencies include the transference number t^+, cell temperature T, electrolyte potential $\phi_{e,k}$, the Faraday constant F, and the universal gas constant R. Homogeneous Neumann boundary conditions for ionic transportation are defined as,

$$i_{e,n}|_{x=0} = i_{e,p}|_{x=L} = 0, \tag{2.15}$$

with the corresponding first order derivation defined as,

$$\frac{\partial i_{e,k}}{\partial x} = \begin{cases} a_k F j_k & k \in \{n, p\} \\ 0 & k \in \{s\}. \end{cases} \tag{2.16}$$

2.1.2.4 Charge kinetics

Coupled charge transfer kinetics in the electrode and electrolyte are captured via the Butler-Volmer relation (2.17) by providing a relation between exchange current density, $j_{0,k}$ and the overpotential difference, η_k between domains.

$$j_k = j_{0,k} \sinh\left(\frac{F\eta_k}{2RT}\right), \tag{2.17}$$

with η_k is defined as,

$$\eta_k = \phi_{s,k} - \phi_{e,k} - U_k^{\mathrm{ocp}} - j_k F R_{film,k}, \tag{2.18}$$

where $\phi_{s,k}$ defines the surface potential, $\phi_{e,k}$ is the electrolyte potential, and $U_{s,k}^{\mathrm{ocp}}$ is the solid-phase open-circuit potential. Finally, the exchange current density, $j_{0,k}$ is introduced as,

$$j_{0,k} = m_k (c_{s,k}|_{r=R_s})^{1/2} (c_{s,k,max} - c_{s,k}|_{r=R_s})^{1/2} (c_{e,k})^{1/2}, \tag{2.19}$$

where m_k is the corresponding reaction rate for the selected k subdomain. Finally, the cell terminal potential is defined as,

$$V = U_{n,p}^{\mathrm{ocp}} \ + \ \eta_p \ - \ \eta_n \ + \ \phi_{e,p} \ - \ \phi_{e,n} \ + \ \Delta\Phi_{n,p}, \tag{2.20}$$

with the electrode Ohmic loss differences defined as,

$$\Delta\Phi_{n,p} = (\phi_{s,x=L} - \phi_{s,x=L_n+L_s}) + (\phi_{s,x=0} - \phi_{s,x=L_n}). \tag{2.21}$$

2.1.2.5 System of differential-algebraic equations

This finalises the DFN derivation, with the introduced equations coupled together and numerically solved being the next step; however, before the numerical solving methods for the DFN are discussed, alignment of the total model domain is completed. To align the domains, the respective dimensionless species lengths are transformed into distinct ranges denoted as,

$$\gamma_n = [0, L_n], \tag{2.22}$$

$$\gamma_s = [L_n, L_n + L_s], \tag{2.23}$$

$$\gamma_p = [L_n + L_s, L]. \tag{2.24}$$

This allows for a summation of the total system variables defined below in this transformed domain.

Table 2.1 Doyle-Fuller-Newman system variables

$c_{e,n},\ \phi_{s,n},\ \phi_{e,n},\ i_{e,n},\ N_{s,n}$	$x \in \gamma_n$
$c_{e,s},\ \phi_{e,s},\ i_{e,s},\ N_{e,s}$	$x \in \gamma_s$
$c_{e,p},\ \phi_{s,p},\ \phi_{e,p},\ i_{e,p},\ N_{s,p}$	$x \in \gamma_p$
$c_{s,n}$	$r \in [0, R_n],\ \ x \in \gamma_n$
$c_{s,p}$	$r \in [0, R_p],\ \ x \in \gamma_p$

As the governing DFN system has now been defined, methods to numerically solve this highly coupled system can be investigated. As this system is two-dimensional in space, i.e. two spatial dimensions, it does not require an unprecedented amount of computational resources; however, this still provides challenges for embedded hardware, as discussed in the above chapter. Multiple approaches to solving this coupled partial differential system have been investigated in literature and include finite elements [91], finite difference [92], orthogonal collocation [93, 94, 95], and Chebyshev polynomials [96]. Popular frameworks for solving the DFN in the electrochemical field mainly feature open-source modelling packages such as PyBaMM [97], PETLION [98], and DUALFOIL [99]. Alternatives that require licensing are available through COMSOL, and MATLAB [100, 101]. Independent of the numerical package used to solve the underlying mathematical equations; the physics-based models require solving a coupled system of partial differential equations representing the electrochemical processes that occur during operation.

Additional complexity is associated with this model due to the parameterisation requirements shown in Table 2.2. While the parameterisation requirement can be costly, both financially and laboratory resources wise, the DFN provides spatial information across each domain not attainable from the single particle models while also maintaining accurate predictions at high ionic rates.

2.1.2.6 Single Particle Reduction

Additional development from the Doyle-Fuller-Newman model has been the creation of the reduced-order single particle model (SPM) [102, 103]. This representation assumes that the electrode dynamics can be captured through a single (average) particle, as opposed to the multi-particle representation of the DFN and as such, assumes the radial dimension, r, can be uncoupled from the spatial thickness dimension, x. Additionally, due to the single particle's limitations, spatial variations in electrode lithiation cannot be predicted. This reduced order representation also forgoes electrolyte dynamics, and as such, this model struggles to predict dynamics at high applied currents accurately. Overall,

Table 2.2 Doyle-Fuller-Newman Model Parameters

Positive Electrode		
	$L_p,\ \epsilon_p,\ r_p,\ U_p^{ocp},\ D_p,\ \sigma_p,\ \theta_p^{100},\ \theta_p^{0}$	$x \in [\gamma_p]$
	$c_{s,p}^{max},\ \alpha_p,\ k_p^{norm},\ R_p^{film},\ A_p$	$x \in [\gamma_p]$
Negative Electrode		
	$L_n,\ \epsilon_n,\ r_n,\ U_n^{ocp},\ D_n,\ \sigma_n,\ \theta_n^{100},\ \theta_n^{0}$	$x \in [\gamma_n]$
	$c_{s,n}^{max},\ \alpha_n,\ k_n^{norm},\ R_n^{film},\ A_n$	$x \in [\gamma_n]$
Electrolyte		
	$D_e,\ \kappa_e,\ C_e^{0},\ t^{+}$	$x \in [\gamma_n, \gamma_s, \gamma_p]$
Separator		
	$L_s,\ \sigma_s,\ D_e^{brug}$	$x \in [\gamma_s]$

this reduction to a single particle lowers the computational requirements at the exchange of limiting the accurate predictions to current rates up to 1C [104].

An alternative reduction of the DFN has been achieved to improve the fidelity of the SPM in higher applied current operation by maintaining the mathematical representation of the electrolyte during reduction. This alternative model is known as the single particle model with electrolyte (SPMe) [65]. This electrolyte representation has been shown to offer close performance to the DFN with a reduction in computation performance; however, it increases numerical complexity compared to the SPM as it requires solving a quasi-linear partial differential equation for the electrolyte ion concentration [105]. These prediction improvements have resulted in the large utilisation of this model for electrochemical degradation modelling; due to the high number of charge and discharge cycles, a computational reduction is attractive. However, during high-applied current operation, when electrolyte depletion occurs, this model deviates from the full-order DFN and has difficulty maintaining fidelity. Due to these applied current constraints for both single particle reductions, they have struggled to gain penetration in high-performance applications. In addition to the operation fidelity limitations, the numerical complexity of these reduced single-particle models still provides a challenge for implementation on low-cost embedded hardware. As such, they have conventionally required linearised state-space formation for embedded deployment [106].

2.1.3 Linear state-space representation

For control applications deployed onto embedded targets, low computation models are required to enable fast system response and stability. The prediction requirements are extremely stringent in critical applications such as safety systems for human interactions or high-frequency dynamic control with multiple system interfaces. A common representation of the system plant model in these applications is in the state-space form to ensure requirements are met. In this work, the plant model is defined as the online predictive model deployed onto the embedded target utilised for augmented information provided to the control structure. For linear systems, the continuous form state-space representation is defined as,

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}u \\ \boldsymbol{y} &= \mathbf{C}\boldsymbol{x} + \mathbf{D}u \end{aligned} \tag{2.25}$$

The A, B, C, and D variables denote arrays and vectors that represent the linear response from the system for a given state parameter. The state vector, $x \in \mathbb{R}^n$, represents the system's dynamic state variables within the defined subspace. The variable u represents the system input, which maps to the predicted system output y. The mathematical calculation is visually represented in Figure 2.3.

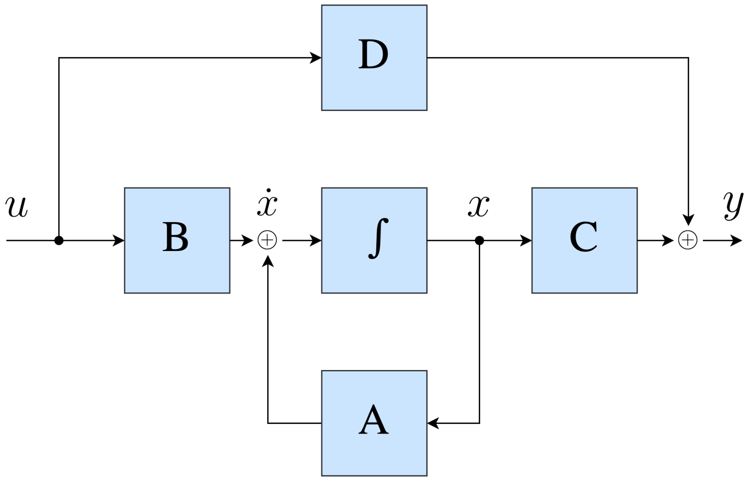


Fig. 2.3 Continuous state-space architecture. The system input, u and the state vector x, are utilised for output predictions of y.

The continuous form state-space representation is discretised for embedded deployments to enable fast numerical integration. This ensures fast system response without complex integration methods and transforms the above continuous form to,

$$\begin{aligned} \boldsymbol{x}[k+1] &= \boldsymbol{A}\boldsymbol{x}[k] + \boldsymbol{B}u[k] \\ \boldsymbol{y}[k] &= \boldsymbol{C}\boldsymbol{x}[k] + \boldsymbol{D}u[k] \end{aligned} \tag{2.26}$$

where this representation is commonly utilised for Kalman-filter estimation [107] and model-predictive control (MPC) [108] to satisfy the given control requirements. In this thesis, a discrete linear state-space formation of the DFN model is presented. This formation structures the output vector for predictions of the underlying DFN variables as presented in Table 2.1, with the applied current as the system input. This formation was selected as it provides an equivalent input-output structure to the DFN and reduces integration for end-users. Alternatively, non-linear state-space representations have been previously utilised to capture non-linear system dynamics. This can provide higher fidelity in non-linear systems at the exchange of high-order terms in the system reduction. Furthermore, Chapter 3 below provides evidence that a linear representation with in-situ re-linearisation accurately captures the electrochemical dynamics.

2.2 Lithium-metal models

Higher energy-dense batteries are needed as electrified transportation expands from passenger vehicles to heavy-duty transportation and aerospace applications such as fixed-wing and vertical take-off and landing aircraft. As previously discussed, one such method to achieve this requirement is anode-free lithium-metal cells, which utilise a porous positive electrode and electrodeposit lithium-ions on the negative current collector during cell charging. For liquid-electrolyte chemistries, the manufactured cell closely matches a conventional lithium-ion format; however, these cells currently face challenges in long-term coulombic efficiency due to active lithium evolution degradation. Multiple industrial companies, such as Cuberg, SES, Quatumscape, and Sion-Power, aim to provide solutions to this cycling inefficiency while maintaining the high energy density that lithium-metal offers. An alternative cell format, utilising a solid-state electrolyte, has been gaining research interest as it can offer improved operational safety [109, 110], the potential for higher cell voltages [110], and an improvement in cycling degradation through resilience to metal oxide reactivity [111]. For both of these formats, the current implementations are quite limited, with a usable cycle life of 500 cycles in ideal operating conditions; however, electrolyte composition investigations have shown promising results for expansion of this limitation in both cases [40, 111, 112].

Currently, most of the research and development of lithium-metal batteries is completed experimentally, as models commonly used for conventional lithium-ion chemistries cannot provide physical insight at a micro-scale level. This results in expensive, time-consuming research developments, which can provide a barrier for small to medium companies as well as academic research groups and results in lower user adoption. To address this issue, work from multiple research groups has been aimed at developing predictive models to improve understanding of electrolyte reactions in electrodeposition [113, 114, 115, 116] as well as understanding mechanical pressure effects [117, 118]. Recently, physics-based lithium-anode models have been developed to improve understanding of the lithium morphology evolution [119, 120] as it has a direct effect on the degradation mechanisms active in the lithium-metal battery. Most of these models aim to provide insight in combination with experimental research that enables a reduction in the invested development cost. These models provide a mechanism to capture lithium morphology; however, due to the solving requirement of the coupled PDE systems, they are conventionally computationally expensive. The work presented in this thesis aims to fill this gap through a fast coupled physics-informed machine-learning framework.

Further opportunities for lithium-metal models capable of providing physical insight include system-level battery management for safe operation. Previous literature has shown that electrodeposited lithium-metal has an increased thermal risk over conventional lithium-ion [121] due to the volatile combustion of the excess lithium under stressed operation. By enabling battery management systems with insight into the physical state of the cell, it should be possible to improve operational performance while maintaining safety.

2.2.1 Electrodeposition phase-field representations

As previously introduced in Section 1.2.3, lithium-metal batteries are one such method to achieve these higher energy densities to enable the feasibility of electrification in many sectors, such as aerospace. However, due to the underlying construction and storage mechanisms of lithium-metal batteries, it is difficult to utilise conventional modelling structures to capture dynamics due to the underlying assumptions of fixed-thickness porous electrode theory. The conventional particle-based models aim to spatially discretise the unit cell thickness, capturing lithium intercalation dynamics through the porous diffusion of lithium ions; however, in the electrodeposition cell, the negative electrode has varying thickness requiring ongoing updates to this spatial discretisation. This is especially difficult to resolve mathematically; without this capability, the lithium composition evolution and failure mechanisms are not represented.

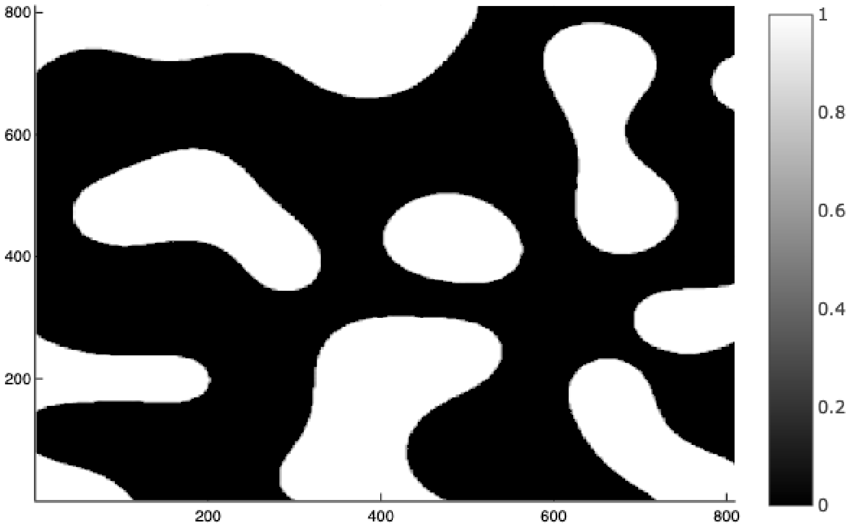


Fig. 2.4 Example phase-field representation, designating the two different phases as numerical values of 0 and 1. Modified and reproduced from [122].

One such method to accurately predict the varying thickness, lithium evolution is a phase-field representation, which is a popular technique in thermodynamics to describe a system of multiple states going through a phase transition. Phase-field representations model the interaction between phases (or states) through diffusion, thus removing sharp interface transitions. The numerical system is divided into a spatial representation of the phase domain, with values of range (0,1) designating the corresponding phase. For the liquid electrolyte lithium-metal anode system, this results in the electro-deposited lithium (solid) forming one phase and the electrolyte (liquid) forming the second, thus enabling the model to predict plating and stripping evolution through phase distinction, which is not possible with conventional lithium-ion methods such as the DFN, or it's reduced

SPM technique. As the phase field represents the interface between phases through a diffuse mechanism, it doesn't capture the solid-electrolyte interphase (SEI) in its standard definition. This has been addressed in recent work by Yurkiv et al. [123] to predict mossy lithium evolution while capturing the SEI effects. Throughout the development of phase-field models capable of predicting lithium morphology evolution for a lithium-metal battery (LMB), the aim is to advance performance in parallel with experimental testing and developments.

2.2.2 Neural PDE solvers

The usage of data-driven machine-learning techniques for lithium-based battery predictions has expanded in previous years, partially due to advancements in computational training hardware, open-source machine-learning frameworks, and the wide expanse of datasets available for model development. These advancements have enabled improvements in cycle degradation modelling for lithium-ion batteries [124, 125, 50], improvements in online state estimation [126, 127], and data-informed material selection for cell manufacturing [128, 129, 130, 131, 132]. In this section, an introduction to partial differential neural solvers will be presented as an alternative method for data-driven machine learning utilisation.

Additional usage for machine-learning techniques has been investigated in partial differential equation solving, commonly denoted as neural PDE solvers. These techniques aim to provide an additional capability to predict the solution of a partial differential system given a training set. This has multiple benefits over conventional solvers, such as fast implementation once the solver has been trained and low-memory requirements when deployed; however, the training cost to create a solver that has generality can be exceedingly expensive. These solvers are commonly divided into two subcategories. The first are neural operators, which aim to map from system inputs and initial conditions, u_0 to the given solution by utilising a neural operator, $\mathcal{M}$.

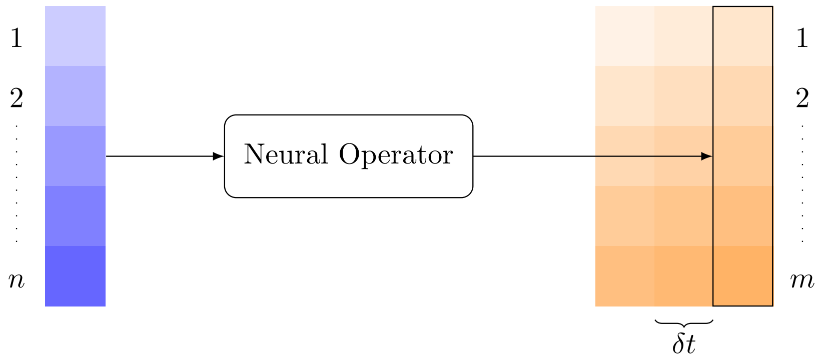


Fig. 2.5 Neural operator mapping initial conditions to system output for the trained time domain.

As these operators cannot generalise boundary, initial conditions, or system shape, their predictions are limited to the system used for training and cannot guarantee predictions outside that domain. One such operator is the physics-informed neural network (PINN), which has been heavily utilised across computer science and engineering fields such as material design [133, 134, 135], stochastic differential equations [136], uncertainty analysis [137, 138], and partial differential equations [139, 140]. Physics-informed neural networks achieve improved results by utilising network architectures that align with the partial differential equation structure; this enables improvements in fitting and performance [141]. These operators are trained to achieve the following equality,

$$\mathcal{M}(t, u_0) = u(t) \tag{2.27}$$

The second type of neural solver utilises autoregressive methods to capture the PDE solution evolution iteratively. This method enables the solver to interact similarly to conventional numerical solving methods. It provides a solution that is not constrained by a predetermined time set, allowing solutions at $\delta t + t$ to be found.

$$u(t + \delta t) = \mathcal{N}(\delta t, u(t)) \tag{2.28}$$

This type of operator provides flexibility over the PINN methods as they commonly encode boundary, initial, and shape information into the solver; however, difficulties in training have been reported, and as such, they are less common. Previously implementations of autoregressive methods have shown the benefit of hybrid-style neural solving, where the time integration aligns with conventional schemes such as the essentially nonoscillatory (ENO), weighted essentially nonoscillatory (WENO), and finite difference. Sinai et al. [142] present a methodology to solve the burger's equation utilising an autoregressive method that results in a time derivative representation similar to the WENO scheme. Greenfield et al. [143] approach multigrid PDE solving with an autoregressive neural network with an unsupervised loss function and showcase improvements over a widely utilised black-box method for selecting operators. Hsieh et al. [144] investigate an iterative finite element scheme for an autoregressive PDE solver with convergence guarantees. Finally, Brandstetter et al. [145] introduce a message-passing graph neural network framework for autoregressive solving of multiple PDE families and present a novel pushforward concept for improvements in training performance. The autoregressive framework provides a mechanism for fast design-space exploration once the underlying system dynamics have been trained, i.e. for trained system evolutions, the domain requirements can be limited to boundary conditions, further expanding the generality of this methodology. This provides a key improvement over the PINN method in design-space exploration, which could be exploited for advanced material discovery.

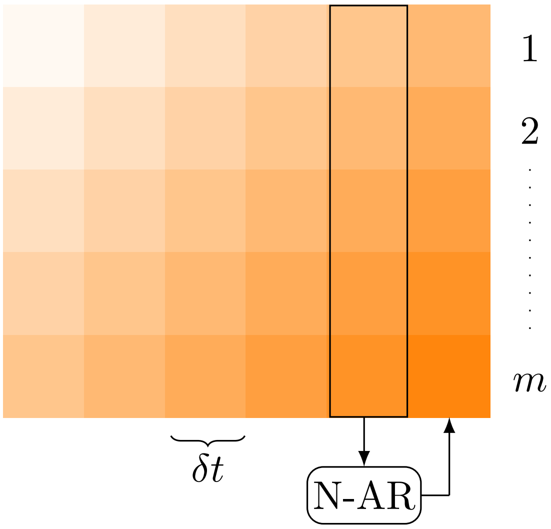


Fig. 2.6 Neural auto-regressive solver predicting the next time-step given the previous output and corresponding boundary and initial conditions.

Given the above neural solver methodologies and the electrodeposition models presented in the previous section, a direction for improvements in fast electrodeposition battery modelling is taking shape. Furthermore, as autoregressive neural methods provide a flexible solution with parallels to conventional time integration schemes, this area provides a logical step to advance the field. Coupling these methods and investigating the challenges posed is the aim of Chapter 5 while showcasing advancements in conventional numerical solvers alongside autoregressive neural solvers.

2.2.3 Scientific programming languages

This thesis presents multiple numerical frameworks with a common theme of high-performance implementations without sacrificing code comprehension. Commonly, interpreted languages such as MATLAB, Python, and R have been heavily used for scientific computing due to their vast libraries, packages, and support, as well as their ease of use for researchers, engineers, and statisticians. These languages offer improved dynamic typesets that enable users to develop high-level source code without worrying about performance optimisation completed underneath[1]. This stems from each language's underlying interpretation of the source code developed by users to the machine code needed to be numerically implemented on the hardware. While this offers multiple user improvements, this interpreter does not offer the same machine code optimisation achievable from a compiled language.

Alternatively, compiled languages such as C/C++, Rust, and Fortran offer high performance due to their compilation of source code to machine code; however, these languages commonly utilise a static typeset and require knowledge of compiler definitions. These languages offer improved numerical performance over the interpreted languages at the expense of user knowledge and additional compilation time before the program is executed. After initial development in interpreted languages, these languages are occasionally utilised for performance improvements for algorithms, packages, and production-ready scientific software.

Finally, there has been a recent push towards just-in-time (JIT) compilation languages as they can provide a performant compromise between the interpreted and compiled languages discussed above. These languages commonly provide benefits such as rich type information and multiple-dispatch, as well as improvements in code comprehension through dynamic language implementation, such as removing the need for user compiling. One such language, Julia [146, 147], is actively developed to provide numerical performance matching the static compiled languages while offering the dynamic language benefits provided in Python, MATLAB, and R. Furthermore, additional aims for this language include multiple-dispatch, enabling even further code reusability and deployment, installation across both x86 and ARM platforms, and straight-forward GPU deployment. Similarly, MATLAB's execution code and Python's Numba

---

[1] While these languages remove some of the optimisations from the user, there are still recommended methods to enable interpreter performance.

package [148] have both recently offered JIT capabilities enabling performance improvements to their pure implementations; however, due to the overhead and previous development goals of these languages, the additions are not as widely supported and seemingly provide performance improvements lower than a pure Julia implementation.

For most of the work presented in this thesis, the Julia language is utilised for package development. This work includes open-source implementations for the presented packages in Chapters 3, 4, and 5.

2.3 Concluding remarks

In this chapter, both lithium-ion and lithium-metal modelling methods aimed at improving performance and underlying system understanding have been introduced. In these methods, the need for improvements in reduced-order modelling is apparent, with benefits including degradation-capable in-situ parameterisation, design space exploration, and micro-scale lithium evolution predictions. Each of these areas benefits heavily from improvements in fast modelling methods; however, care needs to be taken to ensure the underlying physical states are not lost during the order reduction. The second gap that has been assessed relates to electrodeposition modelling, where current lithium-ion models cannot predict lithium evolution. In this case, new models need to be developed that enable insight while maintaining performance and understanding.

The lithium-ion-based DFN model introduces a multi-particle continuum model capable of providing physical state information for the lithium-ion battery. This model will likewise be used as the starting point in the next chapter for real-time capable order reduction to provide physical state information to battery management systems. Expanding on the assessed gaps in the literature, the phase-field methodology and auto-regressive neural solvers presented will be used as the basis for the work completed in Chapter 5 by coupling a ground-truth representation to a physics-informed message passing graph neural network.

In this thesis, the development of methods to improve both the onboard utilisation of these lithium-based batteries and the capability of providing insight into further design directions will be advanced. Both objectives align with fast, accurate modelling improvements as onboard utilisation needs to be deployable onto real-time hardware. In contrast, improvements in design decisions greatly benefit from fast computational solutions that lend themselves to global optimisation and design-space discovery.

2.3.1 Original work

The work presented in this thesis contains novelty published in [1, 2] and work presented at the Gordon Research Conference in Batteries 2022 [149]. This work also presents multiple novel approaches for creating fast, physics-informed battery models utilised in real-time applications and design space exploration. Additionally, this work presents a novel method for in-vehicle model creation that has not been seen in the literature. This method achieves a key improvement for physics-based electrochemical degradation prediction in real-time environments. The originality is as follows,

1. Creation and validation of a computationally informed subspace realisation algorithm capable of in-vehicle model generation. Including the development of a fast, open-source software package with improvements over the conventional discrete realisation algorithm. This work also introduces a method for improved low-frequency linearisation for online performance improvements on embedded hardware. The majority of this work was published in [1].

2. Creation of a multi-scale, high-level software package for electrochemical design decisions. This tool is designed to support design engineers with the initial scoping of an electrochemical energy storage system at the stage where high fidelity is not yet required. This work provides capabilities in moving between length scales while capturing trends across potential geometry and chemistry for lithium-based batteries. An example of a motorsport application is introduced, with an analysis of multiple battery chemistries investigated.

3. An open-source, Julia-based phase-field package for electrodeposition predictions in the lithium-metal anode is presented. This work includes investigating multiple time integration schemes and optimisation through Intel MKL's Pardiso linear algebra solver, memory optimisation, and dynamic language improvements. Performance verification of this work is completed by assessing an alternative Maple language implementation. This work was included in Jang et al. [2].

4. Development of a message-passing graph neural network for physics-informed predictions of phase-field-based lithium evolution in lithium-metal anodes. This work presents an auto-regressive neural solver architecture that has been expanded upon for both single and multi-dimensional phase-field lithium anode evolution predictions. Verification of this architecture is performed through the previously introduced phase-field implementation, with numerical performance also presented.

Chapter 3

Fast, deployable, lithium-ion models

The ability to model lithium-based batteries in real time enables improved performance for online control structures. This chapter presents a novel improvement to realisation algorithms for lithium-ion batteries. A software package, LiiBRA.jl, developed in Julia [147] is also presented for generating numerically reduced real-time capable physics-informed models. LiiBRA.jl enables adaptive physics-informed models to be implemented onto embedded systems, opening up the capabilities of battery management and control systems within energy storage by combining improvements in reduced-order models with the high performance of the Julia language. This work develops a computationally informed realisation algorithm aimed at fast solution generation and investigates in-vehicle model creation as a viable method for degradation-informed models. A sensitivity analysis is performed alongside numerical verification and experimental validation of the model predictions. The reduced-order models are then deployed onto an embedded target to validate real-time capabilities. Finally, improved online performance is presented via usage of neural surrogate models for low-frequency linearisation at varying applied currents, further improving model fidelity.

3.1 Electrochemical Transfer Functions

To achieve the required computational performance for in-vehicle model generation, the nonlinear governing equations of the DFN model shown in Chapter 2 will be reduced and linearised into transfer functions and transformed into a state-space representation. This enables decoupling of the DFN's governing equations for fast, independent predictions. The DFN is selected as the full-order model due to the improved fidelity in high applied current operations as discussed in section 2.1.2.6, the improved spatial resolution of solid-phase lithium concentrations [105], and localised electrolyte depletion predictions. The derivation of these transfer functions is shown below; however, the reader is pointed to Jacobsen and West [150], Smith et al. [151], and Lee et al. [152] for the original derivations. Six electrochemical transfer functions, the electrode surface concentration $C_{\text{surf}}(z,s)$, the electrode surface potential $\Phi_{s,e}(z,s)$, the reaction flux $J(z,s)$, the electrolyte potential $\Phi_e(z,s)$, the electrolyte concentration $C_e(z,s)$, and the electrode potential $\Phi_s(z,s)$ are derived. In this context, z is the spatial electrode location, such that in the negative electrode, $z = x/L_n$ and in the positive electrode, $z = L - x/L_p$. For both definitions, the outer electrode surface is defined as $z = 0$ with the separator surface corresponding to $z = 1$. It is visually displayed in Figure 3.1. Additionally, s is the complex Laplace domain variable, and t is the time domain representation.

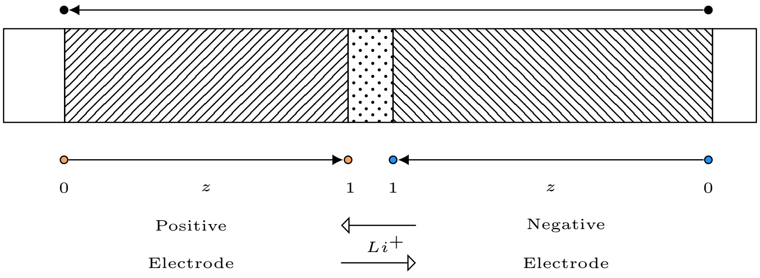


Fig. 3.1 Coordinate system for reduced-order models. z variable defines the unitless electrode position, with x defining the unitless electrolyte domain.

3.1.1 Electrode surface concentration, $C_{\mathbf{surf}}(z,s)$, surface potential, $\Phi_{s,e}(z,s)$, and reaction flux $j(z,s)$ transfer functions

The starting point for this derivation is linearising the Butler-Volmer equation shown previously in equation (2.17) above. This is completed by defining a system point to linearise around; for this work, this point is selected as electrochemical equilibrium, i.e. no system dynamics occurring as per,

$$\rho = [\phi_{s-e} = U_{\text{ocp}}(c_{s,0}), c_{\text{surf}} = c_{s,0}, c_e = c_{e,0}, j = 0], \tag{3.1}$$

where ϕ_{s-e} is defined as $\phi_s - \phi_e$, c_{surf} is the solid surface concentration, with $c_{s,0}$ and $c_{e,0}$ as the corresponding concentrations at time zero. Finally, ρ is introduced as the equilibrium linearisation point. Combining equations (2.17) and (2.19) and linearising about ρ produces,

$$\frac{j_k}{m_k(c_{s,k})^{1/2}(c_{s,k,max} - c_{s,k})^{1/2}(c_{e,k})^{1/2}} = \sinh\left(\frac{\eta_k F}{2RT}\right). \tag{3.2}$$

The left-hand side of the above can then be represented via a two-term Taylor expansion,

$$\begin{aligned} \text{LHS} &\approx \text{LHS}(\rho) + \left.\frac{\partial \text{LHS}}{\partial c_{\text{surf}}}\right|_{\rho}(c_{\text{surf}} - c_{e,0}) + \left.\frac{\partial \text{LHS}}{\partial c_e}\right|_{\rho}(c_e - c_{e,0}) + \left.\frac{\partial \text{LHS}}{\partial j_k}\right|_{\rho} j_k, \\ &\approx \frac{j_k}{j_0}. \end{aligned} \tag{3.3}$$

By expanding the right-hand side of equation (3.2) via equation (2.18) and similarly representing the expansion with a two-term Taylor series, the following result is obtained,

$$\text{RHS}|_{\rho} = \frac{F}{RT}\tilde{\phi}_{s,e} - \frac{F}{RT}\left[\frac{\partial U_{ocp}}{\partial c_{\text{surf}}}\right]\tilde{c}_{\text{surf}} - \frac{F^2 R_{\text{film}}}{RT} j_k, \tag{3.4}$$

where $\sim$ introduces the debiased parameter definition such that $\tilde{c}_e = c_e - c_{e,0}$, and $\widetilde{\phi}_{s,e} = \phi_{s-e} - U_{\text{ocp}}(c_{s,0})$. It is then possible to combine the LHS and RHS above and solve for $\widetilde{\Phi}_{s,e}$ to obtain the following linearised Butler-Volmer representation,

$$\widetilde{\Phi}_{s,e}(z,t) = \underbrace{F\left(\frac{RT}{F^2 j_0} + R_{\text{film}}\right)}_{R_{\text{tot}}} j(z,t) + \left[\frac{\partial U_{ocp}}{\partial c_{\text{surf}}}\bigg|_{c_{s,0}}\right]\widetilde{c}_{\text{surf}}(z,t), \tag{3.5}$$

where the charge transfer is defined as the $RT/(j_0F^2)$ component of R_{tot} in this thesis. This concludes the Butler-Volmer linearisation and allows us to move onto the surface concentration. First, by formally defining the solid surface concentration, $c_{\text{surf}}(z,t) = c_s(R_s,z,t)$ and it's corresponding debiased representation as,

$$\tilde{c}_{\text{surf}}(z,t) = \tilde{c}_s(R_s,z,t) = c_s(R_s,z,t) - c_{s,0}, \tag{3.6}$$

utilising this result, it is possible to modify equation (2.2) to correspond to the debiased representation such that,

$$\frac{\partial \tilde{c}_s(r,z,t)}{\partial t} = \frac{1}{r^2}\frac{\partial}{\partial r}\left(D_s r^2 \frac{\partial \tilde{c}_s(r,z,t)}{\partial r}\right) \tag{3.7}$$

and corresponding initial and boundary conditions defined as,

$$-D_s \frac{\partial \tilde{c}_s(r,z,t)}{\partial r}\bigg|_{r=R_s} = -j(z,t), \qquad \frac{\partial \tilde{c}_s(r,z,t)}{\partial r}\bigg|_{r=0} = 0. \tag{3.8}$$

A derived transfer function representing the form shown in (3.7) has been previously presented by Jacobsen and West [150] and is introduced as,

$$\frac{\widetilde{C}_{\text{surf}}(s)}{J(s)} = \frac{R_s}{D_s}\left(\frac{\tanh(\beta)}{\tanh(\beta) - \beta}\right), \tag{3.9}$$

with $\beta = R_s\sqrt{s/D_s}$. The electrode surface potential is derived below by applying a Laplace transformation to the derived linearised Butler-Volmer equation and the introduced surface concentration transfer function (3.9). First, by representing the solid potential equation (2.13) with the normalised spatial variable, z,

$$a_s F j = \frac{\sigma_{\text{eff}}}{L_n^2}\frac{\partial^2 \phi_s}{\partial z^2}, \tag{3.10}$$

with updated boundary conditions as,

$$\frac{\sigma_{\text{eff}}}{L_n}\frac{\partial \phi_s}{\partial z}\bigg|_{z=0} = -I_{app} \quad \frac{\partial \phi_s}{\partial z}\bigg|_{z=1} = 0. \tag{3.11}$$

Completing a similar transformation of domain for the Stefan-Maxwell electrolyte potential equation (2.14) results in,

$$-a_s F j = \frac{\kappa_{\text{eff}}}{L_n^2}\frac{\partial^2 \phi_e}{\partial z^2}, \tag{3.12}$$

with similar boundary conditions as,

$$\frac{\kappa_{\text{eff}}}{L_n}\frac{\partial \phi_e}{\partial z}\bigg|_{z=0} = 0 \quad \frac{\partial \phi_e}{\partial z}\bigg|_{z=1} = I_{app}. \tag{3.13}$$

Through subtraction of equation (3.10) and (3.12), the following the phase potential difference equation is obtained,

$$\frac{\partial^2}{\partial z^2}\phi_{s-e} = a_s F L_n^2\left(\frac{1}{\sigma_{\text{eff}}} + \frac{1}{\kappa_{\text{eff}}}\right) j, \tag{3.14}$$

the combined boundary conditions then result as,

$$\left(\frac{\sigma_{\text{eff}}}{L_n}\right)\frac{\partial \phi_{s-e}}{\partial z}\bigg|_{z=0} = \left(\frac{\kappa_{\text{eff}}}{L_n}\right)\frac{\partial \phi_{s-e}}{\partial z}\bigg|_{z=1} = -I_{app}. \tag{3.15}$$

As diffusion is captured through the radial domain, r and not the spatial domain, the C_{surf} in equation (3.5) is rewritten with a Laplace transformation completed,

$$\widetilde{\Phi}_{s,e}(z,s) = \left(FR_{tot} + \left[\frac{\partial U_{ocp}}{\partial C_{\text{surf}}}\bigg|_{c_{s,0}} \right] \frac{R_s}{FD_s} \left(\frac{\tanh(\beta)}{\tanh(\beta)-\beta} \right) \right) J(z,s). \tag{3.16}$$

Taking the Laplace transform of equation (3.14) and combining it with equation (3.16) results in,

$$\frac{\partial \widetilde{\Phi}_{s,e}(z,s)}{\partial z} = \frac{a_s F L_n^2 \left(\frac{1}{\sigma_{\text{eff}}} + \frac{1}{\kappa_{\text{eff}}} \right)}{R_{tot} + \left[\frac{\partial U_{ocp}}{\partial C_{\text{surf}}}\bigg|_{c_{s,0}} \right] \frac{R_s}{FD_s} \left(\frac{\tanh(\beta)}{\tanh(\beta)-\beta} \right)} \cdot \widetilde{\Phi}_{s,e}(z,s), \tag{3.17}$$

with corresponding boundary conditions as,

$$\left(\frac{\sigma_{\text{eff}}}{L_n} \right) \frac{\partial \widetilde{\Phi}_{s,e}(z,s)}{\partial z}\bigg|_{z=0} = \left(\frac{\kappa_{\text{eff}}}{L_n} \right) \frac{\partial \widetilde{\Phi}_{s,e}(z,s)}{\partial z}\bigg|_{z=1} = \frac{-i_{app}}{A}. \tag{3.18}$$

To improve reader comprehension and simplify future transfer functions, a dimensionless condensing variable is introduced as,

$$\nu(s) = L_n \sqrt{\frac{a_s\left(\frac{1}{\sigma^{\text{eff}}} + \frac{1}{\kappa^{\text{eff}}} \right)}{R_{\text{tot}} + \left[\frac{\partial U_{k,\text{ref}}}{\partial C_{\text{surf}}} \right] \frac{R_s}{FD_s} \left(\frac{\tanh(\beta)}{\tanh(\beta)-\beta} \right)}}, \tag{3.19}$$

which then produces the following representation of equation (3.17),

$$\frac{\partial \widetilde{\Phi}_{s,e}(z,s)}{\partial z} - \nu^2 \widetilde{\Phi}_{s,e}(z,s) = 0. \tag{3.20}$$

Finally, the above has a previously found solution [152] defined below after enforcing the initial conditions as,

$$\frac{\widetilde{\Phi}_{s,e}(z,s)}{I_{app}(s)} = \frac{L_n}{A\nu(s)\sinh(\nu(s))} \left(\frac{\cosh(\nu(s)z)}{\kappa^{\text{eff}}} + \frac{\cosh(\nu(s)(z-1))}{\sigma^{\text{eff}}} \right). \tag{3.21}$$

Next, the reaction flux transfer function can be found through algebraic manipulation of the surface potential transfer function (3.21) as,

$$\frac{J(z,s)}{I_{app}} = \frac{J(z,s)}{\widetilde{\Phi}_{s,e}(z,s)} \frac{\widetilde{\Phi}_{s,e}(z,s)}{I_{app}}, \tag{3.22}$$

where $J(z,s)/\widetilde{\Phi}_{s,e}(z,s)$ can be obtained from rearranging equation (3.16) above resulting in,

$$\frac{J(z,s)}{I_{app}} = \frac{\nu^2}{a_s F L^2(\frac{1}{\sigma^{\text{eff}}} + \frac{1}{\kappa^{\text{eff}}})} \cdot \frac{\widetilde{\Phi}_{s,e}(z,s)}{I_{app}(s)}. \tag{3.23}$$

By combining the results obtained in equation (3.21), it is possible to introduce the final form for the reaction flux for a given I_{app} input as,

$$\begin{aligned}\frac{J(z,s)}{I_{app}(s)} =& \left(\frac{\nu(s)}{a_s F L_n A(\kappa^{\text{eff}} + \sigma^{\text{eff}})}\right) \\ &+ \left(\frac{\sigma^{\text{eff}}\cosh(\nu(s)z) + \kappa^{\text{eff}}\cosh(\nu(s)(z-1))}{\sinh(\nu(s))}\right).\end{aligned} \tag{3.24}$$

Using the results from the previous steps, the electrode surface concentration transfer function can likewise be derived as,

$$\frac{\widetilde{C}_{\text{surf}}(z,s)}{I_{\text{app}}(s)} = \frac{J(z,s)}{I_{app}(s)} \frac{\widetilde{C}_{\text{surf}}(z,s)}{J(z,s)}. \tag{3.25}$$

By expanding the above right-hand side from results obtained above in equations (3.25) and (3.9), the transfer function can be finalised as,

$$\frac{\widetilde{C}_{\text{surf},n}(z,s)}{I_{\text{app}}(s)} = \left(\frac{\nu(s) R_s \tanh(\beta) \sigma^{\text{eff}} \cosh(\nu(s)\cdot z) + \kappa^{\text{eff}} \cosh(\nu(s)(z-1))}{a_s F L_n D_s A \sinh(\nu(s))(\kappa^{\text{eff}} + \sigma^{\text{eff}})(\tanh(\beta) - \beta)}\right). \tag{3.26}$$

For the positive domain, the above transfer functions are multiplied by -1. This concludes the derivation for the surface concentration and reaction flux transfer functions. In the next subsection, this methodology proceeds with the derivation of the solid potential transfer function, $\phi_s(z,s)$.

3.1.2 Transfer function for solid potential, $\Phi_s(z,s)$

The starting point for the surface potential is the solid charge transfer Laplace domain representation (3.14) which is reintroduced below,

$$a_s F j = \frac{\sigma_{\text{eff}}}{L_n}\frac{\partial^2 \phi_s}{\partial z^2}, \tag{3.27}$$

through integration and a Laplace transformation as introduced in Lee et al. [152], the corresponding solid potential Φ_s transfer function is,

$$\begin{aligned}\frac{\Phi_{s,n}(z,s)}{I_{app}(s)} =& -\frac{L_n \kappa^{\text{eff}}(\cosh((z-1)\nu(s)))}{A\sigma^{\text{eff}}(\kappa^{\text{eff}}+\sigma^{\text{eff}})\nu(s)\sinh(\nu(s))} \\ & -\frac{L_n \sigma^{\text{eff}}(1-\cosh(z\nu(s))+z\nu(s)\sinh(\nu(s)))}{A\sigma^{\text{eff}}(\kappa^{\text{eff}}+\sigma^{\text{eff}})\nu(s)\sinh(\nu(s))}.\end{aligned} \tag{3.28}$$

As above, the positive domain is represented through multiplication by -1.

3.1.3 Transfer function for electrolyte potential, $\Phi_e(x,s)$

For the electrolyte transfer functions, the normalised domain is represented by x and spans the unit cell length. To account for the differing subdomains captured inside of the introduced x domain, subscripts are used to differentiate between positive electrode (p), negative electrode (n), and the separator (s). The subscript k is used when the introduced equations can be utilised for each domain. The first step in the derivation for the electrolyte potential transfer function is integration of the electrolyte charge conservation equation (2.14) with respect to the introduced x domain,

$$\phi_e(x,t) - \phi_e(0,t) = \int_0^x \epsilon_k^b \kappa_e(c_{e,k})\left((1-t^+)\frac{2RT}{F}\frac{\partial \log(c_{e,k})}{\partial x}\right) - i_e(x,t)\;dx. \tag{3.29}$$

Splitting the above integrand into components and introducing a debiased electrolyte potential as $\widetilde{\phi}_e(x,t) = \phi_e(x,t) - \phi_e(0,t)$ produces,

$$[\widetilde{\phi}_{e,1}(x,t)]_1 = \int_0^x \frac{-i_e(x,t)}{\kappa_e} dx, \tag{3.30}$$

$$[\widetilde{\phi}_e(x,t)]_2 = \int_0^x \left((1-t^+)\frac{2RT}{F}\frac{\partial \log(c_{e,k})}{\partial x} \right) dx. \tag{3.31}$$

The summation of the two components produces the final transfer function as,

$$\frac{\widetilde{\Phi}_e(z,s)}{I_{\text{app}}(s)} = [\widetilde{\Phi}_e(z,s)]_1 + [\widetilde{\Phi}_e(z,s)]_2. \tag{3.32}$$

In parallel, it is possible to obtain a transfer function for the ionic current through integration of the reaction flux transfer functions presented in (3.24). For the negative electrode domain of x that results in,

$$\frac{I_{e,n}(x,s)}{I_{app}} = \frac{\sigma^{\text{eff}} \sinh\left(\frac{(L-x)\nu(s)}{L_n}\right) - \kappa^{\text{eff}} \sinh\left(\frac{(L_n-x)\nu(s)}{L_n}\right)}{A(\kappa^{\text{eff}} + \sigma^{\text{eff}}) \sinh(\nu(s))} + \frac{\kappa^{\text{eff}}}{A(\kappa^{\text{eff}} + \sigma^{\text{eff}})}. \tag{3.33}$$

Similarly, with the corresponding positive domain ionic current presented as,

$$\frac{I_{e,p}(x,s)}{I_{app}} = \frac{\sigma^{\text{eff}} \sinh\left(\frac{(L-x)\nu(s)}{L_p}\right) + \kappa^{\text{eff}} \sinh\left(\frac{(L_n+L_s-x)\nu(s)}{L_p}\right)}{A(\kappa^{\text{eff}} + \sigma^{\text{eff}} \sinh(\nu(s)))} + \frac{\kappa^{\text{eff}}}{A(\kappa^{\text{eff}} + \sigma^{\text{eff}})}. \tag{3.34}$$

Solving the first component of the electrolyte potential for the negative domain is possible through substitution of equation (3.30) and integrating. The results are presented as,

$$\begin{aligned}\frac{[\widetilde{\Phi}_{e,1}(x,s)]_1}{I_{\text{app}}(s)} &= \int_0^x \frac{-i_e(x,s)}{\kappa_e}dx \\ &= \frac{L_n(\frac{\sigma_n^{\text{eff}}}{\kappa_n^{\text{eff}}})(1-\cosh(\frac{x\nu_n(s)}{L_n})) - x\nu_n(s)\sinh(\nu_n(s))}{A(\sigma_n^{\text{eff}}+\kappa_n^{\text{eff}})\nu_n(s)\sinh(\nu_n(s))} \\ &+ \frac{L_n(\cosh(\nu_n(s)) - \cosh(\frac{(L_n-x)\nu_n(s)}{L_n}))}{A(\sigma_n^{\text{eff}}+\kappa_n^{\text{eff}})\nu_n(s)\sinh(\nu_n(s))}.\end{aligned} \tag{3.35}$$

For the separator domain, the first component of the transfer function becomes,

$$\frac{[\widetilde{\Phi}_{e,1}(x,s)]_1}{I_{\text{app}}(s)} = \frac{L_n - x}{A\kappa_s^{\text{eff}}} + \frac{L_n((1-\frac{\sigma_n^{\text{eff}}}{\kappa_n^{\text{eff}}})\tanh(\frac{\nu_n(s)}{2}) - \nu_n(s))}{A(\sigma_n^{\text{eff}}+\kappa_n^{\text{eff}})\nu_n(s)}. \tag{3.36}$$

Lastly, the first component of the positive domain transfer function is obtained as,

$$\begin{aligned}\frac{[\widetilde{\Phi}_{e,1}(x,s)]_1}{I_{\text{app}}(s)} =& -\frac{L_s}{A\kappa_s^{\text{eff}}} + \frac{L_n\left(\left(1-\frac{\sigma_n^{\text{eff}}}{\kappa_n^{\text{eff}}}\right)\tanh\left(\frac{\nu_n(s)}{2}\right) - \nu_n(s)\right)}{A(\kappa_n^{\text{eff}}+\sigma_n^{\text{eff}})\nu_n(s)} \\ &- \frac{L_p\left(1+\frac{\sigma_p^{\text{eff}}}{\kappa_p^{\text{eff}}}\cosh(\nu_p(s))\right)}{A(\kappa_p^{\text{eff}}+\sigma_p^{\text{eff}})\sinh(\nu_p(s))\nu_p(s)} \\ &+ \frac{L_p\cosh\left(\frac{(L_n+L_s-x)\nu_p(s)}{L_p}\right)}{A(\kappa^{\text{eff}}+\sigma^{\text{eff}})\sinh(\nu_p(s))\nu_p(s)} \\ &+ \frac{L_p\frac{\sigma_p^{\text{eff}}}{\kappa_p^{\text{eff}}}\cosh\left(\frac{(L-x)\nu_p(s)}{L_p}\right)}{A(\kappa^{\text{eff}}+\sigma^{\text{eff}})\sinh(\nu_p(s))\nu_p(s)} \\ &+ \frac{(L_n+L_s)-x}{A(\sigma_p^{\text{eff}}+\kappa_p^{\text{eff}})},\end{aligned} \tag{3.37}$$

and the second term in the electrolyte transfer function is determined by the value of $C_e(x,s)$ and is shown as,

$$[\widetilde{\Phi}_e(x,t)]_2 = \frac{2RT(1-t_+^0)}{F}\log\left(\frac{C_e(x,t)}{C_e(0,t)}\right). \tag{3.38}$$

This completes the electrolyte potential transfer function derivation with a two-component solution dependent on the spatial location in the x domain. For the numerical implementation, equation (3.35), (3.36), or (3.37) will be computed for $[\widetilde{\Phi}_e(z,s)]_1$ depending on the spatial location, with $[\widetilde{\Phi}_e(x,t)]_2$ computed from the $C_e(x,s)$ transfer function derived in the next section.

3.1.4 Transfer function for the electrolyte concentration, $C_e(x,s)$

Finally, to acquire the electrolyte concentration transfer function, the problem is split into homogenous and non-homogenous components as introduced by Lee et al. [152]. This allows for the homogenous component to be obtained through an orthonormal eigenfunction representation of the ϵ_e weighting function through the separation of variables method. The non-homogenous component performs a projection of the concentration function into ϵ_e to solve for Fourier coefficients. These are then used to derive the electrolyte concentration transfer function. First, the individual component definition of $\epsilon_e(x)$ is defined as,

$$\epsilon(x) = \begin{cases} \epsilon_n & 0 \le x < L_n, \\ \epsilon_s & L_n \le x < L_n + L_s, \\ \epsilon_p & L_n + L_s < x \le L, \end{cases} \tag{3.39}$$

likewise for $D_e^{\text{eff}}(x)$,

$$D_e^{\text{eff}}(x) = \begin{cases} D_{e,n} & 0 \le x < L_n, \\ D_{e,s} & L_n \le x < L_n + L_s, \\ D_{e,p} & L_n + L_s < x \le L. \end{cases} \tag{3.40}$$

3.1.4.1 Homogenous problem

The homogenous problem is presented below with component for $D_e^{\text{eff}}(x)$ and $\epsilon_e(x)$,

$$\frac{\partial c_e(x,t)}{\partial t} = \frac{1}{\epsilon_e(x)} \frac{\partial}{\partial x} \left(D_e^{\text{eff}}(x) \frac{\partial c_e(x,t)}{\partial x} \right), \tag{3.41}$$

with corresponding boundary conditions that enforce continuity between the cell subdomains of x with corresponding $D_e^{\text{eff}}(x)$ and $\epsilon_e(x)$ for each as,

$$\begin{aligned} c_e(L_n^-, t) &= c_e(L_n^+, t), \\ c_e((L_n + L_s)^-, t) &= c_e((L_n + L_s)^+, t), \\ D_{e,n} \frac{\partial c_e(L_n^-, t)}{\partial x} &= D_{e,s} \frac{\partial c_e(L_n^+, t)}{\partial x}, \\ D_{e,m} \frac{\partial c_e((L_n + L_s)^-, t)}{\partial x} &= D_{e,p} \frac{\partial c_e((L_n + L_s)^-, t)}{\partial x}. \end{aligned} \tag{3.42}$$

Furthermore, through the separation of variables, it is possible to split the spatial and time domains such that the form is represented as,

$$c_e(x,t) = \xi(x)\zeta(t). \tag{3.43}$$

Through algebraic rearrangement, it's possible to obtain the separated form as,

$$\frac{d}{dx} \left(D_e^{\text{eff}}(x) \frac{d\xi(x)}{dx} \right) = -\lambda \epsilon_e(x) \xi(x), \tag{3.44}$$

$$\frac{d\zeta(t)}{dt} = -\lambda \zeta(t), \tag{3.45}$$

where the number of solutions to the above equation is dependent on the eigenvalues, λ, that satisfy the relation; this results in an infinite set for this system. Therefore, the solutions depend on the specific eigenvalues, resulting in a variable addition of $\xi(x) \to \xi(x;\lambda)$ and $\zeta(t) \to \zeta(t;\lambda)$. This solution of (3.45) is obtained as,

$$\xi(t;\lambda) = \xi(0;\lambda)e^{-\lambda t}. \tag{3.46}$$

The solution of (3.44) is dependent on the subdomain and results in three different equations orignally presented in [152]. For the negative electrode domain,

$$\xi_n(x;\lambda) = k_1 \cos\left(\sqrt{\frac{\lambda\epsilon_n}{D_{e,n}}}x\right). \tag{3.47}$$

The separator solution is obtained as,

$$\xi_s(x;\lambda) = k_3 \cos\left(\sqrt{\frac{\lambda\epsilon_s}{D_{e,s}}}x\right) + k_4 \sin\left(\sqrt{\frac{\lambda\epsilon_{e,s}}{D_{e,s}}}x\right). \tag{3.48}$$

Finally, the positive electrode solution is,

$$\xi_p(x;\lambda) = k_5 \cos\left(\sqrt{\frac{\lambda\epsilon_p}{D_{e,p}}}x\right) + k_6 \sin\left(\sqrt{\frac{\lambda\epsilon_p}{D_{e,p}}}x\right). \tag{3.49}$$

It is then possible to solve for the variables k_1, k_3, and k_4 through the following linear algebraic system,

$$\begin{aligned}\begin{bmatrix} \cos\left(\sqrt{\frac{\lambda\epsilon_s}{D_{e,s}}}L_n\right) & \sin\left(\sqrt{\frac{\lambda\epsilon_s}{D_{e,s}}}L_n\right) \\ -\sqrt{\lambda\epsilon_s D_{e,s}}\sin\left(\sqrt{\frac{\lambda\epsilon_s}{D_{e,s}}}L_n\right) & \sqrt{\lambda\epsilon_s D_{e,s}}\cos\left(\sqrt{\frac{\lambda\epsilon_s}{D_{e,s}}}L_n\right)\end{bmatrix}\begin{bmatrix} k_3 \\ k_4 \end{bmatrix} \\ = k1\begin{bmatrix} \cos\left(\sqrt{\frac{\lambda\epsilon_n}{D_{e,n}}}L_n\right) \\ -\sqrt{\lambda\epsilon_n D_{e,n}}\sin\left(\sqrt{\frac{\lambda\epsilon_n}{D_{e,n}}}L_n\right)\end{bmatrix}.\end{aligned} \tag{3.50}$$

Likewise, variables k_5, and k_6 can be obtained and are represented in relation to k_3 and k_4 as,

$$\begin{aligned}
&\begin{bmatrix} \cos\left(\sqrt{\frac{\lambda\epsilon_p}{D_{e,p}}}\hat{x}\right) & \sin\left(\sqrt{\frac{\lambda\epsilon_p}{D_{e,p}}}\hat{x}\right) \\ -\sqrt{\lambda\epsilon_p D_{e,p}}\sin\left(\sqrt{\frac{\lambda\epsilon_p}{D_{e,p}}}\hat{x}\right) & \sqrt{\lambda\epsilon_p D_{e,p}}\cos\left(\sqrt{\frac{\lambda\epsilon_p}{D_{e,p}}}\hat{x}\right) \end{bmatrix} \begin{bmatrix} k_5 \\ k_6 \end{bmatrix} \\
&= \begin{bmatrix} \cos\left(\sqrt{\frac{\lambda\epsilon_s}{D_{e,s}}}\hat{x}\right) & \sin\left(\sqrt{\frac{\lambda\epsilon_s}{D_{e,s}}}\hat{x}\right) \\ -\sqrt{\lambda\epsilon_s D_{e,s}}\sin\left(\sqrt{\frac{\lambda\epsilon_s}{D_{e,s}}}\hat{x}\right) & \sqrt{\lambda\epsilon_s D_{e,s}}\cos\left(\sqrt{\frac{\lambda\epsilon_s}{D_{e,s}}}\hat{x}\right) \end{bmatrix} \begin{bmatrix} k_3 \\ k_4 \end{bmatrix},
\end{aligned} \tag{3.51}$$

where $\hat{x} = L_n + L_s$, and the total eigenfunction representation across the spatial x domain is,

$$\xi(x;\lambda) = \begin{cases} \xi_n(x;\lambda) & 0 \leq x < L_n, \\ \xi_s(x;\lambda) & L_n \leq x < L_n + L_s, \\ \xi_p(x;\lambda) & L_n + L_s < x \leq L. \end{cases} \tag{3.52}$$

As these eigenfunctions are by definition orthogonal [153], the next step is to introduced a weighting function, $\alpha(x)$ that fulfils the following,

$$\int_0^L \xi(x;\lambda)^2 \alpha(x) dx = 1. \tag{3.53}$$

With an additional boundary condition of $\partial\xi(x;\lambda)/\partial x = 0$, it is possible to solve for the set of eigenvalues, denoted in the context as λ_k through analysis of the roots in the following equation,

$$\frac{d\xi_p(x;\lambda_k)}{dx} = -k_5 \sin\left(\sqrt{\frac{\lambda_k\epsilon_p}{D_{e,p}}}L\right) + k_6 \cos\left(\sqrt{\frac{\lambda_k\epsilon_p}{D_{e,p}}}L\right) = 0. \tag{3.54}$$

This results in a representation of k_5 and k_6 in terms of λ_k, and the next step is to numerical search across the unit cell width, L, to obtain zero crossings. Further discussion of the numerical implementation will be presented below, as an efficient method for this search is desirable. Once these zero crossings have been obtained, the solution to the homogenous problem is presented as,

$$c_e(x,t) = \sum_{k=0}^{\infty} \zeta(0;\lambda_k)\xi(x;\lambda_k)e^{-\lambda_k t}. \tag{3.55}$$

3.1.4.2 Inhomogeneous problem

In parallel to the above homogenous solution, it is possible to investigate the inhomogeneous component as the expectation is the final transfer function will be a component of both. First, the inhomogenous problem is restated as,

$$\frac{\partial c_e(x,t)}{\partial t} = \frac{1}{\epsilon_e(x)}\frac{\partial}{\partial x}\left(D_e^{\text{eff}}(x)\frac{\partial c_e(x,t)}{\partial x}\right) + \frac{a_s(1-t_+^0)}{\epsilon_e(x)}j(x,t). \tag{3.56}$$

Similarly to the previous subsection, the problem is transformed into a series expansion and then partially derived with respect to t,

$$\frac{\partial c_e(x,t)}{\partial t} = \sum_{k=0}^{\infty}\frac{d\hat{c}_{e,k}(t)}{dt}\xi(x;\lambda_k), \tag{3.57}$$

where $\hat{c}_{e,k}(t)$ represents the set of generalised Fourier coefficients. By substituting equation (3.57) into (3.56) it is possible to obtain,

$$\sum_{k=0}^{\infty}\frac{d\hat{c}_{e,k}(t)}{dt}\xi(x;\lambda_k) = \frac{1}{\epsilon_e(x)}\frac{\partial}{\partial x}\left(D_e^{\text{eff}}(x)\frac{\partial c_e(x,t)}{\partial x}\right) + \frac{a_s(1-t_+^0)}{\epsilon_e(x)}j(x,t). \tag{3.58}$$

To reduce the above equation, a bounded integration with respect to x from 0 to L after multiplying both sides by $\xi(x;\lambda_s)\epsilon_e(x)$ which reduces to,

$$\begin{aligned}\frac{d\hat{c}_{e,k}(t)}{dt} &= -\lambda_k\int_0^L c_e(x,t)\xi(x;\lambda_k)\epsilon_e(x)dx + \lambda_k\int_0^L a_s(1-t_+^0)j(x,t)\xi(x;\lambda_k)dx \\ &= -\lambda_k\hat{c}_{e,k}(t) + \hat{j}_k(t;\lambda_k),\end{aligned} \tag{3.59}$$

where $\hat{j}_k$ has been defined as,

$$\hat{j}_k(t;\lambda_k) = \lambda_k \int_0^L a_s(1 - t_+^0)j(x,t)\xi(x;\lambda_k)dx. \tag{3.60}$$

3.1.4.3 Transfer function combination

Now that both the inhomogeneous and homogenous problems have derived solutions, the final transfer function for $c_e(x,t)$ can be viewed as the summation of $\hat{c}_{e,k}(t)$ components. That is to say, through obtaining a transfer function for $\hat{c}_{e,k}(t)$, it would be possible to obtain one for $c_e(x,t)$. Starting from equation (3.59),

$$\begin{aligned}
\hat{j}_k(t;\lambda_k) =& \lambda_k \int_0^L a_s(1 - t_+^0)j(x,t)\xi(x;\lambda_k)dx, \\
s \cdot \hat{C}_{e,k}(s) =& -\lambda_k \hat{C}_{e,k}(s) + \hat{J}_k(s), \\
\frac{\hat{C}_{e,k}(s)}{I_{app}(s)} =& \frac{1}{s+\lambda_k}\frac{\hat{J}_k(s)}{I_{app}(s)},
\end{aligned} \tag{3.61}$$

where $\hat{j}_k(t)$ is a summation of subdomain components and defined as,

$$\begin{aligned}
\hat{j}_k(t;\lambda_k) =& \lambda_k \int_0^L a_s(1 - t_+^0)j(x,t)\xi(x;\lambda_k)dx, \\
=& \lambda_k \int_0^{L_n} a_s(1 - t_+^0)j(x,t)\xi(x;\lambda_k)dx + \lambda_k \int_{L_n+L_s}^{L} a_s(1 - t_+^0)j(x,t)\xi(x;\lambda_k)dx, \\
=& \hat{J}_{k,n}(s) + \hat{J}_{k,p}(s).
\end{aligned} \tag{3.62}$$

It is then required to investigate these subdomain representations of $\hat{J}_k(s)$, utilising equation (3.60) above,

$$\begin{aligned}
\hat{j}_{k,n}(t) =& \int_0^{L_n} a_s(1 - t_+^0)j(x,t)\xi(x;\lambda_k)dx, \\
\frac{\hat{J}_{k,n}(s)}{I_{app}(s)} =& \int_0^{L_n} a_s(1 - t_+^0)\frac{J(x/L_n,s)}{I_{app}(s)}\xi(x;\lambda_k)dx,
\end{aligned} \tag{3.63}$$

where $J(x/L_n, s)/I_{app}(s)$ is previously defined in (3.24) above. Following the previous derivation presented in Lee et al.[152], equation (3.63) is integrated and provides,

$$\begin{aligned}\frac{\hat{J}_{k,n}(s)}{I_{\mathrm{app}}(s)} =& \frac{k_1(1-t_+^0)\hat{L}_n \sin(\hat{L}_n)(\kappa_n^{\mathrm{eff}}+\sigma_n^{\mathrm{eff}}\cosh(\nu_n(s)))\nu_n(s)}{AF(\kappa_n^{\mathrm{eff}}+\sigma_n^{\mathrm{eff}})(\hat{L}_n^2+\nu_n^2(s))\sinh(\nu_n(s))} \\ &+ \frac{k_1(1-t_+^0)\hat{L}_n \sin(\hat{L}_n)(\kappa_n^{\mathrm{eff}}+\sigma_n^{\mathrm{eff}})\nu_n^2(s)}{AF(\kappa_n^{\mathrm{eff}}+\sigma_n^{\mathrm{eff}})(\hat{L}_n^2+\nu_n^2(s))},\end{aligned} \tag{3.64}$$

where $\hat{L}_n = L_n\sqrt{\epsilon_n\lambda_k/D_{e,n}}$. Likewise, for the positive domain representation,

$$\begin{aligned}\hat{j}_{k,p}(t) &= \int_{L_n+L_s}^{L} a_s(1-t_+^0)j(x,t)\xi(x;\lambda_k)dx, \\ \frac{\hat{J}_{k,p}(s)}{I_{app}(s)} &= \int_{L_n+L_s}^{L} a_s(1-t_+^0)\frac{J(L-x/L_p,s)}{I_{app}(s)}\xi(x;\lambda_k)dx,\end{aligned} \tag{3.65}$$

solving the integration in (3.65) provides the positive electrode reaction flux transfer function as,

$$\begin{aligned}\frac{\hat{J}_{k,p}(s)}{I_{\mathrm{app}}(s)} =& \frac{k_6(1-t_+^0)\hat{L}_p \cos(\hat{L}_p)(\kappa_p^{\mathrm{eff}}+\sigma_p^{\mathrm{eff}}\cosh(\nu_p(s)))\nu_p(s)}{AF(\kappa_p^{\mathrm{eff}}+\sigma_p^{\mathrm{eff}})(\hat{L}_p^2+\nu_p^2(s))\sinh(\nu_p(s))} \\ &- \frac{k_5(1-t_+^0)\hat{L}_p \sin(\hat{L}_p)(\kappa_p^{\mathrm{eff}}+\sigma_p^{\mathrm{eff}}\cosh(\nu_p(s)))\nu_p(s)}{AF(\kappa_p^{\mathrm{eff}}+\sigma_p^{\mathrm{eff}})(\hat{L}_p^2+\nu_p^2(s))\sinh(\nu_p(s))} \\ &+ \frac{k_6(1-t_+^0)\hat{L}_p \cos(\hat{L}_{ns})(\kappa_p^{\mathrm{eff}}+\sigma_p^{\mathrm{eff}}\cosh(\nu_p(s)))\nu_p(s)}{AF(\kappa_p^{\mathrm{eff}}+\sigma_p^{\mathrm{eff}})(\hat{L}_p^2+\nu_p^2(s))\sinh(\nu_p(s))} \\ &- \frac{k_5(1-t_+^0)\hat{L}_p \sin(\hat{L})(\kappa_p^{\mathrm{eff}}+\sigma_p^{\mathrm{eff}}\cosh(\nu_p(s)))\nu_p(s)}{AF(\kappa_p^{\mathrm{eff}}+\sigma_p^{\mathrm{eff}})(\hat{L}_p^2+\nu_p^2(s))\sinh(\nu_p(s))} \\ &- \frac{k_5(1-t_+^0)\sigma_p^{\mathrm{eff}}(\cos(\hat{L}_{ns})\kappa_p^{\mathrm{eff}}+\cos(\hat{L})\sigma_p^{\mathrm{eff}})\nu_p^2(s)}{AF(\kappa_p^{\mathrm{eff}}+\sigma_p^{\mathrm{eff}})(\hat{L}_p^2+\nu_p^2(s))} \\ &- \frac{k_6(1-t_+^0)\sigma_p^{\mathrm{eff}}(\sin(\hat{L}_{ns})\kappa_p^{\mathrm{eff}}+\sin(\hat{L})\sigma_p^{\mathrm{eff}})\nu_p^2(s)}{AF(\kappa_p^{\mathrm{eff}}+\sigma_p^{\mathrm{eff}})(\hat{L}_p^2+\nu_p^2(s))},\end{aligned} \tag{3.66}$$

with, $\hat{L}_p = L_p\sqrt{\epsilon_p\lambda_k/D_{e,p}}$, $\hat{L}_{ns} = (L_n+L_s)\sqrt{\epsilon_p\lambda_k/D_{e,p}}$, and $\hat{L} = L\sqrt{\epsilon_p\lambda_k/D_{e,p}}$. Finally, the generalised Fourier coefficient electrolyte concentration transfer function is presented as,

$$\frac{\hat{C}_{e,k}(s)}{I_{\mathrm{app}}(s)} = \frac{1}{s+\lambda_k}\left[\frac{\hat{J}_{k,n}(s)}{I_{\mathrm{app}}(s)} + \frac{\hat{J}_{k,p}(s)}{I_{\mathrm{app}}(s)}\right]. \tag{3.67}$$

The final electrolyte transfer function can be obtained by integrating equation (3.57) with respect to time and combining with 3.67 as,

$$\frac{C_e(x,s)}{I_{\mathrm{app}}(s)} = \sum_{k=0}^{\alpha} \frac{\hat{C}_{e,k}(x,s)}{I_{\mathrm{app}}(s)}\xi(x;\lambda_k). \tag{3.68}$$

3.1.5 Concluding derived transfer functions

The transfer functions utilised for reduced-order realisation have been derived in this section. Each transfer function represents an internal physical electrochemical variable utilised for online predictions and control later in this chapter. To capture the physical states in a single representation, the transfer functions defined in equations (3.21), (3.24), (3.26), (3.28), (3.32), (3.68), for both positive and negative electrode domains are combined into a single input, multiple output (SIMO) response array, which provides a single mathematical structure comprising the continuous-time cell impulse response, shown as $G(s)$ in (3.69) below. To capture battery terminal voltage, equation (2.20) is utilised from the output parameters of the system. Efficiently translating this formation into a state-space representation is the basis for the computationally informed discrete realisation algorithm defined in Section 3.2.

$$G(s) = \left\{ \begin{matrix} \frac{C_e(x,s)}{I_{app}(s)} \\ \\ \frac{\Phi_e(x,s)}{I_{app}(s)} \\ \\ \frac{\tilde{C}_{\text{surf}}(z,s)}{I_{app}(s)} \\ \\ \frac{\tilde{\Phi}_{s,e}(z,s)}{I_{app}(s)} \\ \\ \frac{\Phi_s(z,s)}{I_{app}(s)} \\ \\ \frac{J(z,s)}{I_{app}(s)} \end{matrix} \right\} \tag{3.69}$$

3.2 Computationally Informed Discrete Realisation Algorithm

A computationally informed discrete realisation algorithm (CI-DRA) is presented in this section to utilise the transfer functions derived in the previous section and create a state-space representation of the system dynamics. This method utilises the sampled impulse response from the continuous transfer functions derived in Section 3.1 above. The CI-DRA provides a mathematical pathway to achieve the linear state-space realisation of the form,

$$\begin{aligned} \boldsymbol{x}[t_c + T_s] &= \mathbf{A} \cdot \boldsymbol{x}[t_c] + \mathbf{B} \cdot u[t_c], \\ \boldsymbol{y}[t_c] &= \mathbf{C} \cdot \boldsymbol{x}[t_c] + \mathbf{D} \cdot u[t_c]. \end{aligned} \tag{3.70}$$

The CI-DRA incorporates the zero-order hold methodology first presented in the conventional DRA [73]; however, to achieve faster system realisation, an improvement has been achieved to reduce the overall realisation time. This is achieved by aligning the sampled transfer function frequency (F_{tf}) and the final state-space sampling frequency, $F_s = 1/T_s$, to be integer multiples, thus removing the DRA's interpolation requirements of the discrete transfer function response [73]. This alignment also enables the CI-DRA to utilise the discrete-time impulse response from the transfer function response without the additional accumulated continuous time step response that is required in the conventional DRA method. This section introduces the conventional discrete realisation algorithm and the improvements achieved with the computationally informed algorithm.

First, the approximate discrete system response of the continuous time transfer function is introduced as [154],

$$G(z) \approx G(s) \mid_{s=\frac{2(z-1)}{(z+1)}F_{tf}} . \tag{3.71}$$

This result is utilised to generate the approximate discrete time response, $G(z)$, from the continuous time transfer functions. Next, by relating the discrete Fourier transformation of a sequence to its z-transform [155], the following equation is formed,

$$G_d[f] = G\left(2F_{tf}\frac{\exp(j2\pi f/N) - 1}{\exp(j2\pi f/N) + 1}\right), \quad 0 < f < N \tag{3.72}$$

where N is defined as the number of points captured in the transfer function response and depends on the response length and sampling frequency, F_{tf}. For this work, N is sized to be memory efficient instead of computationally efficient in the conventional DRA, i.e. the size of N is the exact requested length. In contrast, the conventional DRA selected the next largest power of two. Through this sizing, the CI-DRA provides additional computational performance as the size of N impacts the realisation performance. Next, to acquire the discrete-time impulse response, $g_{imp}[n]$, an inverse discrete Fourier transformation is applied to $G_d[f]$,

$$g_{imp}[n] = \frac{1}{N}\sum_{f=0}^{N-1} G_d[f]\exp(j2\pi fn/N). \tag{3.73}$$

The conventional DRA proceeds with a cumulative summation and interpolation of $g_{imp}[n]$, at the state-space sampling frequency, F_s. This result can recreate the discrete-time impulse response through an element-wise difference operation (i.e. $g_{step}[k] - g_{step}[k-1]$). The cumulative operator is presented as,

$$g_{step} = \frac{1}{F_{tf}} \sum_{i=0}^{k-1} g_d[i]. \tag{3.74}$$

As the relation between F_{tf} and T_s is exploited, the CI-DRA can forgo the steps formed from equation (3.74) that are required in the conventional DRA. The relation allows for the CI-DRA to directly utilise equation 3.73 for realisation, providing a substantial performance improvement with large values of N. To showcase this difference, a reference continuous time transfer function is introduced,

$$G_r(s) = \frac{1}{8s^2 + 2s + 4}. \tag{3.75}$$

The CI-DRA's aligned impulse response is shown in Figure 3.2, with comparison to the conventional DRA. Continuing the realisation process, the Ho-Kalman [156] algorithm is utilised to form the state-space representation. This is completed via exploitation of the Markov parameters that comprise the resultant discrete response impulse response, $\{G_k = g_{imp}[n]\}$. This response can be shown in the

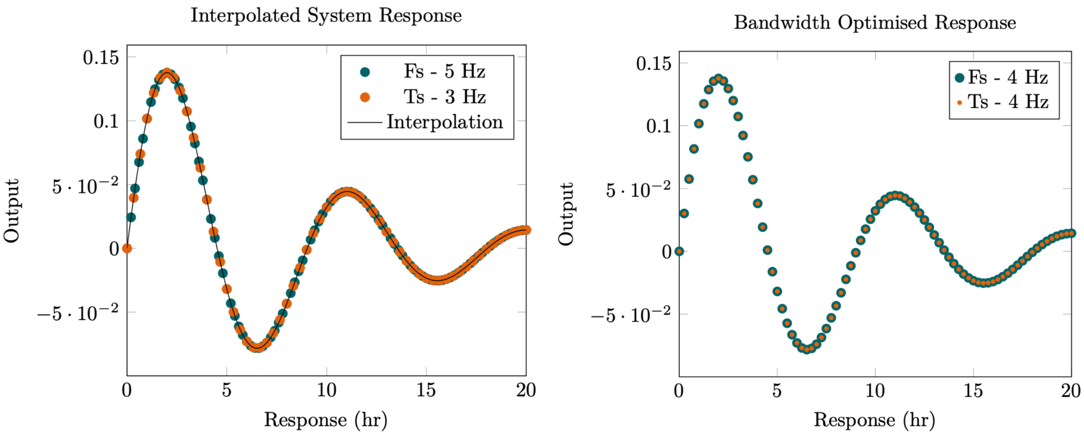


Fig. 3.2 Conventional DRA discrete-time impulse response interpolation from continuous-time transfer function impulse response (left) and CI-DRA's response alignment removing interpolation requirements (right)

following form,

$$\boldsymbol{G}_k = \begin{cases} \boldsymbol{D} & k = 0, \\ \\ \boldsymbol{C}\boldsymbol{A}^{t-1}\boldsymbol{B} & k = 1, 2, 3, .. \end{cases} \tag{3.76}$$

The D array can be gathered from the system response at time-step zero; however, it can also be obtained utilising an initial value theorem for the equivalent discrete-time system definition as,

$$\boldsymbol{D} = G_0 = g_{imp}[0] = \lim_{z \to \infty} G(z). \tag{3.77}$$

This method will be utilised to numerically capture the D array. Further discussion on the reformation of the D array for improvement in online predictions will be discussed in a later section. The remaining transfer function response can then be formulated into a block Hankel matrix (3.78) of corresponding Markov parameters. This block Hankel ($\mathcal{H}_{k,m}$) has indices corresponding to a subset domain of the discrete-time impulse response. The dimensionality of G_k is defined by the system realisation for the CI-DRA; this comprises the vector size of equation (3.69) and the corresponding transfer function response length N.

$$\boldsymbol{\mathcal{H}}_{k,m} = \begin{pmatrix} \boldsymbol{G}_1 & \boldsymbol{G}_2 & \boldsymbol{G}_3 & \cdots & \boldsymbol{G}_m \\ \boldsymbol{G}_2 & \boldsymbol{G}_3 & \boldsymbol{G}_4 & \cdots & \boldsymbol{G}_{m+1} \\ \boldsymbol{G}_3 & \boldsymbol{G}_4 & \boldsymbol{G}_5 & \cdots & \boldsymbol{G}_{m+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{G}_k & \boldsymbol{G}_{k+1} & \boldsymbol{G}_{k+2} & \cdots & \boldsymbol{G}_{m+k-1} \end{pmatrix}. \tag{3.78}$$

The block Hankel has an additional feature that allows for relation to the controllability and observability matrices defined as,

$$\boldsymbol{\mathcal{H}}_{k,m} = \boldsymbol{\mathcal{O}}_k \boldsymbol{\mathcal{C}}_m. \tag{3.79}$$

This relation provides a mechanism to form the A, B, and C matrices from the block Hankel matrix and is shown in the observability ($\mathcal{O}$) and the controllability ($\mathcal{C}$) definitions below. Initially, by exploiting this relation and factoring $\mathcal{H}_{k,m}$ into the two matrices, the first step to obtaining these matrices can be completed.

$$\boldsymbol{\mathcal{O}}_k = \begin{bmatrix} \boldsymbol{C} \\ \boldsymbol{CA} \\ \boldsymbol{CA}^2 \\ \vdots \\ \boldsymbol{CA}^{k-1} \end{bmatrix} \tag{3.80}$$

$$\boldsymbol{\mathcal{C}}_m = \begin{bmatrix} \boldsymbol{B} & \boldsymbol{AB} & \boldsymbol{A}^2\boldsymbol{B} & \cdots & \boldsymbol{A}^{m-1}\boldsymbol{B} \end{bmatrix} \tag{3.81}$$

To accomplish this factoring, singular value decomposition (SVD) provides the mechanism to reduce the block Hankel through truncation of the system order. The truncated SVD is shown in equation (3.82) below, where $\boldsymbol{\Sigma}_s$ captures the highest order singular values of the block Hankel in descending order while $\boldsymbol{\Sigma}_n \approx 0$ captures the remaining values low order values. This approximation is exact when the remaining orders captured by $\boldsymbol{\Sigma}_n$ are zero. Selection of the size of Σ_s compromises numerical performance and final system fidelity. Further discussion on this compromise is presented in Section 3.3.2 below.

$$\boldsymbol{\mathcal{H}}_{k,m} = \begin{bmatrix} \boldsymbol{U}_s & \boldsymbol{U}_n \end{bmatrix} \begin{bmatrix} \boldsymbol{\Sigma}_s & 0 \\ 0 & \boldsymbol{\Sigma}_n \end{bmatrix} \begin{bmatrix} \boldsymbol{V}_s^\dagger \\ \boldsymbol{V}_n^\dagger \end{bmatrix} = \boldsymbol{U}_s \boldsymbol{\Sigma}_s \boldsymbol{V}_s^\dagger \tag{3.82}$$

Next, combining the SVD and the observability and controllability definitions, equations (3.80) and (3.81) become,

$$\boldsymbol{\mathcal{O}}_k = \boldsymbol{U}_s \boldsymbol{\Sigma}_s^{1/2} \boldsymbol{T}, \tag{3.83}$$

$$\boldsymbol{\mathcal{C}}_m = \boldsymbol{T}^{-1} \boldsymbol{\Sigma}_s^{1/2} \boldsymbol{V}_s^{\dagger}, \tag{3.84}$$

where $\boldsymbol{T}$ is introduced as a transformation matrix that defines the basis for $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$. It is presented as the identity matrix in this work, thus simplifying the computation of $\boldsymbol{\mathcal{O}}_k$ and $\boldsymbol{\mathcal{C}}_m$. It is then possible to exploit the original structure of the matrices and utilise the intrinsic Markov parameters and obtain the resulting state-space representation as:

$$\begin{aligned} \boldsymbol{A} &= \mathfrak{O}_k^{\dagger} \mathfrak{H}_{k,m+1} \mathfrak{C}_m^{\dagger}, \\ \boldsymbol{B} &= \mathfrak{C}_m[1{:}M, 1{:}\gamma_{in}], \\ \boldsymbol{C} &= \mathfrak{O}_k[1{:}\gamma_{out}, 1{:}M], \end{aligned} \tag{3.85}$$

where γ_{out} and γ_{in} are the state-space output size and input size, respectively, $\mathfrak{H}_{k,m+1}$ denotes a single index forward shifted of the block Hankel matrix, M is the system order, and $\dagger$ is the matrix transpose. The realisation process is completed with the above equations, and a linear system model of state-space form, as shown previously in equation (3.70), is acquired. A summary of the CI-DRA method is shown in the list below.

Summary of the CI-DRA:

1. Align the final system sampling period and transfer function sampling period, ensuring that T_s is an integer multiple of the transfer function sampling frequency, F_{tf} enabling the removal of the cumulation response and interpolation steps in the conventional DRA.

2. Compute the discrete-time frequency response $G_d[f]$ for the derived transfer functions and generate the discrete-time impulse response, $g_{imp}[n]$ via the inverse fast Fourier transformation equation (3.73).

3. Form the block Hankel matrix from the discrete-time impulse response of values ($t > 1$) and compute the in-place truncated singular value decomposition to acquire the controllability and observability matrices.

4. Perform an in-place element shift operation on the block Hankel matrix to obtain the time-shifted block Hankel matrix.

5. Form a linear state-space system from the Ho-Kalman algorithm with unstable poles replaced by their reciprocals and oscillating poles replaced by their magnitudes.

3.2.1 Software Implementation

LiiBRA.jl, a Julia [147] based package, has been created for the fast computational implementation of the above computationally informed discrete realisation algorithm. This package improves on previously presented implementations [58] of the eigensystem realisation algorithm by providing fast computational solutions while maintaining fidelity. Key improvements include performant truncated SVD support, large array memory optimisation, and performance benefits from the Julia language largely due to the bottleneck from block Hankel matrix formation. Julia provides a high-performance dynamic typeset with just-in-time compilation and multiple dispatch capabilities. These features provide an effective computational language for scientific computing while providing modern syntax. The open-source code repository for LiiBRA.jl can be found at https://github.com/BradyPlanden/LiiBRA.jl. Example usage of LiiBRA.jl for creation and simulation of the reduced-order models is presented in the code shown in Figure 3.3 below. First, the cell variable is constructed, with the Chen et al. [81] parameterisation, followed by the range of state-of-charge and temperature points for the model generation to be completed. This range enables interpolating the generated state-space models for varying states of charge and temperatures. A 75% initialisation state-of-charge is selected, and the model generation is completed by passing through the exported Realise function call, which outputs a tuple comprising of each corresponding state-space array. Finally, a hybrid pulse power characterisation is completed through the exported HPPC function call, with corresponding positive and negative pulse amplitudes of 4A and -3A, respectively.

```
pkg> add LiiBRA

using LiiBRA

Cell = Construct("LG M50") # Construct LG M50 Parameterisation
ψ = collect(0.8:-0.05:0.7) # SOC Range
T = collect (273.15:5.0:318.15) # Temperature Range
SOC = 0.75 # Initial SOC

A,B,C,D = Realise(Cell,ψ,T) # Create Models

HPPC(Cell,ψ,SOC,4.0,-3.0,A,B,C,D) # HPPC at 75% SOC with 4A, and -3A pulses
```

Fig. 3.3 Example usage of LiiBRA.jl, providing a simple package for creating and simulating reduced-order models.

The high-level structure of LiiBRA.jl is shown in Figure 3.4 below. The package dependencies are shown and offer improved code reusability while minimising the size of LiiBRA.jl. Through distributing the codebase and utilising Julia's open-source packages, LiiBRA.jl can be modular and flexible while providing improved algorithm selection for compatibility and performance. These dependencies include TSVD.jl [157] for the truncated SVD, FFTW.jl [158], providing an interface for inverse fast Fourier transforms, and Roots.jl for numerical root finding for the electrolyte concentration transfer function.

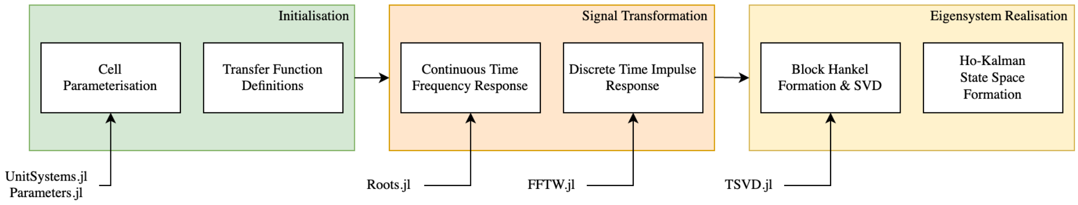


Fig. 3.4 High-level architecture of LiiBRA.jl's implementation of CI-DRA with package dependencies listed.

To provide robust operation, LiiBRA.jl includes a dependency of Interpolations.jl [159] to support the implementation of the conventional DRA when the conditions required for the CI-DRA are not met (i.e. F_{tf} and F_s are not integer multiples); however, N remains sized for memory efficiency as discussed in the previous section. This provides an easy interface for model creation, with feedback to end-users on the computational method being utilised. An additional achievement for LiiBRA.jl is its capability to be numerically solved on ARM hardware which greatly expands utilisation for in-situ model creation. This provides a mechanism for one of the aims of this work, enabling in-vehicle model creation for parameterisation variability. LiiBRA.jl's capabilities are presented in the next section.

3.3 Results

In this section, numerical optimisation of the SVD method used in LiiBRA.jl is presented, followed by an investigation of parametric sensitivities and a verification of the reduced-order models against the full-order system. Finally, experimental validation of the reduced-order model is presented through an automotive drive cycle. The x86 computational results for this work were gathered on a 2019 Macbook Pro 13" Intel i5 with ARM results obtained on a Qualcomm Snapdragon 845 with a Ubuntu 18.04.5 LTS operating system. Both hardware architectures utilised Julia version 1.7.2.

3.3.1 Singular Value Decomposition Methods

During initial platform profiling, the singular value decomposition was shown to impact the total numerical solution time greatly. Optimising this truncated singular value decomposition was required to achieve the online performance targets of this work. Three open-source packages were investigated for their high computational efficiency: Arpack.jl [160], PROPACK.jl [161], and TSVD.jl.[157] Arpack.jl is a Fortran wrapper of the implicitly restarted Arnoldi method [162], reducing to the implicitly restarted Lanczos method for symmetric input matrices. PROPACK.jl is likewise a Fortran wrapper of the Fortran PROPACK software, initially developed by R.M. Larsen [157]. This package implements the Lanczos bidiagonalisation method with partial reorthogonalisation and implicit restart, in which it acts directly on the system matrix without forming the equivalent system in memory. Similarly, TSVD.jl implements the Lanczos bidiagonalisation method with partial reorthogonalisation; however, it is implemented directly with Julia.

To capture each implementations performance, the block Hankel size is varied, as defined in equation (3.78). As the block Hankel determines the length of system response captured in the model formation, it is utilised as the input variable for this investigation. Analysing the results presented in Figure 3.5 below, it is clear that TSVD.jl provides the lowest solution time across the presented range of block Hankel sizes, with PROPACK.jl and Arpack.jl following respectively. PROPACK.jl is shown to use significantly more memory, with the difference between TSVD.jl and Arpack.jl negligible. Therefore, for SVD computations in LiiBRA.jl, the TSVD.jl package was selected as it enables a large range of solutions regarding block Hankel size without compromising performance.

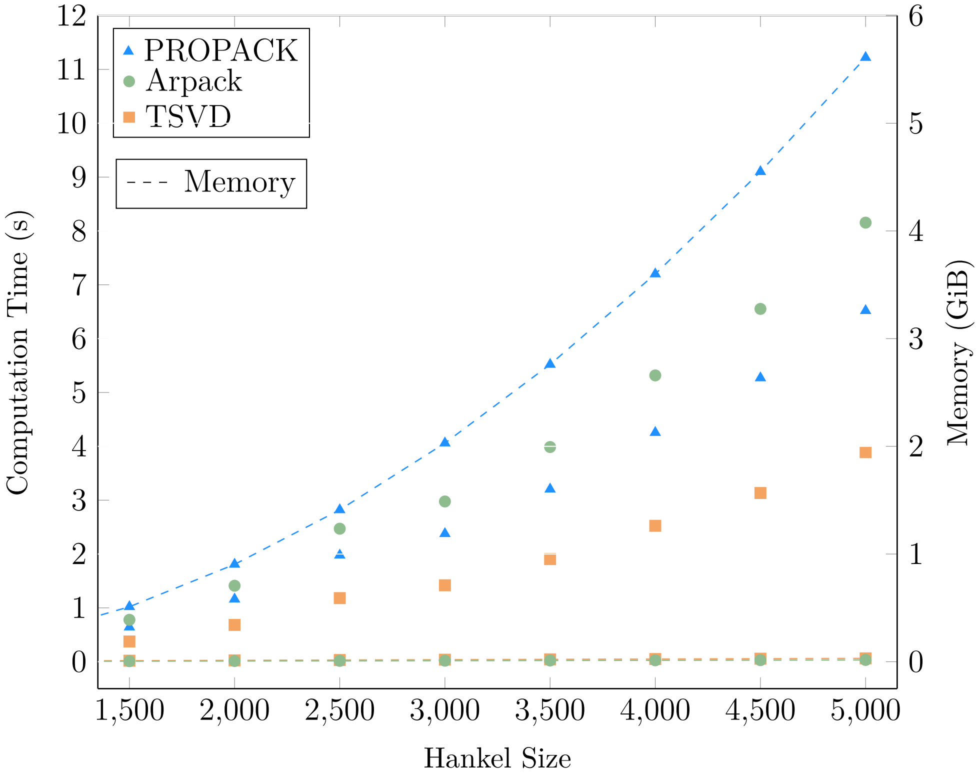


Fig. 3.5 Computation results of PROPACK.jl, TSVD.jl, and Arpack.jl completing SVD of varying block Hankel sizes

Next, to investigate the CI-DRA's improvements over the conventional DRA, a comparison is completed across differing transfer function system response lengths as presented in Figure 3.6 below. As the CI-DRA provides a mechanism to simplify model generation through interpolation removal and sampling response point optimisation, the improvements depend on the total system response length. Figure 3.6 showcases the CI-DRA's ability to capture approximately twice the length of system response over the conventional DRA method for an equivalent computational time from eight hours onwards. This performance improvement enables longer system dynamics to be captured in the reduced-order model generation for improved fidelity in long-term electrochemical predictions. For a 12-hour system response, a 21.7% improvement in model generation time is available through the CI-DRA over the conventional DRA.

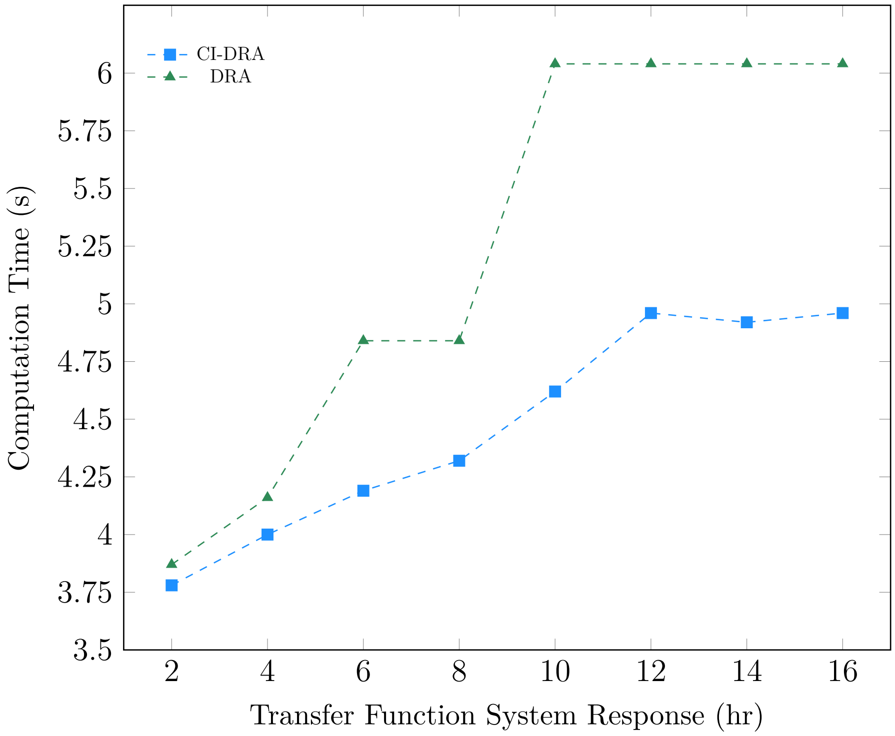


Fig. 3.6 Computation results comparing the CI-DRA to the conventional DRA for varying transfer function system response length utilising LiiBRA.jl.

3.3.2 Computational Sensitivity

A numerical sensitivity analysis was also completed for the CI-DRA framework variables. Table 3.1 lists the variables and the tested ranges. An initial investigation was completed to determine stable confines for the model, which were then selected as the default values to minimise numerical instabilities. The minimum and maximum of each range were tested to determine each variable's sensitivity on the resultant computational time. The benchmarking package BenchmarkingTools.jl [163] was utilised to obtain the relevant statistical results. For this work, the minimum number of simulations for each variable set was selected at six to constrain the total number of simulations while reducing the effect of numerical jitter on the analysis. The median computation time for each variable is shown in Figure 3.7.

Table 3.1 Default values and corresponding ranges for LiiBRA.jl sensitivity analysis of framework variables.

Variable	Definition	Default	Range
$\mathcal{H}$	Size of square Hankel matrix	2500	1500 - 3500
$S_{e,m}$	Number of spatial particles in electrolyte	6	4 - 8
M	System order	8	4 - 12
$S_{s,m}$	Number of spatial particles in electrodes	4	2 - 6
T_{len}	Length of transfer function sampling time [hr]	4.5	1.0 - 8.0
F_{tf} & F_s	System sampling frequencies [Hz]	4	2 - 6

Furthermore, this analysis provides insight towards a minimal package configuration for fast model generation. For this analysis, the transfer function sampling frequency (F_{tf}) and final system sampling time (T_s) are coupled together, as this provides a stable solution for investigating the CI-DRA. The coupled F_s and T_s variables, the number of particles in the electrode ($S_{s,m}$) and transfer function sampling length (T_{len}) have the lowest sensitivities and thus should be selected based on the required model fidelity. The block Hankel size ($\mathcal{H}$), number of particles in the electrolyte ($S_{e,m}$), and model order (M) have large impacts on the total computational time. These variables should be selected based on a compromise between model fidelity and generation time requirements.

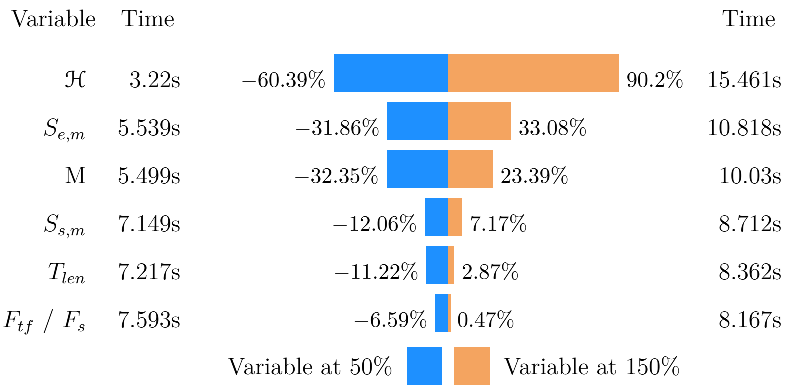


Fig. 3.7 CI-DRA numerical sensitivity for reduced-order model creation at each variable's lower bound (50% default) and higher bound (150% default). Block Hankel size is shown to have the highest sensitivity, with the coupled F_{tf} and F_s having the lowest.

3.3.3 Numerical Verification

A worldwide harmonised light vehicle test procedure (WLTP) 3B [164] has been implemented for numerical verification and experimental validation. This drive-cycle was created from specifications provided for a 2022 Tesla Model 3 long-range vehicle, which are provided in Table 3.2 below. The fitted parameters are denoted by *.

Table 3.2 Tesla Model 3 long-range specifications used for simulated WLTP

Variable	Definition	Value	Unit
M	Total vehicle mass	1931	kg
E	Onboard useable energy	82	kWh
N_s/N_p	Electric system orientation	96s47p*	-
C_{Nom}	Rated single cell capacity	5	Ahr
V_{Lim}	Operational voltage limits	2.5 / 4.2	V
F_D	Vehicle drivetrain losses	[165]	N
η_M	Motor efficiency	0.827*	-

The predicted cycle from the above specifications was then utilised to generate the single-cell scaled power cycle, shown in Figure 3.8 below, for an alternative pack designed with LG Chem. M50 cells instead of Tesla-manufactured cells. This cell was selected as it is widely available and provides a strong reference for the current state of a high-energy intercalation cell with an NCM 811 positive electrode and bi-component Graphite-SiO$_x$ negative electrode. LiiBRA.jl was then parameterised with the LG M50 electrochemical characterisation presented by Chen et al. [81] and was utilised for both model generation and simulation of the WLTP 3B drive-cycle.

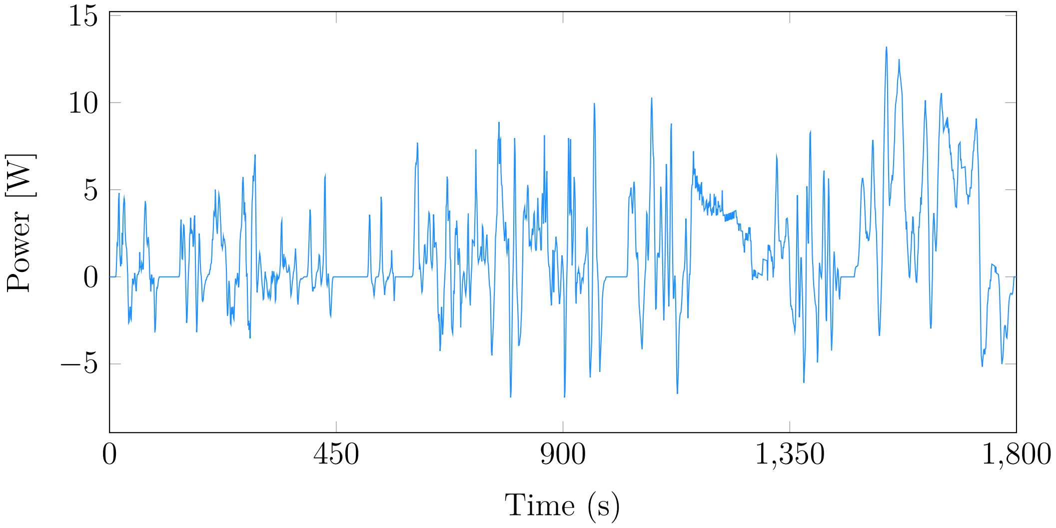


Fig. 3.8 Generated WLTP cycle for a single cell based on a 2022 Tesla Model 3, long-range pack designed with LG M50 cells.

For this verification, the block Hankel was sized at 2500 by 2500 elements, with the transfer function and final system sampling time set to 4 Hz, a reduced system order of six was utilised, and finally, the transfer function sampling length was set to 4.5 hours. Figure 3.9 below displays this verification across the WLTP 3B drive cycle for both predicted terminal voltage and negative electrode concentration. These two variables are shown as references to the underlying physical state prediction capabilities of the CI-DRA and LiiBRA.jl; however, alternatives could have likewise been selected. Additionally, the open-source python battery mathematical modelling package (PyBaMM) [97] was utilised to solve the full-order model with identical parameterisation. Figure 3.9 below outlines the predicted terminal voltage and negative electrode concentration for both the reduced-order and full-order models when initialised at the experimentally aligned 75% state of charge value.

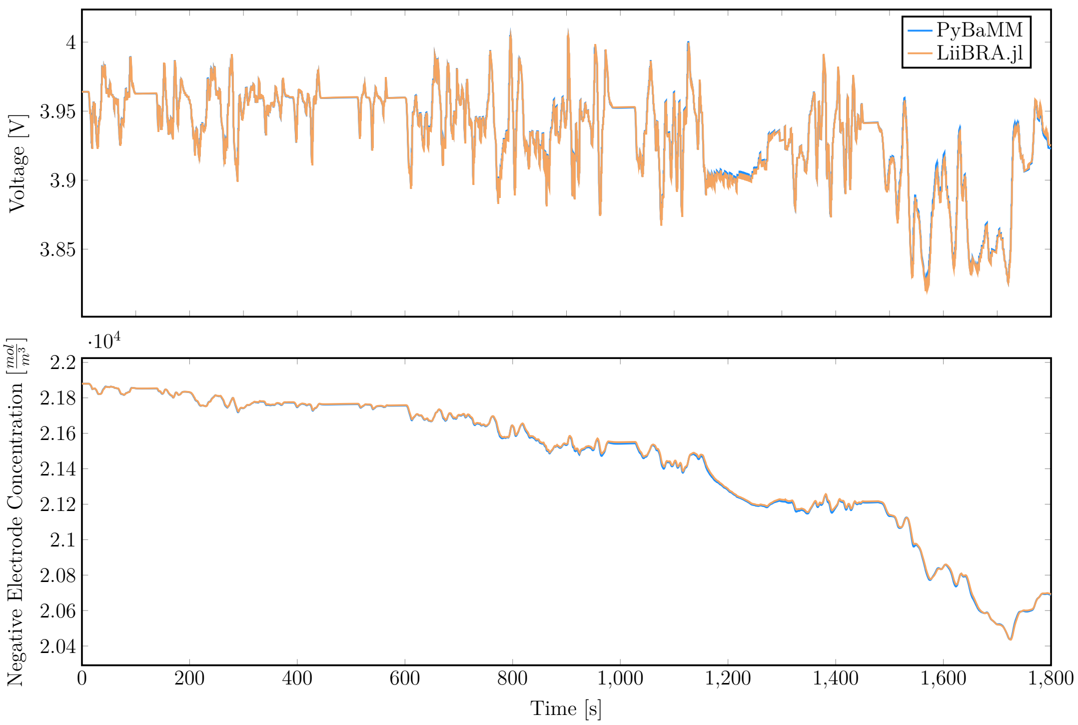


Fig. 3.9 Comparison of LiiBRA.jl to the full order implementation in PyBAMM, WLTP 3B at 75% SOC and 25°C for terminal voltage and negative electrode concentration.

The root-mean-square deviation between the full-order and reduced-order models is 3.64 $[mV]$ and 5.59 $[mol/m^3]$ with an absolute maximum deviation of 46.68 $[mV]$ and 14.24 $[mol/m^3]$ observed. LiiBRA.jl had a total model creation time of 20.2 seconds for five state-of-charge references for a single temperature. The WLTP prediction component for LiiBRA.jl resulted in a mean time of 135.5 milliseconds for a total combined time of 20.34 seconds. In comparison, PyBaMM had a mean runtime of 46.28 seconds. These results also show that LiiBRA.jl has comparable fidelity to the full-order DFN implemented in PyBaMM while enabling significantly faster predictions. The 20.34-second model generation time is only required when first generating the reduced-order model; any additional simulations for the generated models would only require the 135.5 millisecond computation time. Figure 3.10 below showcases the predicted spatial electrolyte salt concentration across the simulation drive cycle.

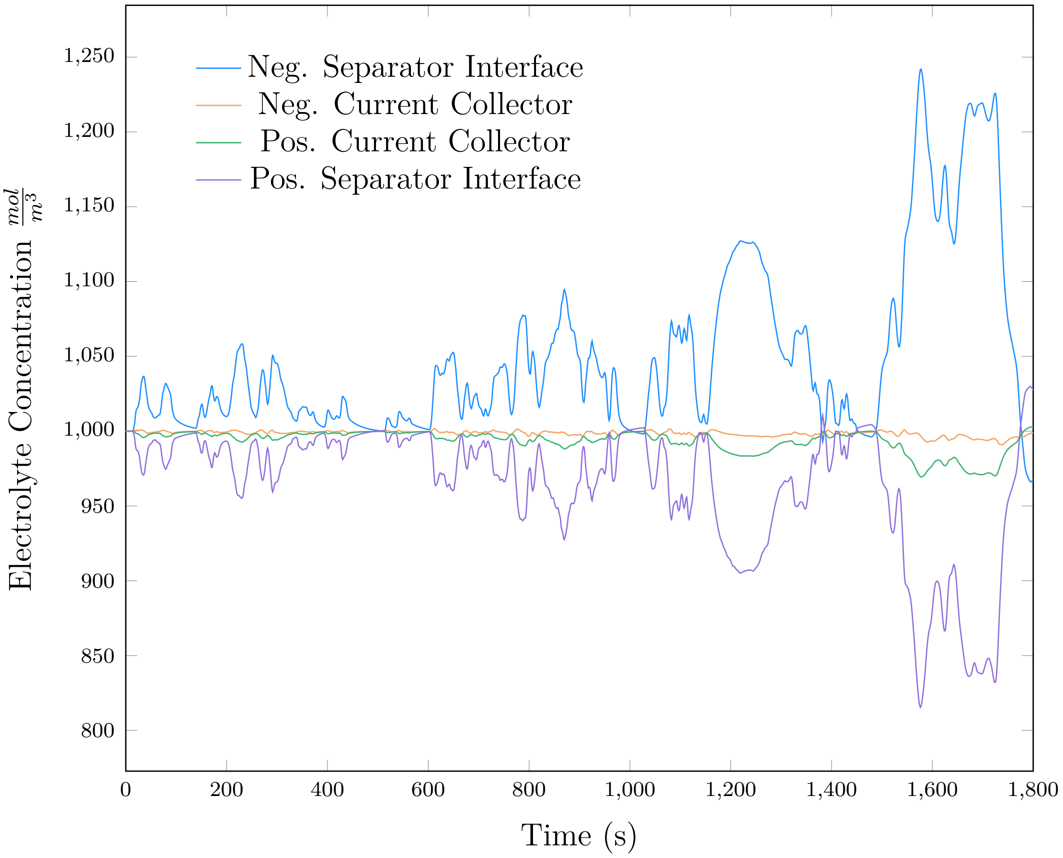


Fig. 3.10 LiiBRA.jl electrolyte concentration for the WLTP 3B drive-cycle at 75% starting SOC and 25°C

An additional performance investigation was performed to compare LiiBRA.jl to a previously reported MATLAB implementation [57, 58] of the DRA. This implementation was modified to reproduce the CI-DRA methodology, allowing for a direct investigation of LiiBRA.jl's performance. Both models were initialised with identical variable and parameterisation sets. The block Hankel size was varied to represent different fidelity and computational timing compromises. An additional ARM implementation of LiiBRA.jl is also presented to confirm capabilities for in-vehicle model generation. The x86 and ARM implementations of LiiBRA.jl perform very similarly, with a mean computational time of 4.04 seconds and 6.06 seconds, respectively. The MATLAB implementation performs significantly worse, with a mean computational time of 13.27 seconds across the simulated block Hankel sizes. A mean computational improvement of 3.51 times is showcased across the varied block Hankel sizes for the x86 results. These results present a three-minute total model generation time for ARM hardware, thus showcasing the viability of LiiBRA.jl for in-vehicle model creation and enabling physics-based model modifications over the lifetime of the battery pack. Computational timings are displayed in Figure 3.11 below.

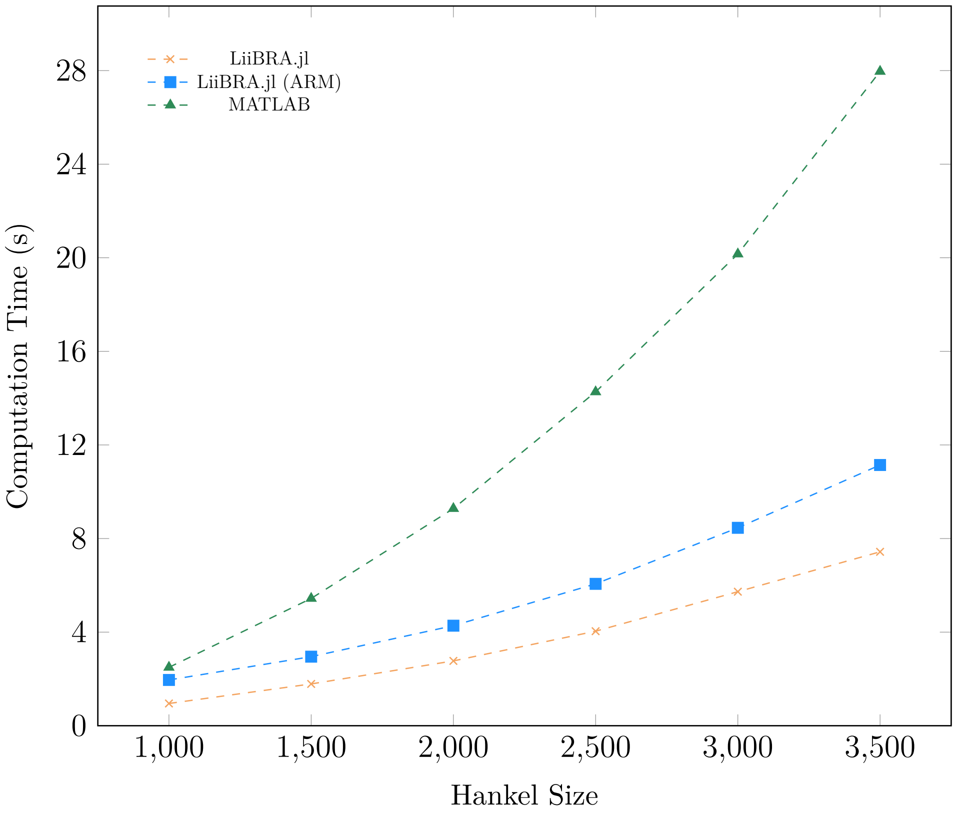


Fig. 3.11 Computation results for CI-DRA using LiiBRA.jl (x86, ARM) and Matlab (x86) for varying block Hankel sizes with identical system parameterisation.

3.3.4 Experimental Validation

Experimental validation of LiiBRA.jl's capabilities is presented below, utilising the parameterisation of an LG Chem. M50 cylindrical 21700 cell [81]. To the author's knowledge, lithium-ion discrete realisation algorithms have been verified from full-order and linearised partial differential implementations [152, 74, 166, 167]; however, an experimental validation has not been presented in the literature.

This section presents an experimental voltage validation of the CI-DRA utilising LiiBRA.jl and the parameterised LG Chem. M50 dataset is presented.

For this validation, three cells were experimentally tested to reduce cell-to-cell variance. This is seen as the minimum requirement, and future investigations are recommended to verify the minimum number of experimentally tested cells required to capture adequate statical variations [168] with respect to LiiBRA.jl. Each cell is initially conditioned at 25°C for five cycles at a 1C discharge rate and a C/2 charge rate utilising an Arbin LBT21084 cycler and a Binder KB115 incubator. A constant current discharge follows this to 75% SOC based on the measured capacity with a WLTP 3B drive-cycle performed based on the specifications shown in Table 3.2. A T-type thermocouple is surface mounted with thermal paste at the body centre of the cell to ensure temperature consistency across the experimentally captured data.

Figure 3.12 below showcases the predicted voltages for the CI-DRA and the experimentally measured cell for the WLTP 3B drive cycle. These results verify the capabilities of the CI-DRA method and LiiBRA.jl for physics-based predictions, with the prediction values producing a root mean square deviation of 7.54 mV to the experimental cell. It should be noted that LiiBRA.jl experiences an increased cell voltage error throughout the length of the drive cycle. This is believed to result from the variation between the experimental and modelled applied current, as well as assumptions made during the experimental cell parameterisation by Chen et al. [81] Online SOC estimation for the conventional DRA has been previously presented [169] which provides a viable correction for this longer-term deviation. Finally, an updated parameterisation of the LG M50 cell is available [82] and includes temperature effects during characterisation, requiring fewer calibration parameters. This parametrisation presents the potential to improve the predicted fidelity without modifying the model architecture.

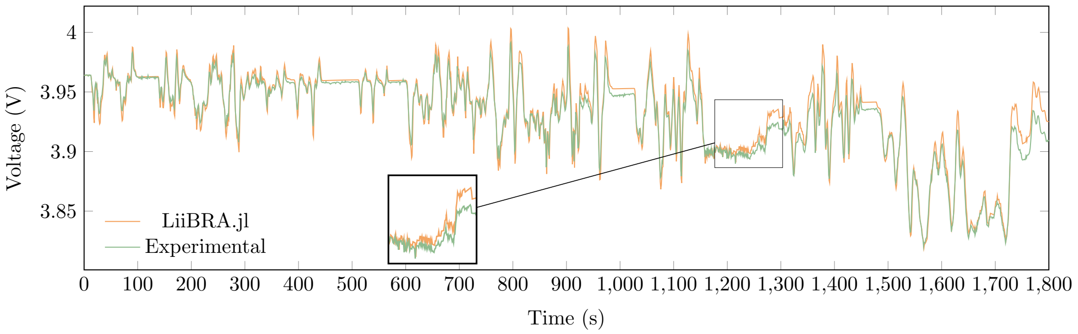


Fig. 3.12 WLTP 3B voltage validation of CI-DRA implemented with LiiBRA.jl to experimentally tested cell. This drive cycle is started at 75% SOC and 25° C initial cell temperature.

3.3.5 Degradation-Informed Predictions

As the in-situ capabilities of the CI-DRA and LiiBRA.jl have been presented alongside an experimental validation of the framework, this section presents a methodology for implementing this framework for degradation-informed predictions. This is completed by exploiting the in-situ capabilities of the presented framework, as the ability for fast realisation on ARM hardware provides a mechanism for reparameterisation of the predictive state-space realisation. The underpinning methods for determining the reparameterisation values are outside of the scope of this thesis; however, there have been multiple works to determine the parameter values for electrochemical models in the literature [51, 170, 171, 172, 173]. These parameter estimation methods utilise non-destructive means to acquire the electrochemical values for the corresponding model predictions; however, due to parameter observability challenges, these methods tend to have lumped parameter values. This subsection shows the degradation-informed capabilities via the predictive changes for corresponding reparameterisation and the corresponding model generation time.

The CI-DRA framework is parameterisation with the Chen et al. [81] dataset for the initial predictions, with a framework variable definition presented in Table A.1. A constant current discharge is predicted for the initial parameterisation, as well as three additional reparameterised models. To showcase the reparameterisation capabilities and its ability to support degradation-informed predictions, the negative electrode stoichiometric concentration limit, θ_n^0, is modified as an example degradation mechanism. This investigation is presented in Figure 3.13 for a 2A discharge at 25° C until the minimum voltage value is reached.

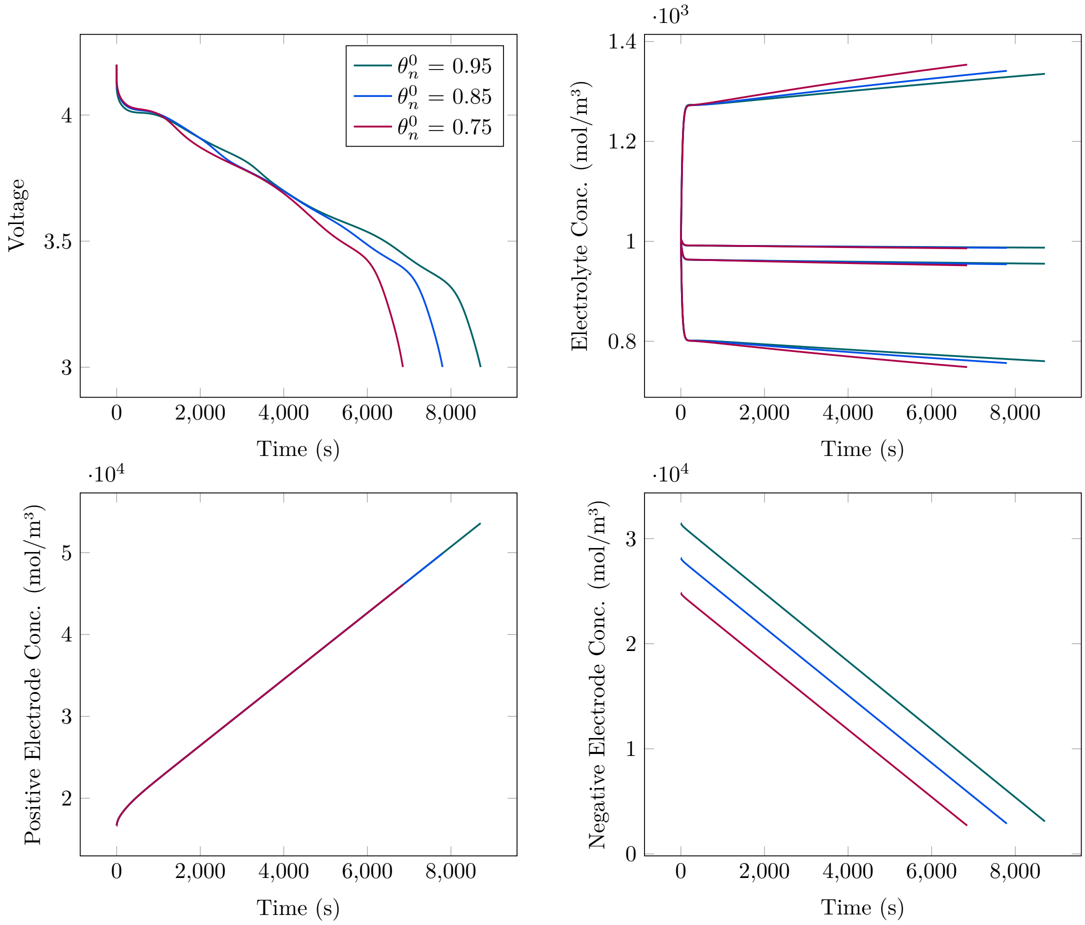


Fig. 3.13 Constant current discharge at 2 A to 3V cell minimum, 100% initial SOC and 25° C initial cell temperature. Three corresponding θ_n^0 values, $0.95, 0.85, 0.75$, are displayed, presenting LiiBRA and the CI-DRA's degradation-informed prediction capabilities.

3.3.6 Online Linearisation & Embedded Deployment

The CI-DRA has shown capability in generating reduced-order models in both offline and online applications; however, a validation of the real-time capabilities of those models is required to confirm the framework's performance. This section confirms this by deploying the generated models from the CI-DRA onto a real-time capable embedded controller. The state-space models generated by the CI-DRA were transferred into a MATLAB Simulink model utilising a fixed-step, discrete solver with a 10ms runtime that was then deployed onto the real-time target through C-code compilation. This toolchain was selected as it is commonly utilised in the automotive industry, and the embedded target supports this compile framework. The process of obtaining the state-space models used for this validation is shown in Appendix A.

The deployed representation is parameterised with an LG Chem. M50 cell, with initialisation at 100% SOC and an operating temperature of 298.15K. The CI-DRA framework variables were parameterised to include a system order of 4, a transfer function system response length of 4.5 hours, a square block Hankel matrix size of 2500 capturing the first 1500 elements of the transfer function response, with the remaining elements capturing an additional 250 elements at locations $[2500 : 2750, ..., 5500 : 5750]$. This methodology improved system predictions by increasing the length of captured response from the transfer functions while maintaining an equivalent memory size. The spatial particle distribution includes four electrolyte particles spaced at the solid electrode boundaries (i.e. $Z = 0, 1$) and two particles in each electrode at the same spatial locations. Finally, the state-space system and transfer function sampling time were selected at 4 Hz.

Given the presented state-space representation in Appendix A and the introduced state-space architecture presented previously in Chapter 2, further validation of the real-time deployment can be completed. As initially presented in Section 3.1 above, the CI-DRA follows the convention introduced by Lee et al. [174] to linearise the transfer functions around a predetermined equilibrium point. This convention has provided accurate predictions, as shown in the previous section; however, if the cell operation becomes highly dynamic (i.e. high applied currents), the linearisation around the point ρ, which is utilised in the model generation, will produce higher deviations from the full-order model. To account for this error term, Lee et al. introduced a methodology to linearise the state-space model for varying flux amplitudes by reforming the D matrix. This has been shown to improve system predictions during highly-dynamic operations. The remainder of this section aims to develop performance improvements for this linearisation.

The conventional linearisation terms required for online system modification are shown below and comprise non-linear hyperbolic functions. These equations utilise the corresponding conductivities for each sub-domain of the cell and the calculated reaction flux for the current time step to update the linearisation. This information enables the creation of an updated D matrix for the next time step, providing an improved prediction for the current operating conditions. First, utilising the initial value theorem to capture the D indices for each corresponding transfer function, the limit of the s-domain approaching ∞ is utilised for a redefinition of the $\nu(s)$ variable for the negative and positive domains. The negative domain representation is shown below, with the positive (ν_p) represented with a corresponding L_p replacement,

$$\nu_n(\infty) = L_n \sqrt{\frac{a_s(\frac{1}{\sigma^{\text{eff}}} + \frac{1}{\kappa^{\text{eff}}})}{R_{\text{tot}}}} \tag{3.86}$$

The next step in the relinearisation of the D matrix is to compute the corresponding contribution for each transfer function. Starting with the electrolyte potential transfer function (3.35), the initial value theorem response is represented below for spatial particles within the negative electrode domain with the subdomain particle location denoted as ψ and as $s \to \infty$,

$$\begin{aligned}\frac{\Phi_e(\psi,\infty)}{I_{\text{app}}(\infty)} =& \frac{L_n(\frac{\sigma_n^{\text{eff}}}{\kappa_n^{\text{eff}}})(1-\cosh(\frac{\psi\nu_n(\infty)}{L_n}))-\psi\nu_n(\infty)\sinh(\nu_n(\infty))}{A(\sigma_n^{\text{eff}}+\kappa_n^{\text{eff}})\nu_n(\infty)\sinh(\nu_n(\infty))} \\ &+\frac{L_n(\cosh(\nu_n(\infty))-\cosh(\frac{(L_n-\psi)\nu_n(\infty)}{L_n}))}{A(\sigma_n^{\text{eff}}+\kappa_n^{\text{eff}})\nu_n(\infty)\sinh(\nu_n(\infty))}\end{aligned} \tag{3.87}$$

For particles within the separator domain, the transfer function is reintroduced from equation (3.36) with the corresponding initial value theorem applied as,

$$\frac{[\widetilde{\Phi}_{e,1}(\psi,\infty)]_1}{I_{\text{app}}(\infty)} = \frac{L_n-\psi}{A\kappa_s^{\text{eff}}} + \frac{L_n((1-\frac{\sigma_n^{\text{eff}}}{\kappa_n^{\text{eff}}})\tanh(\frac{\nu_n(\infty)}{2})-\nu_n(\infty))}{A(\sigma_n^{\text{eff}}+\kappa_n^{\text{eff}})\nu_n(\infty)} \tag{3.88}$$

Finally, the positive electrode domain equation (3.37) is represented with the initial value theorem applied as,

$$\begin{aligned}\frac{[\widetilde{\Phi}_{e,1}(\psi,\infty)]_1}{I_{\text{app}}(\infty)} =& -\frac{L_s}{A\kappa_s^{\text{eff}}}+\frac{L_n((1-\frac{\sigma_n^{\text{eff}}}{\kappa_n^{\text{eff}}})\tanh(\frac{\nu_n(\infty)}{2})-\nu_n(\infty))}{A(\kappa_n^{\text{eff}}+\sigma_n^{\text{eff}})\nu_n(\infty)} \\ &-\frac{L_p(1+\frac{\sigma_p^{\text{eff}}}{\kappa_p^{\text{eff}}}\cosh(\nu_p(\infty)))}{A(\kappa_p^{\text{eff}}+\sigma_p^{\text{eff}})\sinh(\nu_p(\infty))\nu_p(\infty)} \\ &+\frac{L_p\cosh(\frac{(L_n+L_s-\psi)\nu_p(\infty)}{L_p})}{A(\kappa^{\text{eff}}+\sigma^{\text{eff}})\sinh(\nu_p(\infty))\nu_p(\infty)} \\ &+\frac{L_p\frac{\sigma_p^{\text{eff}}}{\kappa_p^{\text{eff}}}\cosh(\frac{(L-\psi)\nu_p(\infty)}{L_p})}{A(\kappa^{\text{eff}}+\sigma^{\text{eff}})\sinh(\nu_p(\infty))\nu_p(\infty)} \\ &+\frac{(L_n+L_s)-\psi}{A(\sigma_p^{\text{eff}}+\kappa_p^{\text{eff}})}\end{aligned} \tag{3.89}$$

This is continued for each of the introduced transfer functions in equation (3.69) with the updated definition of ν such that $s \to \infty$. The corresponding concentration transfer functions have a zero initial value as such, are given by a null vector of length corresponding to the number of spatial points predicted. The electrolyte concentration is defined as,

$$C_e = [0]_{\psi,1} \tag{3.90}$$

Likewise, for the solid phase concentration corresponding to the particle number in these domains (j),

$$C_{\text{surf}} = [0]_{j,1} \tag{3.91}$$

As these functions are numerically complex, with a requirement to compute at each time step, they can contribute to an increased computational time for real-time deployment. An alternative to computing these hyperbolic functions is presented by investigating surrogate machine learning models capturing the non-linear dynamics. A simplistic neural network is utilised for this investigation to enable architecture understanding and fast real-time computation. A feedforward network is created, with data generated through a Sobol design exploration from the hyperbolic system, covering the corresponding up to 5C. The architecture of this network is introduced in Figure 3.14.

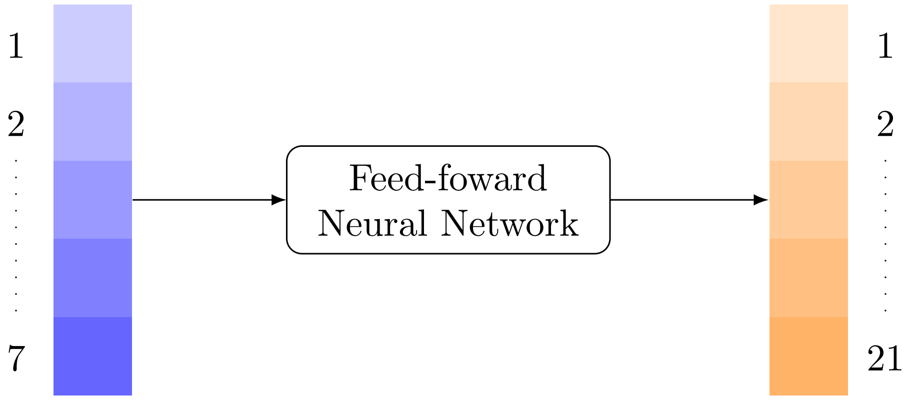


Fig. 3.14 Surrogate feedforward neural network comprising two hidden layers, mapping a seven-dimensional vector to the twenty-one-dimensional output vector.

To ensure a robust final model, the operating range for the exercised dataset extends past the cell operating limits to avoid network extrapolation. This dataset was split into three subsets and utilised for training (70%), testing (15%), and validation (15%) of the network. A seven-dimensional input vector, corresponding to the required parameters for computing equation (3.69), was mapped to two hidden layers comprising ten neurons each. These parameters comprise the domain specific dimensionless condensing variables, ν_n and ν_p, and the solid and electrolyte conductivies for the negative, σ_n^{eff} and κ_n^{eff}, positive, σ_p^{eff} and κ_p^{eff}, and the electrolyte conductivity in the separator, κ_s^{eff}. The output vector comprises the indices of the D matrix, with four electrolyte particles and two solid phase particles for each domain; this results in a twenty-one-dimensional output vector. The fitted performance of this surrogate model is presented in Figure 3.15, showcasing the capability for accurate predictions.

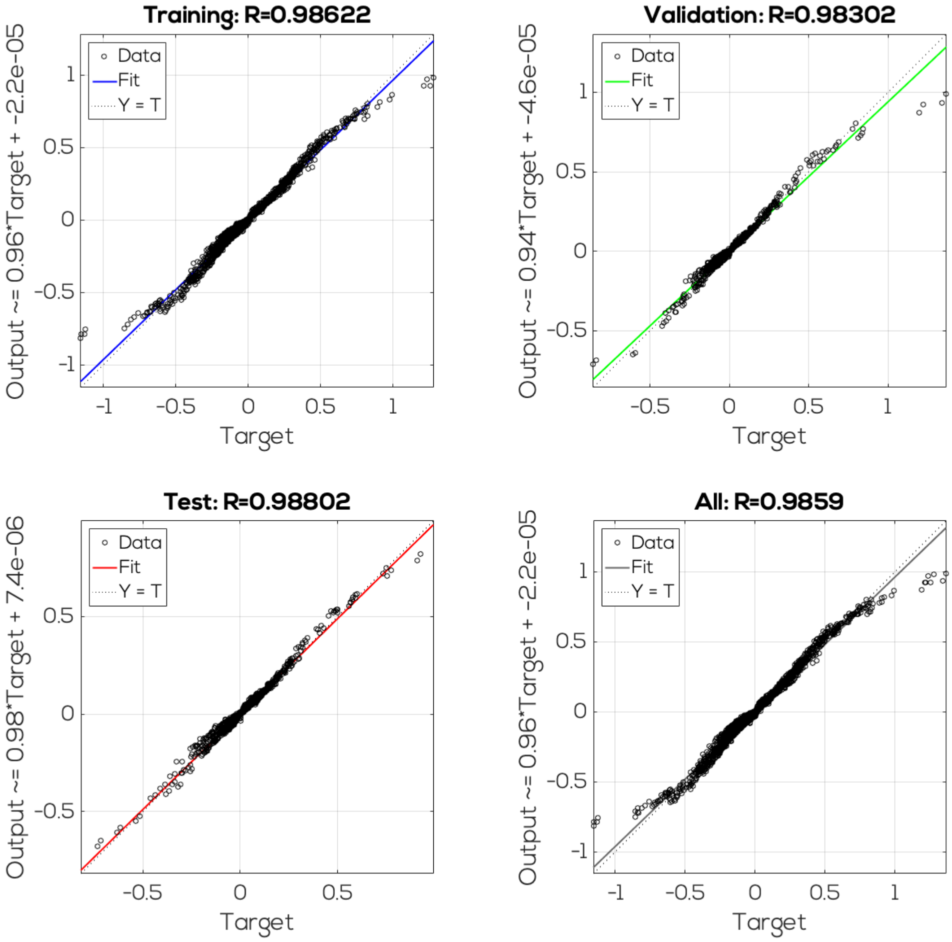


Fig. 3.15 Fitted performance of the surrogate neural network for training, testing, validation, as well as a combined performance for the total dataset.

The generated models coupled with the online linearisation method were compiled onto an ETAS ES910 embedded prototyping and interface module to validate the improved linearisation method. This unit provides an embedded reference for comparison with acceptable computation performance to ensure these models operate without overrunning the real-time requirements. The conventional linearisation and the neural network surrogate model were deployed, as well as an enhanced self-correcting equivalent circuit model [58]. This equivalent circuit model provides a reference to current battery management predictive models, as previously discussed in Chapter 2. Figure 3.16 below displays the computational requirements for each method when computing a 10-millisecond prediction window for a static discharge. This result is reported from the embedded controller for each time step and includes numerical jitter. This is equivalent to a final model sample time of 100 Hz, showcasing a higher operating condition for the CI-DRA. The conventional linearisation method results in a real-time factor of 200 and validates this methodologies capability in creating reduced-order models for real-time deployment without additional optimisations. The ANN linearisation method improves the real-time factor to 2.27×10^2 for this hardware, resulting in an approximate 13.5% reduction in computational requirements. Finally, the equivalent circuit model results in computationally efficient predictions with a real-time factor of 1.79×10^3; however, it does not provide any internal state information.

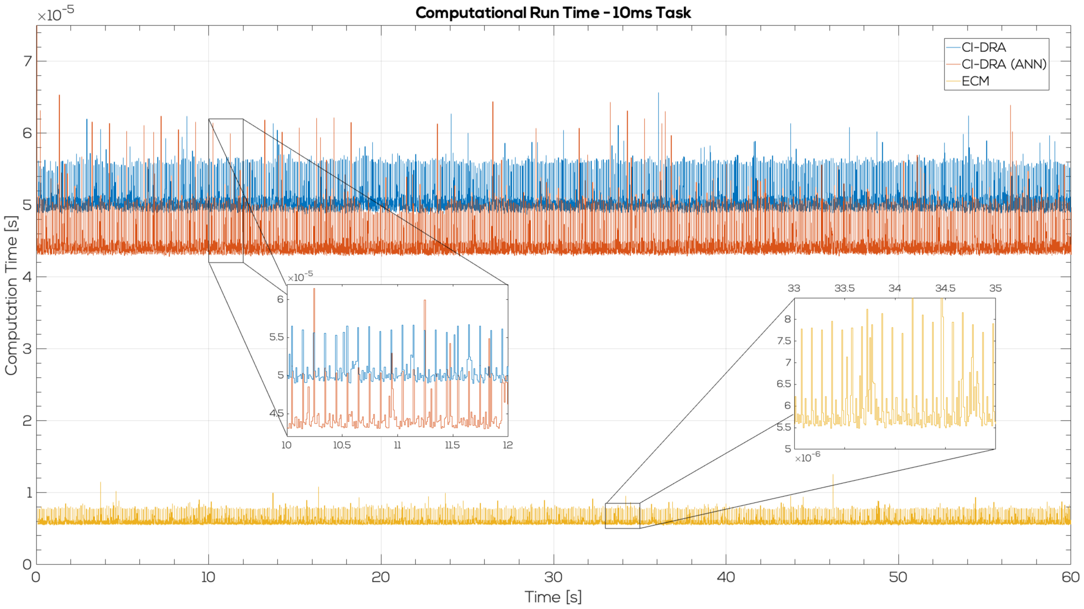


Fig. 3.16 Deployed model computational time for ECM, CI-DRA, and CI-DRA with surrogate linearisation.

With this investigation, the CI-DRA's real-time model creation capabilities are verified while presenting an improvement in the online linearisation method. Compared to the equivalent circuit model, the CI-DRA generation models have a higher computation burden; however, they still exceed the required real-time factor on the deployed hardware. Sizing of the generated model and utilisation of the surrogate linearisation method provide calibratable options to ensure an acceptable solution for varying hardware selection.

3.4 Concluding Remarks

This chapter presents an open-source modelling package, LiiBRA.jl, developed in Julia to create and simulate real-time capable electrochemical models. An improved realisation algorithm (CI-DRA) is presented with computational implementation discussed and results showing improvements over the conventional method. This work presents capabilities in offline model creation and expansion into in-vehicle creation via ARM compilation. This advancement was feasible due to the improvements enabled by the CI-DRA, the high-performance capabilities of LiiBRA.jl, and the ease of ARM-based compilation. This improvement opens the future potential of parameterisation for in-vehicle online models, providing a vital mechanism for individualised pack degradation predictions over its lifetime due to its physics-informed capabilities. This package provides a mean value improvement over the presented MATLAB CI-DRA implementation of 3.51 times. For ARM deployment, this package provides a modest 1.53 times decrease in performance compared to an equivalent x86 characterisation. Investigations showed a computational solution time of 6.06 seconds per model for ARM-based generation providing a total model creation time of three minutes.

An investigation into the CI-DRA's capabilities over the conventional DRA was presented, showcasing a performance improvement of 21.7% for 12 hours of transfer function system response sampling. This was continued into a framework variable sensitivity analysis, which presented the CI-DRA's dependencies on block Hankel size, the number of spatial particles in the electrolyte, and reduced system order. Experimental validation of the CI-DRA was completed, with voltage prediction of a WLTP 3B drive-cycle resulting in an RMSE value of 3.67mV. Finally, the generated models were deployed onto an embedded target to validate their real-time capabilities. This deployment enabled an additional improvement of 13.5% over the conventional DRA via the use of neural network surrogate models to enable online linearisation of the generated model. To conclude, the creation of LiiBRA.jl for this work has provided a performant step towards solving the two-language problem in real-time embedded computing. This is realised through the in-situ capabilities of LiiBRA.jl and the CI-DRA optimisation, enabling the in-place creation of the reduced-order models for embedded deployment. The translation of these models in corresponding C-code or real-time deployed Julia code can be completed through multiple open-source packages in the Julia environment. Further, it reduces resources in software creation and maintenance.

This chapter concludes the real-time realisation algorithm development. In the next chapters, this thesis expands into a fast, physics-informed modelling method for model-based cell design. This work will expand on a fast multi-scale tool for high-level usage and continues into an application for next-generation lithium-metal cells.

Chapter 4

Multi-scale battery modelling

4.1 Multi-scale battery design

Model-informed lithium-based battery design enables improvements in micro-scale parameter effects and faster design iterations and provides a formulation for application-specific battery design. As such, this chapter is implemented as a bridge chapter between the previously introduced improvements in reduced-order battery modelling and lithium-metal anode modelling. The aim is to introduce the open-source multi-scale modelling package, BattCalc.jl, to frame the need for lithium-metal modelling improvements from a systems-level perspective. As such, a methodology to improve the understanding of multi-scale battery design and its impacts on manufacturing targets is presented. This is completed by coupling micro-scale parameters to the system-level battery pack characteristics. This multi-scale predictive framework provides a mechanism for application-specific design and selection, enabling key benefits for high-performance applications with strict system requirements. Finally, an investigation that compares current-generation lithium-ion chemistry to potential next-generation lithium-metal chemistries for a high-energy and power-demanding motorsport application is presented. In this work, four length scales are discussed, starting from the electrode length scale, continuing to multiple electrodes at the stack level, expanding upwards to the full cell scale, and finally to the system level, which comprises multiple cells.

4.1.1 Electrode scale predictions

To start the multi-scale battery characteristic predictions, an introduction to the micro-scale parameters required for the design optimisation of lithium-based batteries is required. As discussed previously in this thesis, lithium-based batteries commonly utilise porous electrodes for lithium intercalation. The parameter governing the ratio of active material to total electrode volume is reintroduced as solid-phase porosity, ϵ_k where k denotes the specific electrode domain, i.e. (p) subscript refers to the positive electrode, (n) the electrode, and (s) the separator. This parameter is utilised in conjunction with the electrode coating thickness, L_k and the active material-specific volume, ν_k to calculate the areal mass loading, ρ_A for the domain-specific electrode as,

$$\rho_A = \frac{(1-\epsilon_k)L_k}{\nu_k}, \tag{4.1}$$

where the active material specific volume can be calculated through a summation of the active material composition introduced as,

$$\nu_k = \sum_{n=1}^{i} \frac{1}{N_n \rho_n}, \ \ i \in [1, N_t], \tag{4.2}$$

where i corresponds to each element in the electrode composition, N is defined as the molecular ratio of species in the active material with subscript t denoting the total composition, and ρ is the corresponding active material density. The total electrode mass can then be defined through the product of the current collector density, ρ_{cc} and thickness, L_{cc} to achieve areal mass for the current collector as,

$$M_k = A_k(\rho_{A,k} + \rho_{cc,k} \cdot L_{cc,k}). \tag{4.3}$$

Likewise, given the electrode loading and composition, it is possible to calculate the areal capacity, Q_A. The first method to accomplish this is through theoretical capacity from the electrode composition introduced as,

$$Q_{A,k} = \rho_{A,k} \cdot Q_{t,k} = \rho_{A,k} \cdot \frac{n_k F}{M_{a,k}}. \tag{4.4}$$

This method utilises Faraday's law of capacitance to predict the reversible specific capacity for a given composition; however, this specific capacity is not commonly achievable in the experimental operation of reversible intercalation electrodes due to multiple difficulties commonly corresponding to structural degradation and electrolyte voltage constraints. Alternatively, the prediction can be directly obtained from measured reversible capacity. This introduces an associated error term, (e_i), attached to the measurement from the experimental equipment; however, it provides a closer representation of the final electrode areal capacity. For this work, the reversible areal capacity values have been acquired through literature from experimental measurements. They have been previously shown in Table 1.1 and Table 1.2 for common cathode and anode compositions, respectively. Equation (4.5) below presents the areal capacity calculation for an experimentally acquired specific capacity, Q_k.

$$Q_{A,k} = \rho_{A,k} \cdot (Q_k + e_i). \tag{4.5}$$

Similarly to the total electrode mass, the total electrode capacity can be calculated utilising the electrode area, A_k,

$$Q_k = A_k \cdot Q_{A,k}. \tag{4.6}$$

The total electrode energy can be calculated via the summation of terminal voltage for a given applied current. This reduces to the nominal electrode voltage, V_{nom} and the previously calculated electrode capacity, Q_k.

$$E_k = Q_k \cdot V_{nom} \tag{4.7}$$

Finally, the electrode direct current resistance can be investigated by utilising a transmission line model (TLM) as previously presented in the literature [175, 176]. For this framework, the methodology presented by Morasch et al. [177] is utilised for predictions dependent on the corresponding limiting mechanism contributing to the total electrode impedance. As introduced by Morasch et al., the low-frequency impedance response, R_{k}, is defined as,

$$R_k = \boldsymbol{Z}_{\omega \to 0} = \sqrt{R_{\mathrm{ion,k}} R_{\mathrm{ct,k}}} \coth(\sqrt{R_{\mathrm{ion,k}} / R_{\mathrm{ct,k}}}), \tag{4.8}$$

where $R_{\mathrm{ion,k}}$ is the ionic resistance, with $R_{\mathrm{ct,k}}$ representing the charge transfer resistance, and $\boldsymbol{Z}_{\omega \to 0}$ denoting the complex impedance of the electrode at zero frequency. $R_{\mathrm{ion,k}}$ and $R_{\mathrm{ct,k}}$ and commonly experimentally fit, and for this work and experiment fit from the data presented in Ogihara et al. [176] is utilised. This experimental fit can be found in the open-source repository . Utilising the representations of $R_{\mathrm{ion,k}}$ and $R_{\mathrm{ct,k}}$, it is possible to predict the low-frequency resistance from equation (4.8) above.

4.1.2 Multi-electrode Scale

Utilising the above electrode definitions, it is possible to expand from a singular electrode scale to a multi-electrode in a stack orientation. This requires additional representations for the separator and current collectors, but as those components are non-active, their contributions only contribute to the mass calculations. Figure 4.1 below presents the stack orientation for three configurations: a lithium-ion cell, a lithium-metal cell with excess lithium, and an anode-free lithium-metal cell. The stack definition includes two positive electrodes, two negative electrodes, and two separators; however, the individual electrode layers are rotated in orientation to minimise material requirements due to the double coating on the electrodes. The double coating reduces the required cell materials via a shared current collector, which improves energy density and reduces manufacturing costs.

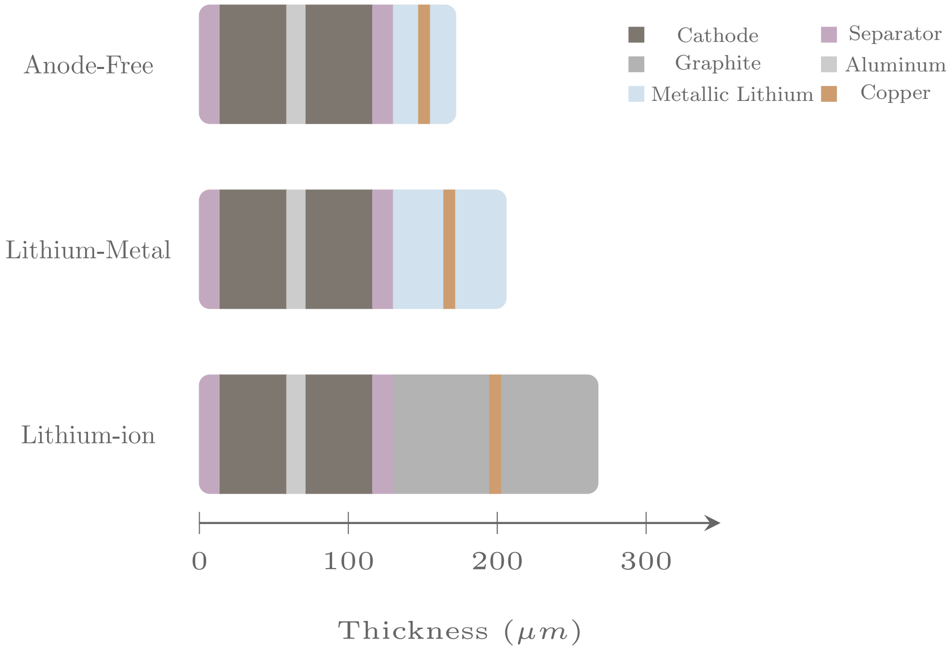


Fig. 4.1 Lithium-based stack definition with anode-free lithium-metal, 2:1 lithium excess lithium-metal, and intercalation-based lithium-ion.

For stack scale predictions, the corresponding mass and volume of the components in Figure 4.1 are included; however, cell casing and tabs are ignored and will be introduced at the cell scale. The total summation of components is shown below for mass-specific summation,

$$M_{st} = \sum_{i=1}^{n} M_i = 2M_{cath} + 2M_{anode} + 2M_{sep} + M_{ccp} + M_{ccn} + M_{electrolyte}, \quad (4.9)$$

and for volume-specific, the component masses are replaced with their corresponding volumes as,

$$V_{st} = \sum_{i=1}^{n} V_i = 2V_{cath} + 2V_{anode} + 2V_{sep} + V_{ccp} + V_{ccn} + V_{electrolyte}. \quad (4.10)$$

Due to the coupled exchange of lithium ions during oxidation and reduction for each electrode, the reversible capacity for the stack is defined as the minimum reversible capacity between the positive electrode, Q_p and the negative electrode, Q_n, as shown below,

$$Q_{st} = \min(Q_p, Q_n). \quad (4.11)$$

Furthermore, the stack energy density can then be defined with respect to component mass or volume as introduced in equations (4.9) and (4.10). The mass-specific energy density is introduced as,

$$E_{st} = \frac{Q_{st} \cdot (V_{nom}^{pos} - V_{nom}^{neg})}{M_{st}}, \quad (4.12)$$

where V_{nom} for each electrode is previously defined in Table 1.1. Similarly, the corresponding stack-scale power density is obtained and shown as,

$$P_{st} = \frac{I_{st,max} \cdot (V_{nom}^{pos} - V_{nom}^{neg})}{M_{st}}. \quad (4.13)$$

where $I_{st,max}$ is defined as the rated maximum constant current for the stack. Finally the stack direct current resistance is defined through a summation of electrode resistances as,

$$R_{st} = \sum_{i=1}^{n} R_i. \tag{4.14}$$

At this point, the stack scale predictions can be completed and can be used for design investigations at a coin-cell scale. Predicted values at this scale are commonly difficult with error terms, including manufacturing defects and equipment uncertainties weighted higher due to the perspective signal-to-noise ratio. To improve these issues, recent literature has aimed to standardise the manufacturing of coin-cells [178]. In the next section, the length scale will be increased further, and cell-level predictions will be investigated.

4.1.3 Cell scale models

Having discussed the electrode and stack length scales, the next step is to expand further into cell-level predictions. The main additions for cell-scale predictions include the addition of non-active mass components required for cell packaging and the coupling of multiple stacks to achieve full-cell predictions. For this work, a methodology to predict pouch cell geometries is presented; however, cylindrical and prismatic geometries follow similar methodologies, with the non-active mass varying due to the differing geometry. For the non-active mass summations, the first step is to define the geometric cell area,

$$A_{cell} = W_{cell} \cdot L_{cell}. \tag{4.15}$$

As well the cell thickness is defined by the stack thickness and the corresponding number of layers,

$$T_{cell} = T_{st} \cdot N_{st}. \tag{4.16}$$

Given the total cell dimensions, it is then possible to calculate the cell casing mass for a given casing thickness T_c,

$$M_{casing} = 2 \cdot (A_{cell} + T_{cell} \cdot (L_{cell} + W_{cell})) \cdot (T_c \cdot \rho_c) \tag{4.17}$$

leading to the full cell mass calculation of,

$$M_{cell} = M_{st} \cdot N_{st} + M_{tab,pos} + M_{tab,neg} + M_{casing}. \tag{4.18}$$

Next, the full cell resistance can be quantified as,

$$R_{cell} = \frac{R_{st}}{N_{st}} \tag{4.19}$$

Cell capacity, (Q_{cell}), energy, (E_{cell}), voltage, (V_{cell}), and corresponding densities are directly scaled from the stack predictions based on the number of stacks included in the cell. As the electrochemical calculations are defined at stack length scale, and lower, from this point and higher the main additions will be including the required non-active material needed for safety, storage, and electrical connections.

4.1.4 Multi-cell definitions

Stepping from cell scale to pack scale involves the inclusion of safety considerations such as protective outer casing, thermal propagation deterrents, and thermal management systems, as well as electrical circuitry such as high-voltage busbars, voltage sensing, relays, and battery management systems. The overall electrical configuration of the pack defines voltage characteristics as such,

$$V_{pack} = V_{cell} \cdot N_{series}, \tag{4.20}$$

where N_{series} defines the number of cells connected in series. The capacity is presented as,

$$Q_{pack} = Q_{cell} \cdot N_{parallel}, \tag{4.21}$$

where $N_{parallel}$ defines the number of parallel strings connected in the pack. Additionally, the full pack energy is introduced as,

$$E_{chem} = \sum_{i=1}^{N} E_i. \tag{4.22}$$

where E_i represents the each full-cell energy contribute such that, $E_i = E_{cell,i}$. Given the total electrical configuration of the pack, it is now possible to summarise the overall DC resistance utilising Thévenin's theorem as,

$$R_{pack} = 1/(1/R_{string} * N_{parallel}), \tag{4.23}$$

where R_{string} is defined as the summation of resistances for the total number of cells in series,

$$R_{string} = \sum_{i=1}^{N} R_i. \tag{4.24}$$

Next, given the total resistance of the pack, the thermal power due to ohmic heating can be calculated as,

$$P_{pack,\Omega} = I_{app}^2 \cdot R_{pack}. \tag{4.25}$$

It is then possible to predict the energy available in the pack by assuming ohmic heating forms the majority of the electrical losses. This is completed by integrating the ohmic power loss over the corresponding time vector, I_{app} as such,

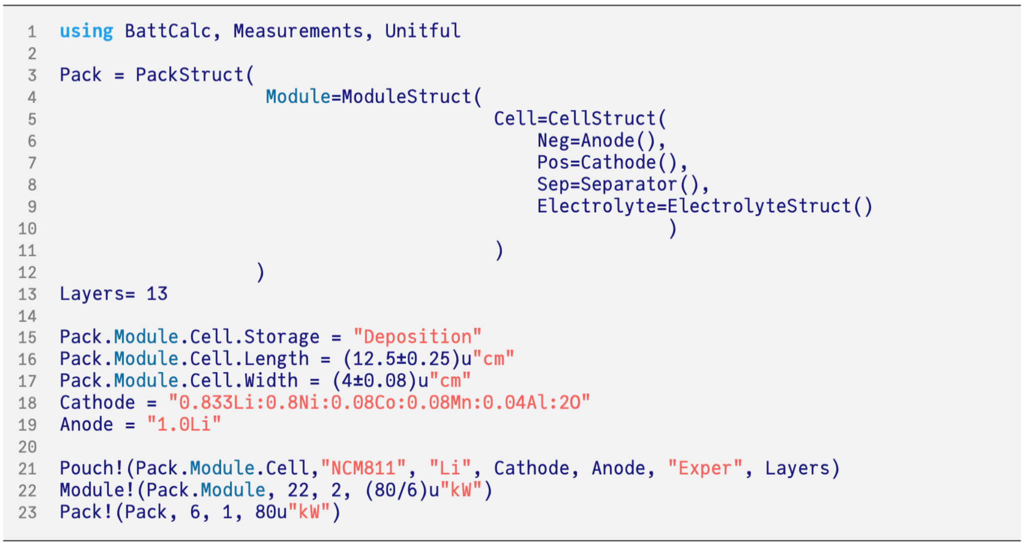
$$E_{pack,usable} = E_{pack,chem} - R_{\Omega} \int I_{app}^2 \, dt. \tag{4.26}$$

At this scale, the remaining calculations mainly comprise summations of the non-active components based on the predicted volume of the electrochemical components. The development of an open-source software package that can meet user-specific requirements for the predictions discussed above is discussed in the next section.

4.2 Software architecture

A Julia package has been developed to implement the above equations and act as an open-source tool for design iterations and predictions. This package, BattCalc.jl provides a fast method to investigate the differing length-scale effects on final cell and pack characterisations. As this type of analysis removes the spatial resolution and variations from the defined variables previously introduced, such as coating thickness, porosity, and loading, care must be taken to ensure predictions are achievable. To provide the user with an understanding of parameter sensitivities and potential non-homogeneity effects, propagated uncertainty analysis spans the predicted scales. While this doesn't guarantee an exact prediction of the full system dynamics or capabilities, it is useful for coupling potential chemistries and differing lithium storage mechanisms when predicting trends. This package also aims to expand dataset understanding and is actively being developed to enable the coupling of experimentally acquired data.

Example usage of the BattCalc.jl package is presented in Figure 4.2 below, with the pack structure built through wrapping the cell structure, which comprises the stack and electrode scale parameterisation. Following that, the number of stack layers for the formed pouch cell is set. At this point, the exported functions, "Pouch!", "Module!" and "Pack!" can be called; however, in the example, the default parameterisation of the lithium storage mechanism is overwritten, as well as the cell length and width, followed by the cathode and anode compositions. The "Pouch!" function is called with the updated variable definitions, followed by the "Module!" function with an electrical layout of 22 cells in series with two parallel strings corresponding to a 100V module utilising high-voltage lithium-metal cells, i.e. 4.5V terminal voltage maximum and a mean power definition of $13.3\bar{3}$ kW. Finally, the system-level pack is formed through the "Pack!" function with a high-voltage electrical configuration of 6 modules in series with 1 parallel string and corresponding power.



```julia
using BattCalc, Measurements, Unitful

Pack = PackStruct(
                   Module=ModuleStruct(
                                        Cell=CellStruct(
                                            Neg=Anode(),
                                            Pos=Cathode(),
                                            Sep=Separator(),
                                            Electrolyte=ElectrolyteStruct()
                                                        )
                                        )
                  )
Layers= 13

Pack.Module.Cell.Storage = "Deposition"
Pack.Module.Cell.Length = (12.5±0.25)u"cm"
Pack.Module.Cell.Width = (4±0.08)u"cm"
Cathode = "0.833Li:0.8Ni:0.08Co:0.08Mn:0.04Al:2O"
Anode = "1.0Li"

Pouch!(Pack.Module.Cell,"NCM811", "Li", Cathode, Anode, "Exper", Layers)
Module!(Pack.Module, 22, 2, (80/6)u"kW")
Pack!(Pack, 6, 1, 80u"kW")
```

Fig. 4.2 Example usage of BattCalc.jl, providing a simple method for predicting battery characteristics.

4.3 Future chemistries for motorsport applications

To introduce BattCalc.jl's capabilities and the benefits of multi-scale modelling for battery design, a comparative investigation of three lithium-based batteries, each with identical positive electrodes but varying negative electrode storage mechanisms, is presented. This study aims to provide insight into the characteristics of varying lithium-based storage cells for a high-power motorsport application. As motorsport applications have higher requirements for onboard power when compared to commercial passenger vehicles, the current utilised lithium-ion solutions are developed with a bias toward power density at the expense of energy density. This has resulted in shorter events due to the reduction of onboard energy and lower velocities compared to internal combustion alternatives. This compromise is constant across multiple applications requiring high-power capabilities, including aerospace and heavy-duty transportation.

This investigation includes energy-dense lithium-metal batteries with the aim to improve understanding of the capabilities for the high-power applications and is achieved through an example application utilising a Formula Student vehicle. The first step in this investigation was to define the cells' electrode and stack level characteristics, ensuring characteristics match the literature and are scaled to match the final application. Next, pack characteristics are predicted with the vehicle level parameters such as mass and energy provided to a transient vehicle dynamic simulation. This simulation is completed utilising AVL VSM [179] software with a parameterised electric Formula Student competition drive-cycle. Figure 4.3 below provides a visual representation of the geometric trajectory of this high-performance drive-cycle obtained from experimental testing.

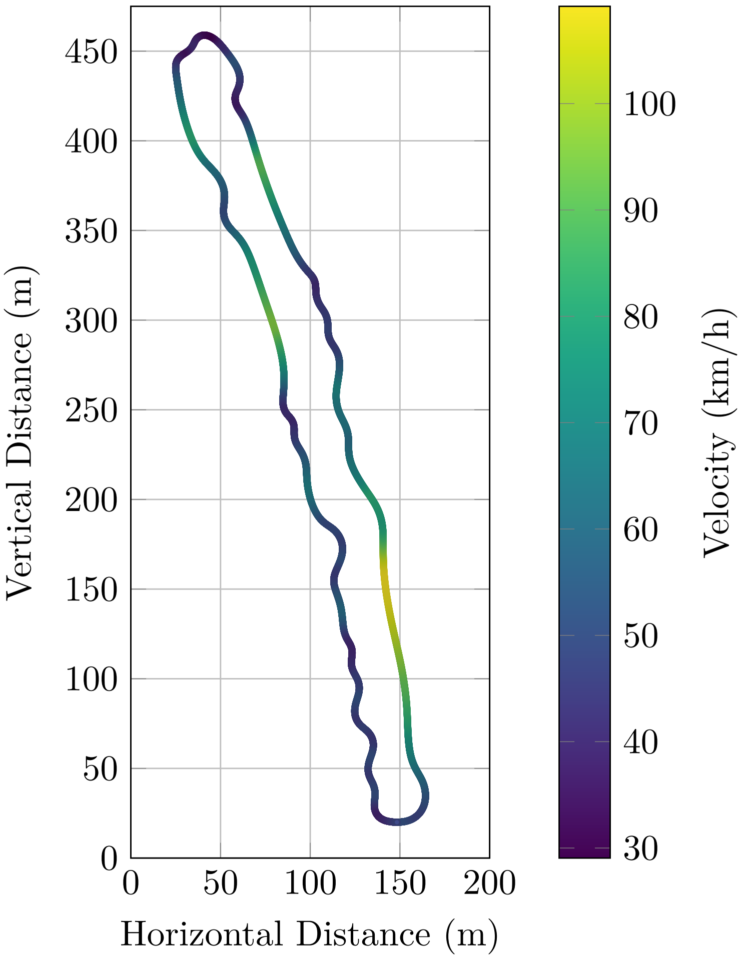


Fig. 4.3 Single lap trajectory of the 22-lap drive-cycle with the corresponding velocity.

The vehicle parameterisation for the numerical vehicle dynamic simulation is presented in Table 4.1 below and represents generalised characteristics for a Formula Student electric vehicle. This parameterisation has completed a validation process ensuring system response aligns with experimental captured data. As the main focus of this thesis is on electrochemical modelling and design, the validation process is left out; however, the reader is pointed to [180] as it includes a range of experimentally recorded times for Formula Student vehicles completing this drive-cycle.

Table 4.1 Default (Lithium-ion) values for dynamic Formula Student vehicle simulation study.

Variable	Definition	Value
Vehicle Mass (kg)	Total vehicle mass excluding driver	180
Pack Energy ($kW \cdot hr$)	Available Onboard Pack Energy	5.5
Vehicle Power (kW)	Maximum Electrical Power Available from Pack	80
Drive-cycle Length (km)	Total length of driven cycle	21.96
Lift/Drag Coefficient	Coefficient of Lift to Coefficient of Drag Ratio	4.5
Pack Voltage (V)	Maximum Pack Voltage	600

Utilising BattCalc.jl, an example cell geometry was selected as a pouch with approximately 5 Ah. This capacity enables multiple electrical configurations to achieve the energy and voltage target shown in Table 4.1 above, providing additional flexibility in the system-level configuration. Next, to ensure direct comparisons between the differing anode storage methods, the cathode parameterisation remains constant and is sized to a high loading of ~ 24 $[mg/cm^2]$ with a 72.5 [µm] thickness. The lithium-ion anode is sized to achieve a negative/positive (N/P) capacity ratio of 1.1, with the lithium-metal cell including a $\sim$ 20µm lithium foil providing an excess of lithium of approximately 2.3 times. The anode-free cell contains electrodeposited lithium from cycling, with the thickness shown as the fully charged value. Each cell includes identical casing materials of 0.21mm thick aluminium and is filled with 1.6 $[g/Ah]$ of electrolyte. From the given electrode parameterisation, Table 4.2 displays the resultant predicted values up to cell scale. These predictions include an initialised uncertainty of two per cent full scale attached to each input variable. As shown, the lithium-metal cell benefits heavily from the removal of the porous anode while utilising electrodeposition for storage. While the anode-free cell provides the highest energy density, it should be noted that the lithium-metal with 2.3 times excess provides a viable solution that could enable high cycle life; however, with the high costs of lithium, this may not apply to every application.

Table 4.2 Theoretical analysis of lithium-based battery characteristics from electrode scale to stack scale.

	Lithium-Ion (NCM811)	Lithium-Metal (NCM811)	Anode-Free Lithium-Metal (NCM811)
Cathode			
Thickness (µm)	72.5 ± 1.45	72.5 ± 1.45	72.5 ± 1.45
Areal Capacity (mAh/cm^2)	4.42 ± 0.18	4.42 ± 0.18	4.42 ± 0.18
Loading (mg/cm^2)	23.9 ± 0.81	23.9 ± 0.81	23.9 ± 0.81
Anode			
Thickness (µm)	85.2 ± 1.70	43.5 ± 0.94	23.5 ± 0.47
Areal Capacity (mAh/cm^2)	3.9 ± 0.78	8.38 ± 0.25	4.53 ± 0.13
Loading (mg/cm^2)	14.4 ± 2.9	2.28 ± 0.049	1.23 ± 0.025
Electrolyte			
Mass (g)	0.62 ± 0.13	0.706 ± 0.038	0.706 ± 0.038
Stack			
Area (cm^2)	50 ± 1.4	50 ± 1.4	50 ± 1.4
Mass (g)	5.21 ± 0.45	$4.07 \pm .16$	3.97 ± 0.16
Thickness (µm)	367 ± 4.5	284 ± 3.5	244 ± 3.1
Nominal Voltage (V)	3.67 ± 0.014	3.93 ± 0.01	3.93 ± 0.01
Capacity (mAh)	390 ± 79	442 ± 22	442 ± 22
Energy (Wh)	1.43 ± 0.29	1.74 ± 0.086	1.74 ± 0.086
Energy Density (Wh/kg)	$\mathbf{275 \pm 34}$	$\mathbf{426 \pm 9.0}$	$\mathbf{437 \pm 9.1}$
Volumetric Energy Density (Wh/L)	780 ± 160	1222 ± 46	1422 ± 52

Utilising the results presented above from BattCalc.jl, it is possible to predict the pack level characteristics and investigate full vehicle performance. The pack level predictions are presented in Table 4.3 below and have been sized to match the vehicle parameterisation introduced in Table 4.1 above. This sizing has resulted in differing cell capacities for the lithium-ion and lithium-metal cells, as the aim is to align the final pack energy for each configuration. The module and outer pack casing characteristics are also held constant across each configuration. This results in the mass differences being propagated from the electrochemical volume requirements and enables the lithium-metal solutions to maintain their improvements predicted at the cell scale. The electrical configuration for each cell variation has been sized to match the maximum vehicle voltage, with module energy limited to 100 volts to maintain safe assembly processes.

Table 4.3 Theoretical analysis of lithium-based battery characteristics for pack scale motorsport applications.

	Lithium-Ion (NCM811)	Lithium-Metal (NCM811)	Anode-Free Lithium-Metal (NCM811)
Cell			
Capacity (Ah)	5.07 ± 0.23	5.30 ± 0.23	5.30 ± 0.23
Nominal Voltage (V)	3.67	3.84	3.84
Mass (g)	74.6 ± 2.1	55.5 ± 1.8	54.2 ± 1.8
DC Resistance ($m\Omega$)	15.5 ± 0.45	23.3 ± 0.7	29.0 ± 0.86
Energy (Wh)	18.6 ± 0.84	20.8 ± 0.89	20.8 ± 0.89
Energy Density (Wh/kg)	249 ± 12	375 ± 7.3	384 ± 7.4
Volumetric Energy Density (Wh/L)	716 ± 39	1088 ± 31	1244 ± 34
Module			
Cell Configuration	24s2p	22s2p	22s2p
Mass (kg)	4.58 ± 0.1	3.44 ± 0.084	3.38 ± 0.08
Energy (kWh)	0.893 ± 0.04	0.916 ± 0.039	0.916 ± 0.039
Energy Density (Wh/kg)	195 ± 9.2	266 ± 6.6	271 ± 6.7
Pack			
Module Configuration	6s1p	6s1p	6s1p
Peak Voltage (V)	605	594	594
Mass (kg)	$\mathbf{36.0 \pm 0.83}$	$\mathbf{27.7 \pm 0.67}$	$\mathbf{26.9 \pm 0.66}$
Energy (kWh)	5.36 ± 0.24	5.50 ± 0.24	5.50 ± 0.24
Energy Density (Wh/kg)	$\mathbf{149 \pm 7.2}$	$\mathbf{199 \pm 5.4}$	$\mathbf{204 \pm 5.5}$
Volumetric Energy Density (Wh/L)	558 ± 30	808 ± 24	904 ± 26

Next, the coupled variables used for the transient simulation gained from the electrochemical prediction include the total pack mass and available energy. The transient simulation is completed across the total drive cycle, defined as the time to complete 22 cycles of the trajectory shown in Figure 4.3. Figure 4.4 below presents a predicted single lap from the total drive-cycle when parameterised with current-generation lithium-ion batteries. This parameterised is used for the initial sizing of the onboard energy requirements.

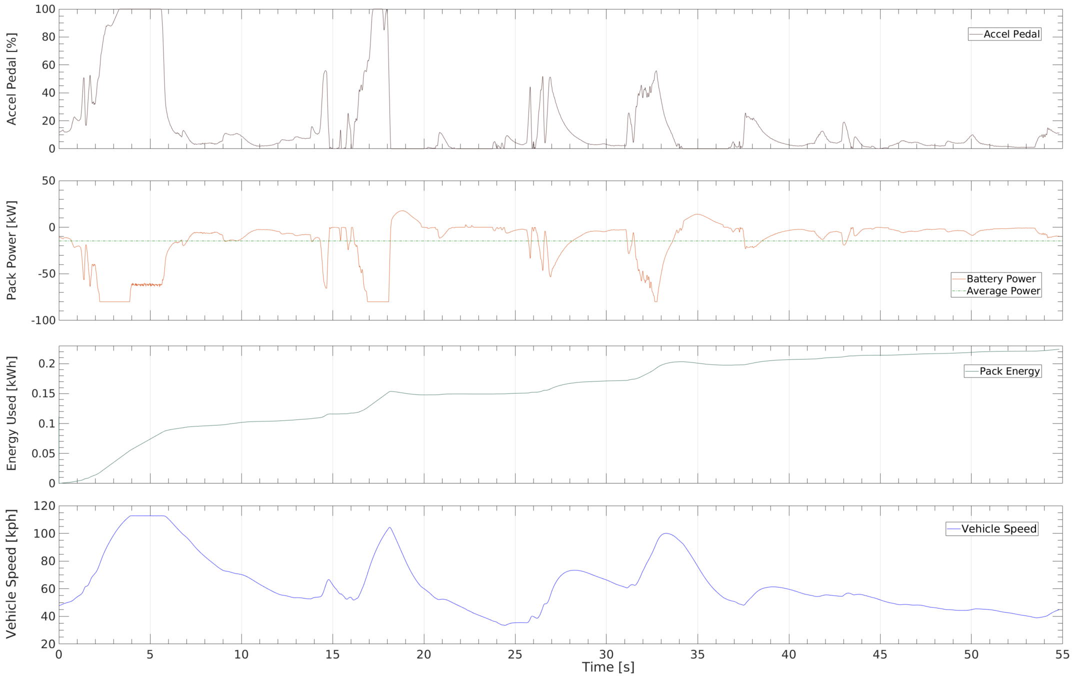


Fig. 4.4 Single lap simulation of endurance drive cycle for a Formula Student electric vehicle. Negative power values denote discharging energy from the battery pack, with the energy convention reversed.

Numerical results from the transient simulation for each cell configuration are displayed in Table 4.4 below. From these results, the anode-free lithium metal cell displays an improvement of 7.52 seconds with a 28.8 Wh reduction of used energy. As this investigation does not include vehicle optimisation for each cell configuration, it is expected that even further improvements will be achieved due to the large volume reduction in the pack for the anode-free configuration. Even without these additional improvements, this investigation provides an initial direction for enabling energy-dense lithium-metal batteries in high-performance applications. Additionally, due to the low volume, lithium foil-based lithium-metal cells may provide a solution that provides a large percentage of the benefits enabled by anode-free configuration while potentially achieving a longer cycle life.

Table 4.4 Results for transient vehicle dynamic simulation for each battery chemistry and configuration relative to the anode-free lithium-metal cell.

	Lithium-Ion (NCM811)	Lithium-Metal (NCM811)	Anode-Free Lithium-Metal (NCM811)
Results			
Mass Delta (kg)	$+9.1 \pm 1.1$	$+0.8 \pm 0.94$	0.0
Used Energy Delta (Wh)	+28.8	+2.53	0.0
Energy Density (Wh/kg)	-55 ± 9.1	-5 ± 7.7	0.0
Time Delta (s)	+7.62	+0.67	0.0

4.4 Concluding Remarks

This chapter presents a multi-scale battery prediction methodology that enables an improved understanding of underlying parametric effects. This methodology was developed utilising fast multi-scale relations with the numerical package, BattCalc.jl developed. This work has presented capabilities for investigating lithium storage mechanisms and the applicability of differing chemistries in demanding applications. Furthermore, an investigation of storage mechanisms for lithium-metal applicability in motorsports was assessed through a transient vehicle dynamic simulation parameterised for a Formula Student vehicle with differing cell chemistries. This investigation presented modest improvements of 7.62 seconds over the 22 km drive-cycle for an anode-free lithium-metal battery pack while consuming 28.81 Wh less energy when compared to the current generation lithium-ion configuration.

Overall, this chapter presented a methodology for high-level design capabilities that will be expanded upon in the next chapter, where a higher fidelity continuum model is presented for lithium-metal anode modelling. This work aims to expand on the presented high-level methodology presented in this chapter, stepping into lower-length scales to predict the evolution of the lithium-metal anode and improve system understanding for improved performance.

Chapter 5

Fast lithium-metal models

Lithium-metal battery models can improve understanding of underlying lithium morphology evolution and corresponding degradation mechanisms, further enabling the electrification of fields such as aerospace and heavy-duty transportation. This chapter presents a novel continuum modelling methodology for predicting lithium-metal anode thickness, potential, and composition evolution. This work presents a fast, open-source coupled partial differential model solved with phase-field methods that enable predictions from a predefined initialised anode formation. As previously discussed in Chapter 4 above, the benefits that lithium-metal cells provide can enable the electrification of many high-performance applications not currently accessible with current-generation lithium-ion batteries. This advancement has been actively under development for the last decade with multiple start-up companies such as SES, Quatumscape, Cuberg, and Sion Power, aiming to deliver commercially viable solutions with energy densities as high 417 Wh/kg [181] with a corresponding capacity of 107 Ah. Additionally, multiple research institutes and funded consortia have been advancing lithium-metal capabilities such as the Battery500 [182], SOLBAT [183], SAFELiMOVE [184], and HIDDEN [185]. Each of these projects has developed experimental methodologies to advance lithium-metal battery manufacturing and design; however, there is still a large gap in model-based solutions for design and performance predictions. This work contributes a coupled physics-based phase-field and neural solver framework for modelling of lithium-metal batteries. While experimental methods are commonly utilised to advance manufacturing, there are multiple benefits from coupling numerical models, such as reducing product iterations, improving underlying system understanding and enabling advanced battery management.

One of the challenges in developing lithium-metal battery models is the varying numerical domain caused by lithium electrodeposition. In an anode-free cell, this can result in a domain size variation of 20 times or more, depending on the cell size, due to the nature of electrodeposition compared to intercalation. This domain change can be complex to mathematically represent as numerical discretisation is commonly developed for fixed domains. An additional difficulty that is experienced when modelling lithium-metal electrodeposition is the spatial variation in lithium morphology across the current collector. This morphology variation has been one of the main challenges in developing commercial lithium-metal cells, as it can result in the isolation of electrodeposited lithium, which exacerbates capacity degradation. Under certain operating conditions, this morphology can evolve into a mossy, tree-like dendritic structure, as shown in Figure 5.1 below.

5.1 Lithium-Anode Electrodeposition Model

The evolution of lithium morphology towards a dendritic, mossy structure is one of the main challenges limiting lithium metal batteries' long cycle life capability. As introduced in Section 2.2.1, phase-field representation can provide a capable method to predict this evolution. As the need for an improved continuum model of lithium-metal evolution has been presented, the next step is to derive the

Fig. 5.1 Mossy lithium morphology (left) and corresponding ideal dense, uniform morphology (right)

system of equations and methodology utilised in this chapter. The derivation of the system follows, with this work referencing the initial derivation in Jang et al. [2].

First, the system represents phases as values of range (0,1), i.e. the first phase corresponds to spatial values of zero, with the second phase represented through values of one. This enables the prediction of the two-phase domain through the translation to the bounded numerical range. Given this spatial representation, the model's initial geometry (Y_0) can be defined spatially as an array of values within this range. This system can be utilised in the two-dimensional case for varying, non-uniform anode geometry aligning with experimental morphology evolution. An example of this two-dimensional geometry is constructed for a gaussian seed rectangular example representing the lithium metal anode; this is mathematically represented as,

$$z_0[x,y] = max\left(eps(), 0.5 + 0.5 * \tanh\left(\frac{(y - t_r(x))}{\sqrt{w*2}}\right)\right), \tag{5.1}$$

where $eps()$ represents the minimum float value on the computational hardware and is used to avoid ill-conditioned calculations. The variable y denotes the spatial location in the vertical domain, and t_r defines an initial gaussian seed in the vertical domain and is defined as,

$$t_r(x) = T + \exp(-B * (x - C)^2) \tag{5.2}$$

where T defines the initial thickness in the corresponding domain and is of range (0,1). B and C are the gaussian input variables defining the location and shape in the domain. Finally, $w = h/2$ where h is the unit cell dimensions for the simulated domain. A visual representation for a domain of length 40 µm by 40 µm, corresponding to $h = 40e - 6$ and a t_r representation for three gaussian seeds is shown in Figure 5.2.

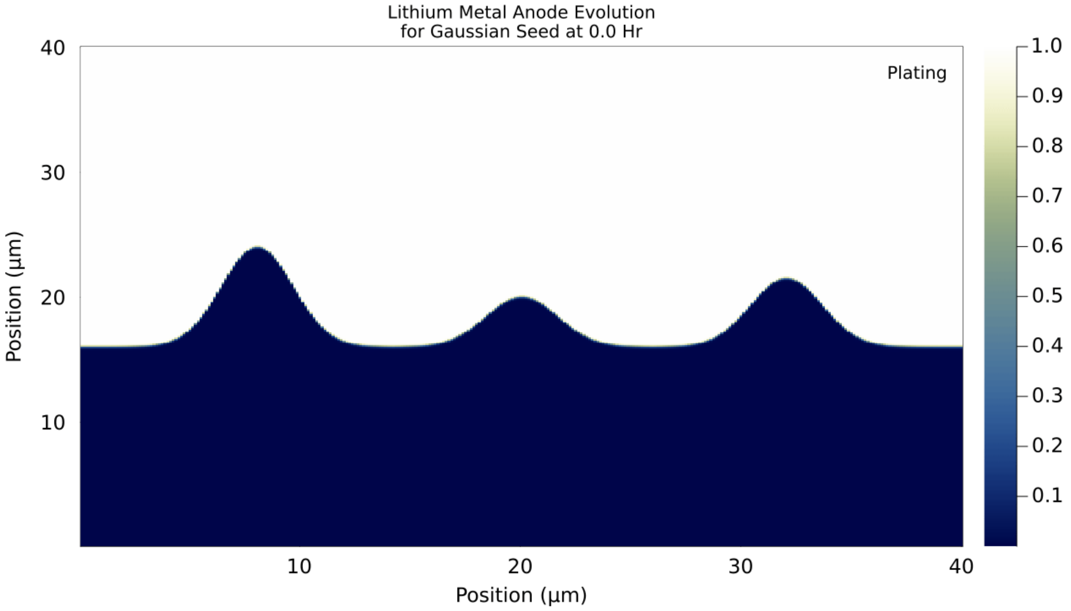


Fig. 5.2 Two-dimensional phase-field representation of a lithium-metal anode with varying gaussian seed geometry using BattPhase.jl

Once the numerical system has been initialised, the system dynamics are represented with linear kinetics and Neumann boundary conditions. Two assumptions are made in the derivation of this model, (1) Only linear reaction kinetics, and (2) The moving phase boundary is proportional to the flux at the interface only. The second assumption follows that concentration gradients are ignored, resulting in the following second-order derivative for dimensionless electrolyte potential,

$$\left(\frac{\partial^2\phi}{\partial y^2}\right) + \left(\frac{\partial^2\phi}{\partial x^2}\right) = 0, \tag{5.3}$$

with a dimensionless applied current given at the cathode surface as δ bounding the first-order derivate of dimensionless potential as,

$$\left.\frac{\partial\phi}{\partial y}\right|_{y=1} = \delta, \tag{5.4}$$

with the corresponding Neumann condition at the current collector of the anode (i.e. y=0),

$$\left.\frac{\partial\phi}{\partial y}\right|_{y=0} = 0, \tag{5.5}$$

as well as δ defined as,

$$\delta = \frac{I_{app}LF}{\kappa RT}, \tag{5.6}$$

where I_{app} is the applied current density, L is the domain length, and κ is the electrolyte conductivity. The horizontal domain is bounded through Neumann conditions at X = 0, and X = 1, shown as,

$$\left.\frac{\partial\phi}{\partial x}\right|_{x=0} = 0,\ \left.\frac{\partial\phi}{\partial x}\right|_{x=1} = 0. \tag{5.7}$$

Assuming linear second-order kinetics, it is possible to represent the boundary condition at the electrolyte and anode interface as,

$$\left.\frac{\partial\phi}{\partial y}\right|_{y=z} = k\phi, \tag{5.8}$$

where k is defined as the dimensionless exchange current density and z is the initial spatial location of the two-phase (anode/electrolyte) interface at initialisation.

The equation governing k is introduced as,

$$k = \frac{i_0 LF}{\kappa RT}, \tag{5.9}$$

where i_0 is the exchange current density. The moving phase boundary can then be represented as,

$$\frac{\partial z}{\partial t} = \nu k \phi, \tag{5.10}$$

where ν is defined as a constant relation between density, ρ, and molecular mass,M_w of the electrodeposited species, Li and is calculated as,

$$\nu = 3600 \cdot \frac{M_w I_{app}}{\rho F L n \delta}, \tag{5.11}$$

where n is the valiance electrons available for the reaction. The initial spatial construction then defines initial conditions for the moving phase boundary as,

$$z(0) = z_0. \tag{5.12}$$

The next step in this derivation is to utilise the Landau transformation [186] to modify the analytic solution for removal of the dependency on the moving boundary,

$$Y = \frac{y - z}{1 - z}, \ X = \frac{x - z}{1 - z}, \ t = \tau, \tag{5.13}$$

with the corresponding change of variable in the time domain introduced as τ, measured in hours. Combining equation (5.13) with (5.3) results in the following formation,

$$\left(\frac{\frac{\partial^2 \phi(Y,\tau)}{\partial Y^2}}{(1 - z(\tau))^2} \right) + \left(\frac{\frac{\partial^2 \phi(X,\tau)}{\partial X^2}}{(1 - z(\tau))^2} \right) = 0, \tag{5.14}$$

with a corresponding transformed of equations (5.4) and (5.8) for the Y domain introduced as,

$$\left.\frac{\frac{\partial \phi(Y,\tau)}{\partial Y}}{1-z(\tau)}\right|_{Y=0} = k\phi(X,\tau),\ \left.\frac{\frac{\partial \phi(Y,\tau)}{\partial Y}}{1-z(\tau)}\right|_{Y=1} = \delta. \tag{5.15}$$

The X domain likewise has similar transformed equations,

$$\left.\frac{\frac{\partial \phi(X,\tau)}{\partial y}}{1-z(\tau)}\right|_{X=0} = 0,\ \left.\frac{\frac{\partial \phi(X,\tau)}{\partial X}}{1-z(\tau)}\right|_{X=1} = 0. \tag{5.16}$$

The moving phase boundary can then be represented in transformed form as,

$$\frac{\partial z(\tau)}{\partial \tau} = \nu k\phi(Y,\tau). \tag{5.17}$$

This summarises the analytical formation of the electrodeposited lithium-metal anode governing system for this work. The phase-field formation is now presented as introduced in [2] through the following steps, with γ representing the phase-field parameter due to the previous usage of the common phase-field parameter ϕ. For this approach, an immersed-boundary method is utilised and enables movement of boundary condition term equation (5.15) to the source term equation (5.14). The two-dimensional potential equation then becomes,

$$\left(\frac{\partial}{\partial Y}\right)\left(\gamma\frac{\partial \phi}{\partial Y}\right) + \left(\frac{\partial}{\partial X}\right)\left(\gamma\frac{\partial \phi}{\partial X}\right) = k\phi\sqrt{\left(\frac{\partial \gamma}{\partial Y}\right)^2 + \left(\frac{\partial \gamma}{\partial X}\right)^2}, \tag{5.18}$$

with the moving phase boundary evolution becoming,

$$\frac{\partial \gamma}{\partial \tau} + \nu k\phi\sqrt{\left(\frac{\partial \gamma}{\partial Y}\right)^2 + \left(\frac{\partial \gamma}{\partial X}\right)^2} = 0, \tag{5.19}$$

with a final set of boundary conditions for the phase field representation defined as,

$$\left.\frac{\partial \phi}{\partial X}\right|_{X=0} = 0, \ \left.\frac{\partial \gamma}{\partial X}\right|_{X=0} = 0, \left.\frac{\partial \phi}{\partial X}\right|_{X=1} = 0, \ \left.\frac{\partial \gamma}{\partial X}\right|_{X=1} = 0, \tag{5.20}$$

$$\left.\frac{\partial \phi}{\partial Y}\right|_{Y=0} = 0, \ \left.\frac{\partial \gamma}{\partial Y}\right|_{Y=0} = 0, \left.\frac{\partial \phi}{\partial Y}\right|_{Y=1} = \delta, \ \left.\frac{\partial \gamma}{\partial Y}\right|_{Y=1} = 0, \tag{5.21}$$

with initial moving phase boundary geometry as defined in 5.1 and mathematically represented as,

$$z_0 = \gamma(0) = \frac{1}{2} + \frac{1}{2} * \tanh\left(\frac{(y - t_r)}{\sqrt{w * 2}}\right). \tag{5.22}$$

In the next section, the numerical methods used to solve equations (5.18 - 5.19) for the given boundary conditions (5.20- 5.21) and initial conditions (5.22) are presented. This work follows the numerical path developed by Jang et al. [2], as this section aims to improve the framework presented numerically.

5.1.1 Numerical Methods

Having defined the lithium-metal anode representation in the previous section, the next step is investigating finite difference methods and numerical time integrators to solve the system of partial differential equations. Previously, many varying finite difference schemes have been utilised to approximate mathematical differentiation numerically. Each of these schemes has characteristics that lend them to improvements for specific system dynamics, such as the computational efficiency of the forward difference scheme. The central difference scheme is utilised in this work for numerical derivative approximation of the dimensionless potential. The phase-field equations are then represented with an upwind scheme providing improvements for non-oscillatory differential predictions around the moving phase boundary. This section utilised an Intel i9 10980-XE workstation with 64 GB of memory for computational performance investigations.

5.1.1.1 Central difference

Utilising the 3-point central difference scheme, the phase-field formation of potential equation (5.18) becomes,

$$\begin{aligned}&\frac{(\gamma_{i+1,j}+\gamma_{i,j})(\phi_{i+1,j}-\phi_{i,j})-(\gamma_{i,j}+\gamma_{i-1,j})(\phi_{i,j}-\phi_{i-1,j})}{2h^2}\\&+\frac{(\gamma_{i,j+1}+\gamma_{i,j})(\phi_{i,j+1}-\phi_{i,j})-(\gamma_{i,j}+\gamma_{i,j-1})(\phi_{i,j}-\phi_{i,j-1})}{2h^2}\\&-k\phi_{i,j}\sqrt{\left(\frac{\gamma_{i+1,j}-\gamma_{i-1,j}}{2h}\right)^2+\left(\frac{\gamma_{i,j+1}-\gamma_{i,j-1}}{2h}\right)^2}=0,\end{aligned} \tag{5.23}$$

with corresponding boundary condition forms as,

$$\begin{aligned}&\left(\frac{\gamma_{1,j}+\gamma_{0,j}}{2}\right)\left(\frac{\phi_{1,j}-\phi_{0,j}}{h}\right)=0,\left(\frac{\gamma_{N+1,j}+\gamma_{N,j}}{2}\right)\left(\frac{\phi_{N+1,j}-\phi_{N,j}}{h}\right)=0,\\&\left(\frac{\gamma_{i,1}+\gamma_{i,0}}{2}\right)\left(\frac{\phi_{i,1}-\phi_{i,0}}{h}\right)=0,\left(\frac{\gamma_{i,N+1}+\gamma_{i,N}}{2}\right)\left(\frac{\phi_{i,N+1}-\phi_{i,N}}{h}\right)=\delta.\end{aligned} \tag{5.24}$$

5.1.1.2 Upwind scheme

For the moving phase boundary parameter, an upwind scheme is utilised to discretise the Hamiltion-Jacobian equation to minimise oscillations during the computation. This is utilised as the discontinuities in the Hamilton-Jacobi would pose oscillatory challenges for the phase-field formation. The upwind formation is summarised below,

$$\begin{aligned}&\frac{d\gamma_{i,j}}{dt}+\nu^2k\phi_{i,j}n_{i,j}=0,\\&\frac{\gamma_{1,j}-\gamma_{0,j}}{h}=0,\ \frac{\gamma_{N+1,j}-\gamma_{N,j}}{h}=0,\\&\frac{\gamma_{i,1}-\gamma_{i,0}}{h}=0,\ \frac{\gamma_{i,N+1}-\gamma_{i,N}}{h}=0,\end{aligned} \tag{5.25}$$

where $j=1,...,N$ and $i=1,...,N$ with initial conditions defined as,

$$\gamma_{i,j}(0)=\frac{1}{2}+\frac{1}{2}*\tanh\left(\frac{(j-t_r)}{\sqrt{w*2}}\right). \tag{5.26}$$

Equations (5.23) - (5.25) represent a differential algebraic system of equations that can be numerically solved via time-discretisation methods shown in the next section. As this system does not have stiff properties, the explicit numerical schemes introduced below are adequate; however, future expansion of this derivation towards nonlinear concentration diffusion will require reassessment.

5.1.1.3 Time discretisation and approximation

Time integration of the differential-algebraic equation (DAE) set is handled through the third-order strong stability preserving Runge-Kutta (SSP-RK3) [187] method for BattPhase.jl. As denoted in the name, these methods provide stability improvements over the conventional Runge-Kutta algorithm for time integration when numerically solving systems of DAEs. These algorithms were initially introduced in literature as the total variation diminishing (TVD) spatial discretisation for use with hyperbolic conservation laws, which contained discontinuous solutions [188, 178]. This method includes a stability property in the forward Euler integration method,

$$\|u^{n+1}\| = \|u^n + f(u^n)\Delta t\| \leq \|u^n\|, \tag{5.27}$$

while maintaining a step size restriction such that,

$$0 \leq \Delta t \leq \Delta t_{lim}. \tag{5.28}$$

In this work, this stability enables improved dimensionless electrolyte potential predictions and large-time discretisation steps when appropriate. Mathematically, for the system of equations defined above, the SSP-RK3 is represented as,

$$\begin{aligned}
&\gamma_{k1} = \gamma_j + f(\gamma_j, \phi_j)\ \Delta t, \\
&0 = g(\gamma_{k1}, \phi_{k1}), \\
&\gamma_{k2} = \frac{3\gamma_j + \gamma_{k1} + f(\gamma_{k1}, \phi_{k1})\ \Delta t}{4}, \\
&0 = g(\gamma_{k2}, \phi_{k2}), \\
&\gamma_{j+1} = \frac{\gamma_j + 2\gamma_{k2} + 2f(\gamma_{k2}, \phi_{k2})\ \Delta t}{3}, \\
&0 = g(\gamma_{j+1}, \phi_{j+1}),
\end{aligned} \tag{5.29}$$

where $f(\gamma, \phi) = d\gamma/dt$ and $g(\gamma, \phi) = 0$. While providing a stable method for time integration, the SSP-RK3 requires a linear solution at each step for ϕ, which greatly increases the computational burden for this method. A numerical reduction of the SSP-RK3 has been previously completed to improve the computational performance further [2]. As this work aims to further improve the computational performance for the above-introduced phase-field model, the third-order approximation introduced by Jang et al. is also presented. This approximation of the strong stability preserving Runge-Kutta third-order is introduced as,

$$\begin{aligned}
&\gamma_{k1} = \gamma_j + f(\gamma_j, \phi_j)\ dt, \\
&0 = g(\gamma_{k1}, \phi_{k1}), \\
&\gamma_{k2} = \frac{3\gamma_j + \gamma_{k1} + f(\gamma_{k1}, \phi_{k1})\ dt}{4}, \\
&\phi_{k2} = \frac{1}{2}\phi_{k1} + \frac{1}{2}\phi_j, \\
&\gamma_{j+1} = \frac{\gamma_j + 2\gamma_{k2} + 2f(\gamma_{k2}, \phi_{k2})\ dt}{3}, \\
&\phi_{j+1} = \phi_{k1}.
\end{aligned} \tag{5.30}$$

This introduced approximation removes the linear solve in the ϕ_{k2} and ϕ_{j+1} steps, enabling the removal of two linear solves from the time integration. Provided the prediction fidelity is maintained, this approximation removes a large computational burden from the numerical solution. To validate BattPhase.jl's implementation of these schemes, both are numerically created in BattPhase.jl with a modular architecture enabling easy selection of the time-descritisation method. Figure 5.3 below presents the numerical prediction difference between the third-order SSP Runge-Kutta and its introduced approximation. The dimensionless root mean square difference between the SSP-RK3 and SSP-RK3 approximation across the two-hour prediction is 3.468×10^{-6}.

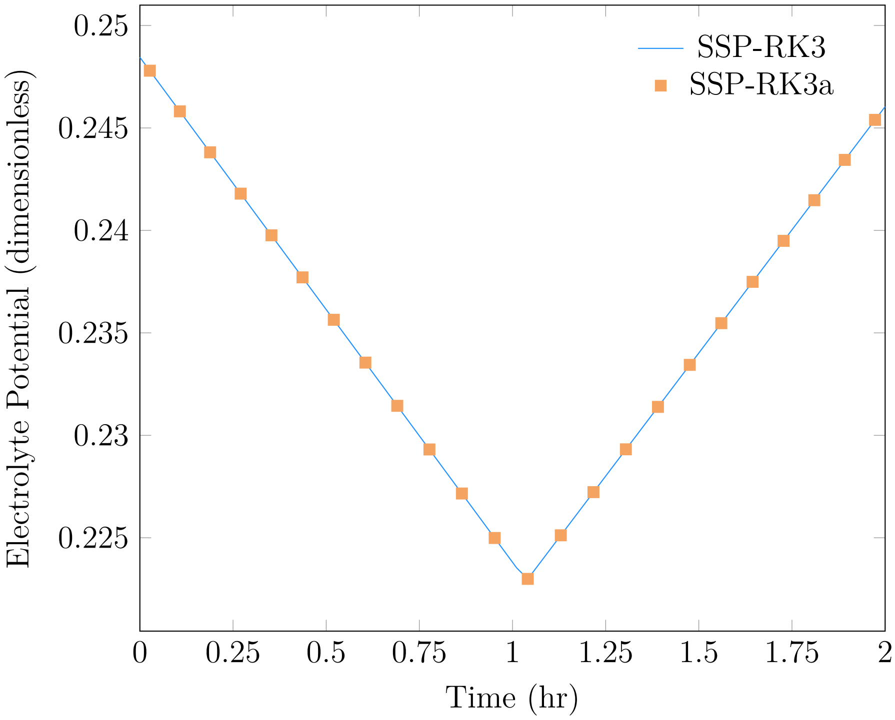


Fig. 5.3 Phase-field predicted electrolyte potential comparison between strong stability preserving Runge Kutta third order (SSP-RK3) and the presented approximation (SSP-RK3a) for $\delta = 0.255$, $k = 1.557$, and $\nu = 0.475$

To further confirm the numerical performance improvement, the approximation and the original SSP-RK3 were benchmarked across a range of grid sizes, with results displayed in Table 5.1 below. For this investigation, the system is configured with the initial seed geometry presented in Figure 5.2 above. The simulation comprises one hour of electrodeposition, followed by a one-hour stripping operation. This is completed without any additional numeric optimisations completed to the Julia 1.7.2 installation, i.e. inclusive of the standard linear algebra package, OpenBLAS.

Table 5.1 Time comparison (in seconds) of the SSP-RK3 time integration scheme to its numerical approximation across a range of domain sizes.

Domain Size	SSP-RK3	SSP-RK3 Approximation
10^2	3.14×10^{-3}	1.03×10^{-3}
20^2	21.6×10^{-3}	7.12×10^{-3}
40^2	129×10^{-3}	44.1×10^{-3}
80^2	1.21×10^{0}	422×10^{-3}
160^2	1.16×10^{1}	4.16×10^{0}
320^2	1.30×10^{2}	4.45×10^{1}
640^2	1.41×10^{3}	4.82×10^{2}

From the above results, the approximated SSP-RK3 offers improved performance across the investigated grid ranges of 65.9% over the full-order implementation. Additionally, the predicted potential aligns well with the full-order scheme. Given these results, the approximation will be utilised in the remainder of this chapter to reduce the computational requirements while maintaining the prediction fidelity. In the next section, the numerical linear algebra implementation is further optimised.

5.1.2 Linear algebra packages

Given the above system derivation and discretisation for both the dimensionless potential and moving phase parameter, a Julia-based computational package, BattPhase.jl, was created to solve the system of differential-algebraic equations numerically. Julia was selected for this to achieve high computational performance while maintaining code readability and comprehension. As the computational requirements to provide solutions for this system grow with the size of N, Julia's JIT compilation enables a final solution that exceeds conventional interpreted languages such as MATLAB and Maple. The open-source code repository for BattPhase.jl can be found at https://github.com/BradyPlanden/BattPhase.jl. To showcase the capabilities of this package, an initial example investigation is presented for a simplistic anode formation. For this example, the anode is simulated for $\tau = 2$ $[hr]$, split into one hour of deposition and one hour of stripping. The exchange current density, δ, is set at 0.1, $\nu = k = 1$, $N = 200$, and with an initial seed thickness of 5 µm. The resultant evolution shown in Figure 5.4 is predicted, with an electrodeposited anode thickness growth of 18 µm resulting in a maximum thickness of 23 µm at the end of the electrodeposition cycle. After the stripping cycle, the thickness reduces back to 5 µm; however, the moving boundary gradient has been reduced, resulting in a larger transition between the two phases. This is expected, with the results still maintaining mass conservation as seen when comparing Figure 5.4.a and Figure 5.4.c below.

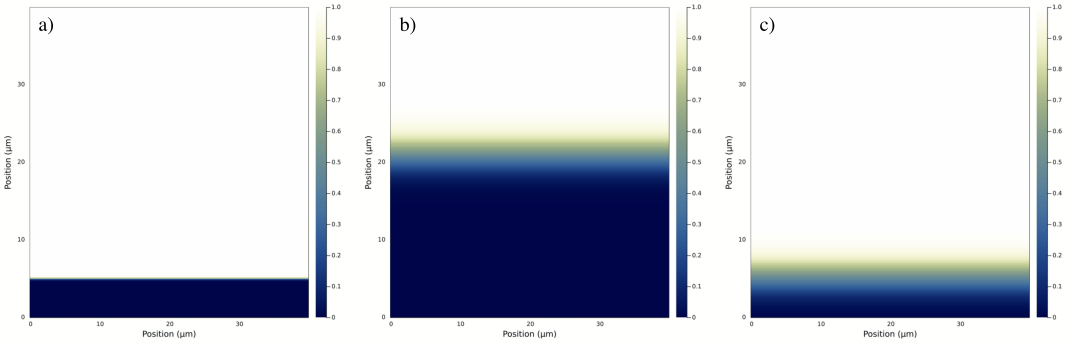


Fig. 5.4 Numerical evolution of lithium-metal phase-field model at time-step zero (a), 1-hour (b), and 2-hours (c).

For the simplistic geometry displayed above, an investigation into the computational time required for various grid sizes is completed. This grid exploration aims to understand further the computational requirements for fine microstructure evolution, such as the previously described mossy lithium composition. In parallel to this investigation, an alternative computational implementation of the linear algebra package utilised by Julia is introduced. Specifically, the Intel-based Pardiso solver was implemented in BattPhase.jl to improve the Jacobian matrix solution time. This implementation was completed through the open-source package Pardiso.jl coupled with the Intel Math Kernal Library (MKL). The results for both the grid investigation and the linear algebra implementations are included in Table 5.2 below. To showcase the performance improvements of BattPhase.jl, these results are also compared to a Maple language implementation [2], which was numerically implemented with similar hardware (Intel i9-11900K, 64 GB RAM) to work presented in this section.

Table 5.2 Comparison (in seconds) of BattPhase.jl utilising OpenBLAS and Pardiso to Maple language implementation across a range of domain grid sizes with the third order SSP Runge-Kutta approximation.

Domain Size	OpenBLAS	Pardiso	Maple Implementation (w/ Pardiso)
10^2	1.03×10^{-3}	1.76×10^{-3}	582×10^{-3}
20^2	7.12×10^{-3}	1.10×10^{-2}	1.00×10^{0}
40^2	4.41×10^{-2}	5.98×10^{-2}	1.99×10^{0}
80^2	4.22×10^{-1}	3.72×10^{-1}	4.15×10^{0}
160^2	4.16×10^{0}	3.11×10^{0}	1.20×10^{1}
320^2	4.45×10^{1}	2.88×10^{1}	5.80×10^{1}
640^2	4.82×10^{2}	2.30×10^{2}	4.01×10^{2}

As shown in the above results, the Pardiso linear algebra implementation provides a mean improvement over the standard OpenBLAS implementation of 11.55 % across the grid range presented above. Due to OpenBLAS' fast implementation at low grid sizes, Pardiso's implementation performance is relatively worse; however, a large improvement over the OpenBLAS occurs for grid sizes larger than 80^2 with a reduction in prediction time of 52.3% at a grid size of 640^2. Given these results, both Pardiso and OpenBLAS are implemented in BattPhase.jl with a selection mechanism developed for the end-user. This concludes the adaptions for the computational time reduction of the conventional BattPhase.jl package with a prediction reduction of 72.8% over the initial OpenBlas/SSP-RK3 implementation for a grid size of 160^2. Finally, the Pardiso implementation of BattPhase.jl provides a 74.1% prediction time reduction over the Maple equivalent implementation for the 160^2 grid size. This confirms the improvements that BattPhase.jl provides for fast, performant lithium-metal anode modelling predictions over alternative approaches.

5.1.3 Concluding Remarks

An open-source Julia-based package has been developed in the work above for fast numerical predictions of lithium-metal anode evolution utilising a phase-field representation. This package provides improvements over alternative languages while maintaining a user-friendly dynamic typeset. A strong stability-preserving Runge-Kutta third-order approximation was introduced as described in [2] aimed to reduce the required linear solutions of the Jacobian matrix and showed improvements in the full-order implementation and equivalent Maple implementation. Furthermore, the presented framework was coupled with the linear algebra package Pardiso through the underlying Intel Math Kernal Library (MKL), which enabled further performance improvements over the conventional OpenBLAS implementation. These implementations, plus the underlying Julia language, have provided a performant framework for lithium-metal phase-field representations and have exceeded the Maple implementation presented in [2]. Further optimisation of this framework is expected through improvements in thread parallelisation and investigation into higher-order Runge-Kutta pairs for potential improvements in fidelity. Future work for this framework would include coupling the phase-field electrodeposition method with a porous electrode representation to create a full-cell model. The first step in this coupled implementation would be utilising the one-dimensional representation with modified cathode boundary conditions for the porous representation. A coupled numerical solver with time-aligned steps would be required, in addition to ensuring the conservation of mass and charge at the interface. Overall, this section has improved phase-field lithium-metal modelling by presenting a fast, performant open-source package.

The next section uses the above numerical model to achieve even higher numerical performance by developing a message-passing graph neural network representation. This work aims to expand the prediction capabilities past conventional numeric solutions while ensuring underlying physics is maintained.

5.2 Neural solver methods

As anode-free lithium-metal cells become available for consumer usage, advancements in online capable electrochemical models are required. These models are needed to inform the battery management system of potential degradation operating states while ensuring the optimal utilisation of these highly performant cells. This poses a challenge; as discussed in the previous section, this system requires applying numerical moving boundaries to the system definition to capture the evolving negative electrode thickness accurately. An alternative to conventional numerical methods must be investigated to achieve fast predictions capable of design space investigation of corresponding (δ, ν, and k), potential onboard predictions, and cell design development.

In this section, an alternative machine-learning-based solution for domain predictions is presented as a method to achieve performance improvements while maintaining underlying system knowledge. As previously discussed in 2.2.2, neural solvers can provide strong capabilities in predicting partial differential solutions for a trained domain. In Chapter 2, an introduction to neural solvers was presented, with a discussion on the benefits and capabilities of each framework. This section presents a framework for coupling the previously introduced lithium-metal phase-field model, BattPhase.jl and auto-regressive neural solvers. This work aims to enable improved usage of lithium evolution models for design-space iterations and reduce the computational burden for lengthy cycle degradation predictions.

5.2.1 Auto-regressive neural solver framework

This section introduces a message-passing graph neural network (MP-GNN) to accomplish the auto-regressive neural solver methodology. Originally introduced by Scarselli et al. [189], graph neural networks have previously shown improved flexibility over alternative neural architectures. The flexibility to construct the underlying graph in such a way that it aligns with the features in the trained dataset has enabled these improvements, which have further resulted in the expansion of GNNs across multiple fields such as visual scene understanding [190, 191], predictions in chemical molecular properties [192, 193, 194], traffic pattern predictions [195, 196], and classification problems for both images and videos [197, 198]. Due to these improvements, MP-GNNs have been previously introduced to capture physical system dynamics [199, 200, 201] to varying success. This section builds off previous work from Brandstetter et al. [145] following the expansion of the open-source Pytorch-based [202] code repository for implementation enabling phase-field architectures.

For a given grid domain of (X, Y), the graph representation can be formed as $\mathcal{G} = (\nu, \varepsilon)$ where $i \in \nu$ is the node mapping and $(i, j) \in \varepsilon$ defines the indices of connected edges. For this grid, the individual cells are represented as graph node features, $\mathbf{f}_i$. In this work, the framework follows the encode-process-decode methodology previously introduced in Battaglia et al. [203] and Sanchez-Gonza et al. [204]. This framework is introduced as,

Encoding for each node of (i) maps, [1] the last K number of solutions for corresponding phase-field element $(\mathbf{u}^{k-K:k})$, [2] the node's corresponding element position $(\mathbf{x}_i)$, [3] current timestep (t_k), and [4] the PDE coefficients, (ν, κ), boundary conditions, (δ), and initial conditions, (T), as $(\boldsymbol{\theta}_{PDE})$. The final node embedding vector comprises each of these elements as $\mathbf{f}_i^0 = \varepsilon([\mathbf{u}^{k-K:k}, \mathbf{x}_i, t_k, \boldsymbol{\theta}_{PDE}]$. Without encoding $\boldsymbol{\theta}_{PDE}$, it would not be possible to predict across PDE system parameterisations; this would then require retraining for each specific case. This encoding is completed with a multilayer perceptron network.

Processing computes M steps forward with intermediate graph representations $\{G^1, G^2, ..., G^M\}$. The message, $\boldsymbol{m}_{i,j}^m$ and feature update,$\boldsymbol{f}_i^{m+1}$ are computed as,

$$\text{edge } j\text{-}>i \text{ message:} \quad \boldsymbol{m}_{i,j}^m = \phi(\boldsymbol{f}_i^m, \boldsymbol{f}_j^m, (\boldsymbol{u}_i^{k-K:k} - \boldsymbol{u}_j^{k-K:k}), (\boldsymbol{x}_i - \boldsymbol{x}_j), \boldsymbol{\theta}_{PDE})$$

$$\text{node } i \text{ update:} \qquad \boldsymbol{f}_i^{m+1} = \psi(\boldsymbol{f}_i^m, \textstyle\sum(\boldsymbol{m}_{ij}^m, \boldsymbol{\theta}_{PDE})),$$

where ϕ and ψ are multi-layer perceptrons, $\boldsymbol{x}_i - \boldsymbol{x}_j$ is the grid location difference and enables relative relation training, and $\boldsymbol{u}_i - \boldsymbol{u}_j$ is the solution difference, much like conventional numerical derivative operators. $\boldsymbol{\theta}_{PDE}$ includes the PDE information for this given step.

Decoding with a shallow 1D convolutional network comprised of shared weights. This CNN is used across spatial locations to output the next time-step predictions at grid-point $\boldsymbol{x}_i$. Each node has a vector $\boldsymbol{f}_i^m$ corresponding to the time predictions utilised in the CNN to decode future spatial values. This benefits the framework's performance by providing a time-series-like input that enforces smoothing. This operation performs similarly to the conventional multistep method but avoids stability issues through the nonlinearity and flexibility of the time-series input. As introduced in [145], a new vector, $\boldsymbol{d}_i^l$, is created through this mechanism and encapsulates the future predicted time-steps, i.e. $\boldsymbol{d}_i^l = \{d_i^1, d_i^2, ..., d_i^K\}$, the updated solution is introduced via,

$$\boldsymbol{u}_i^{k+l} = \boldsymbol{u}_i^k + (t_{k+l} - t_k) \cdot \boldsymbol{d}_i^l \;\; \text{for } 1 < l < K. \tag{5.31}$$

This chapter investigates the one-dimensional and two-dimensional phase-field representations, each offering a different use case. For the one-dimensional representation, the main identified use case is coupling with similar low-dimensional electrochemical models, such as the single particle model (SPM), the single particle with electrolyte model (SPMe), and the Doyle-Fuller-Newman (DFN) model. This couple would create a hybrid representation of the lithium-metal cell, with the neural solver providing the anode solution and the coupled model providing the cathode solution. For the two-dimensional case, the identified utilisation is in lithium anode morphology predictions, as this would enable future insights into degradation mechanisms and material compositions. For this use case, the underlying phase-field model needs to be representative of these system dynamics; however, given improvements in BattPhase.jl, this is achievable and can offer improved design exploration performance of the conventional numerical methods. As the phase-field model developed in the previous section utilises linear kinetics and forgoes concentration gradients, the training dataset won't include dendritic, tree-like structures; however, the training dataset framework developed is expected to be agnostic to those underlying microscale mechanics. Finally, the neural solver framework is visually represented in Figure 5.5.

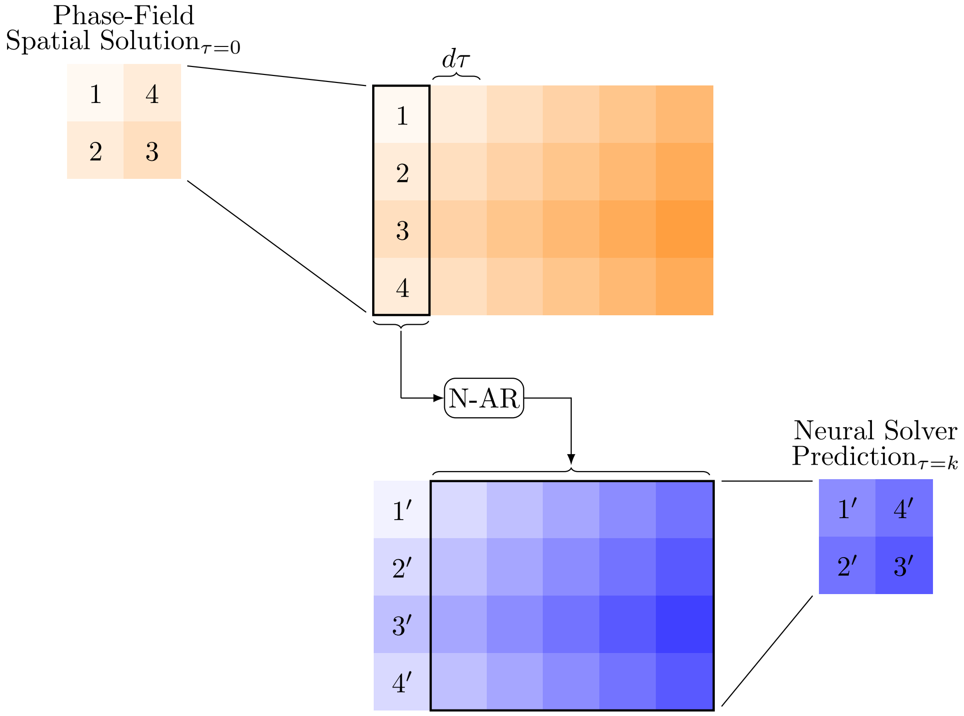


Fig. 5.5 Training and neural solver architecture representation, with an unrolling size of K steps, and corresponding phase-field spatial input data mapped forwards by $d\tau$. The prime notation indicates predicted values for the corresponding spatial location.

5.2.2 Training structure

The training structure for the autoregressive message-passing graph neural network architecture follows previous work completed by Brandstetter et al. [145], with modifications to improve the performance of the phase-field representation. As autoregressive solvers commonly map the solution at time-step k to a future solution $k+1$, the simplest implementation of this method is to train one time-step forward. The loss to minimise for this implementation comprises the random selected initial conditions from the training dataset to form the distribution denoted as $p_0(u_0)$, the corresponding ground truth distribution at time-step k, $[p_k(u_k) = \int p_k(u_k|u_0)\ p_0(u_0)\ du_0]$ and is introduced as,

$$L_{step} = \mathbb{E}_k \mathbb{E}_{\boldsymbol{u}^{k+1}|\boldsymbol{u}^k, \boldsymbol{u}^k \sim p_k} [\mathcal{L}(A(\boldsymbol{u}^k), \boldsymbol{u}^{k+1})], \tag{5.32}$$

where $\boldsymbol{u}^k + 1|\boldsymbol{u}^k$ is the distribution of future predictions given the current solution, $\mathbb{E}_{\mathrm{k}}$ is the expectation function, and $\boldsymbol{\mathcal{L}}$ is the corresponding loss function. For this work, a squared L2 norm loss function is utilised. Challenges included in this implementation include error propagation due to deviations in the predicted solution at $\boldsymbol{u}_{k+1}$ from the ground truth distribution $p_{k+1}(\boldsymbol{u}_{k+1})$ for each iteration of the autoregressive solver, A. This propagation accumulates throughout the prediction time domain for the system and results in poor model performance and overfitting issues from the numerical error. An alternative to improve stability for autoregressive neural solver predictions is to introduce an adversarial-style element to the loss as,

$$L_{stabilty} = \mathbb{E}_k \mathbb{E}_{\boldsymbol{u}^k+1|\boldsymbol{u}^k, \boldsymbol{u}^k \sim p_k} [\mathbb{E}_{\epsilon|\boldsymbol{u}^k} [\boldsymbol{\mathcal{L}}(A(\boldsymbol{u}^k + \boldsymbol{\epsilon}), \boldsymbol{u}^{k+1})]]. \tag{5.33}$$

The final loss is computed as $L_{total} = L_{step} + L_{stability}$ and has been previously introduced as the "push-forward" trick by Brandstetter et al. Next, the data structure methodology for multi-dimensional training is presented. A multi-dimensional concatenation to a single dimension is performed to provide a robust framework. To achieve this while maintaining the structure required for the graph neural network, the two-dimensional grid was vertically concatenated, as shown in Figure 5.6 below, into a single-dimensional input with reverse indexing for the column dimension. The ordering of this vector is critical to ensure encoded nodes align with corresponding nearest neighbours for accurate edge propagation.

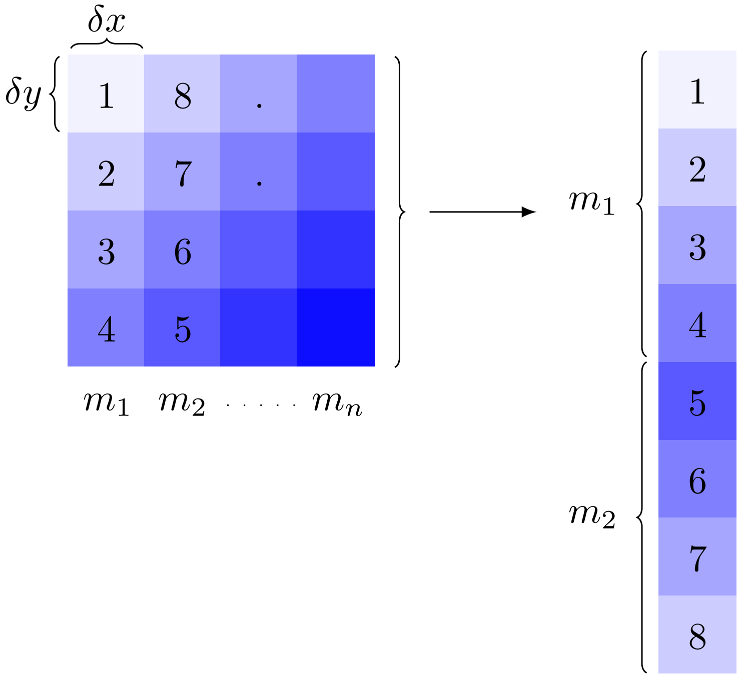


Fig. 5.6 Visual representations of the multi-dimensional grid concatenation for input into the neural solver architecture. Extension of this methodology enables expansion into high-order dimensionality.

In addition to the "push-forward" trick, the temporal bundling method introduced by Brandstetter et al. [145] is coupled into the neural solver framework. Temporal bundling enables coupled forward predictions determined through a time-window range and draws parallels to the optimal time horizon methodology utilised in model predictive control. This is expanded in the next subsection, with further discussion around improvements to the loss function defined above, concluding the remainder of this section. To improve the numerical stability of the autoregressive MP-GNN architecture, the prediction length is unrolled from a single step to multiple forward steps. This multi-step prediction horizon has been shown to reduce numerical error propagation when loss propagated is constrained from the final prediction step.

Additionally, a design space exploration methodology to generate a dataset across varying initial conditions was completed to capture the system dynamics across the operating regime. To ensure this dataset was representative of the operating conditions, the normalised design space for the lithium seed thickness (T), defined in equation (5.2), was initialised dependent on the system operation, i.e. for negative applied current densities, the initial seed thickness space was biased high and vice-verse for positive. The design space algorithm used to generate the datasets was a Latin hypercube, as it provides capabilities in filling expansive spaces given a minimal number of cases. The other parameters included in the model parameterisation include the dimensionless exchange current density, k, the density-molecular mass ratio, ν, set by equation (5.11), the applied current density, δ, the timestep increment, $d\tau$, and the prediction length, τ. Table 5.3 below presents the design space ranges utilised for the loss-function assesment in the next section.

Table 5.3 Latin-hypercube variable ranges to generate training dataset for loss-function performance assesment.

Parameter	Range
$T_{\delta \geq 0}$	0 – 30 µ m
$T_{\delta < 0}$	28 µ m – 38 µ m
δ	-3.0 – 3.0
τ	1.0 hr
$d\tau$	0.66 min
k	1.0
ν	1.0

5.2.3 Results

This section presents the autoregressive message passing graph neural network framework results. This starts with an activation function optimisation for the given dataset and a graph connectivity assessment for the connected neighbour and corresponding prediction performance. A one-dimensional case is presented that compresses the phase-field representation to a single dimension for simplistic lithium-metal predictions. Next, the multi-dimensional data structure presented in the above section is verified to the one-dimensional case. To verify the neural solver's performance with complex morphology, a standardised gaussian seed is included in the training dataset, with location determined from the Latin-hypercube design space. Finally, the computational performance of the neural solver is discussed and compared to the BattPhase.jl method presented in the previous section. For the work presented in this section, a deep-learning workstation was utilised comprising of an Intel Xeon W-2275 14-core processor, 512 gigabytes of memory, and four Nvidia RTX 3090 graphics cards. Unless otherwise specified, the results presented in this section were performed on this system utilising a single 3090 graphics card.

5.2.3.1 Training performance

In this section, the training performance for the framework is assessed, with an investigation into the multi-dimensional dataset structure performed. Utilising the training structure defined in section 5.2.2 above, the first optimisation performed in this framework is modifications in the activation function for the neural architecture. In this work, two different activation functions are compared, the swish [205], and the rectified linear unit (ReLU) [206, 207, 208]. The swish activation function operates as,

$$f(x) = x \cdot \sigma(\beta x), \tag{5.34}$$

where β is a scalar parameter that can be predetermined or integrated into the training regime. The sigmoid function, σ is defined as,

$$\sigma(z) = (1 + \exp(-z))^{-1}, \tag{5.35}$$

where the swish activation function becomes a linear function if $\beta = 0$, a step function when $\beta \to \infty$, and a sigmoid weight linear unit function [209] when $\beta = 1$. This capability provides flexibility while maintaining integration with larger network architecture due to the simplistic definition. The ReLU function is defined as,

$$f(x) = \max(0, x). \tag{5.36}$$

For a given input, x, the ReLU function acts as a step function between the input value and zero, depending on the corresponding input sign. By inspection, it is clear that as $\beta \to \infty$, the swish function closely represents the ReLU function. Due to this trend, using the swish function improves training flexibility over the ReLU, as β can be preselected to match the ReLU response while enabling modifications during training. Figure 5.7 below presents both activation functions,

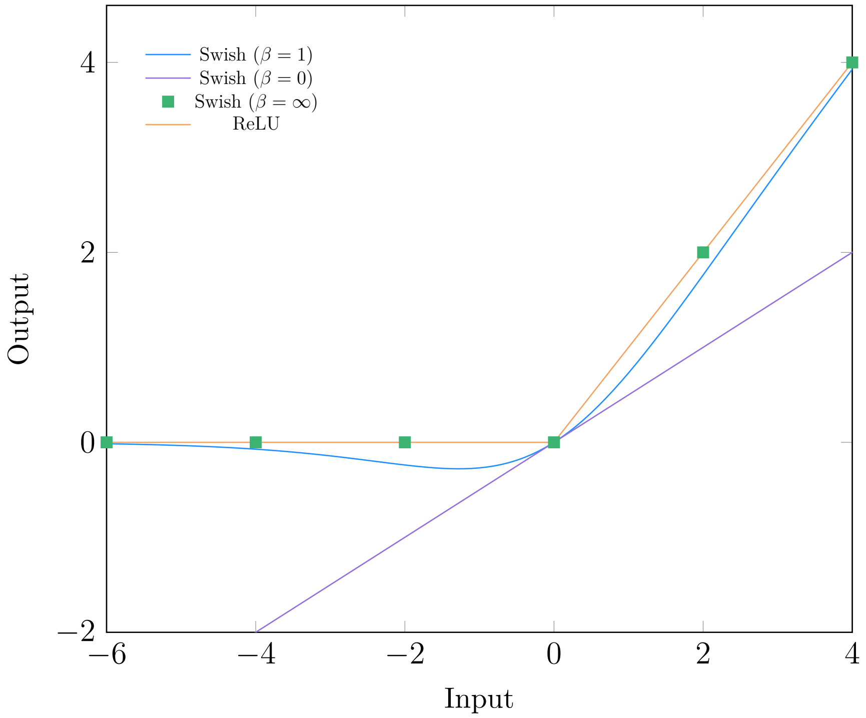


Fig. 5.7 Corresponding outputs for Swish and ReLU activation functions for three given β values.

A comparison of the ReLU and Swish activation functions during the autoregressive neural solver training is completed to quantify potential performance benefits. For this investigation, the training dataset comprised 512 training samples, 64 validation samples, and 64 testing samples completed through an optimised Latin hypercube space-filling algorithm [210]. This Latin hypercube algorithm has been selected to provide an optimised design space-filling routine for a given sampling number. The design space selected for this architecture comprises the phase-field system parameterisation, specifically, the applied current density, δ, initial seed thickness, T, the introduced density-molecular mass constant, ν, and dimensionless exchange current density, k. This training distribution results in a 66.6% partition for training and 16.6% each for testing and validation. Figure 5.8 below displays the training loss for this investigation for 15 training epochs, corresponding to 75 loss function assessments.

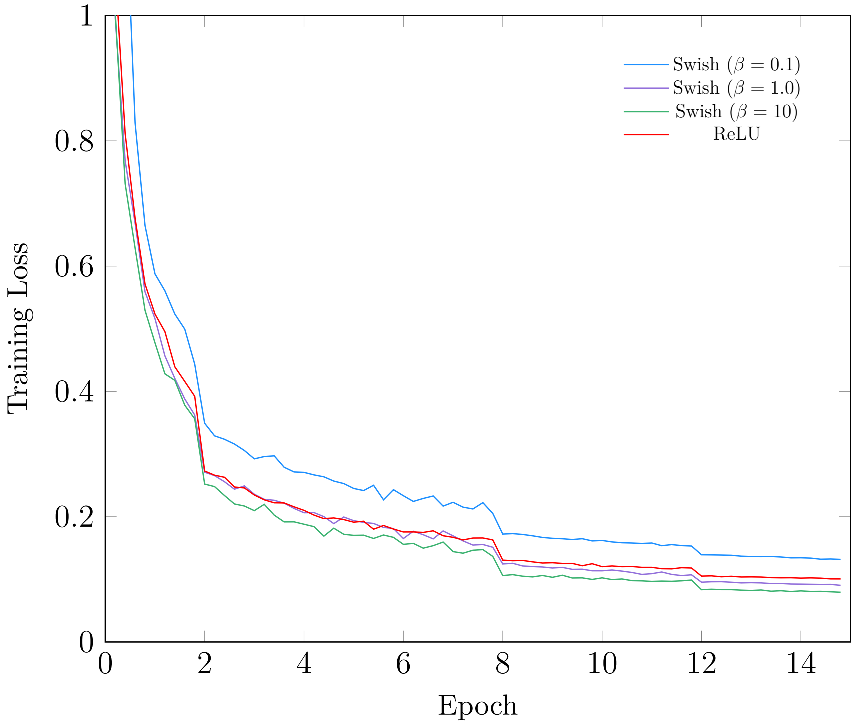


Fig. 5.8 Corresponding outputs for Swish and ReLU activation functions for three given β values.

For the remainder of this chapter, unless specified otherwise, the training regime for the autoregressive neural solver includes the generation of the operating dataset from the BattPhase.jl full-order representation, followed by a loss minimisation across 15 epochs, with a batch size of one, and finally, graph connectivity spanning across three neighbours. The next section presents the neural performance based on this training configuration.

5.2.3.2 The one-dimensional case

First, the one-dimensional phase-field representation is tackled to verify the neural solver's capabilities for predicted microscale evolution. For this verification, the introduced training methodology above is utilised with a spatial grid size of 40, corresponding to $L = 40\ \mu m$, which is fed into the training structure. Each of these spatial locations is augmented to include the system of equation parameterisation, which is generated through the Latin hypercube design space comprising 1024 cases explored. This design space covers parameter ranges introduced in Table 5.3. Figure 5.9 presents the one-dimensional neural solver predicted trajectory unrolled twice corresponding to 50-time steps compared to the ground truth Battphase.jl trajectory.

In Figure 5.9, the neural solver was trained for 15 epochs with a batch size of two and an immediate neighbour connection of three. Due to the reduced complexity of the graph structure for the one-dimensional case, as well as the increased number of training points, this lower number of epochs was required for training. An RMSE of 3.83×10^{-5} between the neural solver and the ground truth model was obtained across the 25 predicted time-steps for the phase-field representation (0,1). A further use case for the one-dimensional model is online capable deployments that provide the anode thickness for state-of-power (SOP) predictions. In high-performance applications, state-of-power is commonly solved by forward integration of a simplistic numerical model, such as an ECM, to

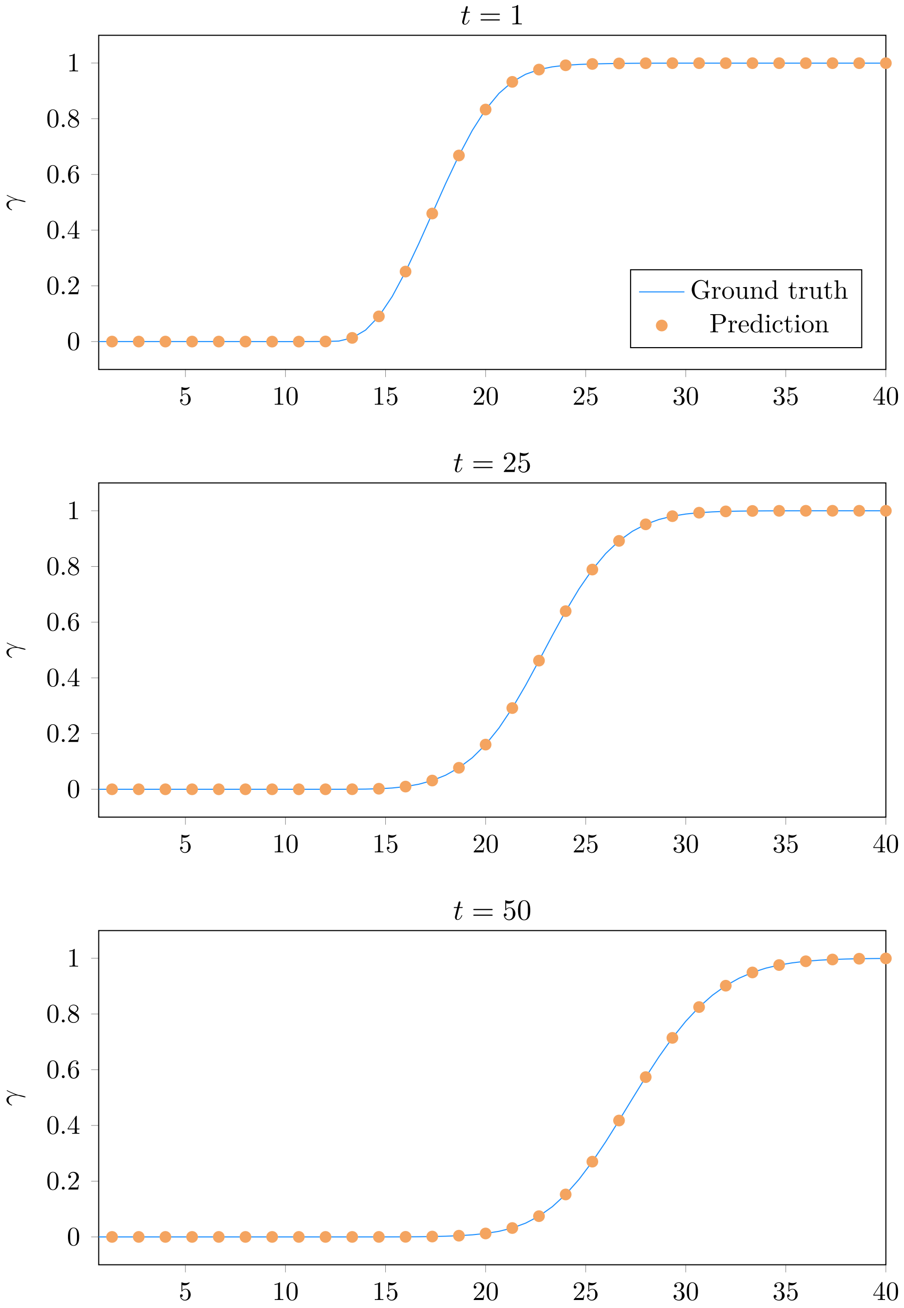


Fig. 5.9 One-dimensional neural solver unrolled twice for a 50 time-step prediction (30 minutes) compared ground truth model for $\nu = 1.0$, $k = 1.0$, $\delta = 0.337$, and an initial seed thickness of $14\mu m$ acquired from the testing allocated dataset.

provide the control structure with future information for a given power request. This time integration, coupled with the control structure interactions, enforces terminal voltage limits for the operated energy storage system; however, for lithium-metal batteries, this terminal voltage doesn't guarantee anode thickness values for optimal coulombic efficiency. By deploying a lithium evolution model, the online control structure would gain access to underlying anode thickness, enabling the potential for optimal control structures for reductions in degradation mechanisms. In the next section, the architectures domain dimensionality is increased to cover the two-dimensional case utilising the methodology introduced in 5.2.2 above.

5.2.3.3 The two-dimensional case

As the one-dimensional case has been introduced, the two-dimensional extension will be discussed in this section. The multi-dimensional training methodology introduced is combined with a Latin hypercube space exploration, similarly completed for the one-dimensional case. Due to the additional numerical cost for the two-dimensional forward model, the number of cases is reduced to 512 across an equivalent parameter space. This dataset is fed into the training regime for 15 epochs with an immediate neighbour connection of three and a domain grid size of 40^2. This verification initialises a seed thickness of 6.0 μm with $\delta = 2.015$, $k = 1.557$ and $\nu = 0.060$ to complete a plating prediction. The results of this prediction are compared to the ground truth model in Figure 5.10 below, comprising of predicted time-steps across the electrodeposited operation from the testing allocation of data. This confirmation also aligns with the one-dimensional case with an RMSE of 9.52×10^{-3} between the ground truth and prediction.

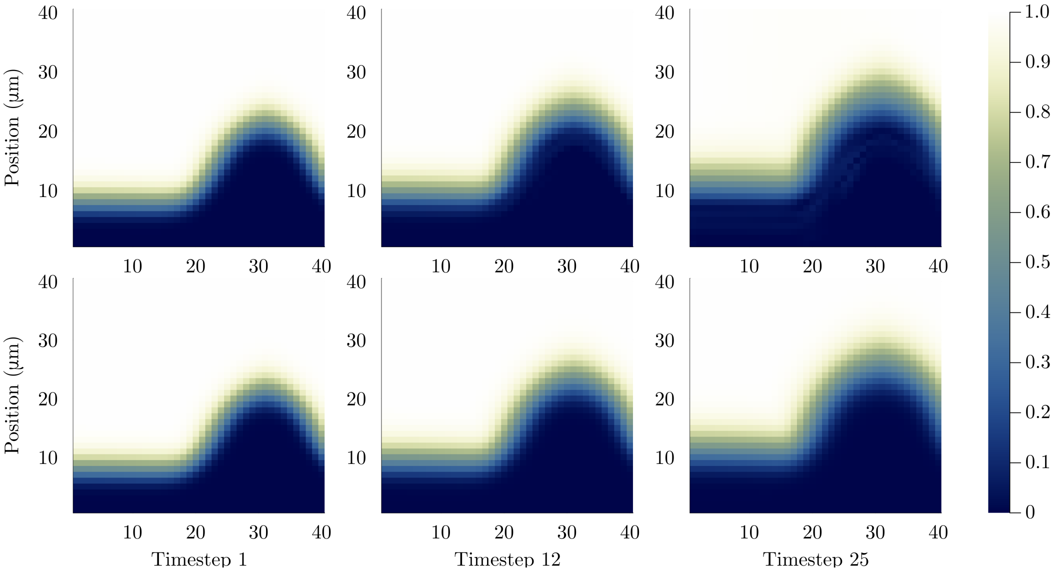


Fig. 5.10 Two-dimensional neural solver unrolled forward prediction (bottom) comparison to ground truth (top) for time-steps (0,12 and 25) corresponding to 15 minutes of forward prediction.

These results confirm the auto-regressive solver's flexibility to initial seed geometry for a given applied current density, exchange current density, and molecular mass-density ratio. The next section presents a discussion on the neural solver's computational performance.

5.2.3.4 Computational performance

As the fidelity for both single and two-dimensional predictions has been presented and verified to the ground truth model, the computational requirements for the autoregressive neural solver will be discussed in this section. The objective for this chapter has been to reduce the computational lithium-metal anode models while maintaining fidelity, aligning with the thesis aims presented in Chapter 1. As the neural solver requires training before deployment, two numerical results are presented and compared to the conventional BattPhase.jl implementation. The training portion of the neural solver presented is computationally more expensive than the conventional Battphase.jl implementation as it requires system data generation, as well as the loss function minimisation training;

however, the trained solver provides extremely fast predictions which are not achievable from BattPhase.jl or conventional numerical solvers. Table 5.4 below presents the training and the final prediction time requirements for the neural architecture presented in the previous section. This is compared to the phase-field representation for an identical parameterisation. For these results, the BattPhase.jl prediction was computed on the Intel i9 10980XE workstation, with the neural solver utilising an Nvidia RTX 3090 graphics card.

Table 5.4 Computational timing for both training and prediction for a grid size of 90^2 *compared to BattPhase.jl*

Method	Training Time per Epoch [hr]	Prediction Time per Step [ms]
Neural Solver	5.25	3.94
BattPhase.jl	N/A	6.6

Overall, the training requirements for the autoregressive neural solver are quite high; however, a 40.3% decrease in computational time per time-step is achieved for the trained solver. This fast prediction time offers valuable capabilities for deployment in low-computation applications. This includes design space exploration for battery material discovery and optimisation, modular coupling to traditional cathode models for cell-scale battery predictions, and deployment in embedded or interactive applications where prediction speed is the highest requirement. It should also be noted that the neural solver is implemented utilising Pytorch, and thus the underlying function call is in Python, minus the CUDA API; however, BattPhase.jl is developed in Julia and is, therefore, fully compiled before execution.

5.2.4 Concluding Remarks

This section presents an autoregressive neural solver framework that enables fast surrogate model capabilities for the evolution of lithium-anode morphology.

This framework utilises an encode-process-decode methodology with temporal bundling and an adversarial style push-forward operator, as presented in the literature. A methodology for physics-informed data generation from the BattPhase.jl phase-field framework was developed as an improvement over conventional methods. This included a design space investigation, with insight into an optimised variable space presented. Additionally, the activation functions' effect on the model predictions was presented with the final model training structure aligned with the findings. For the single-dimension case, the trained architecture was presented with performance matching the ground truth phase-field predictions showcasing the potential for online prediction capabilities, such as state-of-power calculations. Next, the two-dimensional case was assessed utilising a more complex gaussian seed initial geometry showcasing the two-dimensional capabilities of the framework, with an RMSE of 9.52×10^{-3} between the neural prediction and the ground truth across the 25 presented time steps. Finally, the computational requirements for the neural solver were compared to the ground truth model, showcasing improvements post-training that enables both online capabilities and fast design space explorations for future material compositions. Future improvements for this work include further training requirements to reduce the large overhead needed to achieve the performance presented. The development of a computationally informed architecture that could utilise the one-dimensional results for systems with symmetry is also an area for future developments. Overall, this chapter introduced two methods that, when coupled together, improve physics-informed lithium morphology predictions.

Chapter 6

Conclusions

This thesis presented advancements in physics-informed battery models for lithium-ion and lithium-metal batteries. Improvements in operational performance and application-informed battery design are showcased alongside lithium morphology domain predictions for fast degradation investigations. The work presented has delivered novelty in these areas and has provided improvements to real-time capable battery models and micro-scale lithium-metal predictions. Additionally, this thesis has provided improvements in fast open-source software that lowers the barrier of entry to investigating and utilising lithium-metal batteries for researchers and aids in disseminating the novelty to potential end users. This chapter will conclude the presented improvements and comment on limitations and potential future improvements.

6.1 Summary of novelty

Computationally informed discrete realisation algorithm

In Chapter 3, a computationally informed realisation algorithm (CI-DRA) is presented with an open-source implementation (LiiBRA.jl). This work advances the previous work in realisation algorithms for physics-based lithium-ion modelling capable of real-time deployment. First, an assessment of the state-of-the-art methods for realisation algorithms is presented, highlighting a gap in the literature for onboard capable realisation. An assessment of the current methods for generating the reduced order models was undertaken to fill this gap. This assessment provided two achievable improvements in the current state-of-the-art reduction and alignment of the numerical algorithm to optimise the process for faster solutions and implementation of the algorithm in an advanced computational language capable of online compilation.

The improved realisation algorithm was created and numerically verified to the full-system model through comparison to the Python battery mathematical modelling package (PyBAMM), displaying an excellent voltage prediction result of 7.54 mV RMSE to the full-order PyBAMM implementation for a WLTP 3B drive cycle. Further investigations into the architectural initialisation variable sensitivities were completed with a corresponding recommendation for end-user configuration dependent on available computation performance. The CI-DRA is then compared to the conventional DRA, with both implemented in LiiBRA.jl, showcasing a model creation performance increase of 21.7%. This confirms the numerical improvement of the CI-DRA without sacrificing fidelity in the final reduced-order model. The second aim of this thesis was to enable online reduced order model creation, which was verified through the implementation of LiiBRA.jl on ARM-based hardware. This implementation provides a viable method for the deployment of reduced-order models in-situ, enabling battery parameterisation updates throughout the available lifetime.

To complete a final validation of the numerical framework showcased above, the final reduced-order models are deployed onto an embedded target for real-time control. This included an additional investigation to improve the computational performance of the in-situ linearisation of the low-frequency D state-space array. This improved real-time performance as the surrogate model utilised for the linearisation reduced the online computations by 13.5%. The deployed model was compared to the equivalent circuit model, which provides a relative benchmark to the current industry standard. Overall, the work presented improves lithium-ion battery realisation algorithms in three key ways: (1) The CI-DRA provides an improved algorithm to achieve faster realisation and (2) LiiBRA.jl enables fast offline and online realisation and is available freely, and third, the online computational time has been reduced through usage of surrogate machine learning techniques.

Application informed battery design

Multi-scale effects of system-level characterisations have been investigated, and a novel open-source package for model-informed design decisions is presented in Chapter 4. This work enables physics-geometry-based predictions across different lithium storage methods for the anode, providing key benefits for future-generation battery analysis. Utilising BattCalc.jl, the microscale parameter effects can be scaled to system-level characteristics that enable DC internal resistance predictions, energy density, and capacity for individual cells to multiple-cell battery packs.

This work presents fundamental equations that enable the coupling of length scales to achieve multi-scale predictions. First, the electrode length scale is introduced, summarising the lowest scale calculations. At this scale, the fundamental electrochemical parameters are modelled with corresponding assumptions used to reduce the complexity to a manageable level for fast approximations. The electrode scale is expanded to the unit stack, coupling multiple electrodes and the non-active material, such as current collectors and separators. The step to the cell level includes multiple unit stacks and the inclusion of non-active packing material. This scale provides the first benchmark for BattCalc.jl to experimentally tested batteries, with predicted results including an uncertainty analysis to investigate parameter sensitivities. Finally, the system-level battery pack is formed through the addition of non-active material for safety systems and packaging.

As this chapter introduces the benefits of lithium-metal batteries, the first step is to investigate the characteristics of two comparable lithium-based battery packs. An example high-power application is selected as it currently provides difficulty for lithium-ion-based chemistries, with system targets of 600V and 5.5 kWh. Three cell configurations are selected, comparing lithium-ion and lithium-metal storage mechanisms as well as different lithium-metal designs. This investigation resulted in a mass reduction of 9.1 kg for the anode-free lithium-metal cell while achieving an energy density improvement of 55 Wh/kg compared to the lithium-ion configuration. A transient vehicle dynamic simulation was created and validated, capable of predicting drive-cycle completion time dependent on transient vehicle parameters. The corresponding system-level characteristics were imported into this framework, with results displaying a 7.62-second improvement for the anode-free lithium-metal chemistry while reducing energy consumption by 28.8 Wh.

BattPhase and autoregressive neural solver

In Chapter 5, a microscale phase-field-based model is introduced for lithium-metal morphology predictions in single and two-dimensional cases. As presented in Chapter 4, next-generation lithium-metal batteries offer improvements in energy density not previously attainable for conventional lithium-ion chemistries; however, there is a multitude of challenges that need to be solved before they will be able to achieve cycle life's equivalent to lithium-ion. To advance these challenges and provide further insight, the presented phase-field model allows microscale evolution predictions while achieving fast numerical performance. An open-source framework was introduced with numerical optimisation completed. A prediction time reduction of 72.8% over the initial model was achieved while maintaining fidelity, as shown in Figure 5.3.

An autoregressive message passing graph neural network solver was also implemented for lithium-metal phase-field representations. Chapter 5 presented an optimal data alignment methodology for multi-dimensional compression to the neural solver framework, as well as optimisation and further development of the neural architecture. An assessment of the framework's activation functions for phase-field representations was presented, with a further comparison of the hidden layer dimensions. Finally, the predictive performance of this autoregressive solver was presented, showcasing its capabilities for fast, flexible predictions for lithium-metal evolution. The autoregressive neural solver was numerically verified to the phase-field electrodeposition model and showed a performance of 3.83×10^{-5} for the one-dimensional case and 9.52×10^{-3} for the two-dimensional case. The numerical performance was also presented with a 40.3% reduction in computational time per step achieved over the phase-field representation for the trained model; however, it should be noted that the training requirements for the neural solver limit its current usage.

6.2 Perspectives and future directions

Limitations of the presented work above have been discussed in their respective chapters; however, some limitations are consistent between these elements of work. The author believes that improvements in open-source software packages should be completed to achieve field-wide improvements in electrochemical modelling. As previously described, multiple open-source methods exist to solve the Doyle-Fuller-Newman model numerically; however, each provides a different interface and convention to the end user. Integration of naming conventions and standardisation of architectural interfaces would lower the barrier of entry for new researchers and enable larger improvements for the field. For the presented packages in this thesis, future implementation of a shared ontology is planned to enable an easy transition for end-users.

The popularity of physics-based battery realisation algorithms has varied since the first publications. The author suspects that penetration into the field will depend heavily on future cell chemistries and the attainable benefits for physics-informed online plant models. To further lower the barrier to entry for realisation algorithms, the author recommends future work be developed inside previously implemented open-source packages when possible. An automated method to apply the CI-DRA inside of PyBAMM, or PETLION.jl, would provide a single platform for users to model the full system and the reduced-order derivation. This provides a challenge, as an automated derivation of the corresponding transfer functions is not straightforward. Julia's symbolic mathematics package, JuliaSymbolic.jl [211], provides a potential method for this integration; however, difficulties are expected due to the complexity of the derived equations. The currently implemented LiiBRA.jl package utilises the transfer function derivation originally presented in [152], which assumes that the electrolyte potential is independent of the electrolyte concentration when deriving the reaction flux transfer function; however, more recent work by Rodriguez et al. [166] has removed this assumption and shown improved reduced-order fidelity. Future work to integrate this derivation into LiiBRA.jl is planned with a final aim to provide the end-user with the functionality to select the transfer function derivation. Finally, the implementation of degradation models coupled with the state output from the CI-DRA for online capable degradation predictions is planned.

The open-source multi-scale package, BattCalc.jl presented in Chapter 4 provides capabilities in first-principle-based predictions; however, due to the underlying assumptions, this package has a limited design space for accurate predictions. Future work areas for this work include the integration of this methodology with an experimental data framework to improve automated predictions for material discovery and application-informed design. Due to the prediction limitations of this package, integration with a continuum model such as the DFN or SPMe to achieve informed parameterisation from experimental cell characterisation would provide improvement for battery engineers.

The presented work on lithium-metal anode modelling, BattPhase.jl and the corresponding autoregressive neural solver methodology have shown capabilities in fast numerical morphology predictions. As the underlying phase-field derivation assumes linear kinetics and omits concentration gradient effects, further work is required to achieve the recommended experimental validation of the numerical predictions. Future development in automated coupling the BattPhase.jl framework with the neural solver would provide a framework for easier deployment of the neural solver. Additional optimisation of the autoregressive neural solver for phase-field implementation is planned, with the aim of achieving an open-source framework for fast design-space explorations. Finally, verification of the neural solver's capabilities to capture varying length-scale effects, such as dendritic lithium morphology, is recommended for this methodology.

Appendix A

State space model generation

This appendix presents the model generation procedure utilising LiiBRA.jl for the real-time deployment discussed in section 3.3.6. To produce the deployed state-space models, LiiBRA.jl is first parameterised with the following architecture specifications, This architecture definition is consistent with the experimentally validated and numerically verified specifications introduced in Chapter 3. The process for generation follows the LiiBRA.jl example introduced in Figure 3.3, with the corresponding state-space array output from the Realise! function call. The dimensionality for the generated state space models in this configuration results in a single input dimension (applied current), and a twenty-one dimension output vector, comprising four spatial elements for concentration and potential

Table A.1 Architecture variable specification for deployed LiiBRA.jl models.

Variable	Definition	Value
$\mathcal{H}$	Size of square Hankel matrix	2500
$S_{e,m}$	Number of spatial particles in electrolyte	4
M	System order	4
$S_{s,m}$	Number of spatial particles in electrodes	2
T_{len}	Length of transfer function sampling time [hr]	4.5
F_s / T_s	System sampling frequencies [Hz]	4

in the electrolyte and two spatial elements for each solid phase transfer function spanning both the negative and positive domains. The first output is the **A** matrix and is presented as,

$$\boldsymbol{A} = \begin{bmatrix} 1.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.9984 & \text{-}0.002437 & \text{-}0.001114 & 0.001483 \\ 0.0 & \text{-}0.003446 & 0.9945 & \text{-}0.002623 & 0.002791 \\ 0.0 & \text{-}0.002875 & \text{-}0.00674 & 0.9923 & 0.01182 \\ 0.0 & 0.006684 & 0.01605 & 0.02642 & 0.9443 \end{bmatrix} \tag{A.1}$$

with the first index set to a value of one for time integration of the state variables. Next the **B** matrix is,

$$\boldsymbol{B} = \begin{bmatrix} 0.25 \\ 0.3276 \\ 0.4399 \\ 0.2947 \\ -0.6498 \end{bmatrix} \tag{A.2}$$

where $B[1]$ is the time period for the generated model. The **C** matrix is formed as,

$$\boldsymbol{C} = \begin{bmatrix} 0.0 & 0.9485 & 1.521 & 0.4695 & 0.4606 \\ 0.0 & -0.01827 & -0.004873 & 0.08019 & -0.1869 \\ 0.0 & -0.1244 & -0.1896 & -0.03122 & -0.09665 \\ 0.0 & -0.7102 & -1.18 & -0.4931 & -0.09483 \\ 0.0 & -6.529e-9 & 3.37e-9 & -8.425e-10 & -1.929e-10 \\ 0.0 & -6.529e-9 & 3.37e-9 & -8.425e-10 & -1.929e-10 \\ 0.0 & 7.721e-7 & 1.61e-6 & 2.145e-6 & -4.775e-6 \\ 2.007 & 4.342 & 5.003 & 1.965 & -7.01 \\ 2.007 & 4.157 & 4.622 & 1.452 & -5.861 \\ 0.0 & 1.467e-6 & 3.028e-6 & 4.044e-6 & -8.999e-6 \\ 0.0 & -0.0002099 & -0.0002345 & -7.594e-5 & 0.0003002 \\ 0.0 & -0.0002077 & -0.0002299 & -6.975e-5 & 0.0002865 \\ 0.0 & 9.552e-9 & 1.998e-8 & 2.746e-8 & -6.235e-8 \\ 0.0 & 1.069e-9 & 2.386e-9 & 3.718e-9 & -9.122e-9 \\ -1.579 & -0.7813 & -1.194 & -0.9805 & 1.604 \\ -1.579 & -0.9918 & -1.236 & -1.071 & 1.75 \\ 0.0 & 5.544e-12 & -2.862e-12 & 7.155e-13 & 1.638e-13 \\ 0.0 & 1.147e-11 & -5.922e-12 & 1.48e-12 & 3.389e-13 \\ 0.0 & -2.318e-11 & 1.196e-11 & -2.991e-12 & -6.847e-13 \\ 0.0 & -3.045e-9 & 1.574e-9 & -3.917e-10 & -9.223e-11 \\ 0.0 & 3.489e-9 & -1.799e-9 & 4.515e-10 & 1.008e-10 \end{bmatrix} \tag{A.3}$$

and finally, the **D** which corresponds to the low-frequency system response and is linearised online for the deployed system is defined as,

$$\boldsymbol{D} = \begin{bmatrix} 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ \text{-0.003447} \\ \text{-0.003829} \\ \text{-0.005977} \\ 0.0 \\ 0.0 \\ 0.00345 \\ \text{-0.004455} \\ \text{-0.003108} \\ \text{-5.406e-6} \\ \text{-3.771e-6} \\ 0.0 \\ 0.0 \\ \text{-3.013e-6} \\ \text{2.996e-6} \\ \text{3.265e-6} \\ 0.03834 \\ 0.04178 \end{bmatrix} \tag{A.4}$$

References

[1] Brady Planden et al. “A computationally informed realisation algorithm for lithium-ion batteries implemented with LiiBRA.jl”. In: *Journal of Energy Storage* 55 (Nov. 2022), p. 105637. ISSN: 2352152X. DOI: 10.1016/j.est.2022.105637. URL: https://linkinghub.elsevier.com/retrieve/pii/S2352152X22016255 (visited on 09/19/2022).

[2] Taejin Jang et al. “BattPhase—A Convergent, Non-Oscillatory, Efficient Algorithm and Code for Predicting Shape Changes in Lithium Metal Batteries Using Phase-Field Models: Part I. Secondary Current Distribution”. In: *Journal of The Electrochemical Society* (Aug. 3, 2022). ISSN: 0013-4651, 1945-7111. DOI: 10.1149/1945-7111/ac86a7. URL: https://iopscience.iop.org/article/10.1149/1945-7111/ac86a7 (visited on 08/23/2022).

[3] International Energy Agency. *Global EV Outlook 2022: Securing supplies for an electric future*. en. Global EV Outlook. OECD, May 2022. ISBN: 978-92-64-89887-5. DOI: 10.1787/c83f815c-en. URL: https://www.oecd-ilibrary.org/energy/global-ev-outlook-2022_c83f815c-en (visited on 07/11/2022).

[4] International Energy Agency. *Global EV Outlook 2019: Scaling-up the transition to electric mobility*. en. Global EV Outlook. OECD, June 2019. ISBN: 978-92-64-47013-2. DOI: 10.1787/35fb60bd-en. URL: https://www.oecd-ilibrary.org/energy/global-ev-outlook-2019_35fb60bd-en (visited on 07/10/2022).

[5] Fatih Birol. *World Energy Outlook 2020*. Tech. rep. Paris, France: IEA, 2020, p. 464. URL: https://www.iea.org/reports/world-energy-outlook-2020.

[6] Ricardo Faria et al. “Impact of the electricity mix and use profile in the life-cycle assessment of electric vehicles”. en. In: *Renewable and Sustainable Energy Reviews* 24 (Aug. 2013), pp. 271–287. ISSN: 13640321. DOI: 10.1016/j.rser.2013.03.063. URL: https://linkinghub.elsevier.com/retrieve/pii/S1364032113002220 (visited on 10/20/2020).

[7] Troy R. Hawkins et al. “Comparative Environmental Life Cycle Assessment of Conventional and Electric Vehicles”. en. In: *Journal of Industrial Ecology* 17.1 (Feb. 2013), pp. 53–64. ISSN: 1088-1980, 1530-9290. DOI: 10.1111/j.1530-9290.2012.00532.x. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1530-9290.2012.00532.x (visited on 10/20/2020).

[8] Georg Bieker. “A GLOBAL COMPARISON OF THE LIFE-CYCLE GREENHOUSE GAS EMISSIONS OF COMBUSTION ENGINE AND ELECTRIC PASSENGER CARS”. In: 2021.

[9] JongRoul Woo and Christopher L. Magee. “Forecasting the value of battery electric vehicles compared to internal combustion engine vehicles: The influence of driving range and battery technology”. In: *International Journal of Energy Research* 44.8 (June 25, 2020), pp. 6483–6501. ISSN: 0363-907X, 1099-114X. DOI: 10.1002/er.5382. URL: https://onlinelibrary.wiley.com/doi/10.1002/er.5382 (visited on 09/19/2022).

[10] Yayuan Liu, Yangying Zhu, and Yi Cui. “Challenges and opportunities towards fast-charging battery materials”. In: *Nature Energy* 4.7 (July 2019), pp. 540–550. ISSN: 2058-7546. DOI: 10.1038/s41560-019-0405-3. URL: http://www.nature.com/articles/s41560-019-0405-3 (visited on 09/19/2022).

[11] Jorn M. Reniers et al. “Improving optimal control of grid-connected lithium-ion batteries through more accurate battery and degradation modelling”. In: *Journal of Power Sources* 379 (Mar. 2018), pp. 91–102. ISSN: 03787753. DOI: 10.1016/j.jpowsour.2018.01.004. URL: https://linkinghub.elsevier.com/retrieve/pii/S0378775318300041 (visited on 03/17/2022).

[12] E. Miguel et al. “Electrochemical Model and Sigma Point Kalman Filter Based Online Oriented Battery Model”. In: *IEEE Access* 9 (2021), pp. 98072–98090. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2021.3095620. URL: https://ieeexplore.ieee.org/document/9477587/ (visited on 07/04/2022).

[13] *Goals for Advanced High-Performance Batteries for Electric Vehicle (EV) Applications – USCAR*. en. URL: https://uscar.org/wpfd_file/goals-for-advanced-high-performance-batteries-for-electric-vehicle-ev-applications/ (visited on 08/01/2022).

[14] BYD. *BYD’s New Blade Battery Set to Redefine EV Safety Standards - BYD USA*. URL: https://en.byd.com/news/byds-new-blade-battery-set-to-redefine-ev-safety-standards/ (visited on 07/06/2022).

[15] Leif E Asp et al. “Structural battery composites: a review”. In: *Functional Composites and Structures* 1.4 (Nov. 2019), p. 042001. ISSN: 2631-6331. DOI: 10.1088/2631-6331/ab5571. URL: https://iopscience.iop.org/article/10.1088/2631-6331/ab5571 (visited on 07/06/2022).

[16] J.-M. Tarascon and M. Armand. “Issues and challenges facing rechargeable lithium batteries”. en. In: *Nature* 414.6861 (Nov. 2001), pp. 359–367. ISSN: 0028-0836, 1476-4687. DOI: 10.1038/35104644. URL: http://www.nature.com/articles/35104644 (visited on 08/05/2022).

[17] Honghe Zheng et al. “A comprehensive understanding of electrode thickness effects on the electrochemical performances of Li-ion battery cathodes”. en. In: *Electrochimica Acta* 71 (June 2012), pp. 258–265. ISSN: 00134686. DOI: 10.1016/j.electacta.2012.03.161. URL: https://linkinghub.elsevier.com/retrieve/pii/S0013468612005270 (visited on 07/06/2022).

[18] K. Mizushima et al. “LixCoO2 (0<x<-1): A new cathode material for batteries of high energy density”. en. In: *Materials Research Bulletin* 15.6 (June 1980), pp. 783–789. ISSN: 00255408. DOI: 10.1016/0025-5408(80)90012-4. URL: https://linkinghub.elsevier.com/retrieve/pii/0025540880900124 (visited on 02/08/2020).

[19] Bruno Scrosati. “History of lithium batteries”. en. In: *Journal of Solid State Electrochemistry* 15.7-8 (July 2011), pp. 1623–1630. ISSN: 1432-8488, 1433-0768. DOI: 10.1007/s10008-011-1386-8. URL: http://link.springer.com/10.1007/s10008-011-1386-8 (visited on 07/10/2022).

[20] George E. Blomgren. “The Development and Future of Lithium Ion Batteries”. en. In: *Journal of The Electrochemical Society* 164.1 (2017), A5019–A5025. ISSN: 0013-4651, 1945-7111. DOI: 10.1149/2.0251701jes. URL: https://iopscience.iop.org/article/10.1149/2.0251701jes (visited on 07/06/2022).

[21] Hyung-Joo Noh et al. “Comparison of the structural and electrochemical properties of layered Li[NixCoyMnz]O2 (x = 1/3, 0.5, 0.6, 0.7, 0.8 and 0.85) cathode material for lithium-ion batteries”. In: *Journal of Power Sources* 233 (July 2013), pp. 121–130. ISSN: 03787753. DOI: 10.1016/j.jpowsour.2013.01.063. URL: https://linkinghub.elsevier.com/retrieve/pii/S0378775313001110 (visited on 10/26/2020).

[22] Naoki Nitta et al. “Li-ion battery materials: present and future”. en. In: *Materials Today* 18.5 (June 2015), pp. 252–264. ISSN: 13697021. DOI: 10.1016/j.mattod.2014.10.040. URL: https://linkinghub.elsevier.com/retrieve/pii/S1369702114004118 (visited on 09/15/2020).

[23] Yuanli Ding et al. “Automotive Li-Ion Batteries: Current Status and Future Perspectives”. en. In: *Electrochemical Energy Reviews* 2.1 (Mar. 2019), pp. 1–28. ISSN: 2520-8489, 2520-8136. DOI: 10.1007/s41918-018-0022-z. URL: http://link.springer.com/10.1007/s41918-018-0022-z (visited on 10/14/2020).

[24] Richard Schmuch et al. “Performance and cost of materials for lithium-based rechargeable automotive batteries”. en. In: *Nature Energy* 3.4 (Apr. 2018), pp. 267–278. ISSN: 2058-7546. DOI: 10.1038/s41560-018-0107-2. URL: http://www.nature.com/articles/s41560-018-0107-2 (visited on 10/26/2020).

[25] Larry Mickelson et al. “Bulk Stress Evolution during Intercalation of Lithium in Graphite”. en. In: *Journal of The Electrochemical Society* 161.14 (2014), A2121–A2127. ISSN: 0013-4651, 1945-7111. DOI: 10.1149/2.0591414jes. URL: https://iopscience.iop.org/article/10.1149/2.0591414jes (visited on 08/05/2022).

[26] Tao Gao et al. “Interplay of Lithium Intercalation and Plating on a Single Graphite Particle”. In: *Joule* 5.2 (Feb. 2021), pp. 393–414. ISSN: 25424351. DOI: 10.1016/j.joule.2020.12.020. URL: https://linkinghub.elsevier.com/retrieve/pii/S254243512030619X (visited on 09/10/2022).

[27] Huajun Tian et al. “High capacity group-IV elements (Si, Ge, Sn) based anodes for lithium-ion batteries”. en. In: *Journal of Materiomics* 1.3 (Sept. 2015), pp. 153–169. ISSN: 23528478. DOI: 10.1016/j.jmat.2015.06.002. URL: https://linkinghub.elsevier.com/retrieve/pii/S2352847815000477 (visited on 11/02/2020).

[28] Uday Kasavajjula, Chunsheng Wang, and A. John Appleby. “Nano- and bulk-silicon-based insertion anodes for lithium-ion secondary cells”. en. In: *Journal of Power Sources* 163.2 (Jan. 2007), pp. 1003–1039. ISSN: 03787753. DOI: 10.1016/j.jpowsour.2006.09.084. URL: https://linkinghub.elsevier.com/retrieve/pii/S037877530602026X (visited on 11/02/2020).

[29] Hyunjung Kim et al. “Three-Dimensional Porous Silicon Particles for Use in High-Performance Lithium Secondary Batteries”. en. In: *Angewandte Chemie* 120.52 (Dec. 2008), pp. 10305–10308. ISSN: 00448249, 15213757. DOI: 10.1002/ange.200804355. URL: http://doi.wiley.com/10.1002/ange.200804355 (visited on 11/02/2020).

[30] Lynn Trahey et al. “High-Capacity, Microporous Cu[sub 6]Sn[sub 5]–Sn Anodes for Li-Ion Batteries”. en. In: *Journal of The Electrochemical Society* 156.5 (2009), A385. ISSN: 00134651. DOI: 10.1149/1.3094033. URL: https://iopscience.iop.org/article/10.1149/1.3094033 (visited on 11/02/2020).

[31] Guan-Nan Zhu, Yong-Gang Wang, and Yong-Yao Xia. “Ti-based compounds as anode materials for Li-ion batteries”. en. In: *Energy & Environmental Science* 5.5 (2012), p. 6652. ISSN: 1754-5692, 1754-5706. DOI: 10.1039/c2ee03410g. URL: http://xlink.rsc.org/?DOI=c2ee03410g (visited on 11/02/2020).

[32] Kent J. Griffith et al. “Niobium tungsten oxides for high-rate lithium-ion energy storage”. en. In: *Nature* 559.7715 (July 2018), pp. 556–563. ISSN: 0028-0836, 1476-4687. DOI: 10.1038/s41586-018-0347-0. URL: http://www.nature.com/articles/s41586-018-0347-0 (visited on 06/25/2022).

[33] Jiangfeng Qian et al. “Anode-Free Rechargeable Lithium Metal Batteries”. en. In: *Advanced Functional Materials* 26.39 (Oct. 2016), pp. 7094–7102. ISSN: 1616301X. DOI: 10.1002/adfm.201602353. URL: https://onlinelibrary.wiley.com/doi/10.1002/adfm.201602353 (visited on 07/06/2022).

[34] Tesfaye Teka Hagos et al. “Locally Concentrated LiPF $_6$ in a Carbonate-Based Electrolyte with Fluoroethylene Carbonate as a Diluent for Anode-Free Lithium Metal Batteries”. en. In: *ACS Applied Materials & Interfaces* 11.10 (Mar. 2019), pp. 9955–9963. ISSN: 1944-8244, 1944-8252. DOI: 10.1021/acsami.8b21052. URL: https://pubs.acs.org/doi/10.1021/acsami.8b21052 (visited on 07/06/2022).

[35] A. J. Louli et al. “Optimizing Cycling Conditions for Anode-Free Lithium Metal Cells”. In: *Journal of the Electrochemical Society* 168.2 (Feb. 2021), p. 020515. ISSN: 0013-4651, 1945-7111. DOI: 10.1149/1945-7111/abe089. URL: https://iopscience.iop.org/article/10.1149/1945-7111/abe089 (visited on 08/12/2021).

[36] Peng Shi et al. “Electrochemical Diagram of an Ultrathin Lithium Metal Anode in Pouch Cells”. en. In: *Advanced Materials* 31.37 (Sept. 2019), p. 1902785. ISSN: 0935-9648, 1521-4095. DOI: 10.1002/adma.201902785. URL: https://onlinelibrary.wiley.com/doi/10.1002/adma.201902785 (visited on 07/10/2022).

[37] Yang Lu et al. “Critical Current Density in Solid-State Lithium Metal Batteries: Mechanism, Influences, and Strategies”. en. In: *Advanced Functional Materials* 31.18 (May 2021), p. 2009925. ISSN: 1616-301X, 1616-3028. DOI: 10.1002/adfm.202009925. URL: https://onlinelibrary.wiley.com/doi/10.1002/adfm.202009925 (visited on 07/10/2022).

[38] Fei Ding et al. “Effects of Carbonate Solvents and Lithium Salts on Morphology and Coulombic Efficiency of Lithium Electrode”. en. In: *Journal of The Electrochemical Society* 160.10 (2013), A1894–A1901. ISSN: 0013-4651, 1945-7111. DOI: 10.1149/2.100310jes. URL: https://iopscience.iop.org/article/10.1149/2.100310jes (visited on 07/10/2022).

[39] Dingchang Lin, Yayuan Liu, and Yi Cui. “Reviving the lithium metal anode for high-energy batteries”. en. In: *Nature Nanotechnology* 12.3 (Mar. 2017), pp. 194–206. ISSN: 1748-3387, 1748-3395. DOI: 10.1038/nnano.2017.16. URL: http://www.nature.com/articles/nnano.2017.16 (visited on 07/10/2022).

[40] Rochelle Weber et al. “Long cycle life and dendrite-free lithium morphology in anode-free lithium pouch cells enabled by a dual-salt liquid electrolyte”. en. In: *Nature Energy* 4.8 (Aug. 2019), pp. 683–689. ISSN: 2058-7546. DOI: 10.1038/s41560-019-0428-9. URL: http://www.nature.com/articles/s41560-019-0428-9 (visited on 07/07/2022).

[41] A. J. Louli et al. “Exploring the Impact of Mechanical Pressure on the Performance of Anode-Free Lithium Metal Cells”. en. In: *Journal of The Electrochemical Society* 166.8 (2019), A1291–A1299. ISSN: 0013-4651, 1945-7111. DOI: 10.1149/2.0091908jes. URL: https://iopscience.iop.org/article/10.1149/2.0091908jes (visited on 07/07/2022).

[42] A. J. Louli. “DEVELOPING ANODE-FREE LITHIUM METAL CELLS WITH LIQUID ELECTROLYTES”. PhD thesis. Halifax, Nova Scotia: Dalhousie University, Aug. 2021.

[43] *Faraday Multi-scale Modelling Project.* en-GB. URL: https://www.faraday.ac.uk/research/lithium-ion/battery-system-modelling/ (visited on 08/05/2022).

[44] Gregory L. Plett. “Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs”. en. In: *Journal of Power Sources* 134.2 (Aug. 2004), pp. 252–261. ISSN: 03787753. DOI: 10.1016/j.jpowsour.2004.02.031. URL: https://linkinghub.elsevier.com/retrieve/pii/S0378775304003593 (visited on 11/02/2020).

[45] Hongwen He et al. “State-of-Charge Estimation of the Lithium-Ion Battery Using an Adaptive Extended Kalman Filter Based on an Improved Thevenin Model”. In: *IEEE Transactions on Vehicular Technology* 60.4 (May 2011), pp. 1461–1469. ISSN: 0018-9545, 1939-9359. DOI: 10.1109/TVT.2011.2132812. URL: http://ieeexplore.ieee.org/document/5739545/ (visited on 11/02/2020).

[46] Dave Andre et al. “Advanced mathematical methods of SOC and SOH estimation for lithium-ion batteries”. en. In: *Journal of Power Sources* 224 (Feb. 2013), pp. 20–27. ISSN: 03787753. DOI: 10.1016/j.jpowsour.2012.10.001. URL: https://linkinghub.elsevier.com/retrieve/pii/S0378775312015303 (visited on 11/02/2020).

[47] Shi Zhao, Stephen R. Duncan, and David A. Howey. “Observability Analysis and State Estimation of Lithium-Ion Batteries in the Presence of Sensor Biases”. In: *IEEE Transactions on Control Systems Technology* 25.1 (Jan. 2017), pp. 326–333. ISSN: 1063-6536, 1558-0865. DOI: 10.1109/TCST.2016.2542115. URL: http://ieeexplore.ieee.org/document/7464285/ (visited on 11/02/2020).

[48] Wladislaw Waag, Stefan Käbitz, and Dirk Uwe Sauer. “Application-specific parameterization of reduced order equivalent circuit battery models for improved accuracy at dynamic load”. en. In: *Measurement* 46.10 (Dec. 2013), pp. 4085–4093. ISSN: 02632241. DOI: 10.1016/j.measurement.2013.07.025. URL: https://linkinghub.elsevier.com/retrieve/pii/S0263224113003242 (visited on 07/04/2022).

[49] Leo Wildfeuer, Philipp Gieler, and Alexander Karger. “Combining the Distribution of Relaxation Times from EIS and Time-Domain Data for Parameterizing Equivalent Circuit Models of Lithium-Ion Batteries”. en. In: *Batteries* 7.3 (Aug. 2021), p. 52. ISSN: 2313-0105. DOI: 10.3390/batteries7030052. URL: https://www.mdpi.com/2313-0105/7/3/52 (visited on 07/04/2022).

[50] Penelope K. Jones, Ulrich Stimming, and Alpha A. Lee. “Impedance-based forecasting of lithium-ion battery performance amid uneven usage”. In: *Nature Communications* 13.1 (Aug. 16, 2022), p. 4806. ISSN: 2041-1723. DOI: 10.1038/s41467-022-32422-w. URL: https://www.nature.com/articles/s41467-022-32422-w (visited on 09/03/2022).

[51] Adrien M. Bizeray et al. “Identifiability and Parameter Estimation of the Single Particle Lithium-Ion Battery Model”. In: *IEEE Transactions on Control Systems Technology* 27.5 (Sept. 2019), pp. 1862–1877. ISSN: 1063-6536, 1558-0865, 2374-0159. DOI: 10.1109/TCST.2018.2838097. URL: https://ieeexplore.ieee.org/document/8386442/ (visited on 01/27/2021).

[52] Antti Aitio et al. “Bayesian Parameter Estimation Applied to the Li-ion Battery Single Particle Model with Electrolyte Dynamics”. In: *arXiv:2001.09890 [cs, eess]* (Jan. 2020). arXiv: 2001.09890. URL: http://arxiv.org/abs/2001.09890 (visited on 03/21/2020).

[53] Yu Merla et al. “An easy-to-parameterise physics-informed battery model and its application towards lithium-ion battery cell design, diagnosis, and degradation”. en. In: *Journal of Power Sources* 384 (Apr. 2018), pp. 66–79. ISSN: 03787753. DOI: 10.1016/j.jpowsour.2018.02.065. URL: https://linkinghub.elsevier.com/retrieve/pii/S0378775318301861 (visited on 07/04/2022).

[54] Renato G. Nascimento et al. “Hybrid physics-informed neural networks for lithium-ion battery modeling and prognosis”. en. In: *Journal of Power Sources* 513 (Nov. 2021), p. 230526. ISSN: 03787753. DOI: 10.1016/j.jpowsour.2021.230526. URL: https://linkinghub.elsevier.com/retrieve/pii/S0378775321010259 (visited on 07/04/2022).

[55] Junchuan Shi, Alexis Rivera, and Dazhong Wu. “Battery health management using physics-informed machine learning: Online degradation modeling and remaining useful life prediction”. en. In: *Mechanical Systems and Signal Processing* 179 (Nov. 2022), p. 109347. ISSN: 08883270. DOI:

10.1016/j.ymssp.2022.109347. URL: https://linkinghub.elsevier.com/retrieve/pii/S0888327022004824 (visited on 07/04/2022).

[56] Jacqueline S. Edge et al. “Lithium ion battery degradation: what you need to know”. In: *Physical Chemistry Chemical Physics* 23.14 (2021), pp. 8200–8221. ISSN: 1463-9076, 1463-9084. DOI: 10.1039/D1CP00359C. URL: http://xlink.rsc.org/?DOI=D1CP00359C (visited on 02/28/2022).

[57] Xiaosong Hu, Shengbo Li, and Huei Peng. “A comparative study of equivalent circuit models for Li-ion batteries”. In: *Journal of Power Sources* 198 (Jan. 2012), pp. 359–367. ISSN: 03787753. DOI: 10.1016/j.jpowsour.2011.10.013. URL: https://linkinghub.elsevier.com/retrieve/pii/S0378775311019628 (visited on 11/02/2020).

[58] Gregory L. Plett. *Battery management systems: battery modeling. Volume 1*. en. OCLC: ocn909081842. Boston : London: Artech House, 2015. ISBN: 978-1-63081-023-8.

[59] Gregory L. Plett. *Battery management systems. Vol. 2: Equivalent-circuit methods*. en. Artech House power engineering and power electronics. OCLC: 948540631. Boston: Artech House, 2016. ISBN: 978-1-63081-027-6.

[60] Martin Petit, Eric Prada, and Valérie Sauvant-Moynot. “Development of an empirical aging model for Li-ion batteries and application to assess the impact of Vehicle-to-Grid strategies on battery lifetime”. In: *Applied Energy* 172 (June 2016), pp. 398–407. ISSN: 03062619. DOI: 10.1016/j.apenergy.2016.03.119. URL: https://linkinghub.elsevier.com/retrieve/pii/S0306261916304500 (visited on 03/17/2022).

[61] Joris de Hoog et al. “Combined cycling and calendar capacity fade modeling of a Nickel-Manganese-Cobalt Oxide Cell with real-life profile validation”. In: *Applied Energy* 200 (Aug. 2017), pp. 47–61. ISSN: 03062619. DOI: 10.1016/j.apenergy.2017.05.018. URL: https://linkinghub.elsevier.com/retrieve/pii/S0306261917305251 (visited on 03/17/2022).

[62] M. Schimpe et al. “Comprehensive Modeling of Temperature-Dependent Degradation Mechanisms in Lithium Iron Phosphate Batteries”. In: *Journal of The Electrochemical Society* 165.2 (2018), A181–A193. ISSN: 0013-4651, 1945-7111. DOI: 10.1149/2.1181714jes. URL: https://iopscience.iop.org/article/10.1149/2.1181714jes (visited on 03/17/2022).

[63] Marc Doyle. “Modeling of Galvanostatic Charge and Discharge of the Lithium/Polymer/Insertion Cell”. en. In: *Journal of The Electrochemical Society* 140.6 (1993), p. 1526. ISSN: 00134651. DOI: 10.1149/1.2221597. URL: https://iopscience.iop.org/article/10.1149/1.2221597 (visited on 02/08/2020).

[64] Thomas F. Fuller, Marc Doyle, and John Newman. “Simulation and Optimization of the Dual Lithium Ion Insertion Cell”. In: *Journal of The Electrochemical Society* 141.1 (Jan. 1994), pp. 1–10. ISSN: 0013-4651, 1945-7111. DOI: 10.1149/1.2054684. URL: https://iopscience.iop.org/article/10.1149/1.2054684 (visited on 03/09/2022).

[65] Scott G. Marquis et al. “An asymptotic derivation of a single particle model with electrolyte”. In: *arXiv:1905.12553 [physics]* (Nov. 2019). arXiv: 1905.12553. URL: http://arxiv.org/abs/1905.12553 (visited on 06/16/2020).

[66] Taejin Jang et al. “Towards Real-Time Simulation of Two-Dimensional Models for Electrodeposition/Stripping in Lithium-Metal Batteries”. In: *ECS Transactions* 104.1 (Oct. 2021), pp. 131–152. ISSN: 1938-5862, 1938-6737. DOI: 10.1149/10401.0131ecst. URL: https://iopscience.iop.org/article/10.1149/10401.0131ecst (visited on 02/20/2022).

[67] Lubhani Mishra et al. “Perspective—Mass Conservation in Models for Electrodeposition/Stripping in Lithium Metal Batteries”. In: *Journal of The Electrochemical Society* 168.9 (Sept. 2021), p. 092502. ISSN: 0013-4651, 1945-7111. DOI: 10.1149/1945-7111/ac2091. URL: https://iopscience.iop.org/article/10.1149/1945-7111/ac2091 (visited on 02/20/2022).

[68] Jorn M. Reniers, Grietus Mulder, and David A. Howey. “Review and Performance Comparison of Mechanical-Chemical Degradation Models for Lithium-Ion Batteries”. en. In: *Journal of The Electrochemical Society* 166.14 (2019). Number: 14, A3189–A3200. ISSN: 0013-4651, 1945-7111. DOI: 10.1149/2.0281914jes. URL: https://iopscience.iop.org/article/10.1149/2.0281914jes (visited on 01/10/2021).

[69] Joel C. Forman et al. “Reduction of an Electrochemistry-Based Li-Ion Battery Model via Quasi-Linearization and Padé Approximation”. In: *Journal of The Electrochemical Society* 158.2 (2011), A93. DOI: 10.1149/1.3519059. URL: https://doi.org/10.1149/1.3519059.

[70] Kandler A. Smith, Christopher D. Rahn, and Chao-Yang Wang. “Model Order Reduction of 1D Diffusion Systems Via Residue Grouping”. In: *Journal of Dynamic Systems, Measurement, and Control* 130.1 (Jan. 2008). ISSN: 0022-0434. DOI: 10.1115/1.2807068. URL: https://doi.org/10.1115/1.2807068.

[71] Venkatasailanathan Ramadesigan et al. “Efficient Reformulation of Solid-Phase Diffusion in Physics-Based Lithium-Ion Battery Models”. In: *Journal of The Electrochemical Society* 157.7 (2010), A854. DOI: 10.1149/1.3425622. URL: https://doi.org/10.1149/1.3425622.

[72] Venkat R. Subramanian, Vinten D. Diwakar, and Deepak Tapriyal. “Efficient Macro-Micro Scale Coupled Modeling of Batteries”. In: *Journal of The Electrochemical Society* 152.10 (2005), A2002. DOI: 10.1149/1.2032427. URL: https://doi.org/10.1149/1.2032427.

[73] James L. Lee, Andrew Chemistruck, and Gregory L. Plett. “Discrete-time realization of transcendental impedance models, with application to modeling spherical solid diffusion”. In: *Journal of Power Sources* 206 (May 2012), pp. 367–377. ISSN: 03787753. DOI: 10.1016/j.jpowsour.2012.01.134. URL: https://linkinghub.elsevier.com/retrieve/pii/S0378775312002789 (visited on 02/07/2020).

[74] Albert Rodríguez, Gregory L. Plett, and M. Scott Trimboli. “Comparing four model-order reduction techniques, applied to lithium-ion battery-cell internal electrochemical transfer functions”. en. In: *eTransportation* 1 (Aug. 2019), p. 100009. ISSN: 25901168. DOI: 10.1016/j.etran.2019.100009. URL: https://linkinghub.elsevier.com/retrieve/pii/S2590116819300098 (visited on 02/21/2022).

[75] Xing Jin et al. “Physically-based reduced-order capacity loss model for graphite anodes in Li-ion battery cells”. en. In: *Journal of Power Sources* 342 (Feb. 2017), pp. 750–761. ISSN: 03787753. DOI: 10.1016/j.jpowsour.2016.12.099. URL: https://linkinghub.elsevier.com/retrieve/pii/S037877531631802X (visited on 02/23/2022).

[76] Sangwoo Han, Yifan Tang, and Saeed Khaleghi Rahimian. “A numerically efficient method of solving the full-order pseudo-2-dimensional (P2D) Li-ion cell model”. In: *Journal of Power Sources* 490, 229571 (Apr. 2021), p. 229571. DOI: 10.1016/j.jpowsour.2021.229571.

[77] C. R. Birkl et al. “A Parametric Open Circuit Voltage Model for Lithium Ion Batteries”. In: *Journal of The Electrochemical Society* 162.12 (2015), A2271–A2280. ISSN: 0013-4651, 1945-7111. DOI: 10.1149/2.0331512jes. URL: https://iopscience.iop.org/article/10.1149/2.0331512jes (visited on 09/26/2022).

[78] Madeleine Ecker et al. “Parameterization of a Physico-Chemical Model of a Lithium-Ion Battery: II. Model Validation”. In: *Journal of The Electrochemical Society* 162.9 (2015), A1849–A1857. ISSN: 0013-4651, 1945-7111. DOI: 10.1149/2.0541509jes. URL: https://iopscience.iop.org/article/10.1149/2.0541509jes (visited on 09/26/2022).

[79] Madeleine Ecker et al. “Parameterization of a Physico-Chemical Model of a Lithium-Ion Battery: I. Determination of Parameters”. In: *Journal of The Electrochemical Society* 162.9 (2015), A1836–A1848. ISSN: 0013-4651, 1945-7111. DOI: 10.1149/2.0551509jes. URL: https://iopscience.iop.org/article/10.1149/2.0551509jes (visited on 09/26/2022).

[80] Johannes Schmalstieg et al. “Full Cell Parameterization of a High-Power Lithium-Ion Battery for a Physico-Chemical Model: Part I. Physical and Electrochemical Parameters”. In: *Journal of The Electrochemical Society* 165.16 (2018), A3799–A3810. ISSN: 0013-4651, 1945-7111. DOI: 10.1149/2.0321816jes. URL: https://iopscience.iop.org/article/10.1149/2.0321816jes (visited on 09/26/2022).

[81] Chang-Hui Chen et al. “Development of Experimental Techniques for Parameterization of Multi-scale Lithium-ion Battery Models”. In: *Journal of The Electrochemical Society* 167.8 (May 2020), p. 080534. ISSN: 1945-7111. DOI: 10.1149/1945-7111/ab9050. URL: https://iopscience.iop.org/article/10.1149/1945-7111/ab9050 (visited on 10/31/2020).

[82] Kieran O’Regan et al. *Thermal-electrochemical parameters of a high energy lithium-ion cylindrical battery.* preprint. Chemistry, Jan. 2022. DOI: 10.26434/chemrxiv-2022-d2q4n. URL: https://chemrxiv.org/engage/chemrxiv/article-details/61e2fd304a603d7d99314784 (visited on 05/04/2022).

[83] Kandler Smith and Chao-Yang Wang. “Power and thermal characterization of a lithium-ion battery pack for hybrid-electric vehicles”. In: *Journal of Power Sources* 160.1 (2006), pp. 662–673. ISSN: 0378-7753. DOI: https://doi.org/10.1016/j.jpowsour.2006.01.038. URL: https://www.sciencedirect.com/science/article/pii/S0378775306001017.

[84] Marcelo A. Xavier et al. “A Computational Framework for Lithium Ion Cell-Level Model Predictive Control Using a Physics-Based Reduced-Order Model”. In: *IEEE Control Systems Letters* 5.4 (2021), pp. 1387–1392. DOI: 10.1109/LCSYS.2020.3038131.

[85] Tedjani Mesbahi et al. “Dynamical modeling of Li-ion batteries for electric vehicle applications based on hybrid Particle Swarm–Nelder–Mead (PSO–NM) optimization algorithm”. In: *Electric Power Systems Research* 131 (Feb. 2016), pp. 195–204. ISSN: 03787796. DOI: 10.1016/j.epsr.2015.10.018. URL: https://linkinghub.elsevier.com/retrieve/pii/S0378779615003144 (visited on 09/02/2022).

[86] KaiChin Lim et al. “Fading Kalman filter-based real-time state of charge estimation in LiFePO4 battery-powered electric vehicles”. In: *Applied Energy* 169 (May 2016), pp. 40–48. ISSN: 03062619. DOI: 10.1016/j.apenergy.2016.01.096. URL: https://linkinghub.elsevier.com/retrieve/pii/S0306261916300848 (visited on 09/02/2022).

[87] Haifeng Dai et al. “Adaptive model parameter identification for large capacity Li-ion batteries on separated time scales”. In: *Applied Energy* 184 (Dec. 2016), pp. 119–131. ISSN: 03062619. DOI: 10.1016/j.apenergy.2016.10.020. URL: https://linkinghub.elsevier.com/retrieve/pii/S0306261916314441 (visited on 09/02/2022).

[88] Ximing Cheng et al. “Novel Parametric Circuit Modeling for Li-Ion Batteries”. In: *Energies* 9.7 (July 14, 2016), p. 539. ISSN: 1996-1073. DOI: 10.3390/en9070539. URL: http://www.mdpi.com/1996-1073/9/7/539 (visited on 09/02/2022).

[89] John Newman and William Tiedemann. “Porous-electrode theory with battery applications”. en. In: *AIChE Journal* 21.1 (Jan. 1975), pp. 25–41. ISSN: 0001-1541, 1547-5905. DOI: 10.1002/aic.690210103. URL: https://onlinelibrary.wiley.com/doi/10.1002/aic.690210103 (visited on 03/09/2022).

[90] John Newman and Karen E Thomas-Alyea. *Electrochemical systems*. John Wiley & Sons, 2012.

[91] Kandler Smith and Chao-Yang Wang. “Solid-state diffusion limitations on pulse operation of a lithium ion cell for hybrid electric vehicles”. In: *Journal of Power Sources* 161.1 (Oct. 2006), pp. 628–639. ISSN: 03787753. DOI: 10.1016/j.jpowsour.2006.03.050. URL: https://linkinghub.elsevier.com/retrieve/pii/S0378775306006161 (visited on 11/16/2021).

[92] Venkat R. Subramanian et al. “Mathematical Model Reformulation for Lithium-Ion Battery Simulations: Galvanostatic Boundary Conditions”. In: *Journal of The Electrochemical Society* 156.4 (2009), A260. ISSN: 00134651. DOI: 10.1149/1.3065083. URL: https://iopscience.iop.org/article/10.1149/1.3065083 (visited on 11/16/2021).

[93] A. M. Bizeray et al. “Lithium-ion battery thermal-electrochemical model-based state estimation using orthogonal collocation and a modified extended Kalman filter”. In: *Journal of Power Sources* 296 (Nov. 2015). arXiv: 1506.08689, pp. 400–412. ISSN: 03787753. DOI: 10.1016/j.jpowsour.2015.07.019. URL: http://arxiv.org/abs/1506.08689 (visited on 02/08/2020).

[94] Long Cai and Ralph E. White. "Lithium ion cell modeling using orthogonal collocation on finite elements". In: *Journal of Power Sources* 217 (Nov. 2012), pp. 248–255. ISSN: 03787753. DOI: 10.1016/j.jpowsour.2012.06.043. URL: https://linkinghub.elsevier.com/retrieve/pii/S0378775312010439 (visited on 02/07/2020).

[95] Paul W. C. Northrop et al. "Coordinate Transformation, Orthogonal Collocation, Model Reformulation and Simulation of Electrochemical-Thermal Behavior of Lithium-Ion Battery Stacks". In: *Journal of The Electrochemical Society* 158.12 (2011), A1461. ISSN: 00134651. DOI: 10.1149/2.058112jes. URL: https://iopscience.iop.org/article/10.1149/2.058112jes (visited on 06/21/2020).

[96] Ross Drummond et al. "A Feedback Interpretation of the Doyle–Fuller–Newman Lithium-Ion Battery Model". In: *IEEE Transactions on Control Systems Technology* 28.4 (July 2020), pp. 1284–1295. ISSN: 1063-6536, 1558-0865, 2374-0159. DOI: 10.1109/TCST.2019.2909722. URL: https://ieeexplore.ieee.org/document/8694827/ (visited on 11/16/2021).

[97] Valentin Sulzer et al. *Python Battery Mathematical Modelling (PyBaMM)*. Version 22.1. June 2021. DOI: 10.5334/jors.309. URL: https://github.com/pybamm-team/PyBaMM.

[98] Marc D Berliner et al. "Methods—PETLION: Open-Source Software for Millisecond-Scale Porous Electrode Theory-Based Lithium-Ion Battery Simulations". In: *Journal of The Electrochemical Society* 168.9 (2021), p. 090504.

[99] JS Newman. "Fortran programs for simulation of electrochemical systems, dualfoil. f program for lithium battery simulation". In: *Available on: www. cchem. berkeley. edu/jsngrp/fortran. html* (2004).

[100] Marcello Torchio et al. "LIONSIMBA: A Matlab Framework Based on a Finite Volume Model Suitable for Li-Ion Battery Design, Simulation, and Control". en. In: *Journal of The Electrochemical Society* 163.7 (2016), A1192–A1205. DOI: 10.1149/2.0291607jes. URL: https://iopscience.iop.org/article/10.1149/2.0291607jes (visited on 06/22/2020).

[101] Scott J. Moura. *fastDFN*. https://github.com/scott-moura/fastDFN. 2016.

[102] Bala S. Haran, Branko N. Popov, and Ralph E. White. "Determination of the hydrogen diffusion coefficient in metal hydrides by impedance spectroscopy". en. In: *Journal of Power Sources* 75.1 (Sept. 1998), pp. 56–63. ISSN: 03787753. DOI: 10.1016/S0378-7753(98)00092-5. URL: https://linkinghub.elsevier.com/retrieve/pii/S0378775398000925 (visited on 11/01/2020).

[103] Shriram Santhanagopalan et al. "Review of models for predicting the cycling performance of lithium ion batteries". en. In: *Journal of Power Sources* 156.2 (June 2006), pp. 620–628. ISSN: 03787753. DOI: 10.1016/j.jpowsour.2005.05.070. URL: https://linkinghub.elsevier.com/retrieve/pii/S0378775305007810 (visited on 11/02/2020).

[104] Scott G. Marquis et al. “A Suite of Reduced-Order Models of a Single-Layer Lithium-ion Pouch Cell”. In: *arXiv:2008.03691 [physics]* (Aug. 2020). arXiv: 2008.03691. URL: http://arxiv.org/abs/2008.03691 (visited on 10/19/2020).

[105] F Brosa Planella et al. “A continuum of physics-based lithium-ion battery models reviewed”. In: *Progress in Energy* 4.4 (Oct. 1, 2022), p. 042003. ISSN: 2516-1083. DOI: 10.1088/2516-1083/ac7d31. URL: https://iopscience.iop.org/article/10.1088/2516-1083/ac7d31 (visited on 12/14/2022).

[106] Liuying Li et al. “Lithium-ion battery cathode and anode potential observer based on reduced-order electrochemical single particle model”. In: *Journal of Energy Storage* 44 (Dec. 2021), p. 103324. ISSN: 2352152X. DOI: 10.1016/j.est.2021.103324. URL: https://linkinghub.elsevier.com/retrieve/pii/S2352152X21010161 (visited on 12/15/2022).

[107] Fan Zhang et al. “State-of-charge estimation based on microcontroller-implemented sigma-point Kalman filter in a modular cell balancing system for Lithium-Ion battery packs”. In: *2015 IEEE 16th Workshop on Control and Modeling for Power Electronics (COMPEL)*. 2015 IEEE 16th Workshop on Control and Modeling for Power Electronics (COMPEL). Vancouver, BC: IEEE, July 2015, pp. 1–7. ISBN: 978-1-4673-6847-6. DOI: 10.1109/COMPEL.2015.7236525. URL: https://ieeexplore.ieee.org/document/7236525/ (visited on 09/30/2022).

[108] Marcelo A. Xavier et al. “A Low-Cost MPC-Based Algorithm for Battery Power Limit Estimation”. In: *2020 American Control Conference (ACC)*. 2020 American Control Conference (ACC). Denver, CO, USA: IEEE, July 2020, pp. 1161–1166. ISBN: 978-1-5386-8266-1. DOI: 10.23919/ACC45564.2020.9147337. URL: https://ieeexplore.ieee.org/document/9147337/ (visited on 09/30/2022).

[109] Jinqiu Zhou et al. “High-Safety All-Solid-State Lithium-Metal Battery with High-Ionic-Conductivity Thermoresponsive Solid Polymer Electrolyte”. In: *Nano Letters* 19.5 (May 8, 2019), pp. 3066–3073. ISSN: 1530-6984, 1530-6992. DOI: 10.1021/acs.nanolett.9b00450. URL: https://pubs.acs.org/doi/10.1021/acs.nanolett.9b00450 (visited on 09/08/2022).

[110] Shuixin Xia et al. “Practical Challenges and Future Perspectives of All-Solid-State Lithium-Metal Batteries”. In: *Chem* 5.4 (Apr. 2019), pp. 753–785. ISSN: 24519294. DOI: 10.1016/j.chempr.2018.11.013. URL: https://linkinghub.elsevier.com/retrieve/pii/S2451929418305308 (visited on 09/08/2022).

[111] Yunpeng Jiang et al. “Modified solid-electrolyte interphase toward stable Li metal anode”. In: *Nano Energy* 77 (Nov. 2020), p. 105308. ISSN: 22112855. DOI: 10.1016/j.nanoen.2020.105308. URL: https://linkinghub.elsevier.com/retrieve/pii/S2211285520308855 (visited on 09/08/2022).

[112] Mingguang Wu et al. “Perspective on solid-electrolyte interphase regulation for lithium metal batteries”. In: *SmartMat* 2.1 (Mar. 2021), pp. 5–11. ISSN: 2688-819X, 2688-819X. DOI: 10.1002/smm2.1015. URL: https://onlinelibrary.wiley.com/doi/10.1002/smm2.1015 (visited on 09/08/2022).

[113] Charles Monroe and John Newman. “Dendrite Growth in Lithium/Polymer Systems”. In: *Journal of The Electrochemical Society* 150.10 (2003), A1377. ISSN: 00134651. DOI: 10.1149/1.1606686. URL: https://iopscience.iop.org/article/10.1149/1.1606686 (visited on 09/25/2022).

[114] Charles Monroe and John Newman. “The Effect of Interfacial Deformation on Electrodeposition Kinetics”. In: *Journal of The Electrochemical Society* 151.6 (2004), A880. ISSN: 00134651. DOI: 10.1149/1.1710893. URL: https://iopscience.iop.org/article/10.1149/1.1710893 (visited on 09/25/2022).

[115] Charles Monroe and John Newman. “The Impact of Elastic Deformation on Deposition Kinetics at Lithium/Polymer Interfaces”. In: *Journal of The Electrochemical Society* 152.2 (2005), A396. ISSN: 00134651. DOI: 10.1149/1.1850854. URL: https://iopscience.iop.org/article/10.1149/1.1850854 (visited on 09/25/2022).

[116] Luis E. Camacho-Forero, Taylor W. Smith, and Perla B. Balbuena. “Effects of High and Low Salt Concentration in Electrolytes at Lithium–Metal Anode Surfaces”. In: *The Journal of Physical Chemistry C* 121.1 (Jan. 12, 2017), pp. 182–194. ISSN: 1932-7447, 1932-7455. DOI: 10.1021/acs.jpcc.6b10774. URL: https://pubs.acs.org/doi/10.1021/acs.jpcc.6b10774 (visited on 09/25/2022).

[117] Pallab Barai, Kenneth Higa, and Venkat Srinivasan. “Impact of External Pressure and Electrolyte Transport Properties on Lithium Dendrite Growth”. In: *Journal of The Electrochemical Society* 165.11 (2018), A2654–A2666. ISSN: 0013-4651, 1945-7111. DOI: 10.1149/2.0651811jes. URL: https://iopscience.iop.org/article/10.1149/2.0651811jes (visited on 09/25/2022).

[118] Xin Zhang et al. “Pressure-Driven Interface Evolution in Solid-State Lithium Metal Batteries”. In: *Cell Reports Physical Science* 1.2 (Feb. 2020), p. 100012. ISSN: 26663864. DOI: 10.1016/j.xcrp.2019.100012. URL: https://linkinghub.elsevier.com/retrieve/pii/S266638641930013X (visited on 09/25/2022).

[119] Zijian Hong and Venkatasubramanian Viswanathan. “Phase-Field Simulations of Lithium Dendrite Growth with Open-Source Software”. In: *ACS Energy Letters* 3.7 (July 13, 2018), pp. 1737–1743. ISSN: 2380-8195, 2380-8195. DOI: 10.1021/acsenergylett.8b01009. URL: https://pubs.acs.org/doi/10.1021/acsenergylett.8b01009 (visited on 09/25/2022).

[120] Liting Gao and Zhansheng Guo. “Phase-field simulation of Li dendrites with multiple parameters influence”. In: *Computational Materials Science* 183 (Oct. 2020), p. 109919. ISSN: 09270256. DOI: 10.1016/j.commatsci.2020.109919. URL: https://linkinghub.elsevier.com/retrieve/pii/S0927025620304109 (visited on 09/25/2022).

[121] Yalun Li et al. “Thermal Runaway Triggered by Plated Lithium on the Anode after Fast Charging”. In: *ACS Applied Materials & Interfaces* 11.50 (Dec. 18, 2019), pp. 46839–46850. ISSN: 1944-8244, 1944-8252. DOI: 10.1021/acsami.9b16589. URL: https://pubs.acs.org/doi/10.1021/acsami.9b16589 (visited on 09/25/2022).

[122] David Montes de Oca Zapiain, James A. Stewart, and Rémi Dingreville. "Accelerating phase-field-based microstructure evolution predictions via surrogate models trained by machine learning methods". In: *npj Computational Materials* 7.1 (Dec. 2021), p. 3. ISSN: 2057-3960. DOI: 10.1038/s41524-020-00471-8. URL: http://www.nature.com/articles/s41524-020-00471-8 (visited on 09/05/2022).

[123] Vitaliy Yurkiv et al. "Phase-field modeling of solid electrolyte interface (SEI) influence on Li dendritic behavior". en. In: *Electrochimica Acta* 265 (Mar. 2018), pp. 609–619. ISSN: 00134686. DOI: 10.1016/j.electacta.2018.01.212. URL: https://linkinghub.elsevier.com/retrieve/pii/S0013468618302809 (visited on 07/28/2022).

[124] Yunwei Zhang et al. "Identifying degradation patterns of lithium ion batteries from impedance spectroscopy using machine learning". In: *Nature Communications* 11.1 (Dec. 2020), p. 1706. ISSN: 2041-1723. DOI: 10.1038/s41467-020-15235-7. URL: http://www.nature.com/articles/s41467-020-15235-7 (visited on 09/03/2022).

[125] Kristen A. Severson et al. "Data-driven prediction of battery cycle life before capacity degradation". In: *Nature Energy* 4.5 (May 2019), pp. 383–391. ISSN: 2058-7546. DOI: 10.1038/s41560-019-0356-8. URL: http://www.nature.com/articles/s41560-019-0356-8 (visited on 09/03/2022).

[126] Ephrem Chemali et al. "State-of-charge estimation of Li-ion batteries using deep neural networks: A machine learning approach". In: *Journal of Power Sources* 400 (Oct. 2018), pp. 242–255. ISSN: 03787753. DOI: 10.1016/j.jpowsour.2018.06.104. URL: https://linkinghub.elsevier.com/retrieve/pii/S0378775318307080 (visited on 09/03/2022).

[127] Darius Roman et al. "Machine learning pipeline for battery state-of-health estimation". In: *Nature Machine Intelligence* 3.5 (May 2021), pp. 447–456. ISSN: 2522-5839. DOI: 10.1038/s42256-021-00312-3. URL: http://www.nature.com/articles/s42256-021-00312-3 (visited on 09/03/2022).

[128] Parker Liu et al. "High throughput materials research and development for lithium ion batteries". In: *Journal of Materiomics* 3.3 (Sept. 2017), pp. 202–208. ISSN: 23528478. DOI: 10.1016/j.jmat.2017.07.004. URL: https://linkinghub.elsevier.com/retrieve/pii/S2352847817300527 (visited on 09/03/2022).

[129] Yoichi Takagishi, Takumi Yamanaka, and Tatsuya Yamaue. "Machine Learning Approaches for Designing Mesoscale Structure of Li-Ion Battery Electrodes". In: *Batteries* 5.3 (Aug. 1, 2019), p. 54. ISSN: 2313-0105. DOI: 10.3390/batteries5030054. URL: https://www.mdpi.com/2313-0105/5/3/54 (visited on 09/03/2022).

[130] Lei Cheng et al. "Accelerating Electrolyte Discovery for Energy Storage with High-Throughput Screening". In: *The Journal of Physical Chemistry Letters* 6.2 (Jan. 15, 2015), pp. 283–291. ISSN: 1948-7185, 1948-7185. DOI: 10.1021/jz502319n. URL: https://pubs.acs.org/doi/10.1021/jz502319n (visited on 09/03/2022).

[131] Abhishek Khetan, Alan Luntz, and Venkatasubramanian Viswanathan. "Trade-Offs in Capacity and Rechargeability in Nonaqueous Li–O $_2$ Batteries: Solution-Driven Growth versus Nucleophilic Stability". In: *The Journal of Physical Chemistry Letters* 6.7 (Apr. 2, 2015), pp. 1254–1259. ISSN: 1948-7185, 1948-7185. DOI: 10.1021/acs.jpclett.5b00324. URL: https://pubs.acs.org/doi/10.1021/acs.jpclett.5b00324 (visited on 09/03/2022).

[132] Austin D. Sendek et al. "Machine Learning-Assisted Discovery of Solid Li-Ion Conducting Materials". In: *Chemistry of Materials* 31.2 (Jan. 22, 2019), pp. 342–352. ISSN: 0897-4756, 1520-5002. DOI: 10.1021/acs.chemmater.8b03272. URL: https://pubs.acs.org/doi/10.1021/acs.chemmater.8b03272 (visited on 09/03/2022).

[133] Zhiwei Fang and Justin Zhan. "Deep Physical Informed Neural Networks for Metamaterial Design". In: *IEEE Access* 8 (2020), pp. 24506–24513. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2019.2963375. URL: https://ieeexplore.ieee.org/document/8946546/ (visited on 09/03/2022).

[134] Dehao Liu and Yan Wang. "Multi-Fidelity Physics-Constrained Neural Network and Its Application in Materials Modeling". In: *Journal of Mechanical Design* 141.12 (Dec. 1, 2019), p. 121403. ISSN: 1050-0472, 1528-9001. DOI: 10.1115/1.4044400. URL: https://asmedigitalcollection.asme.org/mechanicaldesign/article/doi/10.1115/1.4044400/956256/MultiFidelity-PhysicsConstrained-Neural-Network (visited on 09/03/2022).

[135] Yuyao Chen et al. "Physics-informed neural networks for inverse problems in nano-optics and metamaterials". In: *Optics Express* 28.8 (Apr. 13, 2020), p. 11618. ISSN: 1094-4087. DOI: 10.1364/OE.384875. URL: https://opg.optica.org/abstract.cfm?URI=oe-28-8-11618 (visited on 09/03/2022).

[136] Dongkun Zhang, Ling Guo, and George Em Karniadakis. "Learning in Modal Space: Solving Time-Dependent Stochastic PDEs Using Physics-Informed Neural Networks". In: *SIAM Journal on Scientific Computing* 42.2 (Jan. 2020), A639–A665. ISSN: 1064-8275, 1095-7197. DOI: 10.1137/19M1260141. URL: https://epubs.siam.org/doi/10.1137/19M1260141 (visited on 09/03/2022).

[137] Yibo Yang and Paris Perdikaris. "Adversarial uncertainty quantification in physics-informed neural networks". In: *Journal of Computational Physics* 394 (Oct. 2019), pp. 136–152. ISSN: 00219991. DOI: 10.1016/j.jcp.2019.05.027. URL: https://linkinghub.elsevier.com/retrieve/pii/S0021999119303584 (visited on 09/03/2022).

[138] Yinhao Zhu et al. "Physics-constrained deep learning for high-dimensional surrogate modeling and uncertainty quantification without labeled data". In: *Journal of Computational Physics* 394 (Oct. 2019), pp. 56–81. ISSN: 00219991. DOI: 10.1016/j.jcp.2019.05.024. URL: https://linkinghub.elsevier.com/retrieve/pii/S0021999119303559 (visited on 09/03/2022).

[139] Jiequn Han, Arnulf Jentzen, and Weinan E. "Solving high-dimensional partial differential equations using deep learning". In: *Proceedings of the National Academy of Sciences* 115.34 (Aug. 21, 2018), pp. 8505–8510. ISSN: 0027-8424, 1091-6490. DOI: 10.1073/pnas.1718942115. URL: https://pnas.org/doi/full/10.1073/pnas.1718942115 (visited on 09/03/2022).

[140] Justin Sirignano and Konstantinos Spiliopoulos. “DGM: A deep learning algorithm for solving partial differential equations”. In: *Journal of Computational Physics* 375 (Dec. 2018), pp. 1339–1364. ISSN: 00219991. DOI: 10.1016/j.jcp.2018.08.029. URL: https://linkinghub.elsevier.com/retrieve/pii/S0021999118305527 (visited on 09/03/2022).

[141] Salvatore Cuomo et al. *Scientific Machine Learning through Physics-Informed Neural Networks: Where we are and What’s next*. June 7, 2022. arXiv: 2201.05624[physics]. URL: http://arxiv.org/abs/2201.05624 (visited on 09/26/2022).

[142] Yohai Bar-Sinai et al. “Learning data-driven discretizations for partial differential equations”. In: *Proceedings of the National Academy of Sciences* 116.31 (July 30, 2019), pp. 15344–15349. ISSN: 0027-8424, 1091-6490. DOI: 10.1073/pnas.1814058116. URL: https://pnas.org/doi/full/10.1073/pnas.1814058116 (visited on 09/03/2022).

[143] Daniel Greenfeld et al. “Learning to Optimize Multigrid PDE Solvers”. In: *Proceedings of the 36th International Conference on Machine Learning*. Ed. by Kamalika Chaudhuri and Ruslan Salakhutdinov. Vol. 97. Proceedings of Machine Learning Research. PMLR, Sept. 2019, pp. 2415–2423. URL: https://proceedings.mlr.press/v97/greenfeld19a.html.

[144] Jun-Ting Hsieh et al. *Learning Neural PDE Solvers with Convergence Guarantees*. June 4, 2019. arXiv: 1906.01200[cs,stat]. URL: http://arxiv.org/abs/1906.01200 (visited on 09/03/2022).

[145] Johannes Brandstetter, Daniel Worrall, and Max Welling. *Message Passing Neural PDE Solvers*. Mar. 26, 2022. arXiv: 2202.03376[cs,math]. URL: http://arxiv.org/abs/2202.03376 (visited on 09/03/2022).

[146] Jeff Bezanson et al. “Julia: A Fast Dynamic Language for Technical Computing”. In: *arXiv:1209.5145 [cs]* (Sept. 23, 2012). arXiv: 1209.5145. URL: http://arxiv.org/abs/1209.5145 (visited on 02/18/2022).

[147] Jeff Bezanson et al. “Julia: A fresh approach to numerical computing”. In: *SIAM Review* 59.1 (2017), pp. 65–98. DOI: 10.1137/141000671. URL: https://epubs.siam.org/doi/10.1137/141000671.

[148] Siu Kwan Lam et al. *numba/numba: Version 0.56.2*. Sept. 5, 2022. DOI: 10.5281/zenodo.7051340. URL: https://zenodo.org/record/7051340 (visited on 09/14/2022).

[149] Brady Planden, Katie Lukow, and Denise Morrey. “Methodology Modelling Anode-Free Lithium Metal Batteries Utilising Message Passing Graph Neural Networks Background”. en. In: Publisher: Unpublished. 2022. DOI: 10.13140/RG.2.2.36344.32001. URL: https://rgdoi.net/10.13140/RG.2.2.36344.32001 (visited on 08/01/2022).

[150] Torben Jacobsen and Keld West. “Diffusion impedance in planar, cylindrical and spherical symmetry”. In: *Electrochimica Acta* 40.2 (Feb. 1995), pp. 255–262. ISSN: 00134686. DOI: 10.1016/0013-4686(94)E0192-3. URL: https://linkinghub.elsevier.com/retrieve/pii/0013468694E01923 (visited on 03/19/2020).

[151] Kandler A. Smith, Christopher D. Rahn, and Chao-Yang Wang. "Control oriented 1D electrochemical model of lithium ion battery". In: *Energy Conversion and Management* 48.9 (Sept. 2007), pp. 2565–2578. ISSN: 01968904. DOI: 10.1016/j.enconman.2007.03.015. URL: https://linkinghub.elsevier.com/retrieve/pii/S0196890407000908 (visited on 02/08/2020).

[152] James L. Lee, Andrew Chemistruck, and Gregory L. Plett. "One-dimensional physics-based reduced-order model of lithium-ion dynamics". In: *Journal of Power Sources* 220 (Dec. 2012), pp. 430–448. ISSN: 03787753. DOI: 10.1016/j.jpowsour.2012.07.075. URL: https://linkinghub.elsevier.com/retrieve/pii/S0378775312012104 (visited on 02/07/2020).

[153] Ravi P. Agarwal and Donal O'Regan. *An Introduction to Ordinary Differential Equations*. New York, NY: Springer New York, 2008. ISBN: 978-0-387-71275-8 978-0-387-71276-5. DOI: 10.1007/978-0-387-71276-5. URL: http://link.springer.com/10.1007/978-0-387-71276-5 (visited on 09/12/2022).

[154] Gene F Franklin, J David Powell, Michael L Workman, et al. *Digital control of dynamic systems*. Vol. 3. Addison-wesley Reading, MA, 1998, pp. 187–210.

[155] Alan V Oppenheim, John R Buck, and Ronald W Schafer. *Discrete-time signal processing. Vol. 2*. Upper Saddle River, NJ: Prentice Hall, 2001.

[156] BL HO and Rudolf E Kálmán. "Effective construction of linear state-variable models from input/output functions". In: *at-Automatisierungstechnik* 14.1-12 (1966), pp. 545–548.

[157] Rasmus Munk Larsen. "Lanczos Bidiagonalization With Partial Reorthogonalization". In: *DAIMI Report Series* 27.537 (Dec. 1998). ISSN: 2245-9316, 0105-8517. DOI: 10.7146/dpb.v27i537.7070. URL: https://tidsskrift.dk/daimipb/article/view/7070 (visited on 01/19/2022).

[158] Matteo Frigo and Steven G. Johnson. "The Design and Implementation of FFTW3". In: *Proceedings of the IEEE* 93.2 (2005). Special issue on "Program Generation, Optimization, and Platform Adaptation", pp. 216–231. DOI: 10.1109/JPROC.2004.840301.

[159] Paul Dierckx. *Curve and surface fitting with splines*. Oxford University Press, 1995. URL: https://github.com/kbarbary/Dierckx.jl.

[160] R. B. Lehoucq, D. C. Sorensen, and C. Yang. *ARPACK Users Guide: Solution of Large Scale Eigenvalue Problems by Implicitly Restarted Arnoldi Methods*. 1997.

[161] Dominique et al. *JuliaSmoothOptimizers/PROPACK.jl: v0.4.0*. Version v0.4.0. July 2021. DOI: 10.5281/zenodo.5090075. URL: https://doi.org/10.5281/zenodo.5090075.

[162] Walter Edwin Arnoldi. "The principle of minimized iterations in the solution of the matrix eigenvalue problem". In: *Quarterly of applied mathematics* 9.1 (1951), pp. 17–29.

[163] Jiahao Chen and Jarrett Revels. "Robust benchmarking in noisy environments". In: *arXiv e-prints*, arXiv:1608.04295 (Aug. 2016). arXiv: 1608.04295 `[cs.PF]`.

[164] United Nations Economic Commission for Europe. *Global Technical Regulation - ECE/TRANS/180/Add.15/Amend.6.* https://unece.org/transport/standards/transport/vehicle-regulations-wp29/global-technical-regulations-gtrs. 2021.

[165] U.S. Environmental Protection Agency. *2022 Tesla Model 3 Long Range AWD - Certification Summary Information Report.* Certification Summary Information Report. Oct. 2021.

[166] Albert Rodríguez, Gregory L. Plett, and M. Scott Trimboli. "Improved transfer functions modeling linearized lithium-ion battery-cell internal electrochemical variables". In: *Journal of Energy Storage* 20 (Dec. 2018), pp. 560–575. ISSN: 2352152X. DOI: 10.1016/j.est.2018.06.015. URL: https://linkinghub.elsevier.com/retrieve/pii/S2352152X18302032 (visited on 02/08/2020).

[167] Krishnakumar Gopalakrishnan, Teng Zhang, and Gregory J. Offer. "A Fast, Memory-Efficient Discrete-Time Realization Algorithm for Reduced-Order Li-Ion Battery Models". In: *Journal of Electrochemical Energy Conversion and Storage* 14.1 (Feb. 2017), p. 011001. ISSN: 2381-6872, 2381-6910. DOI: 10.1115/1.4035526. URL: https://asmedigitalcollection.asme.org/electrochemical/article/doi/10.1115/1.4035526/380537/A-Fast-MemoryEfficient-DiscreteTime-Realization (visited on 11/02/2021).

[168] Philipp Dechent et al. "Estimation of Li-Ion Degradation Test Sample Sizes Required to Understand Cell-to-Cell Variability**". In: *Batteries & Supercaps* 4.12 (Dec. 2021), pp. 1821–1829. ISSN: 2566-6223, 2566-6223. DOI: 10.1002/batt.202100148. URL: https://onlinelibrary.wiley.com/doi/10.1002/batt.202100148 (visited on 01/05/2022).

[169] Kirk D. Stetzel et al. "Electrochemical state and internal variables estimation using a reduced-order physics-based model of a lithium-ion cell and an extended Kalman filter". en. In: *Journal of Power Sources* 278 (Mar. 2015), pp. 490–505. ISSN: 03787753. DOI: 10.1016/j.jpowsour.2014.11.135. URL: https://linkinghub.elsevier.com/retrieve/pii/S0378775314020023 (visited on 03/18/2022).

[170] Weihan Li et al. "Parameter sensitivity analysis of electrochemical model-based battery management systems for lithium-ion batteries". In: *Applied Energy* 269 (July 2020), p. 115104. ISSN: 03062619. DOI: 10.1016/j.apenergy.2020.115104. URL: https://linkinghub.elsevier.com/retrieve/pii/S0306261920306164 (visited on 12/28/2022).

[171] Huiyong Chun et al. "Real-Time Parameter Estimation of an Electrochemical Lithium-Ion Battery Model Using a Long Short-Term Memory Network". In: *IEEE Access* 8 (2020), pp. 81789–81799. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2020.2991124. URL: https://ieeexplore.ieee.org/document/9081953/ (visited on 12/28/2022).

[172] Marc D. Berliner et al. "Nonlinear Identifiability Analysis of the Porous Electrode Theory Model of Lithium-Ion Batteries". In: *Journal of The Electrochemical Society* 168.9 (Sept. 1, 2021), p. 090546. ISSN: 0013-4651, 1945-7111. DOI: 10.1149/1945-7111/ac26b1. URL: https://iopscience.iop.org/article/10.1149/1945-7111/ac26b1 (visited on 12/28/2022).

[173] E. Miguel et al. “Review of computational parameter estimation methods for electrochemical models”. In: *Journal of Energy Storage* 44 (Dec. 2021), p. 103388. ISSN: 2352152X. DOI: 10.1016/j.est.2021.103388. URL: https://linkinghub.elsevier.com/retrieve/pii/S2352152X2101077X (visited on 12/28/2022).

[174] James L Lee. “Reduced-order physics-based model of lithium-ion batteries”. en. Thesis. University of Colorado: Colorado Springs, 2012. URL: https://mountainscholar.org/handle/10976/250.

[175] Nobuhiro Ogihara et al. “Theoretical and Experimental Analysis of Porous Electrodes for Lithium-Ion Batteries by Electrochemical Impedance Spectroscopy Using a Symmetric Cell”. In: *Journal of The Electrochemical Society* 159.7 (2012), A1034–A1039. ISSN: 0013-4651, 1945-7111. DOI: 10.1149/2.057207jes. URL: https://iopscience.iop.org/article/10.1149/2.057207jes (visited on 01/04/2023).

[176] Nobuhiro Ogihara et al. “Impedance Spectroscopy Characterization of Porous Electrodes under Different Electrode Thickness Using a Symmetric Cell for High-Performance Lithium-Ion Batteries”. In: *The Journal of Physical Chemistry C* 119.9 (Mar. 5, 2015), pp. 4612–4619. ISSN: 1932-7447, 1932-7455. DOI: 10.1021/jp512564f. URL: https://pubs.acs.org/doi/10.1021/jp512564f (visited on 07/23/2022).

[177] Robert Morasch et al. “Methods—Understanding Porous Electrode Impedance and the Implications for the Impedance Analysis of Li-Ion Battery Electrodes”. In: *Journal of The Electrochemical Society* 168.8 (Aug. 1, 2021), p. 080519. ISSN: 0013-4651, 1945-7111. DOI: 10.1149/1945-7111/ac1892. URL: https://iopscience.iop.org/article/10.1149/1945-7111/ac1892 (visited on 01/04/2023).

[178] Brandon R. Long et al. “Enabling High-Energy, High-Voltage Lithium-Ion Cells: Standardization of Coin-Cell Assembly, Electrochemical Testing, and Evaluation of Full Cells”. In: *Journal of The Electrochemical Society* 163.14 (2016), A2999–A3009. ISSN: 0013-4651, 1945-7111. DOI: 10.1149/2.0691614jes. URL: https://iopscience.iop.org/article/10.1149/2.0691614jes (visited on 09/20/2022).

[179] AVL. *VSM*. Version 4.3.17. Aug. 20, 2022. URL: https://www.avl.com/-/avl-vsm-4-.

[180] Formula SAE. *FSAE Michigan 2017 Results*. May 16, 2017. URL: https://www.sae.org/binaries/content/assets/cm/content/attend/2018/student-events/fsae/results/fsae_mi_2017_result.pdf (visited on 08/20/2022).

[181] *Battery World 2021*. SES. URL: https://ses.ai/us-battery-world-2021/ (visited on 09/24/2022).

[182] Ping Liu and Shirley Meng. “Battery500 Consortium: Development of High Capacity Cathodes and Robust Solid Electrolytes”. In: (Jan. 2022). DOI: 10.2172/1838452. URL: https://www.osti.gov/biblio/1838452.

[183] *SOLBAT - Next generation solid-state battery technology*. SOLBAT. URL: https://www.solbat-faraday.org (visited on 09/15/2022).

[184] *The SAFELiMOVE Project*. SAFELiMOVE. URL: https://safelimove.eu/ (visited on 09/15/2022).

[185] *The HIDDEN Project.* The Hidden Project. URL: https://hidden-project.eu/ (visited on 09/15/2022).

[186] H. G. Landau. “Heat conduction in a melting solid”. In: *Quarterly of Applied Mathematics* 8.1 (Jan. 1, 1950), pp. 81–94. ISSN: 0033-569X, 1552-4485. DOI: 10.1090/qam/33441. URL: https://www.ams.org/qam/1950-08-01/S0033-569X-1950-33441-3/ (visited on 08/23/2022).

[187] Sigal Gottlieb, Chi-Wang Shu, and Eitan Tadmor. “Strong Stability-Preserving High-Order Time Discretization Methods”. In: *SIAM Review* 43.1 (Jan. 2001), pp. 89–112. ISSN: 0036-1445, 1095-7200. DOI: 10.1137/S003614450036757X. URL: http://epubs.siam.org/doi/10.1137/S003614450036757X (visited on 08/26/2022).

[188] Chi-Wang Shu. “Total-Variation-Diminishing Time Discretizations”. In: *SIAM Journal on Scientific and Statistical Computing* 9.6 (Nov. 1988), pp. 1073–1084. ISSN: 0196-5204, 2168-3417. DOI: 10.1137/0909073. URL: http://epubs.siam.org/doi/10.1137/0909073 (visited on 09/22/2022).

[189] F. Scarselli et al. “The Graph Neural Network Model”. In: *IEEE Transactions on Neural Networks* 20.1 (Jan. 2009), pp. 61–80. ISSN: 1045-9227, 1941-0093. DOI: 10.1109/TNN.2008.2005605. URL: http://ieeexplore.ieee.org/document/4700287/ (visited on 09/06/2022).

[190] Adam Santoro et al. *A simple neural network module for relational reasoning.* June 5, 2017. arXiv: 1706.01427[cs]. URL: http://arxiv.org/abs/1706.01427 (visited on 09/23/2022).

[191] David Raposo et al. *Discovering objects and their relations from entangled scene representations.* Feb. 16, 2017. arXiv: 1702.05068[cs]. URL: http://arxiv.org/abs/1702.05068 (visited on 09/23/2022).

[192] David Duvenaud et al. *Convolutional Networks on Graphs for Learning Molecular Fingerprints.* Nov. 3, 2015. arXiv: 1509.09292[cs,stat]. URL: http://arxiv.org/abs/1509.09292 (visited on 09/23/2022).

[193] Justin Gilmer et al. *Neural Message Passing for Quantum Chemistry.* June 12, 2017. arXiv: 1704.01212[cs]. URL: http://arxiv.org/abs/1704.01212 (visited on 09/23/2022).

[194] Rhys E. A. Goodall and Alpha A. Lee. “Predicting materials properties without crystal structure: deep representation learning from stoichiometry”. In: *Nature Communications* 11.1 (Dec. 2020), p. 6280. ISSN: 2041-1723. DOI: 10.1038/s41467-020-19964-7. URL: http://www.nature.com/articles/s41467-020-19964-7 (visited on 09/23/2022).

[195] Yaguang Li et al. *Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting.* Feb. 22, 2018. arXiv: 1707.01926[cs,stat]. URL: http://arxiv.org/abs/1707.01926 (visited on 09/23/2022).

[196] Zhiyong Cui et al. *Traffic Graph Convolutional Recurrent Neural Network: A Deep Learning Framework for Network-Scale Traffic Learning and Forecasting.* Nov. 4, 2019. arXiv: 1802.07007[cs,stat]. URL: http://arxiv.org/abs/1802.07007 (visited on 09/23/2022).

[197] Xinlei Chen et al. *Iterative Visual Reasoning Beyond Convolutions.* Mar. 29, 2018. arXiv: 1803.11189[cs]. URL: http://arxiv.org/abs/1803.11189 (visited on 09/23/2022).

[198] Xiaolong Wang et al. *Non-local Neural Networks.* Apr. 13, 2018. arXiv: 1711.07971[cs]. URL: http://arxiv.org/abs/1711.07971 (visited on 09/23/2022).

[199] Michael B. Chang et al. *A Compositional Object-Based Approach to Learning Physical Dynamics.* Mar. 4, 2017. arXiv: 1612.00341[cs]. URL: http://arxiv.org/abs/1612.00341 (visited on 09/23/2022).

[200] Nicholas Watters et al. "Visual Interaction Networks: Learning a Physics Simulator from Video". In: *Advances in Neural Information Processing Systems.* Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper/2017/file/8cbd005a556ccd4211ce43f309bc0eac-Paper.pdf.

[201] Sjoerd van Steenkiste et al. *Relational Neural Expectation Maximization: Unsupervised Discovery of Objects and their Interactions.* Feb. 28, 2018. arXiv: 1802.10353[cs]. URL: http://arxiv.org/abs/1802.10353 (visited on 09/23/2022).

[202] Adam Paszke et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems 32.* Ed. by H. Wallach et al. Curran Associates, Inc., 2019, pp. 8024–8035. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[203] Peter W. Battaglia et al. *Relational inductive biases, deep learning, and graph networks.* Oct. 17, 2018. arXiv: 1806.01261[cs,stat]. URL: http://arxiv.org/abs/1806.01261 (visited on 09/06/2022).

[204] Alvaro Sanchez-Gonzalez et al. *Learning to Simulate Complex Physics with Graph Networks.* Sept. 14, 2020. arXiv: 2002.09405[physics,stat]. URL: http://arxiv.org/abs/2002.09405 (visited on 09/06/2022).

[205] Prajit Ramachandran, Barret Zoph, and Quoc V. Le. *Searching for Activation Functions.* Oct. 27, 2017. arXiv: 1710.05941[cs]. URL: http://arxiv.org/abs/1710.05941 (visited on 09/13/2022).

[206] Richard H. R. Hahnloser et al. "Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit". In: *Nature* 405.6789 (June 22, 2000), pp. 947–951. ISSN: 0028-0836, 1476-4687. DOI: 10.1038/35016072. URL: http://www.nature.com/articles/35016072 (visited on 09/13/2022).

[207] Kevin Jarrett et al. "What is the best multi-stage architecture for object recognition?" In: *2009 IEEE 12th International Conference on Computer Vision.* 2009 IEEE 12th International Conference on Computer Vision (ICCV). Kyoto: IEEE, Sept. 2009, pp. 2146–2153. ISBN: 978-1-4244-4420-5. DOI: 10.1109/ICCV.2009.5459469. URL: http://ieeexplore.ieee.org/document/5459469/ (visited on 09/13/2022).

[208] Vinod Nair and Geoffrey E. Hinton. "Rectified Linear Units Improve Restricted Boltzmann Machines". In: *ICML.* 2010, pp. 807–814. URL: https://icml.cc/Conferences/2010/papers/432.pdf.

[209] Stefan Elfwing, Eiji Uchibe, and Kenji Doya. *Sigmoid-Weighted Linear Units for Neural Network Function Approximation in Reinforcement Learning.* Nov. 1, 2017. arXiv: 1702.03118[cs]. URL: http://arxiv.org/abs/1702.03118 (visited on 09/13/2022).

[210] Magnus Urquhart, Emil Ljungskog, and Simone Sebben. “Surrogate-based optimisation using adaptively scaled radial basis functions”. In: *Applied Soft Computing* 88 (Mar. 2020), p. 106050. ISSN: 15684946. DOI: 10.1016/j.asoc.2019.106050. URL: https://linkinghub.elsevier.com/retrieve/pii/S1568494619308324 (visited on 09/24/2022).

[211] Shashi Gowda et al. “High-Performance Symbolic-Numerics via Multiple Dispatch”. In: *ACM Commun. Comput. Algebra* 55.3 (Jan. 2022), pp. 92–96. ISSN: 1932-2240. DOI: 10.1145/3511528.3511535. URL: https://doi.org/10.1145/3511528.3511535.