

Reem Haile Kidane  
Jørgen Myklebust  
Bjarte Steinsund

# LiB Ageing Mechanisms and Energy Storage Replacement/Expansion in Marine Vessels

Modelling of Battery Ageing with PyBaMM

Bachelor's thesis in Renewable Energy

Supervisor: Jacob Joseph Lamb

Co-supervisor: Trude Birgitte Byre & Håvard Lefdal Hove

May 2022



Reem Haile Kidane  
Jørgen Myklebust  
Bjarte Steinsund

# **LiB Ageing Mechanisms and Energy Storage Replacement/Expansion in Marine Vessels**

Modelling of Battery Ageing with PyBaMM

Bachelor's thesis in Renewable Energy  
Supervisor: Jacob Joseph Lamb  
Co-supervisor: Trude Birgitte Byre & Håvard Lefdal Hove  
May 2022

Norwegian University of Science and Technology  
Faculty of Engineering  
Department of Energy and Process Engineering

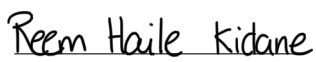







# Bachelor's Thesis

<b>Project title:</b> LiB Ageing Mechanisms and Energy Storage Replacement/Expansion in Marine Vessels  <b>Oppgåva sin tittel:</b> Aldringsmekansimar i LiB og utskifting/utviding av energilagringssystem i marine fartøy	<b>Given date:</b> 10.01.2022 <b>Submission date:</b> 30.05.2022 <b>Number of report/appendix pages:</b> 88/14
<b>Project participants:</b> Reem Haile Kidane Jørgen Myklebust Bjarte Steinsund	<b>Internal supervisor:</b> Jacob Joseph Lamb E-mail: <a href="mailto:jacob.j.lamb@ntnu.no">jacob.j.lamb@ntnu.no</a> Tel.: 902 38 329
<b>Program of study:</b> Bachelor in Engineering, Renewable Energy	<b>Project number:</b> 22BIFOREN-012
<b>Client:</b> Siemens Energy AS	<b>External supervisors:</b> Trude Birgitte Byre E-mail: <a href="mailto:trude-birgitte.byre@siemens-energy.com">trude-birgitte.byre@siemens-energy.com</a> Tel.: 948 58 400 Håvard Lefdal Hove E-mail: <a href="mailto:haavardlefdal.hove@siemens-energy.com">haavardlefdal.hove@siemens-energy.com</a> Tel.: 417 68 906

Free for publication: Available after agreement with client: Free to be published after: Trondheim, 30<sup>th</sup> of May, 2022

Reem H. Kidane



Jørgen Myklebust



Bjarte Steinsund



## Preface

In the Renewable Energy Study Programme at the Norwegian University of Science and Technology, all students must complete a bachelor's thesis. Its completion fulfils the requirements for the degree of Bachelor in Engineering at NTNU. This report was written during the spring semester of 2022, and is the basis for the evaluation of the subject FENT2900. As part of FENT2900 a preparation project, which accelerated the progression of this report, was conducted in the beginning of this semester. Further oral presentations, as well as a popular science poster, were constructed as part of the 20 ECTS FENT2900 subject. During this project, the group members have been given the opportunity to take a deep dive into a most interesting subject of choice.

The conduction of this report is based on the knowledge and skills which have been acquired over the three last years of study. The conduction of this report has provided the group members with new knowledge, as well as valuable insights on cooperation, communication, planning and project management. The Renewable Energy programme in Trondheim gives a broad, multidisciplinary education. This has provided the group members with a solid starting foundation in multiple fields of study, as well as a set of tools on how to acquire cross-domain knowledge. These tools have definitely helped the group overcome the steep learning curve, which had to be traversed, during the battery simulation- and modelling deep dive.

Siemens Energy provided the project problem, which was immediately an easy sell for all the group members. The group members all originate from parts of the country where ferries, and lately electric ferries, carry people over the fjords. In addition to this, the group members are all very intrigued by the major investments which have been made in the Norwegian battery industry as of late. Siemens Energy provided a real problem that needs a solution. It has both been very challenging, rewarding and motivating to work with an actual, real life problem throughout this project. Especially since its solution could have real life implications on how batteries, a technology which defines society, are used in portable applications. This report explores the ageing of lithium-ion batteries, with a special emphasis on ferry conditions, and how these can be optimized within practical constraints, to prolong battery life time. This study was conducted through simulations with the battery modelling package PyBaMM in Python.

The group would like to direct a most sincere thank you to our supervisor and study programme leader, Associate Professor Jacob Joseph Lamb, for his excellent guidance, thoughtful insights, and enthusiasm. Next, the group would like to express gratitude to Trude Brigitte Byre and Håvard Lefdal Hove at Siemens Energy for providing the project problem, for their guidance and time, and for the exciting guided battery factory tour, which we were all welcomed to. Next the group would like to recognize the PyBaMM community at large, which have been both very welcoming and helpful. Two members of the community; co-creator of PyBaMM, Postdoctoral Research Fellow Valentin Sulzer from Carnegie Mellon University, and PhD student Yannick Kuhn from the German Aerospace Center and the Helmholtz Institute Ulm, deserves extra recognition for their willingness to answer questions, and provide explanations on challenging concepts. Lastly the group would like to thank PhD candidate Silje Nornes Bryntesen for demonstrating the scanning electron microscope, and for providing images produced by this characterization technique.



## Abstract

In order to counteract climate change, develop technology, and reduce costs, a number of electric ferries have been put into service along the Norwegian coastline. Onboard these vessels, lithium-ion batteries are used as the means of storing energy. In many ferries, these batteries have been in use for years. Hence, they have had to withstand both time and cycling. Consequently, their capacity has gradually faded. For ferry companies, this presents a challenge. In light of this challenge, the principal objective of this thesis was to improve the lifetime of lithium-ion batteries on board ferries. A good approach to this can both reduce expenses and preserve the environment. If the correct strategy is used, batteries can be used for a prolonged period of time. This reduces the use of resources on production and recycling. A general solution can also help enable an increase in marine electrification, which can have a significant positive impact on the environment.

Therefore, the group used PyBaMM, an open-source tool that offers unique flexibility to model lithium-ion batteries and their ageing. In PyBaMM, the battery models are separated from the parameters, and physical models that describe their own unique part of the battery physics can be selected as required. Users can also control the discretisation and solver. A study on the battery models which are implemented in PyBaMM was conducted. According to this study, the fast battery models gave results comparable to those of the more complex Doyle-Fuller-Newman model. To approximate real and future ferry operations, an NMC-811 parameter set was selected. The constructed battery model used these parameters and all the ageing models, which are implemented in PyBaMM. With this approach, capacity loss from SEI, lithium plating, and loss of active material could be accounted for. A mesh refinement study and solver tolerance refinement study were conducted to ensure that the most appropriate simulation settings were used.

The process of building a custom battery model was time-consuming. As PyBaMM is a relatively young software, it can be difficult to find solutions to some problems. Compared to the older software, there are fewer solutions available online. Additionally, to the best of the authors' knowledge, this is the first project report that uses PyBaMM at NTNU. Therefore, the authors constructed methods, discussions, results, and appendices, in a manner such that readers are able to be familiarised with PyBaMM effectively. Several explanations and solutions to issues are also provided. This was intended to facilitate future scientific work with PyBaMM.

The initial results were not aligned with the literature, it then became evident that troubleshooting was required. This process resolved one of the two known issues, as it was affirmed that the `interstitial-diffusion limited SEI` submodel should be used. Additionally, a number of possible problems were ruled out, and the authors have outlined some possible next steps toward identifying the solution to the last problem.

The most significant result in this thesis is the proposal of an optimal SoC value algorithm. Compared to static SoC cycling, it has been shown to decrease capacity loss. More cycles are needed to confirm these results. The SoC cycling method also needs to be validated. The PyBaMM developers have created a powerful tool that could provide significant scientific value. In spite of the program being time consuming to become familiar with, it proved to be the most suitable, as it provided a deeper appreciation of both batteries, ageing and modelling.

## Samandrag

For å motverke klimaendringane, utvikle teknologi og senke kostnader, så har ei rekkje elektriske ferjer vore sett i drift langs den norske kysten. Litium-ion batteri er teknologien som vert nytta til å lagre energi om bord i desse fartøya. Fleire av desse batteria har vore nytta i mange år, og har dermed blitt belasta både med omsyn på tal sykklar og tid, noko som har ført til at kapasiteten til batteria har blitt redusert. Dette er difor ei dagsaktuell utfordring for ferjeselskapa.

Hovudmålet med denne oppgåva var i lys av denne utfordringa, å undersøkje korleis levetida til litium-ion batteri som er nytta om bord i ferjer kan aukast. Gode føringar på dette vil både kunne senke utgifter og spare miljøet. Med riktig strategi kan batteri nyttast lengre, og dermed redusere ressursbruken på både resirkulering og produksjon. Generelle løysingar kan også vere med å påverke mot ei aukande marin elektrifisering, noko som kan skape store miljøgevinstar.

Gruppa modellerte difor litium-ion batteri, og korleis desse aldri ved hjelp av open kjeldekode programvara PyBaMM, som tilbyr ein unik fleksibilitet. Batteri modellen er skild frå parametrane, og fysiske modellar som skildrar kvar sin del av batterifysikken kan veljast etter behov. Vidare har brukarar kontroll over diskretisering og løysingsprogram. Det vart gjennomført ein studie på batteri modellane, som er implementert i PyBaMM. Denne studien viste at dei raske batteri modellane gav samanliknbare resultat med den tyngre Doyle-Fuller-Newman modellen. For å tilnærme reell og framtidig ferjedrift vart eit NMC-811 parameter sett valt. Den konstruert batteri modellen nytta desse parametrane saman med modellane for aldring som er implementert i PyBaMM. På denne måten blei kapasitetstapet som følgje av SEI, «lithium plating» og tap av aktivt materiale teke høgde for. Studiar på både parametrar, grid oppløysing og toleranse blei også gjennomført slik at dei mest føremålstenlege innstillingane blei nytta i simuleringane.

Byggjinga av batteri modellen var ein tidkrevjande prosess. PyBaMM er ei relativt ung programvare, og det kan difor vere utfordrande å finne svar på problem. Grunnen til dette er at eit redusert tal løysingar eksisterer på nettet, samanlikna med eldre programvare. Så vidt forfatarane er kjend, så er også dette den fyrste prosjekt oppgåva som nyttar PyBaMM ved NTNU. Forfatarane har difor konstruert metode, diskusjon, resultat og vedlegg slik at ein skal kunne setje seg inn i PyBaMM på ein effektiv måte. Ei rekkje forklaringar, og løysingar på problem er presentert. Tanken er at dette kan leggje til rette for god vitskap med PyBaMM.

Dei fyrste resultatata som blei produsert var ikkje foreinleg med litteraturen, det var dermed tydeleg at ei feilsøking måtte gjennomførast. Dette løyste éin av to kjende problem, då det vart stadfesta at SEI modellen *interstitial-diffusion limited* bør nyttast. I tillegg til dette vart ei rekkje moglege problem utelukka, forfatarane har staka ut nokre steg vidare mot å avdekke løysinga på det siste problemet under vidare arbeid.

Det mest interessante resultatet i denne oppgåve er truleg forslaget til ei optimal SoC verdi algoritme. Ved bruk av denne algoritmen har det blitt vist at kapasitetstapet kan bli redusert, samanlikna med ei statisk SoC sykling. Resultatet må bli vidare bekrefta ved å gjennomføre fleire sykklar, metoden for SoC sykling må også vidare validerast. PyBaMM utviklarane har skapt eit kraftig verktøy, som truleg vil føre til stor vitskapleg verdi. Programmet var tidkrevjande å setje seg inn i, men definitivt det riktige valet for denne oppgåva, då det gav ei djupare forståing for både batteri, aldring og modellering.



# Contents

<b>Preface</b>	<i>i</i>
<b>Abstract</b>	<i>iii</i>
<b>Samandrag</b>	<i>iv</i>
<b>Figures</b>	<i>ix</i>
<b>Tables</b>	<i>ix</i>
<b>Listings</b>	<i>x</i>
<b>Abbreviations</b>	<i>xi</i>
<b>1 Introduction</b>	<b>1</b>
1.1 Background and Motivation . . . . .	1
1.2 Objective . . . . .	2
1.3 Scope of the Report . . . . .	3
1.4 Outline of the Report . . . . .	3
1.5 Software . . . . .	4
<b>2 Theory</b>	<b>5</b>
2.1 Ferry Structure . . . . .	5
2.1.1 Main Components . . . . .	5
2.1.2 Cooling System . . . . .	7
2.1.3 Grid Infrastructure . . . . .	8
2.1.4 Electrical Prices versus Fuel Prices in Norway . . . . .	8
2.1.5 Alternative Fuels and Retrofitting . . . . .	9
2.1.6 Load Profile and Sea Conditions . . . . .	10
2.2 Energy Storage . . . . .	11
2.2.1 Motivation for the Implementation of Energy Storage . . . . .	11
2.2.2 Methods of Classification and Properties of Energy Storage Solutions . . . . .	12
2.2.3 Energy Storage Trends . . . . .	15
2.3 Batteries . . . . .	17
2.3.1 Primary and Secondary Batteries . . . . .	18
2.3.2 Battery Geometries . . . . .	18
2.4 Lithium-ion Batteries . . . . .	19
2.4.1 Structure and Working Principle . . . . .	19
2.4.2 Chemistries . . . . .	20
2.4.3 Electrode Structure . . . . .	21
2.4.4 Cathode . . . . .	22
2.4.5 Anode . . . . .	23
2.4.6 Electrolyte . . . . .	23
2.4.7 Separator . . . . .	24
2.4.8 Current Collectors and Tabs . . . . .	24



2.5	Battery Figures of Merit . . . . .	24
2.5.1	Capacity . . . . .	24
2.5.2	C-rate . . . . .	25
2.5.3	State of Charge, SoC Window and Depth of Discharge . . . . .	25
2.5.4	State of Health and Internal Resistance . . . . .	26
2.6	Degradation Factors and Mechanisms . . . . .	27
2.6.1	Solid Electrolyte Interphase . . . . .	28
2.6.2	Graphite Exfoliation . . . . .	29
2.6.3	Particle Cracking . . . . .	29
2.6.4	Lithium Plating . . . . .	30
2.6.5	Loss of Electrical Contact . . . . .	31
2.6.6	Other Degradation Mechanisms . . . . .	31
2.7	Degradation Modes . . . . .	31
2.8	Degradation in Battery Packs . . . . .	32
<b>3</b>	<b>Methods</b>	<b>35</b>
3.1	Simulating Battery Cells and Battery Packs . . . . .	35
3.2	Battery Simulation Software . . . . .	35
3.3	PyBaMM . . . . .	35
3.3.1	Using Pre-Built Battery Models . . . . .	36
3.3.2	Choosing Submodel Options . . . . .	37
3.3.3	Choosing Parameter Values . . . . .	38
3.3.4	Choosing Standard Cell Geometries . . . . .	39
3.3.5	Simulation of Experiments and Logging Level . . . . .	39
3.3.6	Choosing Solver and Solver Options . . . . .	40
3.3.7	Example of a Simple Experiment Simulation and Citations . . . . .	41
3.3.8	Choosing Mesh Resolution . . . . .	41
3.3.9	Choosing Submesh Types . . . . .	42
3.4	Issues and Troubleshooting in PyBaMM . . . . .	43
3.4.1	Memory Management . . . . .	44
3.4.2	LineSearch Algorithm . . . . .	45
3.4.3	Minimum and Maximum Voltage . . . . .	45
3.4.4	IDACalcIC Unable to Recover . . . . .	45
3.5	Liionpack . . . . .	46
<b>4</b>	<b>Results and Discussion</b>	<b>47</b>
4.1	Base Lithium-ion Model Study . . . . .	47
4.2	Selected Submodels . . . . .	51
4.2.1	SEI Submodels . . . . .	52
4.2.2	Lithium Plating Submodels . . . . .	52
4.2.3	Particle Mechanics Submodel . . . . .	52
4.2.4	Loss of Active Material Submodel . . . . .	53
4.2.5	Thermal Submodel . . . . .	54
4.3	Choosing Parameter Values . . . . .	54
4.3.1	Base Parameter Set . . . . .	54

4.3.2	SEI Parameters . . . . .	55
4.3.3	Lithium Plating Parameters . . . . .	56
4.3.4	Particle Mechanics and Loss of Active Material Parameters . . . . .	56
4.4	Mesh Refinement Study . . . . .	57
4.5	Solver Tolerance Refinement Study . . . . .	58
4.6	Initial Results . . . . .	60
4.7	Troubleshooting Initial Results . . . . .	61
4.8	Final Results . . . . .	70
4.9	Limitations . . . . .	74
<b>5</b>	<b>Conclusion</b>	<b>76</b>
<b>6</b>	<b>Further Work</b>	<b>77</b>
	<b>Bibliography</b>	<b>78</b>
	<b>Appendix A</b>	<b>A-1</b>
A.1	Utility and Utility Functions . . . . .	A-1
	<b>Appendix B</b>	<b>B-1</b>
B.1	Customizing and Conducting the SoC Window Simulation . . . . .	B-1
	<b>Appendix C</b>	<b>C-1</b>
C.1	Proposal for an Optimal SoC Window Pathfinder Algorithm . . . . .	C-1

## Figures

1.1	Analysis of expected ferry demand . . . . .	2
2.1	Principal sketch of a hybrid ferry . . . . .	5
2.2	An illustration on how the cooling system is connected . . . . .	7
2.3	Electricity prices over the last few years, data presented by Nord Pool . . . . .	9
2.4	Chart of different fuel prices . . . . .	9
2.5	An estimated loadprofile of a ferry crossing a fjord . . . . .	10
2.6	Energy storage systems classified by form of energy . . . . .	12
2.7	Growth of installed PV capacity from 2010-2020 . . . . .	15
2.8	Share of global energy storage installed capacity, by technology . . . . .	16
2.9	Energy storage capacity price history and forecast. . . . .	17
2.10	Conventional cell geometries . . . . .	18
2.11	LiB cell structure and movement of charges . . . . .	19
2.12	Composition of an electrode . . . . .	21
2.13	SEM image of an aged cathode . . . . .	21
2.14	An illustration over the most common ageing mechanisms which occurs in LiBs . . . . .	27
2.15	Illustration of the desired electrochemical reaction of lithium intercalation . . . . .	29
2.16	Illustration of an anode in which particle cracking leads to SEI growth . . . . .	30
2.17	An illustration of lithium plating at its final degradation step . . . . .	30
2.18	Root causes that provokes several degradation mechanisms and modes associated with the two losses that occurs on LiB . . . . .	32
2.19	Scaling from the material level to the pack level . . . . .	33
3.1	Steps to solve a battery model in PyBaMM . . . . .	36
3.2	Typical Python traceback displayed due to a NameError. . . . .	43
3.3	Image from a PyBaMM training video workshop series . . . . .	44
4.1	Capacity and LLI over time by base model . . . . .	48
4.2	Average SEI- and average lithium plating growth over time by base model . . . . .	49
4.3	Cell power and ECM resistance by base model . . . . .	50
4.4	Selected results from the mesh refinement study . . . . .	58
4.5	Selected results from the solver tolerance refinement study . . . . .	59
4.6	Capacity fade as a function of C-rates. . . . .	60
4.7	Comparison of MATLAB and Python figures . . . . .	61
4.8	Basic chen2020_plating base parameter set simulations . . . . .	62
4.9	Results from the SoC cycling submodel review . . . . .	63
4.10	Flowchart of the different lithium plating combinations in PyBaMM . . . . .	64
4.11	Results from the lithium plating review . . . . .	64
4.12	Flowchart of the voltage and the SoC based cycling schemes. . . . .	65
4.13	Results from the troubleshooting of the SoC cycling scheme . . . . .	66
4.14	Results from the submodel review with voltage cycling method . . . . .	67
4.15	Comparison of simulation data plotted against cycle number and time . . . . .	68
4.16	Comparison of the interstitial-diffusion limited SEI submodel . . . . .	69
4.17	Simulation results from differently cooled cells . . . . .	70
4.18	Results from the initial version of the proposed optimal SoC pathfinder algorithm . . . . .	72
4.19	Results from the final version of the proposed optimal SoC pathfinder algorithm . . . . .	73

## Tables

2.1	Technical features of different energy storage technologies . . . . .	13
2.2	Technical features of different energy storage technologies (continues) . . . . .	14
2.3	Technical features of different energy storage technologies (continues) . . . . .	14
2.4	Rated power capacity in GW and number of energy storage projects by project stage around the world . . . . .	17
2.5	Technical features of different LiB chemestries . . . . .	23
4.1	Base battery model solve time comparison . . . . .	47
4.2	The effect of adding custom parameters- and submodels on solve time . . . . .	50
4.3	Summary of NMC base parameter sets . . . . .	55
4.4	Summary of SEI parameter sets . . . . .	55
4.5	Summary of lithium plating parameter sets . . . . .	56
4.6	Summary of particle mechanics and LAM parameters . . . . .	57
4.7	Mesh refinement study results . . . . .	57
4.8	Solver tolerance refinement study results . . . . .	59

## Listings

3.1	Initializing a pre-built battery model . . . . .	37
3.2	Setting submodel options . . . . .	37
3.3	Defining parameter values . . . . .	38
3.4	Defining cell geometry . . . . .	39
3.5	Setting logging level and simulating experiments . . . . .	40
3.6	Setting solver and solver options, and finding simulation time . . . . .	40
3.7	Solving a basic experiment simulation . . . . .	41
3.8	Defining a finer mesh . . . . .	42
3.9	Defining submesh types . . . . .	42
3.10	Using an optional keyword argument in solve to reduce memory usage . . . . .	44
3.11	Changing solver tolerances . . . . .	45
3.12	Changing the lower- and upper cut-off voltage . . . . .	45
4.1	The customized submodel options which were used in the simulations in this report	51
A.1	Required imports . . . . .	A-1
A.2	Processing and saving simulation data to a CSV file . . . . .	A-1
A.3	Memory utilization tracking function . . . . .	A-3
A.4	Memory clearing function . . . . .	A-4
B.1	Customizing the simulation . . . . .	B-1
B.2	SoC window simulation function . . . . .	B-5
C.1	Proposed pathfinder algorithm . . . . .	C-1

## Abbreviations

Abbreviation	Description
BoL	Beginning of life
CAES	Compressed air energy storage
CasADi	Computer-algebra system Algorithmic Differentiation
C-rate	Normalised current rate
DFN	Doyle-Fuller Newman
DoD	Depth of discharge
ECM	Equivalent circuit model
EoL	End of life
FES	Flywheel energy storage
GiB	Gibibyte, $2^{30}$ bytes
HES	Hydrogen energy storage
IEA	International Energy Agency
LAM	Loss of active material
LCO	Lithium cobalt oxide battery
LiB	Lithium-ion battery
LFP	Lithium iron phosphate battery
LLI	Loss of lithium inventory
LMO	Lithium manganese oxide battery
NaS	Sodium-sulfur battery
NaNiCl <sub>2</sub>	Sodium-nickel chloride battery
NCA	Lithium nickel cobalt aluminium oxide battery
NiCd	Nickel-cadmium battery
NMC	Lithium nickel manganese cobalt oxide battery
NT	Newman-Tobias
PbA	Lead-acid battery
PHES	Pumped hydroelectric energy storage
PyBaMM	Python Battery Mathematical Modelling

---

<b>Abbreviation</b>	<b>Description</b>
RAM	Random access memory
SCES	Super capacitor energy storage
SEI	Solid electrolyte interphase
SMES	Superconducting magnetic energy storage
SoC	State of charge
SoH	State of health
SPM	Single Particle Model
SPMe	Single Particle Model with electrolyte
VRFB	Vanadium redox flow battery

---



# 1 Introduction

To uphold the Paris Agreement of keeping the global average temperature rise below 2°C compared to preindustrial levels, action must be taken in society at large [1]. The adaptation of renewable energy sources has become a measure to counteract climate change. Advances have also been made in the transition towards an electric transport sector, and energy storage is poised to play a key role in enabling these developments [2, 3].

Battery manufacturing is emission intensive, therefore, emissions can be reduced by improving manufacturing or increasing battery lifespan [4]. Hence, batteries which are deployed in ferries, and how their lifespan can be extended is investigated in this thesis.

## 1.1 Background and Motivation

At the UN's 26th climate conference, also known as COP26, Norwegian Premier Jonas Gahr Stre stated that Norway is positioned to take a leading role in the development of sea-based solutions. Solutions such as green shipping. "The transition will make a difference in Norway, but our ambitions are larger: We will develop and export new technology that can be used outside of our borders." MF Ampere, the world's first all-electric battery-powered ferry, has served the Lavik - Oppedal ferry crossing over the Sognefjord since 2015, and by 2022 more than a third of Norway's car ferries will use electric propulsion systems, according to the Government's action plan for green shipping from 2019 [5].

Several years have passed since the first battery electric ferries were put into service. The lithium-ion batteries (LiBs) in these vessels have aged, and as a result of this, lost capacity. Because of this, many ferries are in need of replacing their energy storage solutions in the short and medium term. In addition to this challenge, there exist vessels with energy storage systems which need to be expanded in terms of capacity to accommodate new operational requirements. Capacity requirements brought on by ferries having to change fjords, either due to an end of their contract period, or due to demand by the shipowners. An alternative to replacing the entire energy storage system, both due to ageing or new system requirements, is to install new battery modules or packs into a preexisting system. This can both be environmentally and economically friendly. Therefore, the industry seeks to investigate possible solutions on how this can be accomplished, thus providing the starting point for this bachelor thesis.

The group members preferred to address a current, real-world issue. Such as the ageing of battery energy storage systems on board ferries. Even though this report addresses ferries, the group still strives towards finding general solutions which also provide the possibility of being applied in a wide range of energy storage systems, and in marine vessels at large. A solution could contribute towards better utilisation of batteries, such that emissions are cut, and need of investments are eased.



As of August of 2021, 61 new charging facilities distributed among 27 ferry connections were planned to be completed within 2024. The electrification of ferries are a central part of the Norwegian Government's ambition to cut emissions by domestic shipping in half by 2030. To reach climate goals, solutions and technologies must be developed and continually advanced so that more categories of vessels can adapt similar climate change mitigation strategies [6].

Figure 1.1 shows the Norwegian Public Roads Administration's ferry demand forecast. Sizeable demand has been forecast for both the short, medium and long term. The long term demand is lesser than the medium term, as long term bridge and tunnel projects are taken into account. Because of these long term construction projects, ferries will be relieved of duty to an estimated 16 million passenger car units. The passenger car unit is a normalisation unit used to describe how many passenger cars a given vehicle equals, such that traffic volume can be accurately described with a single unit [7].

During the last decade there has been an exciting development of technologies that has advanced the electrification of marine vessels. Car ferries, commonly Roll-on/roll-off ferries, is the marine vessel of focus in this thesis. Thus, this thesis addresses energy storage in ferries that are designed primarily to transport vehicles with passengers over a relatively short stretch of water.

Larger vessels have a higher energy demand and often sail much longer routes than RoRo ferries. Hence, the implementation of energy storage solutions is less viable. Research on climate mitigation strategies for these types of vessels has, for instance, focused on alternative fuel sources [8].

When designing energy storage systems, the amount of energy storage which is required is one of the main figures to dimension. Thus, the distance between ferry harbours is an important figure to consider. In Norway these distances range from 200 metres in Drammen to upwards of 90 kilometres in Lofoten [9]. The number of round trips a ferry makes on a given day is defined in its schedule, which should be adapted to the traffic in the area. Due to differences in routes, operating conditions and traffic, some ferries could benefit from electrification, while others do not.

## 1.2 Objective

The objective of this thesis is to assemble and simulate an accurate battery model by combining numerous physical models, choosing appropriate battery parameters, and making sure that suitable discretization, solver and solver options are used. The goal is to combine

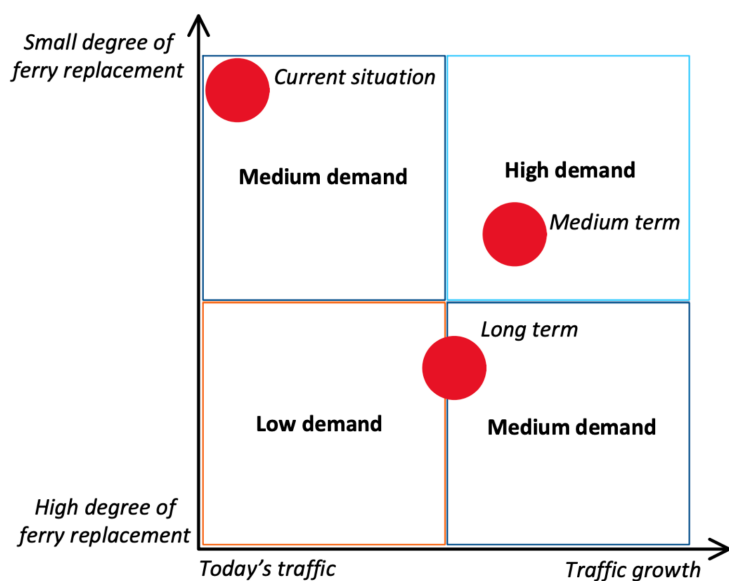


Figure 1.1: Analysis of expected ferry demand [7].

physical submodels, such that the most amount of ageing mechanisms are accounted for in the same battery simulation. Thereafter the ageing behaviour of LiBs can be analyzed through simulations. Further, simulations are applied in the effort of investigating and providing answers to the following questions.

- How to assemble and simulate an accurate LiB model?
- How can the lifespan of LiBs used in ferries be extended?
- How do different operational conditions provoke degradation in LiBs?
- How can pre-existing energy storage systems be expanded?
- What are the challenges of combining aged and unaged LiBs?
- Are any limitations imposed on ferries which replace LiBs?

### 1.3 Scope of the Report

This report exclusively explores ferries as defined in Section 1.1. The report also focuses on LiBs as they are the state-of-the-art energy storage technology, with regards to portable energy storage solutions [10, 11]. The report is mainly technical in nature, as the three group participants are all engineering students. Note that the economics of a technical solution always has to be considered, this is however left out of this report, as there is a vast amount of technical subject matter to be discussed. When constructing a battery model a range of choices has to be made. The determination of simulation settings should be conducted through rigorous studies, research, and communication with supervisors, even so some well argued assumptions has to be made in order to secure the progression of the project work. This leads into the time aspect, wherein the authors unfortunately only had one semester to conduct the work presented in this report.

The group wished to compare and validate simulation results with a laboratory experiment, due to time constraints this were however deemed unpractical, as to little time were made available to obtain comparable data. In the preliminary project it were also asserted that it was unattainable to obtain data from industry, given that the project remained free for publication. The group participants nonetheless decided that the report should remain free for publication, with the thought that the information presented in this report could prove useful to someone in the future which might want to build on this, or similar work. This decision were further supported by, as far as the authors are aware, this report being the first thesis at NTNU which utilizes the simulation software PyBaMM (Python Battery Mathematical Modelling).

### 1.4 Outline of the Report

Chapter 2 provides a theoretical background on ferries, energy storage, batteries, and in particular LiBs and their ageing mechanisms. This theoretical background is a prerequisite to model LiBs, located in specific energy storage systems, such as those found in ferries, accurately. Chapter 3 gives an introduction to how the simulation software is utilised in this report. It was constructed by collecting information from multiple sources, such as the official website, the documentation, and GitHub repository, into a single compressed unit. Chapter 3 also provides information on how to troubleshoot, further common issues are compiled into four sections for ease of access to those facing similar issues, they are also presented together with information on how they can be circumvented. Chapter 4 discusses the construction of the battery model, and the results which were produced by the simulations which were conducted with this model.

The next to last chapter, Chapter 5 concludes the project work. Whilst Chapter 6 provides an extensive list of ideas and suggestions for further work.

## 1.5 Software

The simulation software which were utilized to build a custom battery model, and simulate LiB degradation at the unit cell level in this report, is called PyBaMM. PyBaMM is an open-source tool, which is built for both fast, reliable, robust and flexible simulations of battery models. PyBaMM is elaborated on at length in Section 3.3 through Section 3.4.4.

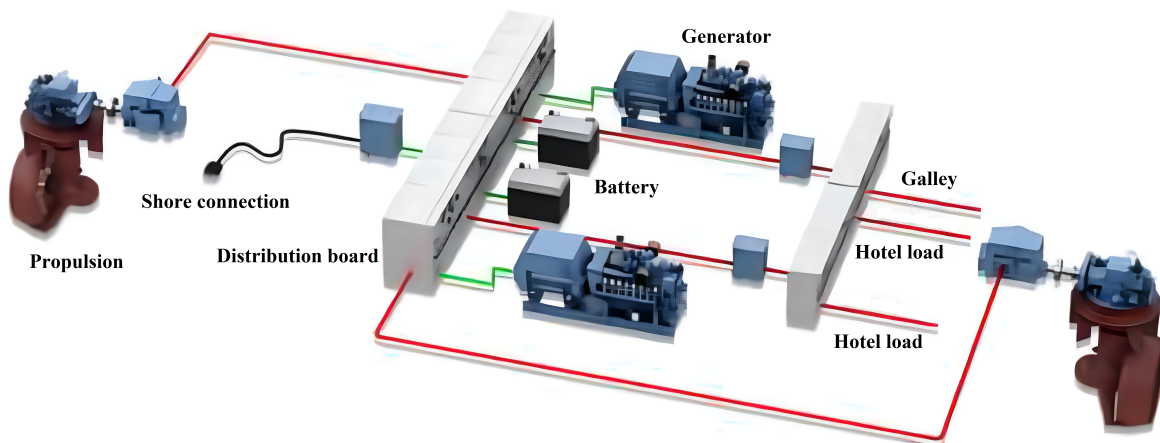
## 2 Theory

This chapter provides a background on ferries and energy storage at large. The sections not only provides a independent take on the subjects but also provides context on ferries with regard to energy storage. The subsequent sections in the theory go into batteries, and in particular the LiB and its figures of merit, and ageing mechanisms. The concepts which are found in these sections are of special importance to fathom in order to be able to model batteries, and their ageing properly. Furthermore, the ferry and energy storage sections provide an understanding of LiB application, which is needed to accurately conduct application based simulations.

### 2.1 Ferry Structure

Maritime vessels such as electric ferries has large power needs. The demand from both the propulsion system, and the hotel load has to be met by the energy storage system. The energy storage systems which are found inside ferries varies, but they should always be secured with some form of redundancy, like a diesel generator, in order to withstand charging issues for instance.

Figure 2.1 shows the main components of the power system which is found inside a hybrid ferry, which is similar to a small power distribution grid. It consists of means of electricity generation, such as batteries and generators, reshapers, which changes and distribute the form of energy, energy storage, and means of transport to end usage. Batteries and generators are both connected to the main busbar. The busbar remains of a large fuse box in Figure 2.1, and it is also shown to be the means of connection to power from the shore. A distribution board supplies the electricity which is generated by the engines.



**Figure 2.1:** Principal sketch of a hybrid ferry [12].

#### 2.1.1 Main Components

In a conventional propulsion system the propeller is directly connected on to the shaft of the engine. Earlier ferries, which are not hybrid or all-electric, have a much smaller electricity demand since the propulsion system is not directly fed by electricity.

The propeller can either have a fixed or a controllable pitch. Some advantages with the controllable pitch propeller is that it can be used together with a non-reversible engine to move the ship both forwards and backwards, which can reduce weight. It is however more complex, and thus more expensive, but it is well suited for vessels that need precise maneuvering.

For a diesel-electric propulsion system, the engine's shaft is directly connected to the diesel generator. Which is different from conventional propulsion, wherein a single axle connects the diesel motor and the propeller. In the diesel-electric design the shaft is replaced with electric machines. As a result, the engine room does not need to be as strategically placed in the vessel, since no axle needs free passage between the two end points.

In a fully electric system, energy can be stored through the usage of batteries. The ferry is also equipped with diesel generators as a backup in case of emergency. Typically, hybrid systems are designed in a way such that the electrical system may relieve some of the load. In some circumstances however, hybrid vessels may run exclusively on electrical power. Typically, conventional fuel is the primary energy source. Figure 2.1 provides a schematic for how a hybrid ferry could be designed. Hybrid system has been proven to offer a range of advantages over diesel only system. Some of these advantages are compiled in the list below [8].

- Greater redundancy
- Reduced fuel consumption
- Reduced emission of CO<sub>2</sub> and other pollutants
- Not impacted as much by uncertainty in future fuel costs
- Insurance against increasing environmental legislation
- Noise reduction
- Possibility of operating in a zero emission mode when the ship is docked
- Less maintenance

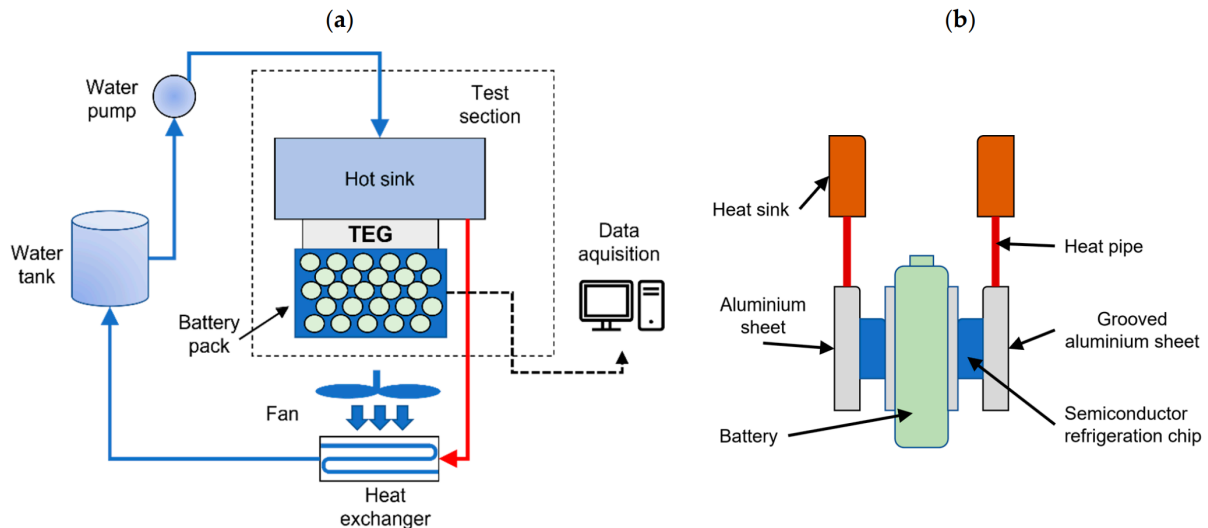
At the time of this report the internal combustion engine, and electric generators are used to transform energy in vessels. Fuel cells are however being developed, and are likely to become more prevalent in the future. The internal combustion engines which are found inside ferries typically run on oil based fuels such as marine diesel, this common to find both in conventional and hybrid solutions [13]. The following list presents the different components and their efficiencies.

- **Internal combustion engine:** Typically, uses petrol and diesel engines, with an efficiency of up to 45% depending on the engine. Mostly, diesel engines are used in larger vessels, because it is cheaper. Has the advantage of having a superior energy density per liter stored. Tanking is also easy, making this system reliable [8].
- **Generator:** Generators convert mechanical energy into electric energy by means of rotating magnetic fields, which induces a voltage in the stationary coils. The efficiency of a generator typically ranges between 90 % and 99 %, depending on its size. Generators are often connected to an internal combustion engine to form a diesel-electric system.
- **Fuel cell:** Fuel cells convert chemical energy into electrical energy when the cell is fed by a reactant gas. The fuel cell has a membrane which allows passage for ions, whilst electrons are moved through an external circuit. The current and voltage which is produced depend

on both the electrochemical reaction in the fuel cell, and external load impedance. This technology has an efficiency of around 50% [8].

### 2.1.2 Cooling System

As the battery is charged and discharged, heat is generated. When the temperature is too low or too high, it will affect the battery and accelerate its ageing. It is therefore important to regulate the temperature and to limit the temperature increase. In a maritime vessel, a similar cooling system, as presented in Figure 2.2, is used to regulate the temperature of the cells of a battery pack.



**Figure 2.2:** An illustration on how the cooling system is connected (a) illustrate a overall system and (b) is a closer look on how heat transferring materials are implemented around a cell [14].

Three types of fluids can be used for the cooling system,

- **Air:** Although air has been found to be effective in removing excess heat generated by battery packs, its use is often limited. Onboard a marine vessel, the surrounding air is sea air, which creates a challenging atmosphere. One of the advantages of this is that the ventilation can be passive, and thus not dependent on electricity, and that it occupies less space. The heat generating components might need additional external fans, air conditioning and cooling ducts, which require electricity.
- **Liquid:** Most vessels are cooled using sea chests and heat transfer through contact surfaces within the sea chest. The reason for this is that the seawater temperature is approximately constant, and as a result of its characteristics, it is also more efficient at dispersing heat than air. These systems are innately more complex, partly since the water cannot directly contact the battery cells, and must be directed by a sea chest or a temperature regulator to these batteries. A power supply is also necessary in order to power the pumps. Furthermore, there exists cooling systems that uses liquids which are directly in contact with the cells. The liquid must be dielectric, as it should not conduct electricity. The most common material used for this purpose is mineral oil or silicone [15].

- **Phase change material (PCM):** PCMs are materials that change state with temperature, typically from solid to liquid or from liquid to gas. In this case, the heat is absorbed by a phase changing material to change its state. It is especially useful if there are brief periods which produces excessive heat, for example during fast charging. Which could prevent the battery from running at its optimal temperature level. If PCMs are used to counteract this, it can lead to increased battery life. Figure 2.2 illustrates how a liquid and PCM hybrid cooling system might appear [15].

### 2.1.3 Grid Infrastructure

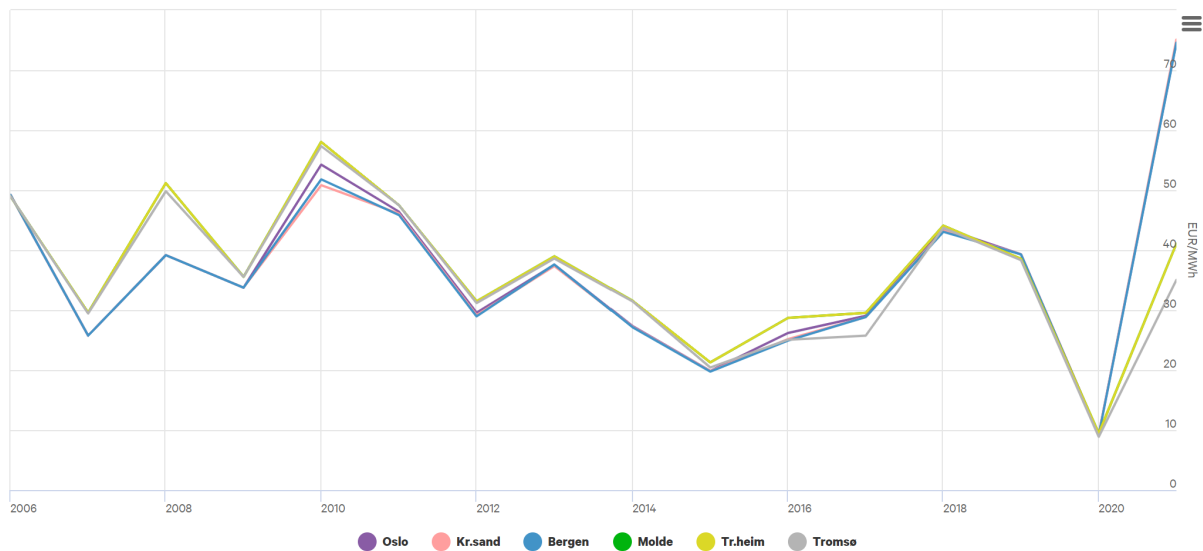
When a ferry charges its batteries needs a large amount energy in a short amount of time, this puts a strain on the grid, which in some cases is insufficient to meet this additional power demand. To reduce the load on the power grid, battery packs can be installed.

A poor grid can benefit from being aided by land based battery packs at the locations of ferry harbours, more precisely, electric ferry chargers. The charging period of a ferry might last 5 minutes, and it can draw a power of 4.5 MW for instance. For the next hour or so, the ferry charger might not be engaged in any charging. This means that the grid has to supply 375 kWh of energy through the course of an hour, to recharge the amount of energy which were dispended in 5 minutes when the ferry charged. From this back-of-the-envelope calculation it is found that the peak power which is drawn from the grid due to the electric ferry, could be reduced as much as twelve times. Which is favourable, if the grid already lacks in capacity.

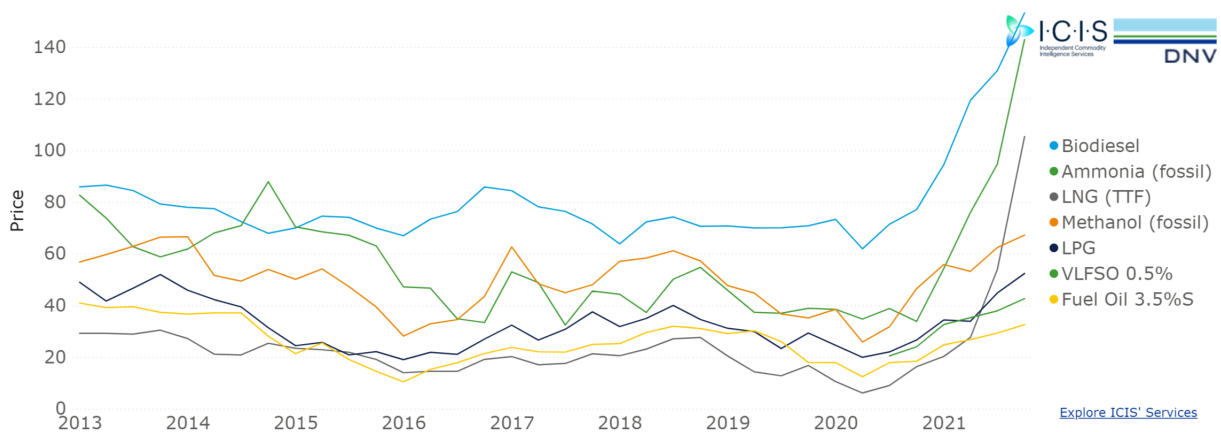
According to the ferry's operational requirements and the system which is installed, it should be able to charge at 1-4 MW for 5 minutes, and the grid needs to be able to accommodate that [16]. There are as of today no standardizations on charging facilities. Because they are relatively new, technological advancement has been preferred over settling for a possibly outdated standard. Plug, pantograph, and contactless charging are the most common connection types.

### 2.1.4 Electrical Prices versus Fuel Prices in Norway

According to the Figure 2.3, electricity prices have remained relatively affordable, and are in line with fuel oil prices, illustrated in Figure 2.4. According to the figures, prices seem to have a positive correlation, or develop similarly. This could indicate that there are no advantages to using electric over diesel, with regards to the development of prices in the near future. Costs could however be reduced by using electricity over conventional diesel engines, as the inefficient fuel usage can be drastically lowered. Additionally, CO<sub>2</sub> taxes are avoided, this also reduces costs [17].



**Figure 2.3:** Electricity prices over the last few years, data presented by Nord Pool [EUR/MWh] [17].



**Figure 2.4:** Chart of different fuel prices by DNV [EUR/MWh] [18].

### 2.1.5 Alternative Fuels and Retrofitting

Worldwide, there are 6 615 alternative fuel vessels registered to be in operation, and 625, or almost 8% of these vessels have, or are scheduled to have battery packs installed. At the beginning of 2022, Norway will have 187 of these vessels in service, and an additional 67 are in order. Despite its small population of approximately 5.5 million, Norway accounts for 40% of the world's battery fleet [13, 19].

This report is based on electrochemical energy storage, but in several upcoming vessels it might not necessarily be the best technological solution. In this section, alternatives to batteries, which can replace marine diesel in the future, are presented.

Ammonia has a higher energy density than hydrogen, as well as a higher evaporation point, which makes it an easier fuel to store. There is, however, a disadvantage to this process, which is that it is highly corrosive and that the nitrogen content creates reactions that can produce  $\text{NO}_x$  exhaust gases, which is unideal [8].



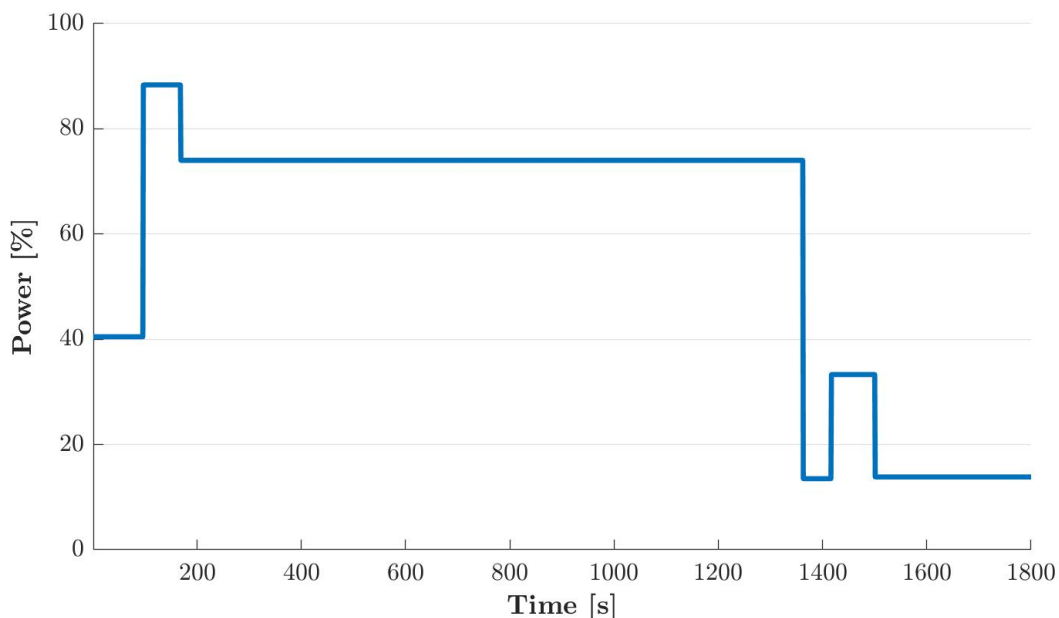
Alternatively, hydrogen may be used since the biproducts is water. The downside to hydrogen, however, is its minuscule size, which makes it very difficult to store efficiently [20].

Liquefied natural gas can be used together with renewable liquefied biogas, which can increase sustainability. A benefit with liquefied gas is that it barely takes any new equipment, and readjustments to convert existing engines. Liquefied natural gas is porven, and contains less carbon per unit of energy compared to fuel oil. It is however unclear if this reduces greenhouse gas emissions on a life cycle basis [21].

Another alternative is to use nuclear power. Nuclear power is not a renewable energy source, but it does fall under the zero-emission energy source. Both projects and political will to commercialize have, however, been lacking. The technology would require large insurances, but is successfully applied in military vessels [8].

### 2.1.6 Load Profile and Sea Conditions

The general energy requirement for a ferry crossing can be found, and is in general predictable during normal operating conditons. Figure 2.5 shows a typical load profile for a single ferry crossing. The first phase of the load profile indicates the load when the ferry is manoveroing away from the dock. The first load spike is connected to the acceleration of the ferry, whilst the long continous line indicates the main fjord crossing. This is followed by manoeuvring, deacceleration, and the docking period. Wherein the ferry only has a short amount of time to charge, which is shown by the last line segment. The load profile will however have some variability depending on weather conditons and the number of vehicles on each transport.



**Figure 2.5:** An estimated loadprofile of ferry crossing a fjord

## 2.2 Energy Storage

Energy can be stored in different forms and through the utilization of a vast array of methods, technologies, and devices. All energy storage solutions are based on the principle of saving energy such that it can be accessed in the future when demand arises. This is analogous to preparing a lunchbox in the morning for an energising meal during work. In engineering energy storage is all about saving energy in the right form, at the right place and at the right time [22]. The examples in the following section, Section 2.2.1, highlight the main applications of energy storage, i.e., energy management, power bridging, and power quality management [23].

### 2.2.1 Motivation for the Implementation of Energy Storage

Renewable energy has emerged, among other things, to combat climate change and improve global energy security. Renewable energy can both reduce greenhouse gas emissions and be deployed to meet increasing energy demands. These cleaner sources of energy have also become the most widely applied solutions for addressing issues such as the inevitable depletion of oil reserves [24]. The majority of sustainable energy sources are both intermittent and unpredictable in nature. This can be managed by the flexibility that energy storage can provide within an energy system. The excess energy provided for by intermittent sources can be consumed and then later be discharged when demand once again rises above supply. The need of energy storage grows with electrification of society. Energy storage can also enable intermittent renewables to achieve greater penetration in a wide range of sectors such as transportation, cooling and heating of buildings and electricity generation [2, 3, 25].

Energy storage can for instance, in addition to allowing for the storage of renewable energy, provide a range of grid services. Electrical grid developers, amongst other things, must ensure that voltage and frequency levels are acceptable such that electrical appliances are not damaged [26]. Grid developers have to make sure that the electrical distribution network adheres to regulations and that it is upgraded to meet future power demands, this leads to a need of enhancements in old grids such as the Norwegian distribution grid. Enhancements are conventionally done through expensive grid reinforcements. Conventional upgrades will no doubt be needed at many locations. Another alternative is to use energy storage devices that draw power from the grid in order to be used for peak shaving when demand surges [27]. Energy storage can ease investments in grid enhancements at suitable locations. Power demand can surge periodically in locations with a low population density and low grid capacity utilization, locations like cabin destinations and ferry harbours with electric ferry chargers. Energy storage can function as equally viable technical solutions, negate over dimensioning of the electrical grid, and save capital. This could explain why energy storage systems has become increasingly popular in frequency control, power quality control and voltage regulation [28–30].

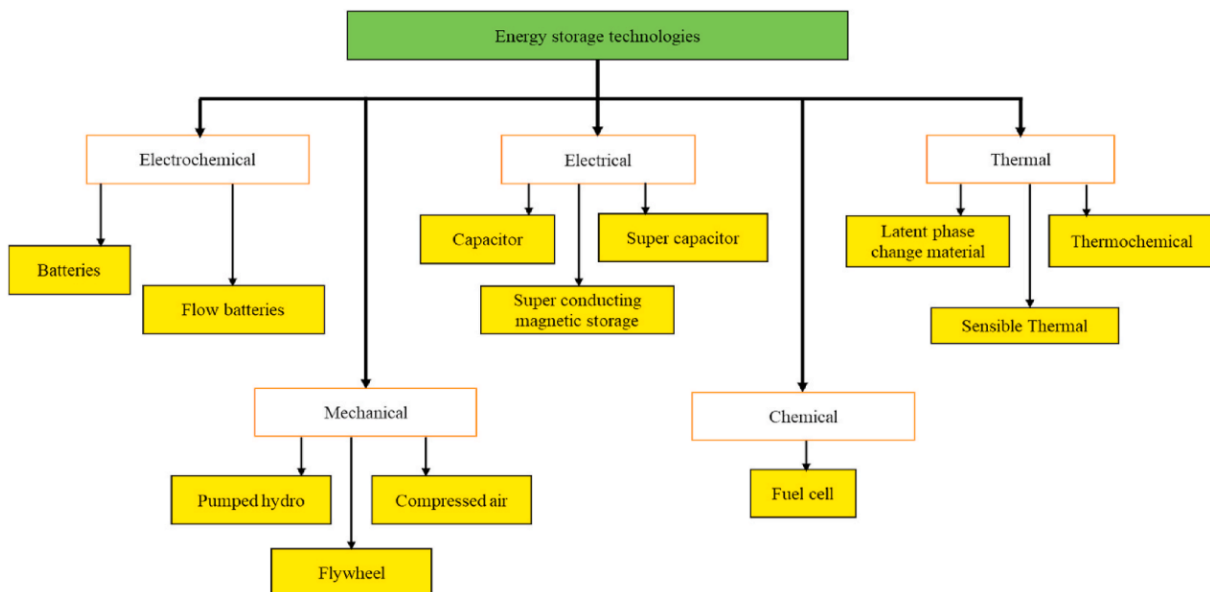
The last decade has been shaped by portable consumer electronics such as the smartphone, an invention which could not have been without energy storage. In a report, in part written for the Office of Electric Transmission and Distribution, it was estimated that \$ 80 billion is lost in the U.S. every year, mainly due to short power interruptions [31]. A more resilient electrical grid equipped with better backup power, with a higher degree of energy storage, could alleviate some of these losses. Energy storage can also play a key role in securing delivery of power to critical loads in facilities such as hospitals. A strategy consisting of a microgrid, containing

both renewables, diesel generators and energy storage, feeding power to a hospital was found to increase security and be cost-effective. It was reported that some \$ 400 000 could be saved on average over a 20-year period through energy savings and avoided losses during power outages [32].

Energy storage can play a key role in enabling the societal transformation of decarbonization [33]. It can therefore be argued that driving energy storage technologies forward, either through investments in research and development, or through the deployment of more mature technologies is an apt motivation in and of itself, even if better solutions exist, since these investments could contribute to the commercialization of transformative energy storage solutions.

## 2.2.2 Methods of Classification and Properties of Energy Storage Solutions

For any application of energy storage engineers has to assess which type of energy storage is most suitable. This can in part be determined through an inspection of which solutions have properties aligned with the demands of the system. It is therefore apparent that it is useful to define certain properties which describes the characteristics of a given energy storage solution. There exists a wide range of methods of storing energy, and to choose the correct solution, these needs to be benchmarked against each other. Some methods store electrical energy whilst other methods store electrochemical-, chemical-, mechanical- or thermal energy. Figure 2.6 presents the technologies used to store different forms of energy.



**Figure 2.6:** Energy storage systems classified by form of energy [34].

No matter the form of energy, it is important to define parameters that make it possible to compare different technologies more rigorously. It is useful to normalise the power and energy of an energy storage system. The most widely used units for this are given on a per-mass basis. Specific energy, defined as energy per mass, describes how energy dense an energy storage device is. It allows easy comparisons between different energy storage systems, as seen in Table 2.1. Specific power is the measure used to evaluate the amount of power a system can deliver on

a per mass basis [22]. The term round trip efficiency describes how much energy one can pull out of a system, relative to how much energy is input to store the same amount of energy. In general, the higher the round trip efficiency of a system is, the lower its losses are. Systems with high round trip efficiencies have to use less energy to store and access equal amounts of energy relative to systems with lower round trip efficiencies. The lifetime of a system can be described by both calendar lifetime and cycle lifetime, each important in its own respect. Calendar lifetime describes the longevity of a system with respect to time, i.e., how many years a given system is expected to last. Whilst cycle lifetime describes how many cycles a system can endure. One complete cycle consists of both the charging and discharging phase [34–40].

The following tables: Table 2.1, Table 2.2 and Table 2.3 provides some key energy storage metrics for a variety of energy storage technologies. The tables showcase key figures of pumped hydroelectric energy storage (PHES), flywheel energy storage (FES), compressed air energy storage (CAES), lead-acid batteries (PbA), nickel–cadmium batteries (NiCd), sodium–sulfur batteries (NaS), Sodium–nickel chloride zebra batteries (NaNiCl<sub>2</sub>), LiB, vanadium redox flow batteries (VRFB), super capacitor energy storage (SCES), superconducting magnetic energy storage (SMES), and hydrogen energy storage (HES) [34–40].

**Table 2.1:** Technical features of different energy storage technologies [34–40]

<b>Technology</b>	<b>Specific power</b> [ <i>W/kg</i> ]	<b>Specific energy</b> [ <i>Wh/kg</i> ]	<b>Round trip efficiency</b> [%]	<b>Calendar lifetime</b> [ <i>yr</i> ]	<b>Cycle lifetime</b> [–]
<b>PHES</b>	-	100-400	65-87	40-60	50 000
<b>FES</b>	11 900	5-100	90-95	20+	20 000
<b>CAES</b>	-	30-60	50-89	20-60	10 000+
<b>PbA</b>	74-415	15-40	65-90	5-15	200-1 800
<b>NiCd</b>	100-300	30-80	60-90	5-20	1 500-3 000
<b>NaS</b>	150-230	100-240	75-90	10-20	2 500-4 500
<b>NaNiCl<sub>2</sub></b>	150-200	100-120	85-92.5	10-14	2 500+
<b>Li-ion</b>	80-370	75-200	78-98	5-20	1 000-10 000
<b>VRFB</b>	80-166	10-35	60-85	5-20	12 000+
<b>SCES</b>	500-10 000	0.5-20	65-99	8-20	500 000
<b>SMES</b>	500-2 000	1-75	80-98	20-30	100 000+
<b>HES</b>	500+	150-10 000	45-66	5-15	1 000+

Table 2.2 is a continuation of Table 2.1, and it includes the key metrics of discharge time, response time, daily self-discharge and suitable storage duration. The discharge time describes how long it takes for the accessible energy in a device to be discharged. For comparisons, the nominal discharge time, which is defined as the discharge time at the rated power, is useful. Response time describes how fast a system is able to react, i.e., how fast a system is able to deliver power. Another important figure, daily self-discharge, describes how much energy a charged system loses without power being drawn from it on a daily basis. Suitable storage duration simply remarks which time periods a given solution is suitable for [34, 37, 38, 40].

**Table 2.2:** Technical features of different energy storage technologies (continues) [34, 37, 38, 40]

Technology	Discharge time [ <i>ms – h</i> ]	Response time [ <i>ms – min</i> ]	Daily self- discharge [%]	Suitable storage duration
<b>PHEs</b>	6-24 h+	sec-min	Very small	Mid-long term
<b>FES</b>	ms-1 h	4 ms-sec	100	Short term
<b>CAES</b>	1-24 h+	1-15 min	Very small	Mid-long term
<b>PbA</b>	s-4 h	1-10 ms	0.03-0.6	Short-mid term
<b>NiCd</b>	s-h	20 ms-sec	0.07-0.6	Short-long term
<b>NaS</b>	1-24 h	sec-2 min	0.05-20	Long term
<b>NaNiCl<sub>2</sub></b>	s-h	sec	12-15	Mid-long term
<b>Li-ion</b>	min-1 h	20 ms-s	0.04-0.33	Short-mid term
<b>VRFB</b>	s-10 h	sec-10 min	0.2	Long term
<b>SCES</b>	ms-h	8-10 ms	5-40	Short term
<b>SMES</b>	ms-1 min	10-100 ms	10-15	Short term
<b>HES</b>	s-24 h+	sec	Almost zero	Long term

Table 2.3 further elaborates on the power cost, energy cost, technological maturity, and environmental impact. Power cost describes how cheap or expensive installed power is for a given energy storage technology, whilst energy cost describes the cost of the installed energy storage capacity. The two last columns of Table 2.3, technological maturity and environmental impact, respectively, describe how well developed a storage technology is and how bad it is for the environment [34, 37, 38, 40].

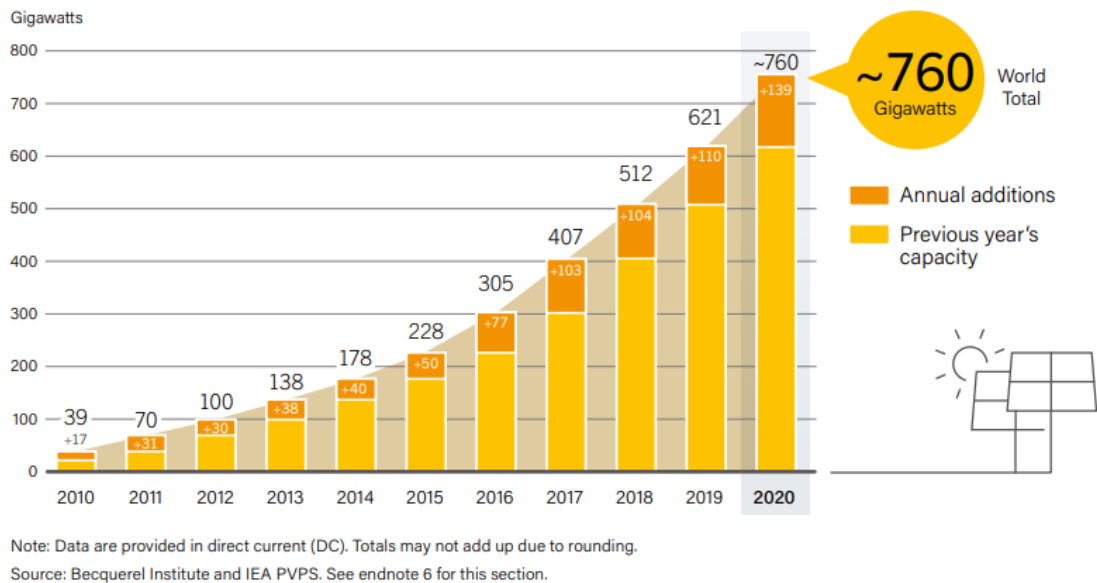
**Table 2.3:** Technical features of different energy storage technologies (continues) [34, 37, 38, 40]

Technology	Power cost [\$/ <i>kW</i> ]	Energy cost [\$/ <i>kWh</i> ]	Technological maturity	Environmental impact
<b>PHEs</b>	500-4 340	5-270	Very mature	High/medium
<b>FES</b>	250-380	500-14 000	Mature	Very low
<b>CAES</b>	400-1 630	2-270	Proven	Medium/low
<b>PbA</b>	200-650	50-400	Very mature	High
<b>NiCd</b>	500-1 500	400-2 400	Very mature	High
<b>NaS</b>	380-3 260	300-540	Proven	High
<b>NaNiCl<sub>2</sub></b>	150-300	100-350	Proven	Medium/low
<b>Li-ion</b>	900-4 340	300-2 500	Proven	Medium/low
<b>VRFB</b>	600-1 630	150-1 090	Proven	Medium/low
<b>SCES</b>	100-480	300-2 000	Proven	Very low
<b>SMES</b>	200-490	1 090-10 850	Developing	Low
<b>HES</b>	500-3 000	10-20	Developing	Low

The most up-to-date review articles were used to quantify the parameters presented in Table 2.1, Table 2.2 and Table 2.3. The field of energy storage is however rapidly evolving. Therefore, even the most current sources are bound to get outdated sooner rather than later. The information should nevertheless outline the present characteristics and traits of different energy storage technologies.

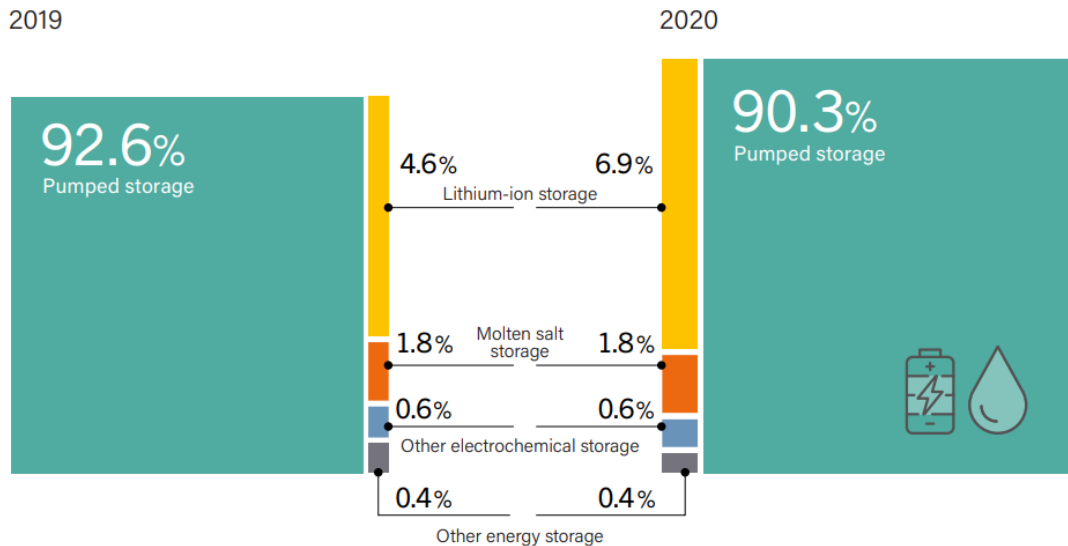
### 2.2.3 Energy Storage Trends

Global electricity demands are constantly rising, and intermittent energy sources are playing an increasing role in the global energy mix [41]. The yearly status reports from REN21 on renewables makes the sector's growth apparent. The status report for 2021 presents this overarching growth trend, which amongst other things is manifested as Figure 2.7.



**Figure 2.7:** Growth of installed PV capacity from 2010-2020 [42].

Figure 2.7 shows tremendous growth in installed photovoltaic capacity over the last decade. The bar chart also shows that the growth has accelerated. Figure 2.8 is also presented in the latest global status report on renewables by REN21. With increased variable energy production, such as solar photovoltaics (PV), the need for energy storage rises. Figure 2.8 shows the installed global energy storage capacity by technology and the respective shares of the major technologies.

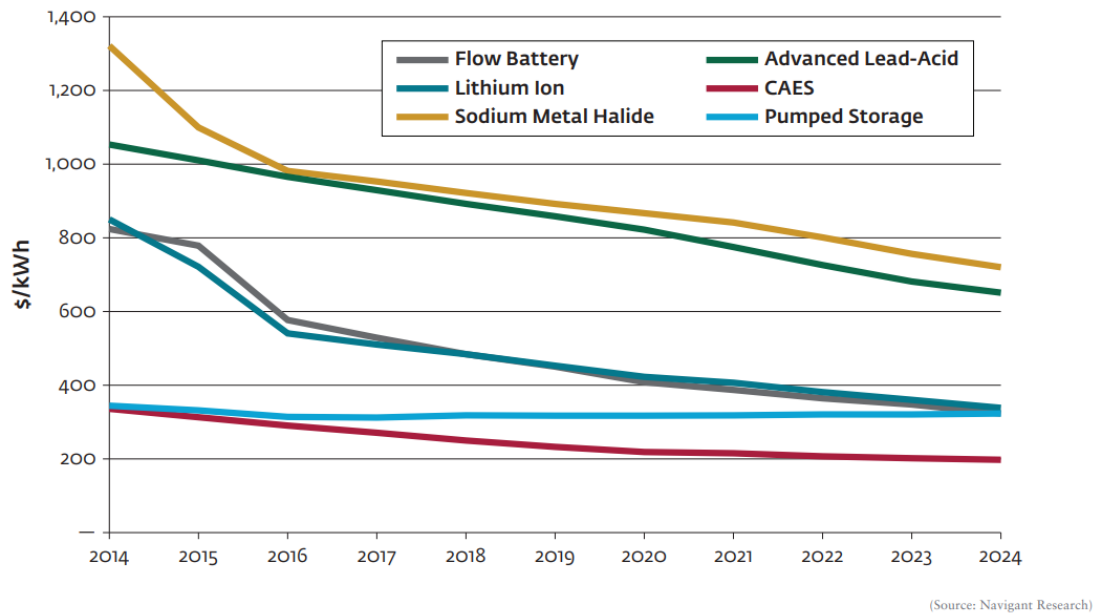


**Figure 2.8:** Share of global energy storage installed capacity, by technology [42].

Figure 2.8 highlights the dominance of pumped hydroelectric energy storage in large scale energy storage. The International Energy Agency (IEA) also states that by 2025 more than half of the new hydropower capacity additions in Europe and China will be pumped hydro [43]. This will further cement pumped hydro's position in large scale energy storage. It is also noteworthy that lithium-ion battery storage is the fastest growing energy storage technology by capacity. Furthermore, the IEA projects that battery storage will grow to more than 150 GWh by 2026, up from some 20 GWh in 2020. The IEA also predicts that the global energy storage power capacity will increase by 56%, to 270 GW, by the end of 2026 [44]. The price per kWh of stored energy is also forecast to drop further in the coming years for different kinds of energy storage. This might lead to increased adoption like what has been observed with solar- and wind power, when their prices dropped respectively. Figure 2.9 shows the installed capacity, as well as the historical and forecast prices for a range of energy storage methods.

Figure 2.9 conveys some trends such as pumped hydro being projected to cost as much lithium-ion and flow batteries on average by 2024. Projected prices are steadily decreasing for the depicted energy storage solutions, with the exception of pumped hydro, where the most cost-effective projects have in all likelihood have been conducted at an earlier time. Table 2.4 has been adapted from Rahman et al.'s Assessment of energy storage technologies: A review [36]. Table 2.4 shows the sum of the rated power capacity and the number of energy storage projects by technology and project status.

The mechanical energy storage category consists of PHES, FES, and CAES. Chemical storage only accounts for HES. Table 2.4 agrees with the previous statements on both the domination of PHES in large-scale energy storage and the growth of LiBs. This is conveyed through the rated power, which is the largest for mechanical storage for every project stage in Table 2.4. Electrochemical installations outpace the other technologies by number of projects, which is indicative of the current electrochemical, and in particular LiB, energy storage growth.



**Figure 2.9:** Energy storage capacity price history and forecast [45].

**Table 2.4:** Rated power capacity in GW and number of energy storage projects by project stage around the world [36]

Technology	Operational	Contracted	Announced	Under construction
<b>Mechanical</b>	166.2 (372)	2.31 (11)	11.63 (26)	5.95 (7)
<b>Electrochemical</b>	2.03 (695)	0.95 (67)	0.63 (136)	0.70 (10)
<b>Chemical</b>	0.01 (7)	0.003 (3)	0.001 (1)	-
<b>Thermal</b>	3.21 (193)	0.13 (3)	0.16 (5)	0.12 (2)

### 2.3 Batteries

Batteries are energy storage devices that, when demanded, directly convert chemical energy into electrical energy. Electrochemical oxidation-reduction reactions in the battery convert stored chemical energy into accessible electrical energy. The amount of energy that a given battery can supply is bound by the number of chemical reactants contained within it [46, 47].

Battery as a term is somewhat ambiguous; therefore, it is helpful to define a guiding battery nomenclature. The smallest electrochemical unit in a battery is known as a unit cell. The electrochemical unit cells are the fundamental building blocks of batteries, and a single battery cell can consist of either one or multiple of these unit cells put together in a single unit. To construct a cell, the unit cells are connected in series, parallel, or in a combination of both. Battery manufacturers can tailor this to get a certain capacity at a given output voltage. A collection of battery cells makes up a battery module, these modules can further be combined into a battery pack. Cells are for the purpose of this report lithium-ion unit cells, if not stated otherwise [47].



Many battery technologies have been proposed, and some have been commercialized. Many technologies are still actively researched, in development, or still on the road to commercialization. Researchers try to create batteries which can be characterized by large specific- power or energy, or both. Costumers also wants batteries that are safe, have a long lifetime, and are made from abundant materials, such that costs are low. The ethics of production are also important. For the time being, no single battery technology is the best for every use case. Therefore, different batteries that fulfil required characteristics are chosen for dissimilar applications.

### 2.3.1 Primary and Secondary Batteries

Batteries, can by their ability of being electrically recharged, be categorized into two main categories. These are primary and secondary batteries, also known as non-rechargeable and rechargeable batteries. In a primary battery, the chemical reactions feed off one of the electrodes, such that they cannot be electrically recharged after a full discharge. Primary batteries are because of this usually disposed of after one discharge [47]. Some primary batteries can however be mechanically recharged. Some versions of the Zn-air primary battery are for instance mechanically rechargeable through the usage of replaceable anode cassettes [48].

Rechargeable batteries, or secondary batteries, are innately electrically rechargeable without the need of replacing any components after a discharge. Simply imposing an external voltage and passing a reverse current through these batteries results in a reversal toward the charged battery state [47]. The process of charging and discharging is however unfortunate, not completely reversible. Section 2.6 elaborates on this.

### 2.3.2 Battery Geometries

Different battery geometries exist, each with their own use cases, drawbacks, and advantages. Figure 2.10 shows some conventional cell geometries.

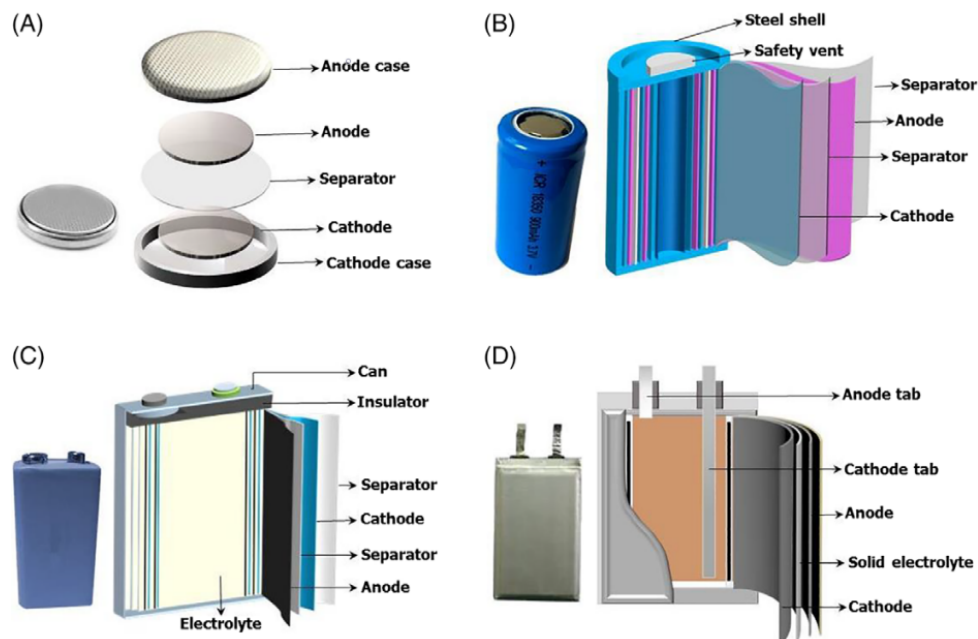


Figure 2.10: Conventional cell geometries [49].

The button, or coin cell, is both small and inexpensive. Its consistency and compactness make it ideal for small portable devices. The cylindrical cell has good mechanical and thermal stability, but a larger number of cells are needed to store the same amount of energy as other cell types. The prismatic cell can be expensive, and its main advantage is that it is volume efficient. Pouch cells are also volume efficient, simple, and lightweight. However, the cell needs room into which it can expand, as swelling is normal [50].

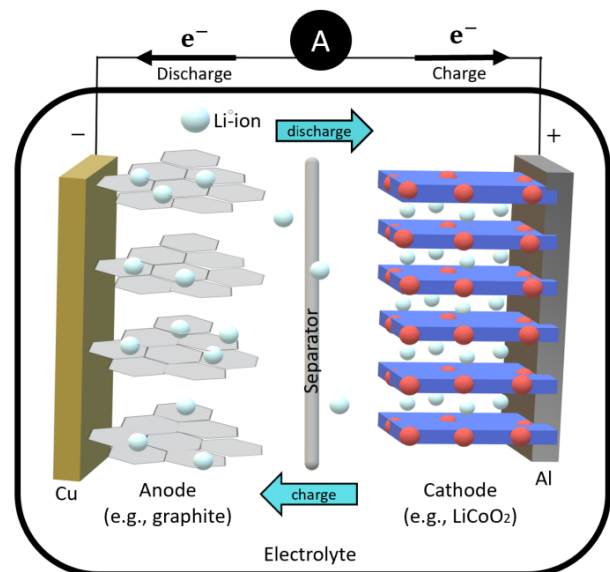
## 2.4 Lithium-ion Batteries

The rechargeable LiB was developed and commercialised in the early 1990s as a response to the growing need for improved batteries. In the following decades, a great deal of effort has been put into improving LiBs, and consequently, impressive progress has been achieved. Today, the LiB battery is the secondary battery which has the highest energy density, it is also one of the most advanced and the most dominant rechargeable batteries. A dominance it is suspected to hold on to in the coming years. LiBs are generally considered the state-of-the-art battery for electric vehicles, power tools, and portable electronics [10, 11].

Lithium (Li) has the atomic number three and is because of this the least dense metal element. Another favourable characteristic of lithium is the fact that lithium can be used to generate high electrode potentials. In essence, this means that a large amount of voltage is generated per cell. This reduces the required number of cells in battery package to achieve a set output voltage. In Figure 2.8 LiBs are shown to both already be widely applied, as well as gaining in market share in the energy storage market. As the world transitions towards cleaner energy the intermittent nature of renewables has to be smoothed out with energy storage such as LiBs, as brought up in Section 2.2.1. The transport sector has also encountered a major shift due to the introduction of LiBs. Especially with regards to the automotive industry. LiBs are commonly used in portable energy storage systems due to their high specific energy capacity, energy efficiency, rapid changeability, light weight, long cycle life, high reliability, flexibility, and low self-discharge rates [10, 11, 51, 52].

### 2.4.1 Structure and Working Principle

Each LiB unit cell consists of the same key components; two electrodes, a separator, an electrolyte, and two current collectors, as shown in Figure 2.11. On top of the number of unit cells each LiB is made up from, fully-fledged LiBs also consists of two tabs and a casing. LiB cells have two electrodes, which has a potential difference that depends on the materials that are used. The electrodes store the chemical energy, and they are respectively named the anode and the cathode. Strictly speaking the names of the electrodes switches if the LiB is charging or discharging. The electrode which is oxidised is known as the anode, or negative electrode, because it has electrons leaving from it. Whilst

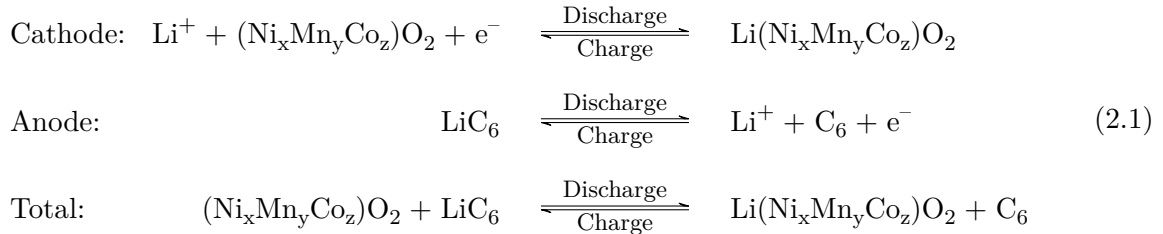


**Figure 2.11:** LiB cell structure and movement of charges [53].

the electrode which accepts electrons is reduced and is known as the cathode or the positive electrode. However, a more convenient naming convention stems from primary batteries [54, 55].

Primary batteries can only be discharged electrically, electrode names are therefore constant. The anode is, according to this naming scheme, always known as the electrode which gets oxidised during discharge, whereas the cathode is the electrode which gets reduced during discharge. This naming scheme is most widely used and the most convenient for the purpose of this report, and is thus used. The separator hinders contact between the two electrodes, preventing short circuit. The electrolyte facilitates transport of ions, as well as forcing electrons towards the external circuit. The two current collectors and tabs ensure the transport of electrons from the electrodes to the external circuit. Whilst the casing inculcates and protects the LiB components [56].

In this report LiBs which uses graphite anodes and NMC cathodes are put an extra emphasis on. The chemical reactions that convert chemical potential to electrical potential and vice versa in this specific LiB are summarised in Equation 2.1 [57].



The basic principle of rechargeable LiBs is shown with Figure 2.11 and Equation 2.1. The power source is active during charge and is switched to a load during discharge in Figure 2.11. Further the illustration and the reactions show the listed cell components, as well as the action of electrons and ions during discharge and charge. During discharge, the difference in potential between the two electrodes causes cyclable  $\text{Li}^+$  to be extracted from the anode, and the ions are transported by the electrolyte through the separator, finally intercalating into the cathode. At the same time, spontaneous reductions in the cathode cause electrons, eager to conduct work, to flow from the cathode to the anode, through the load in the external circuit. This ensures charge neutrality. This split mode of transportation of electrons and ions is what enables work to be done. It is made possible by the fact that the electrolyte is both ionically conductive and insulating. During charge, an external voltage is applied such that the  $\text{Li}^+$  diffuses towards the anode, and are reduced back to neutral Li atoms. At the same time electrons move through the external circuit towards the cathode. Thus, the reaction is reversed, and energy is stored again in the form of chemical potential [56, 58].

#### 2.4.2 Chemistries

LiBs are batteries that intercalate lithium-ions in their electrodes. They are further characterised by their chemistry and geometry. Note that different chemistries refer to different LiB chemistries in this report, and not different battery chemistries, such as nickel-cadmium or lead-acid. Geometry influences the performance of a LiB but is in general overshadowed by the effects of using different chemistries. Graphite based anodes are usually used in LiBs. These anodes

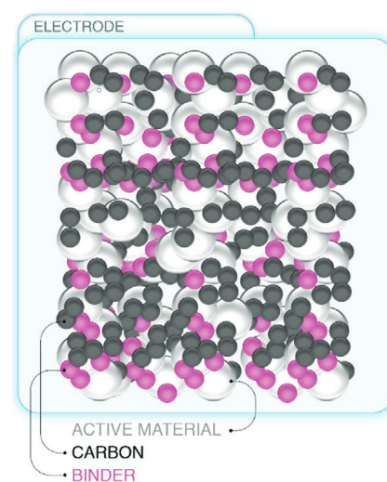
have higher theoretical specific energy capacities than cathodes. This has led to the cathode lagging behind the anode in energy density. LiBs are thus named by their cathode chemistry, since this varies between LiBs. More precisely, LiBs are named by the active material which their cathode uses. The active material of a battery refers to the materials which chemically reacts to produce an electrical current when a LiB discharges. For NMC chemistry, these materials can be found in Equation 2.1 [54].

### 2.4.3 Electrode Structure

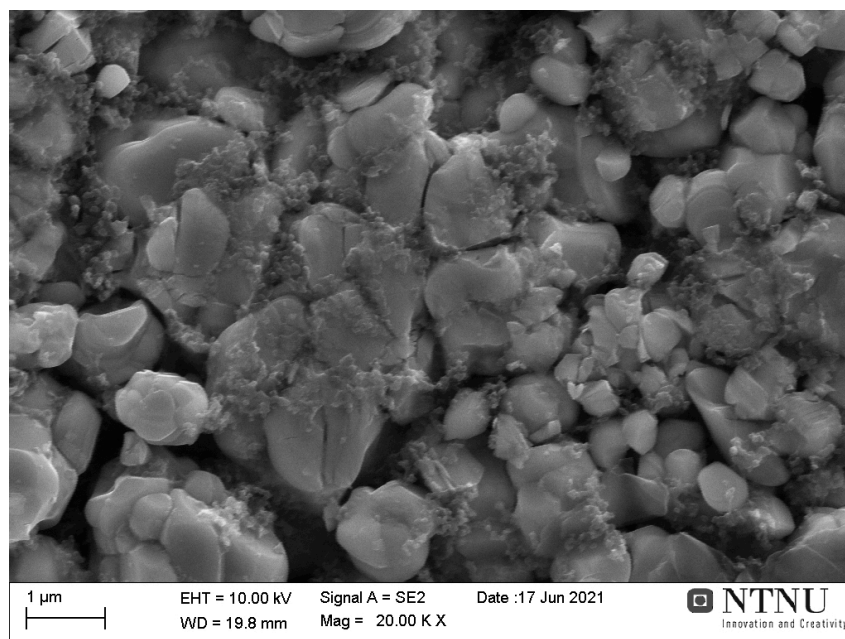
The electrode structure consists of both active- and inactive materials. The inactive materials which are found in electrodes are known as the electrode conductive materials and binders. These materials are of utmost importance for the capability of LiBs. The carbon black material is commonly used to provide conductivity to the electrode [60]. This combination of materials is shown in Figure 2.12.

In Figure 2.12 the binder material is shown to be responsible for maintaining the active material of an electrode together with the conductive agent, carbon. The binder material also ensures electrical connection between the electrode and the current collector by providing adequate adhesion. The electrode particles are shown on top of this collector. The binder has to be steadfast

against the temperatures and electrochemistry it faces in its surroundings, and this has to be done without compromising conductivity [60]. The electrode consists of millions of particles, this allows for pores, and increases surface area compared to continuous objects. Figure 2.13 shows these particles, it is an image taken of the electrode surface of an aged cathode. In the bottom left corner of Figure 2.13 a reference distance of 1  $\mu\text{m}$  can be located.



**Figure 2.12:** Composition of an electrode [59].



**Figure 2.13:** Scanning electron microscope image of an aged cathode [61].

The loading, compression, and pore structure of an electrode can also be changed to achieve certain characteristics. Loading describes the mass of the electrode per current collector area. Cells with electrodes which are heavily loaded are for instance generally known as energy cells, since they have higher specific energies. Whilst less loaded electrodes have lower resistances and are consequently able to generate more power, and are therefore classified as power cells [62, 63]. How compressed the electrode structure is changes the pore structure, which is paramount for ionic and electrical conductivity. Low electrode compression allows easier access to the electrolyte for the lithium-ions, while electrons are more easily conducted at larger compressions, since this packs the carbon additive more closely together, thereby reducing electrical resistance [63].

#### 2.4.4 Cathode

The cathode, or positive electrode, is the host structure that releases lithium when a LiB is being charged. The cathode should be able to chemically receive and deliver lithium-ions at ease. To store as much energy as possible, the cathode should also be able to store as large a quantity of lithium-ions.

In addition to good ionic conductivity, the cathode also needs to be able to easily conduct electrons. Commercially viable cathodes also contain qualities of longevity, stability, affordability, safety, lightness and sustainability [54].

The transition metal oxide cathodes: NMC, NCA, and LCO are all limited by the fact that they use cobalt, as signaled by C in their abbreviation. Cobalt is somewhat scarce, which pushes prices up. There are also ethical concerns regarding its extraction from mines in Congo. Wherein children, and miners work in extremely hazardous conditions, some so poor that they dig by hand [64, 65].

#### The NMC LiB

Adding cobalt to NMO enhances its structural stability, and thus prolongs its cycle life. This results in a new chemistry known as NMC [66]. NMCs have as of late risen to popularity due to its potential for application in portable energy storage systems, such as those found in electrical vehicles and ferries. Where both low costs, high specific energies and cycle life are of paramount importance [67].

The NMC chemistry can also be tailored to fit other applications. This is because the electrochemical properties of an NMC battery depend on its ratio of nickel, manganese, and cobalt and these can vary. It should also be noted that substituents also effects cathode properties [68]. Nickel concentration can, for instance, be increased to achieve higher specific energies. Nickel content can however not be increased indefinitely as thermal stability and life cycle issues will ensue. Whilst higher ratios of cobalt increases costs and ethical calamities. On the other hand, power generation will be reduced if too little cobalt is used. Manganese is both abundant, environmentally friendly and offers increased specific power due to its low resistance. Negatively, manganese decreases the specific energy [66, 68].

Focus has been put towards increasing nickel and manganese content in NMCs to address costs and ethics. The NMC-442 uses a reduced cobalt content compared to the NMC-333, which consists of equal parts of nickel, manganese, and cobalt. The numbers which follow the chemistry



are derived from the subscripts of their respective chemical formulas. And these are different for different variations of the NMC battery. This is why  $x$ ,  $y$ , and  $z$  are used in Equation 2.1. Efforts has also been put towards using larger contents of nickel, than manganese, such that energy densities can be increased, NMC-811 is an example of this [67, 69].

### Overview of Some LiB Chemestries

Table 2.5 shows a comparison of some key technical features of the NMC-, NCA- LMO- and LFP LiB chemestries. Table 2.5 shows that the NMC battery ranks second or first in all tabulated categories.

**Table 2.5:** Technical features of different LiB chemestries [70, 71]

Chemistry	Specific power [ $W/kg$ ]	Specific energy [ $Wh/kg$ ]	Cycle lifetime [-]	Monthly self-discharge [%]
NMC	500-3 000	150-220	1 000-4 000	1%
NCA	1 500-1 900	200-260	>1 000	2-10%
LMO	1000	100-150	300-700	5%
LFP	1400-2400	90-130	1 000-2 000	<1%

#### 2.4.5 Anode

The anode is the negative electrode. LiBs are as known, most capped by their cathode. The anode will therefore not be elaborated on to the same extent as the cathode. Nevertheless, LiBs are generally known to use carbon-based anodes, often as graphite or hard carbon. Despite their low cost, carbon anodes can offer high specific capacities and long lifespans [72]. Carbon anodes can also be combined with silicon in order to increase energy densities. Silicon can however for now, only make up a few percent of the anode content. The reason behind this is that the anodes, which utilise silicon, suffer from large volume expansion and contraction during charge and discharge. Casimir et al. has for instance reported on silicon nanoparticles, wherein a 400% electrode volume expansion were observed [73, 74]. This large volume change is very strenuous and leads to irreversible capacity fade, which limits its usage. Silicon-, stibium-, and selenium alloy anodes, could all because of their high energy densities, low costs and sustainability, serve as the next generation of anodes. Unfortunately, the large volume expansions, which they all face, must be dealt with first [75].

#### 2.4.6 Electrolyte

The electrolyte is crucial to the function of an electrochemical cell. An electrolyte needs to possess good ionic conductivity to facilitate the transport of lithium-ions during charge and discharge. However, the electrolyte should not bind excessively with lithium-ions, such that lithium-ions are inhibited from intercalating in the electrodes. Further, the electrolyte also has to be a good electronic insulator, such that transport of opposite charges are seperated. It is also crucial that the electrolyte is compatible with the separator, such that lithium-ions can be effectively transported between electrodes. It is also important to note that the electrolyte has to be compatible with both the anode and the cathode, such that the electrolyte is not reduced or oxidised, in place of either electrode [76, 77].

### 2.4.7 Separator

The cathode and the anode are separated by a thin insulating layer. This layer is called the separator, and it prevents a short-circuit between the electrodes. In addition to isolating the electrodes, the separator must also allow for the transport of lithium-ions. Thus, the separator has to be compatible with the electrolyte. If the separator is not compatible with the electrolyte, the electrolyte will not be able to facilitate ionic transport [78, 79].

### 2.4.8 Current Collectors and Tabs

Two current collectors are found inside the electrochemical unit cell. They are indispensable components that provide a low resistance path for electrons moving to and from the electrodes. The anode is usually placed on a copper foil. This foil then works as the anode current collector, which secures a link between the anode and the external circuit. Likewise, the cathode is normally coated with an aluminum foil to close the circuit. The aluminium foil collects the current, which goes to and from the cathode, and is therefore known as the cathode current collector. Current collector materials are chosen such that they are conductive and electrochemically stable under their working conditions [79, 80].

The tabs serve the purpose of collecting the current from all of the electrodes within a battery. Importantly, the tabs also provide two terminals, which can be connected to an external circuit, such that power can be delivered to a load. Thus, a single terminal is connected to all the electrodes of similar polarity. Copper that has been treated with nickel and aluminium is generally used for the negative and positive terminals, respectively [81].

## 2.5 Battery Figures of Merit

The following sections introduce some of the basic terminology and concepts that should be understood when modelling and studying LiB ageing. Understanding these terms makes the report, as well as similar literature, easier to follow.

### 2.5.1 Capacity

Capacity is one of the cornerstone terms used to describe batteries. Capacity is the term which is used to describe how many electrical charges are stored and made available for delivery before the voltage drops below a given cut-off value. In other words, the total capacity, measured in Ampere hours (Ah), dictates how many hours a battery can be discharged at a given discharge current. The capacity of a battery depends on the capacity of the active materials. In particular, the composition of the anode and cathode, which has to be optimised in order to obtain large battery capacities. Therefore, the battery capacity ( $Q$ ) can be expressed in Equation 2.2 as the time variant current ( $i$ ) which flows out of a battery during a full discharge, over a time frame ( $\Delta t$ ).

$$Q = \int_0^{\Delta t} i(t) dt \quad (2.2)$$

The term energy capacity is more familiar to the layman and is measured in units of watt hours (Wh). It can be found similarly to capacity by integrating the discharge voltage times the discharge current over a complete discharge. However, Ah is more widely used by battery

manufacturers to describe battery capacity. Ah can be a more convenient unit than Wh when voltage levels are determined beforehand. This is because the unit of Ah then allows easy calculations on the number of hours a given discharge current can be sustained.

### 2.5.2 C-rate

The normalised current rate, also called the C-rate, is used in the literature to compare battery performance regardless of capacity. The C-rate describes how fast or slow a battery charges or discharges and is determined by the ratio between the current ( $I$ ) and the battery capacity, as shown in Equation 2.3.

$$\text{C-rate} = \frac{I}{Q/1\text{ h}} \quad (2.3)$$

Equation 2.3 shows that a battery will fully charge in less than an hour if the C-rate is above 1C. Note the division of 1 h, which is semantics used to make the C-rate unitless. It should be noted that charging and discharging C-rates often are differentiated between since they can vary. The nature of the application of a battery determines if the battery must charge fast, i.e., charge at large C-rates, or discharge fast, or neither. It is, nevertheless, useful to differentiate between these current rates to avoid confusion. This also allows for mapping of the impact of using different charging and discharging currents.

A numerical example explains why the term C-rate is worth using. A given cell has an arbitrary capacity of 10 Ah and is cycling at 1 ampere. Another cell also cycles at 1 ampere, but has a lower capacity of 1 Ah. Compared to its capacity, the last cell faces a more strenuous load than the first cell. This is made clear by their respective C-rates, which are 0.1C and 1C. The data collected from this simple experiment cannot be used to compare their performances fairly. If, however, the C-rates of these cells are similar such that the cycle time is equal and the currents differ, the obtained data could be used to compare performances.

### 2.5.3 State of Charge, SoC Window and Depth of Discharge

State of Charge (SoC) is a term which defines the present energy, remaining within a battery. It is, in Equation 2.4, defined as the ratio of the residual, or remaining capacity to the maximum usable capacity ( $Q_{max.}$ ).

$$\text{SoC}(t) = \frac{Q(t)}{Q_{max.}(t)} \quad (2.4)$$

In this report, Equation 2.4 serves as a suitable definition of SoC. Some definitions of SoC however, refer to the nominal battery capacity. To further complicate the matter, battery manufacturers often choose their own reference points for SoC 0% and 100% SoC. This redefinition offsets SoC, such that the new 0% manufacturer SoC lies above the true 0% SoC. This is also done, admittedly in the opposite direction, for 100% SoC. These measures are taken to prevent customers from cycling in the upper and lower charge echelons, since this imposes an accelerated pace of ageing. However, this convention is disregarded in this report. To know of its existence is however important to avoid misconceptions [82].



In the real world, however, it is not so easily determined, as there are no ways of directly measuring the SoC within a cell. Therefore, a range of estimation techniques have been conceived. Espedal et al. wrote a comprehensive review on these techniques in 2021 [83]. This review is a recommended read for those who are interested in an update on current SoC estimation techniques.

A SoC range, or SoC window ( $\Delta\text{SoC}$ ) refers to the upper and lower SoC values between which a cell is cycled. How wide a range is, defines the amount of energy which can be accessed per cycle. A range which is 40% wide, is for instance used when 40% of the battery capacity is needed per cycle. A battery which, for instance, is cycled between a lower SoC value ( $\text{SoC}_{\text{lower}}$ ) of 30% and an upper SoC value ( $\text{SoC}_{\text{upper}}$ ) of 70% has a SoC range that is 40% wide. Ranges can both be wide and narrow, which respectively defines deep and shallow cycles. To find the capacity within a given SoC range,  $\Delta\text{SoC}$  can simply be multiplied with the present maximum capacity.

$$\Delta\text{SoC} = \text{SoC}_{\text{upper}} - \text{SoC}_{\text{lower}} \quad (2.5)$$

Depth of Discharge (DoD) is a measure that is interchangeably used to define both how discharged a battery is and how wide a SoC range is. For clarity the SoC range term is used to describe SoC windows in this report, whilst DoD is used to refer to how discharged a given battery is, such that 0% DoD implies an SoC value of 100%. The following relationship, presented in Equation 2.6, therefore follows between SoC and DoD.

$$\text{DoD}(t) = 1 - \text{SoC}(t) \quad (2.6)$$

#### 2.5.4 State of Health and Internal Resistance

A battery degrades as time passes, and therefore its capacity fades. It should therefore be noted that the capacity, which is observed at a battery's beginning of life (BoL), can develop to become significantly different from the nominal capacity ( $Q_{\text{nom.}}$ ). The fraction of these two quantities, i.e., current maximum capacity and nominal capacity, can be used to describe battery health, and is thus coined as the term State of Health (SoH) and is defined in Equation 2.7.

$$\text{SoH}(t) = \frac{Q_{\text{max.}}(t)}{Q_{\text{nom.}}} \quad (2.7)$$

This definition implies that a new, unaged battery has a state of health equal to 100% since the two capacity terms are equal. For example, a battery at 90% SoH has lost 10% of its nominal capacity. End of life (EoL) is defined when a battery has a SoH which makes it unusable. Where this value sits on the SoH scale differs between each battery application. Deviations on this scale between applications could therefore give a battery at its EoL, a chance at a second life in another application which is less restrictive with regards to SoH. EoL is nonetheless commonly defined after cell degradation accelerates from linear to nonlinear degradation [84].

The internal cell resistance ( $R_{c,i}$ ) can also be used as a proxy for SoH determination, due to internal resistance increasing with cell age. Thus, ageing can be indicated by comparing initial- and present cell resistances [85]. In this report, the simulation results on the cell equivalent circuit model resistances ( $R_{ECM}$ ) are used to describe the development of the internal cell resistance.

It has been assumed that they develop similarly, some terms are however only present in  $R_{ECM}$ , and not in  $R_{c,i}$ , which might introduce some inaccuracy.

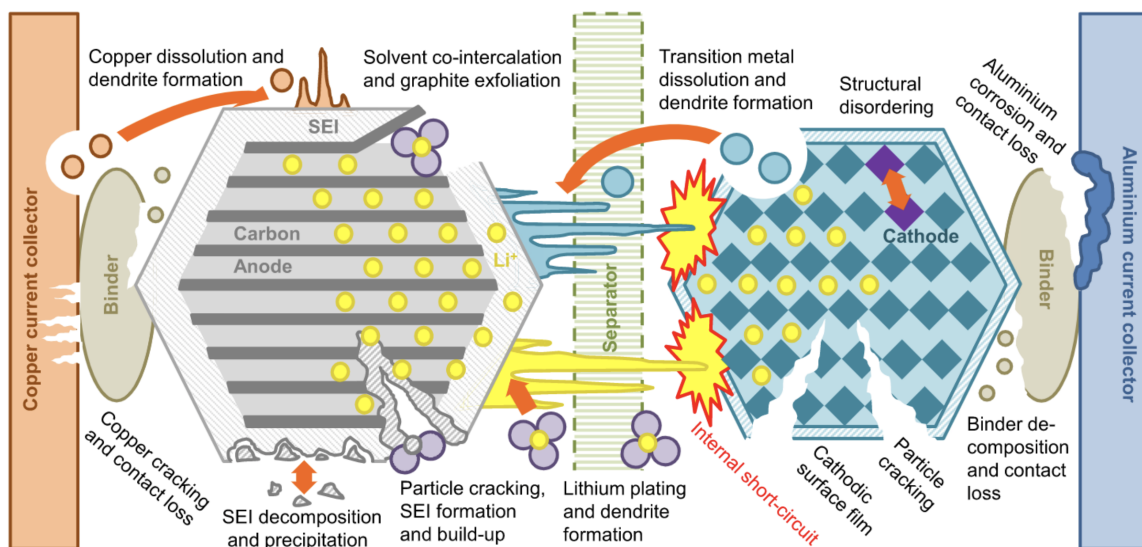
## 2.6 Degradation Factors and Mechanisms

LiBs age due to many factors, whether these are a subject of environmental effects or usage, they all result in degradation that effects different battery components [86, 87]. These degradation effects occur when a battery is either charging or discharging, or when it is at rest. This gives rise to two degradation modes which are referred to as cycle ageing and calendar ageing.

When a battery is at rest, irreversible internal chemical reactions occur. This consumes cyclable lithium-ions. As a result, the reactions develop a solid electrolyte interphase (SEI), which is known to be one of the main contributors to calendar ageing [88]. The rate of degradation at rest can be reduced by controlling storage conditions. In particular, SoC and ambient temperature. Even though the rate can be reduced, degradation is nevertheless unavoidable [89]. Secondary reactions, such as corrosion, occur at high temperatures. This induces loss of lithium inventory, which results in irreversible capacity fade. Colder temperatures limit this type of degradation but can induce other degradation mechanisms, which can lead to altered battery chemistry [87].

Cycle ageing occurs when a battery is either in charge or discharge mode. The factors discussed to effect calendar ageing also apply to cycle ageing. When the battery is in use, however, it is prone to exothermic reactions and large currents, which both produce heat. This increases the cell temperature, which in time causes battery ageing. Subsequently, it can be challenging to distinguish whether the degradation was caused by excessive heat or current [89].

The degradation of LiBs manifests itself as increased rates of self-discharge, capacity fade, and power fade, which is also referred to as impedance growth. As a battery ages, its usable capacity decreases, limiting how much energy can be stored within it. The internal resistance of a battery also increases with its age, which produces more losses in the form of heat, consequently less power can be delivered from the cell [54].



**Figure 2.14:** An illustration over the most common ageing mechanisms which occurs in LiBs [87].

There are wide variations in the lifespans of different types of batteries, designs, and technologies. Figure 2.14 illustrates some of the complexity which is involved in battery ageing, where components are shown to be affected by multiple different mechanisms, which all influences LiB longevity [54].

### 2.6.1 Solid Electrolyte Interphase

One of the major degradation mechanisms in LiBs is the growth of a SEI layer. The SEI is formed by the products of electrochemical reactions that occur when the electrode particles contact the electrolyte solvent [54, 90].

During the initial cycles the SEI layer forms. This consumes a large amount of lithium, and consequently capacity. After its formation, the SEI layer functions as a protective layer on the electrode. The SEI also reduces electrolyte decomposition, by impeding electron transport, while still allowing for lithium-ion transport, and intercalation [54, 90].

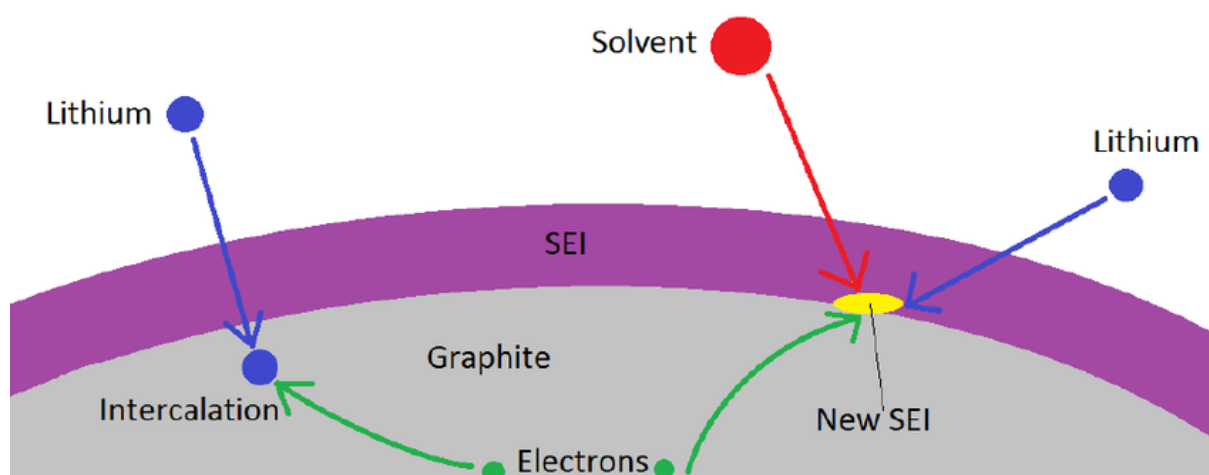
The formation of SEI progresses after its initial formation, resulting in continued capacity loss and structural changes in the electrolyte. The SEI growth rate is however reported to decline with time, such that the capacity loss from SEI growth slows down with time [54, 90].

Impedance growth has been reported to be linked with the thickness of the SEI layer, and in general degradation of electrodes. This additional resistance further decreases cycle life. Increased SEI layer thickness also adds resistance to lithium-ion transport. As the SEI grows, it consumes what was cyclable lithium-ions, such that capacity fades [54, 90]. To slow down SEI growth, root causes must be understood and, if possible, avoided. Root causes such as cycling with large charge currents, and high temperatures [54, 86].

Figure 2.15 shows that the SEI layer is permeable for lithium-ions, it also illustrates the redox reaction which takes place during charge, as shown in Equation 2.1. Figure 2.15 also illustrates the chemical reaction that leads to the formation of the SEI. Equation 2.8 illustrates how a general solvent, which exists in the electrolyte, as well as lithium-ions ( $\text{Li}^+$ ) and electrons ( $\text{e}^-$ ), are consumed in the formation of SEI.



SEI formation is irreversible, resulting in a permanent drain of lithium-ions, consequently, an irreversible degradation of electrodes and capacity loss [54, 86].



**Figure 2.15:** Illustration of the desired electrochemical reaction of lithium intercalation, and the side reaction wherein the electrolyte and lithium is consumed to form SEI [91].

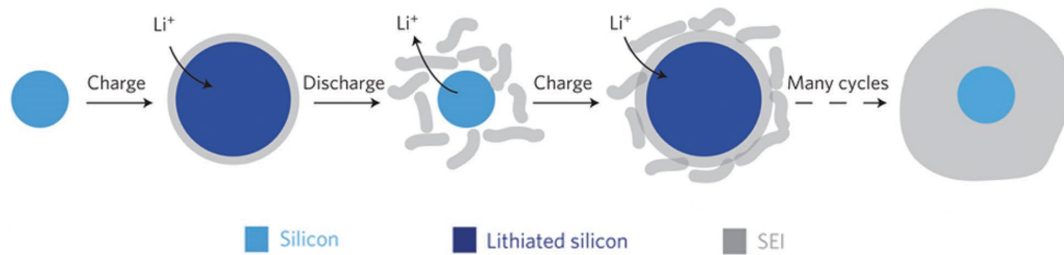
Both the negative electrode-separator interface and the positive electrode-separator interface forms layers. The impact on cell degradation is generally greater at the negative electrode. The cathode electrolyte interphase is therefore often overlooked when considering lithium-ion battery degradation [92].

### 2.6.2 Graphite Exfoliation

The negative electrode is commonly graphite based in LiBs, as mentioned in Section 2.4.5. Lithium-ions may not completely dissolve from the electrolyte solvent prior to entering the negative electrode during charge mode, when they are transported from the electrolyte, through the SEI layer, and into the negative electrode. If the lithium-ion is not fully dissolved, it can cause damage to the graphite layers. This process is known as graphite exfoliation. If the negative electrode is damaged, lithium-ions will be less accessible, or sometimes not accessible at all, which causes loss of active material, resulting in capacity fade and power loss. For graphite to exfoliate, the charge current must be high. The likelihood of exfoliation also increases with large states of charge [86, 93].

### 2.6.3 Particle Cracking

Significant mechanical stress can result from excessive currents, which causes volume expansion. This can cause particle cracking in the electrodes, consequently resulting in the creation of islands and further SEI formation. Islands are created when portions of the electrode becomes detached, resulting in the formation of an isolated nonactive island, this reduces the active electrode material, in addition to removing cyclable lithium-ions from it. As particles cracks, new regions are also directly exposed to the electrolyte, which allows for more SEI growth [94, 95]. Figure 2.16 is a graphical illustration of how the SEI layer grows due to particle cracking.



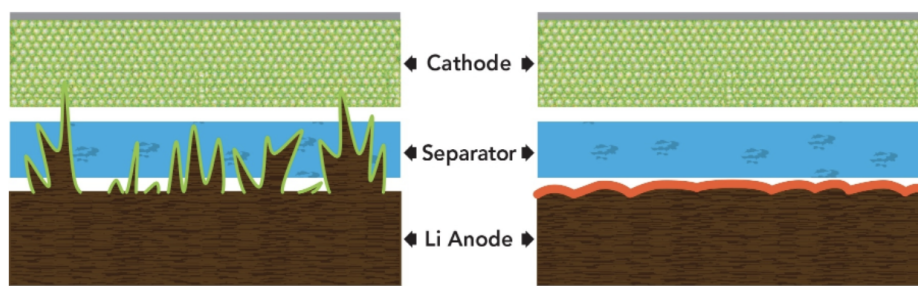
**Figure 2.16:** Illustration of a silicon anode in which particle cracking leads to new electrode regions being exposed to further SEI formation [95].

Figure 2.16 shows how a silicon anode gradually gets a thicker SEI layer. Note that this can happen to graphite anodes, which also experiences volume change during cycling. In addition, as outlined in the previous section, when the electrode material is directly exposed to the electrolyte, the SEI formation further consumes lithium-ions, which contributes towards more capacity fade [96].

#### 2.6.4 Lithium Plating

The formation of lithium dendrites contributes to loss of lithium inventory and safety issues. As lithium-ions move from the positive electrode to the negative electrode during charge, lithium-ions can accumulate onto the surface of the electrode particles. This happens when the rate at which lithium-ions are transferred to the negative electrode surpasses the rate at which they can diffuse into it. Rather than inserting themselves into the saturated negative electrode, lithium-ions which approach the negative electrode, can bond with the lithium metal on the surface. This process is known as lithium plating and is shown in Figure 2.17 [87].

A lithium metal fiber on the electrode surface can grow outward and form long, spiky stems, resulting in both degradation and safety concerns [79]. Safety concerns are due to the fact that lithium metal dendrites can puncture the separator, effectively creating an internal short circuit. As a result of the short circuit, large internal currents cause rapid temperature rise and potentially thermal runaway, fire, and explosions [54].



**Figure 2.17:** An illustration of lithium plating at its final degradation step, forming lithium dendrite formation penetrating the separator, causing an electrical short circuit [97].

### 2.6.5 Loss of Electrical Contact

Insertion and removal of lithium-ions cause volume expansions and contractions, especially when the DoD is great. Such volume variations can result in a reduction in electrical contact between the negative electrode material and the current collector [98]. It has, by Peyman Taheri et al., been shown that poor electrical contact leads to significant heat generation in lithium-ion batteries [88]. The loss of electrical contact can thus cause safety issues and trigger other degradation mechanisms. Additionally, if the battery is over-discharged, corrosion can occur on the current collectors [99]. By avoiding large depths of discharge cycles, i.e., low states of charge, better mechanical connections between the electrodes and current collectors can be sustained. This reduces the impact of the degradation caused by loss of electrical contact.

### 2.6.6 Other Degradation Mechanisms

The ageing mechanisms affects, not only the active materials of the anode and cathode, but also the electrolyte, separator, binder materials and current collectors.

- **Binder decomposition:** Under high temperatures and large SoC, the binder material in the electrodes reacts with the electrolyte and decomposes. Resulting in the loss of active electrode material [87].
- **Structural disordering:** In the process of charging and discharging, the positive electrode material undergoes phase transitions. Under high current rates, phase transitions cause structural changes in the active positive electrode material that causes the particles to misalign. Such structural changes can lead to the loss of active positive electrode material [99].
- **Electrolyte decomposition:** In high charging temperatures, with large SoC, the electrolyte can decompose and lead to the loss of cyclable lithium-ions [86].
- **Transition metal dissolution:** The transition metal ions found in transition metal cathodes can dissolve in the electrolyte and transfer to the negative electrode, resulting in the loss of positive electrode active material. Additionally, transition metal dissolution will cause lithium dendrite formation, as lithium-ions will be drawn toward transition metals deposited on the negative electrode [100].

## 2.7 Degradation Modes

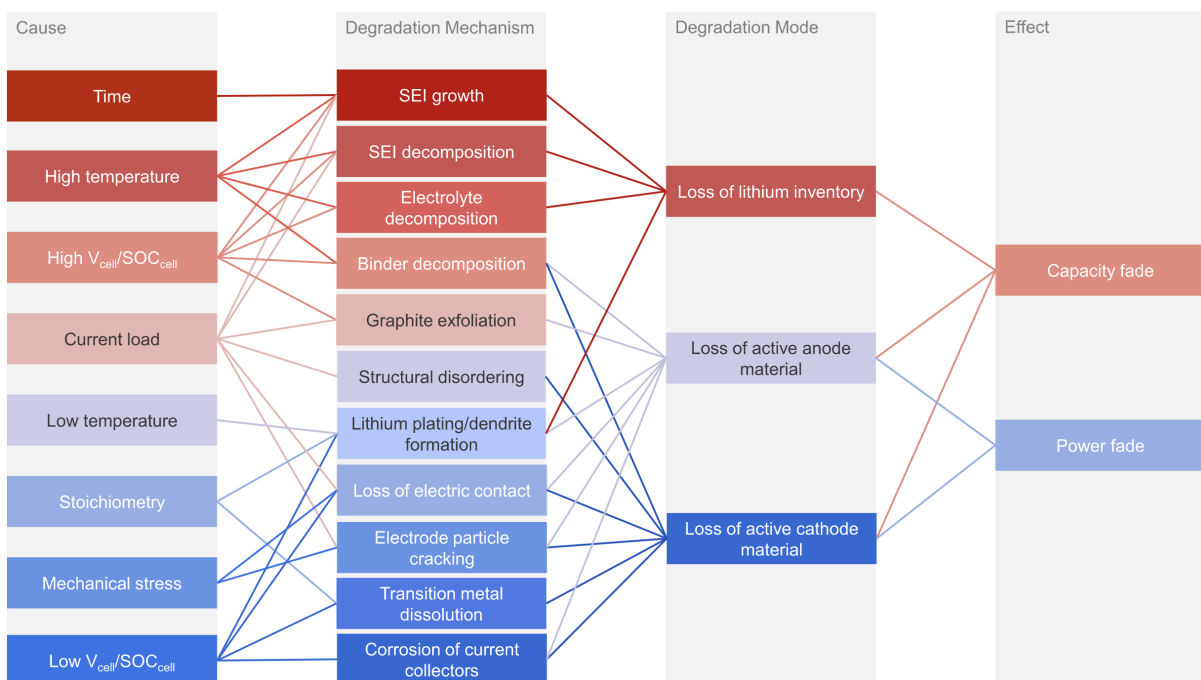
The effect of every degradation mechanism on the cell and its components are not uniform, rather several mechanisms can be grouped into degradation modes, which have different impacts on the different cell components [86]. Degradation mechanisms can be grouped into three degradation modes.

- **Loss of lithium inventory (LLI):** By parasitic reactions such as surface film formation, e.i., SEI growth, decomposition reactions, and lithium plating, lithium-ions are consumed, making them unattainable for cycling between the positive and negative electrodes. Additionally, surface films cause power fade. It is also possible to lose lithium-ions if they are trapped in electrically isolated particles [86].



- **Loss of active material of the negative electrode ( $LAM_{NE}$ ):** Due to particle cracking, loss of electrical contact, or the blocking of active sites by resistive surface layers, parts of the active mass of the anode is no longer available for the insertion of lithium. Both capacity and power can be affected by these processes [86].
- **Loss of active material of the positive electrode ( $LAM_{PE}$ ):** When it is no longer possible to insert lithium into parts of the cathode material, due to structural disordering, particle cracking, or loss of electrical contact. Throughout these processes, power and capacity fade [86].

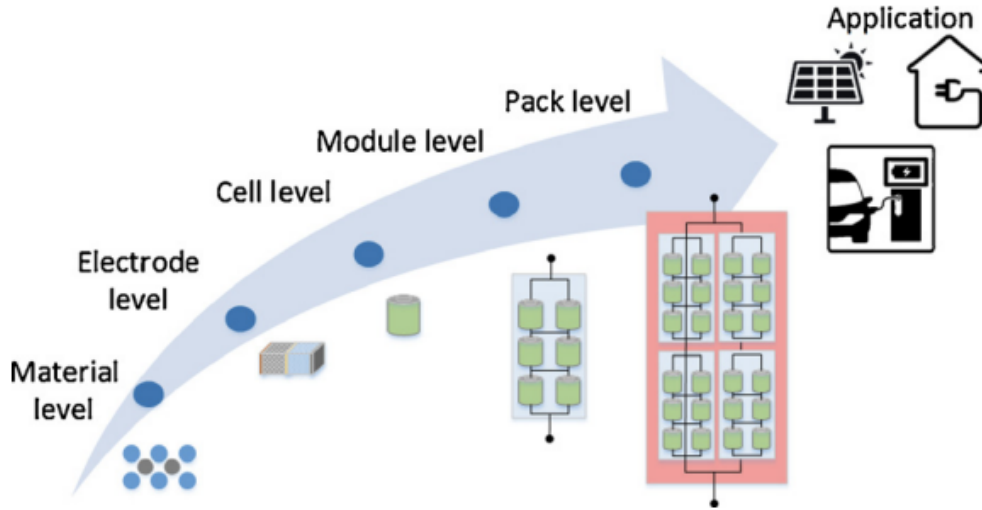
To summarize, a range of root causes with their coupled degradation mechanisms, grouped by degradation mode and effect, are illustrated in Figure 2.18. Lines between boxes indicate relationships between causes, mechanisms, modes, and effects. Blue boxes mainly impacts power fade, whilst red boxes impacts capacity fade.



**Figure 2.18:** Root causes that provokes several degradation mechanisms and modes associated with the two losses that occurs on LiB [86].

## 2.8 Degradation in Battery Packs

To recap from Section 2.3, a battery cell consist of unit cells, which are indicated by the electrode level in Figure 2.19. A battery cell typically consists of multiple unit cells which are either stacked together, like in a pouch cell, or stacked and rolled together, like in a cylindrical cell. Battery cells have limited capacity and power. Hence, for high power and energy applications battery cells are typically connected in series or parallel to form a battery module. The module thus has higher voltage and current output, as well as a larger capacity for storing energy. Larger system requirements can be met by combining battery modules into a battery pack. Figure 2.19, shows the steps from material to battery pack.



**Figure 2.19:** Scaling from the material level to the pack level [101].

At the cell level, ageing occurs during storage and utilization, as mentioned in Section 2.6. Capacity fade due to calendar ageing ( $Q_{cal.}$ ) is caused by the ageing of a battery in storage and is a function of the storage temperature ( $T_s$ ), the SoC of the battery, and time ( $t$ ), as presented in Equation 2.9. The degradation in battery capacity that originates from its usage ( $Q_{cyc.}$ ), is known as cycle ageing, and is, among other things, affected by ambient temperature ( $T_{amb}$ ), SoC, DoD, charge and discharge currents ( $I$ ), and the number of cycles ( $N$ ), as shown in Equation 2.10 [102].

$$Q_{cal.} = f(T_s, SoC, t) \quad (2.9)$$

$$Q_{cyc.} = g(T_{amb}, SoC, DoD, I, N) \quad (2.10)$$

There will always be differences within each cell in a module and in a pack, this makes modelling ageing difficult. Differences may be due to minute differences within the cell manufacturing process, or the different working conditions such as temperature [103]. In a study by Paul et al. the initial capacity of 20 000 fresh cells was fit to a normal distribution with a standard deviation of 1.3%. The internal resistances of these cells were also fit to a normal distribution with a 5.8% standard deviation [104]. This illustrates some of the differences that exist between cells within larger units.

In an integrated module or pack, these differences can lead to accelerated ageing in cells. For example, it has been reported that with a 20% degradation difference, in two cells connected in parallel, the two cells could experience up to 40% higher peak currents [105]. These increased currents result in an increase in the cell's heat production and subsequently increased ageing. Furthermore, the location of a cell within a module, or pack, has been found to affect heat dissipation and thus changes cell temperature [106]. Thus, ageing processes can be accelerated in some cells due to both elevated currents and elevated temperature.



There has yet to be developed a set way to estimate the degradation that occurs in cells inside a pack. A study by Cordoba-Arenas et al. found that the simplest method was to define the most aged ( $Q_{max.}$ ) cell's parameters for the whole pack and then calculate the pack capacity fade, by using Equation 2.11 [107]. Here,  $Q_{cell,i}$  represents the capacity loss in a given cell  $i$ , while  $N$  represents the number of cells.

$$Q_{max.} = \max(Q_{cell}, \dots, Q_{cell,i}) \in [1, N] \quad (2.11)$$

Equation 2.11 is exclusively used for series connected cells. With this method, the overall pack capacity is overestimated. In a different study, a probabilistic approach was applied for ageing estimation. Using this approach, the estimation error of the entire capacity of the pack was improved, although it still slightly underestimated the total ageing of the pack [103, 106].

### 3 Methods

The methods section elaborates on battery simulations, rapidly honing in on PyBaMM, its advantages, how it can, and why it should be used. Furthermore, the presented methods aspires to serve as a startup guide. Some simulation issues and their respective solutions are discussed such that they can be overcome with less of a struggle for future users of PyBaMM. Note that a rudimentary prior knowledge of Python is expected, the method therefore focusses the most on PyBaMM.

#### 3.1 Simulating Battery Cells and Battery Packs

Simulations are important tools which can be utilized to efficiently optimize both processes and general performance. In light of batteries, simulations can, for instance, be used to examine how different parameters, such as C-rate and ambient temperature, of a given battery cell effect ageing. Simulating the behaviour of battery cells can be less time consuming than conducting physical experiments. Simulations can also serve as a cost-effective alternative to real life experiments, especially when considering large battery packages. This paper focuses on LiB system performance in light of degradation and capacity fade. Simulations of batteries, however, have a much broader application such as exploring new design schemes and new battery chemistries.

#### 3.2 Battery Simulation Software

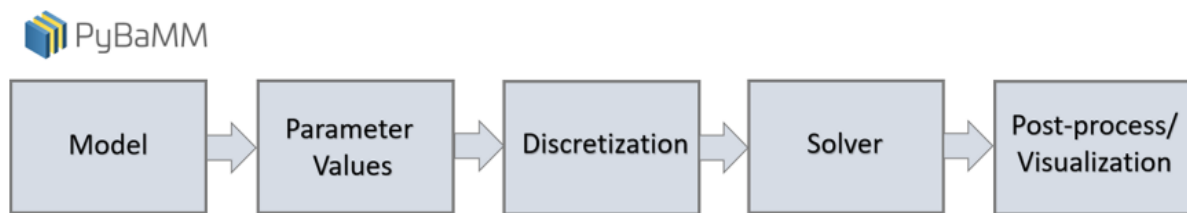
Batteries are today most commonly modeled with the COMSOL Batteries and Fuel Cell module. The main drawbacks of COMSOL with regard to battery modelling are its unforgivable licence fee and limited flexibility [108]. Expensive licence fees can be avoided with open-source software. Open-source is innately transparent, which means that the source code can be examined. Using open-source software can, therefore, with some effort allocated to understanding the source code, allow for a deeper appreciation of the underlying mechanics as well as a better understanding of how results are obtained and their limitations. Some open-source battery modelling software such as fastDFN, LIONSIMBA and M-PET exists. Yet these modelling tools also lack flexibility. They are, for instance, built with the focus on one specific battery model. PyBaMM was conceived to address this in order to accelerate battery research [108].

#### 3.3 PyBaMM

The Python package PyBaMM, which stands for Python Battery Mathematical Modelling, is an open-source tool built for fast, reliable, robust and flexible simulations of battery models. The package is built to be a modular framework in which battery models, parameter values, spatial discretisations, and solvers are set independently from one another. The PyBaMM developers have also made it easy to incorporate new functionality. Battery modellers are therefore not bound by the functionality that commercial developers implement. Anyone can for instance create their own models, and immediately use them within the existing framework. Battery models are created by combining submodels which describe their own particular part of the physics. This structure allows simple customisation of battery models by either swapping or adding submodels such that the desired physical phenomena can be modelled [108, 109]. PyBaMM can be run on Windows, Linux, and MacOS systems that have Python 3.7, 3.8 or 3.9

installed. Installation instructions are available on the PyBaMM website and in the PyBaMM repository on GitHub [110, 111]. The package is very much still under active development, and lately official releases have been monthly. A working development branch also exists, this can be cloned to be more up-to-date. PyBaMM simulations are on the battery cell level. The PyBaMM team is also working on a framework in which PyBaMM is the foundation used to simulate battery packages, this framework is called Liionpack. Liionpack is further elaborated on in Section 3.5.

Figure 3.1 shows the simulation process in PyBaMM. Users have control of each step in the process and can customise each component of the simulation to their liking. Additionally users can at any stage insert self-made components, which works together with the premade framework.



**Figure 3.1:** Steps to solve a battery model in PyBaMM [108].

The following sections elaborate on the simulation steps shown in Figure 3.1. The examples presented in the following listings have been constructed to show how simple it is to use PyBaMM. The examples were also designed to show the existing modularity and to give an introduction into how this can be leveraged to create highly customised simulations. The examples shows how to customize and create simulations which should fit the need for most PyBaMM beginners. There is, however, a lot more to the package than what these examples explore, and a lot of the complexity which can be introduced is not covered in these listings. Post-processing and plotting are left out as this can be done with Matplotlib, and built-in PyBaMM functions in Python, or be retrieved for further processing in MATLAB, for instance. This should be fairly familiar to users who are getting into battery modelling. Some parts of the discretization is left out for advanced users to explore for themselves. For further exploration of this, and more, the authors of this report would like to refer to the PyBaMM website, which both contains the PyBaMM documentation, as well as a series of enlightening training videos, and a getting started guide. The PyBaMM team’s GitHub repository, Google Colab pages and Slack channel, are all also great sources of information. PyBaMM could not have been utilized by the authors without these resources. The following examples were therefore made possible and heavily inspired by these resources. All of the named resources can be accessed either through the PyBaMM website or the team’s GitHub repository [110, 111].

### 3.3.1 Using Pre-Built Battery Models

A range of lithium-ion battery models are implemented by default in PyBaMM. Listing 3.1 shows that the pre-defined models can be initialised by calling `lithium_ion` class. Necessary imports are, as usual, found at the top of the Python script.

**Listing 3.1:** Initializing a pre-built battery model

```

1 import pybamm
2
3 # Examples of lithium-ion models implemented in PyBaMM:
4 # Single Particle Model
5 spm = pybamm.lithium_ion.SPM()
6
7 # Single Particle Model with Electrolyte
8 spme = pybamm.lithium_ion.SPMe()
9
10 # Doyle-Fuller-Newman
11 dfn = pybamm.lithium_ion.DFN()
12
13 # Newman-Tobias
14 nt = pybamm.lithium_ion.NewmanTobias()

```

### 3.3.2 Choosing Submodel Options

PyBaMM lithium-ion battery models are construed with submodels that describe their own particular part of the battery physics, together the submodels make up a coherent multiphysics battery model. Users of PyBaMM can set how a given submodel within a model behaves by altering its option. Written differently, users can set how a particular part of the physics in a battery model behaves. This can be done by altering the submodel options. The already implemented submodels allows for on the fly customization of pre-existing battery models. The submodel that describes lithium plating can for instance be set to `none`, `reversible` or `irreversible`. Changing the options of a submodel is simply done by creating a dictionary which states which submodel options to change, and which setting they should be set to. To know which submodels to change, one needs to know which submodels, and which options, exist. This can be done by inspecting the PyBaMM documentation and by printing out the submodels that make up a given lithium-ion model. Listing 3.2 shows how submodel options are set and implemented into a pre-defined battery model, in essence resulting in a customized Doyle-Fuller Newman model.

**Listing 3.2:** Setting submodel options

```

1 # The following code prints out the submodels which make up the DFN model:
2 dfn = pybamm.lithium_ion.DFN()
3 print(dfn.submodels)
4
5 # After looking through the submodels and the documentation, a
6 # dictionary with submodels and submodel options can be defined:
7 submodel_options = {
8     "intercalation kinetics" : "asymmetric Butler-Volmer",
9     "loss of active material" : "stress and reaction-driven",
10    "SEI" : "solvent-diffusion limited",
11    "SEI film resistance" : "distributed",
12    "SEI porosity change" : "true",
13    "particle mechanics" : "swelling and cracking",
14    "lithium plating" : "irreversible",
15    "thermal" : "lumped"
16 }
17
18 # Thereafter, the submodel options are fed into the optional keyword
19 # argument "options" in a given lithium-ion model:
20 custom_dfn = pybamm.lithium_ion.DFN(options = submodel_options)

```

### 3.3.3 Choosing Parameter Values

A range of parameters can be chosen. The parameter values describe the characteristics of the battery cell which is simulated. Parameters are either defined as constants or functions, and all pre-defined parameters can be found in CSV files in the PyBaMM GitHub repository. Parameters are chosen to describe which battery chemistry is simulated, which for the purpose of this paper is lithium-ion. Further parameters are chosen to describe the anode, separator, cathode, electrolyte, a charging or discharging experiment, SEI and lithium-plating properties. These categories of parameters are collected in what is known as parameter sets for ease of use. How parameter sets are chosen and edited is detailed in Listing 3.3. To further understand the role of parameters, one can inspect the many CSV parameter files in the PyBaMM GitHub repository. Looking further into the negative electrode parameters, for instance, reveals parameters which describe general electrode properties, electrode microstructure, interfacial reactions, density, thermal parameters, mechanical properties, crack model properties, and loss of active materials. Note that every parameter set does not cover every submodel. This is made evident with existence of the `Chen2020` and `Chen2020_plating`, wherein `Chen2020_plating` also includes the `yang2017_Li_plating` parameters [112–114].

**Listing 3.3:** Defining parameter values

```

1 # List of premade parameter sets in PyBaMM:
2 param_sets = ["Ai2020", "Chen2020", "Chen2020_plating",
3               "Ecker2015", "Marquis2019", "Mohtat2020",
4               "NCA_Kim2011", "ORegan2021", "Prada2013",
5               "Ramadass2004"]
6
7 # The Chen2020 parameter set is chosen as a base to modify from:
8 custom_params_set = pybamm.parameter_sets.Chen2020
9
10 # Printing out the Chen2020 parameter set,
11 # such that its parameters can be inspected:
12 print(custom_params_set)
13
14 # The parameter set is easily mutated to simulate
15 # different battery chemistries. If another cathode than
16 # the default cathode, the "nmc_Chen2020", is needed for
17 # instance, it can easily be replaced with another cathode like so:
18 custom_params_set["positive electrode"] = "nmc_ORegan2021"
19
20 # Once the parameter set has been aptly mutated to reflect what is wanted.
21 # The parameter set can be chosen by calling the ParameterValues
22 # class and setting the optional keyword argument "chemistry", which sets
23 # the parameter values for the simulation, like so:
24 custom_params = pybamm.ParameterValues(chemistry = custom_params_set)
25
26 # If one the other hand no changes are needed to the Chen2020 parameters:
27 chen2020_params = pybamm.ParameterValues(chemistry = pybamm.parameter_sets.Chen2020)
28 # If preferred parameters can be defined with the strings from param_sets:
29 chen2020_params = pybamm.ParameterValues(chemistry = "Chen2020")
30
31 # Previously, all the parameters associated with the "positive electrode"
32 # were changed. Look at the CSV files that describe positive electrode
33 # parameters for more information on the parameters that were changed.
34 # To change a singular parameter, first print out all the parameters and
35 # find which parameter to change and what to set them to. The following
36 # code shows how the ambient temperature is changed:
37 print(chen2020_params)
38 chen2020_params["Ambient temperature [K]"] = 263.15

```

```

39 # Parameters can also be changed with the update method:
40 chen2020_params.update({"Ambient temperature [K]" : 263.15})
41
42 # To simulate with the custom parameters, they are simply put into the optional
43 # keyword argument, "parameter_values", in the simulation class like so:
44 sim = pybamm.Simulation(dfn, parameter_values = chen2020_params)

```

Another method than the one shown in Listing 3.3 can be used to edit the parameter values. This method is most relevant for users who want to change and customise many parameters. Fully custom parameter sets can then be implemented by creating a CSV file which consists of every parameter and what the parameters are set to, respectively. The CSV file is then fed into the optional keyword argument, `values`, in `ParameterValues`. The simulation can then commence with unique parameters.

### 3.3.4 Choosing Standard Cell Geometries

Listing 3.4 shows how either pouch or cylindrical cell geometries are chosen in PyBaMM. Both pouch and cylindrical cell geometries are predefined in PyBaMM. This makes it very simple to simulate with these battery geometries. Further customisation of cell geometry is possible, but is not elaborated on in Listing 3.4.

**Listing 3.4:** Defining cell geometry

```

1 # Pouch and cylindrical cell geometries can be set by calling the
2 # battery_geometry convenience function and assigning the strings
3 # shown below to the optional keyword argument form_factor:
4 pouch_geometry = pybamm.battery_geometry(form_factor = "pouch")
5 cylindrical_geometry = pybamm.battery_geometry(form_factor = "cylindrical")
6
7 # Simulating with pouch geometry is done by inputting the geometry
8 # into the optional keyword argument "geometry" as shown below:
9 sim = pybamm.Simulation(dfn, geometry = pouch_geometry)

```

### 3.3.5 Simulation of Experiments and Logging Level

Experiments can easily be defined and simulated by using the `Experiment` class. In PyBaMM experiments are defined with tuples consisting of strings, as shown in Listing 3.5. These strings encode the operating conditions. The experiment defined in Listing 3.5 is defined for illustrative purposes and does not reflect a real life experiment. Experiments that last longer than one cycle can be defined with strings and then be multiplied by the number of cycles to be simulated. Multiple cycle types can also be combined as shown in Listing 3.5. Conditions for early termination of the experiment can also be set. This example experiment is set to stop after every instruction is completed or when 80% of the initial capacity remains.

When simulating computationally expensive experiments, which will take some time, it can be useful to have an idea of how the simulation is progressing. In these situations, the logging level can be leveraged by setting it to "NOTICE". This allows for easy inspection of simulation progress and error messages.

**Listing 3.5:** Setting logging level and simulating experiments

```

1 # The logging level is changed by calling:
2 pybamm.set_logging_level("NOTICE")
3 # "DEBUG" logging level could also be useful
4
5 # The following example experiment showcases acceptable string formats,
6 # termination criterion, cycle combination, and cycle multiplication.
7 arbitrary_experiment = pybamm.Experiment(
8     [
9         "Charge at 1C for 2 hours",
10        "Charge at 200 mW for 45 minutes or until 4.2 V",
11        "Hold at 4.2 V for 50 seconds",
12        "Hold at 4.2 V until C/50",
13        "Discharge at C/20 for 0.5 hours",
14        "Discharge at 1 A for 90 seconds",
15        "Rest for 1 hour"] * 40 +
16     [
17        "Charge at 3C for 30 minutes",
18        "Discharge at 5C for 10 minutes"] * 20,
19     termination = "80% capacity"
20 )
21
22 # The defined experiment is then simulated by feeding it
23 # into the optional keyword argument experiment like so:
24 sim = pybamm.Simulation(dfn, experiment = arbitrary_experiment)

```

### 3.3.6 Choosing Solver and Solver Options

After a simulation is set up it has to be solved in order to obtain results. Both solver options and solver can be specified before solving the simulation. The solver tolerances can be changed from the default value of  $10 \cdot 10^{-6}$ . To obtain more accurate results, the tolerances are tightened and the opposite is done to save solve time, but it is done at the cost of reduced accuracy. It can be a good idea to try several solvers in order to solve the simulation scheme in an efficient manner. Refer to the documentation for the different solvers that have been implemented in PyBaMM. In the example presented in Listing 3.6 the CasADi solver is chosen for its speed, to further speed up the simulation the absolute and relative tolerances have been made less restrictive.

**Listing 3.6:** Setting solver and solver options, and finding simulation time

```

1 # The CasADi solver is chosen, and solver options are set. Possible solver
2 # modes are presented in the documentation for each solver:
3 solver_fast = pybamm.CasadiSolver(atol = 1e-4, rtol = 1e-4, mode = "fast")
4
5 # Next, the specified solver is implemented into the simulation by assigning
6 # it to the optional keyword argument "solver" in the Simulation class:
7 sim_fast = pybamm.Simulation(dfn, solver = solver_fast)
8
9 # The simulation can now be solved by calling the solve method.
10 # Simulations can be defined as custom experiments and drive
11 # cycles. The optional keyword argument "t_eval" can also be
12 # used to simulate constant current discharge at the value of the
13 # Current function [A] parameter. (Set to 1C by default.) "t_eval" sets
14 # the duration with an initial and final time in seconds:
15 sim_fast.solve(t_eval = [0, 3600]) # 1 hour
16
17 # Simulation time can be printed out with the following code:
18 print(f"Solve time: {format(sim_fast.solution.solve_time)}")

```

### 3.3.7 Example of a Simple Experiment Simulation and Citations

The example presented in Listing 3.7 follows the simulation steps presented in Figure 3.1. Little to none customization is done in this example. The example therefore highlights how simple it is to set up a basic simulation in PyBaMM. Discretization and post-processing/visualisation are left out of the example. The developers of PyBaMM has also created a convenience function which prints out citations. It is elaborated on at the end of Listing 3.7.

**Listing 3.7:** Solving a basic experiment simulation

```

1 # Choosing the DFN model:
2 dfn = pybamm.lithium_ion.DFN()
3
4 # Choosing the Chen2020 parameter set:
5 params = pybamm.ParameterValues(chemistry = pybamm.parameter_sets.Chen2020)
6
7 # Selecting the predefined cylindrical geometry:
8 cylindrical_geometry = pybamm.battery_geometry(form_factor = "cylindrical")
9
10 # Choosing a fast solver:
11 solver_fast = pybamm.CasadiSolver(atol = 1e-4, rtol = 1e-4, mode = "safe")
12
13 # Defining an experiment to solve:
14 experiment = pybamm.Experiment([
15     ("Discharge at 2.5C until 3V",
16     "Rest for 5 minutes",
17     "Charge at 2.5C until 4.2V",
18     "Hold at 4.2V until C/15")] * 50)
19
20 # The simulation can now be defined:
21 sim = pybamm.Simulation(dfn, parameter_values = params, solver = solver_fast,
22     experiment = experiment, geometry = cylindrical_geometry)
23
24 # Solving the previously defined simulation:
25 sol = sim.solve()
26
27 # Finally, the citation process can be simplified
28 # by calling the PyBaMM function, which prints out
29 # the research papers that have been used
30 # to put the simulation, which have just been
31 # defined, together. Simply, call print_citations:
32 pybamm.print_citations()
33
34 # This report is written with LaTeX, it is therefore
35 # convenient that PyBaMM can print out the citations
36 # in different formats, such as BibTex:
37 pybamm.print_citations(output_format = "bibtex")

```

### 3.3.8 Choosing Mesh Resolution

In PyBaMM simulations the geometry is meshed and it is split into different subdomains. The mesh resolution which is used in a simulation can be changed in a similar manner as to how submodel options are set. Mesh resolution can be reduced to make a simulation faster. On the other hand, meshes are made finer to acquire more accurate solutions, at the expense of computing time. Listing 3.8 shows how the mesh resolution is increased and implemented in a simulation.



**Listing 3.8:** Defining a finer mesh

```

1 # Start by choosing a model and printing out the
2 # default number of points in the model's mesh:
3 dfn = pybamm.lithium_ion.DFN()
4 print(dfn.default_var_pts)
5
6 # The mesh can be made finer by increasing its number of points.
7 # This is done by defining a dictionary which contains which parts
8 # of the mesh should be made finer. A dictionary "var_pts" which
9 # sets the number of points is defined like so:
10 var_pts = {
11     "x_n" : 40, # x-direction, length, negative electrode
12     "x_s" : 40, # x-direction, length, separator
13     "x_p" : 40, # x-direction, length, positive electrode
14     "r_n" : 40, # number of volumes in the radial direction, negative particle
15     "r_p" : 40, # number of volumes in the radial direction, positive particle
16     "y" : 40, # y-direction, depth
17     "z" : 40, # z-direction, height
18     "R_n" : 30, # negative particle radius (kept at default value)
19     "R_p" : 30 # positive particle radius (kept at default value)
20 }
21
22 # To simulate with the finer mesh it is passed to
23 # the optional keyword argument "var_pts" like so:
24 sim = pybamm.Simulation(dfn, var_pts = var_pts)

```

### 3.3.9 Choosing Submesh Types

In Section 3.3.8 the mesh resolution was made finer. This is a totally acceptable method to obtain more precise results. The method can be incorporated in a mesh refinement study, where the number of points in each subdomain is gradually increased until the results converge within a given threshold of change. There are also more targeted solutions to improve the accuracy of the results. Mesh types can be defined for each subdomain of the cell geometry. A range of submesh types can be intelligently chosen for given parts of the geometry to get more accurate results, some of these types are presented in a list in Listing 3.9. The dimensionality of the mesh is chosen in addition to mesh type. The 0D "meshes" depend on x, 1D on x and z, and 2D on x, z, and y, as defined in Listing 3.8. Exceptions to this being the negative and positive particles which are defined radially, and the current collectors, which do not depend on x. In the exponential submesh for instance, the points are concentrated towards one or both of the boundaries, depending on submesh parameters. This submesh type can in other words be leveraged to form a non-uniform grid which has a higher resolution towards the most interesting parts, i.e., the surface, where the most change or largest gradients are found. The less interesting parts, where smaller gradients are found, can be set to have rougher grids. This can improve result accuracy in an efficient manner if implemented correctly. Listing 3.9 shows how submesh types are defined for the negative particle subdomain.

**Listing 3.9:** Defining submesh types

```

1 # List of some submesh types implemented in PyBaMM:
2 submesh_examples = ["Uniform1DSubMesh", "Exponential1DSubMesh",
3                     "Chebyshev1DSubMesh", "ScikitUniform2DSubMesh"
4                     "ScikitExponential2DSubMesh", "ScikitChebyshev2DSubMesh"]
5
6 # Start by initialising the default mesh. The default mesh contains all
7 # the subdomains and their respective submesh types:

```

```

8 submesh_types = dfn.default_submesh_types
9
10 # To examine which submesh types are assigned to which subdomain by
11 # default, simply print out submesh_types:
12 print(submesh_types)
13
14 # Next change given submesh types within the mesh. The following code
15 # replaces the mesh for the negative particle(s) with an exponential
16 # mesh. New submeshes can be set for specified subdomains with the
17 # MeshGenerator class by defining which submesh type is to be
18 # generated and by inputting the parameters that are required by that
19 # particular submesh. The submesh parameter "side" is defined to be right
20 # such that the grid is the finest towards the particle surface, and
21 # less so towards the particle centre:
22 submesh_types["negative particle"] = pybamm.MeshGenerator(
23     pybamm.Exponential1DSubMesh, submesh_params = {"side" : "right"}
24 )
25
26 # Finally, once the subdomains are meshed as wanted, they are incorporated
27 # into the simulation with the optional argument "submesh_types". Note that
28 # the previous customisation from Listing 3.8, which made the grid finer in
29 # general, can work both alone and in tandem with submesh type specification:
30 sim = pybamm.Simulation(dfn, var_pts = var_pts, submesh_types = submesh_types)

```

### 3.4 Issues and Troubleshooting in PyBaMM

This section has been constructed with the purpose of highlighting how to troubleshoot, as well as some of the issues that the group encountered and eventually overcame when using PyBaMM. Most issues can usually be resolved fairly easily by inspecting the traceback. A traceback, or stack trace, is displayed when Python detects an error and an exception gets raised. The stack trace contains exceptions, error messages, and shows the list of calls which the code were in the middle of when something went wrong. Figure 3.2 shows a typical Python traceback. Effectively utilizing the information presented in the traceback in combination with the documentation should allow for swift corrections of most issues.

```

-----
NameError                                Traceback (most recent call last)
Input In [3], in <cell line: 3>()
      1 # # # UTILITY SECTION # # #
      2 # setting the logging level to notice for visualization of simulation progress:
----> 3 pybamm.set_logging_level("NOTICE")
      7 # # # SIMULATION SETTTINGS SECTION # # #
      8 # defining important variables for the experiments which are conducted.
      9 # how many cycles for each experiment, which SoC ranges to test and
      (...)
     14 # will be created. It can be ID'd by its value.
     15 # first value: charging C-rate, second value: discharging C-rate
     16 C_rates = np.array([[0.1, 0.1], [0.5, 0.1], [1.0, 0.1], [2.0, 0.1], [4.0, 0.1],
     17                     [0.5, 0.1], [1.0, 0.1], [2.0, 0.1], [4.0, 0.1], [0.5, 0.5],
     18                     [1.0, 1.0], [2.0, 2.0], [4.0, 4.0]])

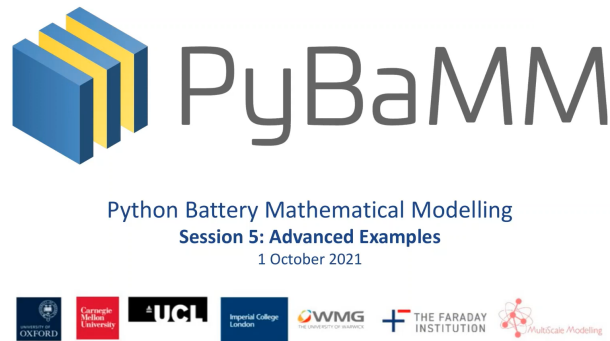
NameError: name 'pybamm' is not defined

```

**Figure 3.2:** Typical Python traceback displayed due to a NameError.

The information presented in Figure 3.2 allows for two possibilities. One; either PyBaMM has not yet been imported, or two; PyBaMM has been imported, but the keyword `as` was used

when the module was imported to give it another name. Other issues, which might not be so readily resolved often has multiple videos, forum posts, explanations, and solutions made about them online. The PyBaMM module is, however, quite new. The vast number of questions and answers one anticipates finding online, therefore, simply does not exist. Users of new software must utilize what is made available to them, such as the resources presented at the end of Section 3.3, to solve issues. Figure 3.3 shows one of the resources which can be used to gain a better handle on PyBaMM.



**Figure 3.3:** Image from a PyBaMM training video workshop series [110].

Some issues, such as memory management, were uncharted territory for the group at large. Having a better understanding of the root cause of a problem, as well as possible solutions and their implementations, could save hours of debugging and digging through old abandoned forums with partly related posts and discussions. Sections 3.4.1 through Section 3.4.4 were put together in an effort of effectively aiding future PyBaMM users.

### 3.4.1 Memory Management

Listing 3.7 shows how a simple experiment can be conducted. The example experiment only consists of 50 cycles. Long experiments with thousands of cycles, experiments meant to model a cell through its entire lifespan, for instance, can also be conducted. The amount of data which gets stored in the RAM throughout these experiments might get insurmountable. This can however be addressed. Listing 3.10 shows PyBaMM's built in solution. It should reduce memory usage in long experiments.

**Listing 3.10:** Using an optional keyword argument in solve to reduce memory usage

```

1 # save the 1st, and thereafter every 5th cycle:
2 sol = sim.solve(save_at_cycles = 5)
3 # save the 1st, 100th, 250th, 300th and 500th cycle:
4 sol = sim.solve(save_at_cycles = [100, 250, 300, 500])

```

In Listing 3.10 two alternatives for the same memory usage reduction technique are shown. Memory is simply freed by specifying which cycles to save. This can be achieved through the usage of an integer. When this is done, only cycles which are an integer multiple of that number will be saved. Another alternative to this is using a list which includes the specific cycles which should be saved. The first cycle is always saved, regardless of how the keyword argument `save_at_cycles` is specified.

Due to the nature of how some of the simulations in this report were conducted, the solution presented in Listing 3.10 was not viable. The solution found in Listing 3.10 is in other words

not compatible with every simulation method. Therefore, a custom memory clearing function was constructed which seeks to replicate what the original solution does. This function can be found in Listing A.4 in Appendix .

### 3.4.2 Linesearch Algorithm

The group found "The linesearch algorithm failed with too small a step" to be a reoccurring solver error. A simple solution was found to handle this. Change the convergence criterion of the solver as shown in Listing 3.11.

**Listing 3.11:** Changing solver tolerances

```

1 # default tolerance settings
2 solver = pybamm.CasadiSolver(atol = 1e-6, rtol = 1e-6)
3
4 # loosened tolerances, preferably lowered just enough for the solver
5 # to converge such that the least amount of accuracy is lost
6 solver = pybamm.CasadiSolver(atol = 1e-3, rtol = 1e-3)

```

Listing 3.11 shows how this error can be overcome by loosening the absolute and relative tolerances. For more information on the solvers implemented in PyBaMM and how they can be customised, consult the documentation [110]. Since tolerances control the maximum error allowed per step in the simulation [115], it is, in order to obtain the most accurate results, advised to loosen tolerances as little as possible.

### 3.4.3 Minimum and Maximum Voltage

The minimum and maximum voltage error is raised when voltage boundaries of a cell is exceeded. This error can come down to not properly controlling the operating conditions within an experiment which is being simulated, such that the voltage exceeds a given cut-off value. This can happen, for example, if the operating condition of an experiment instructs a charge until the cell voltage is above the upper cut-off voltage. The parameters which control the cut-off levels can however, if required, be changed from the default values to simulate custom operating voltages. It should, however, be noted that the default cut-off voltages of a cell are set such that an appropriate amount of lithium-ions are always found in both electrodes. Cycling within the boundaries of the default operating voltage, which is set by the upper and lower cut-off values, prevents the formation of unstable structures. Listing 3.12 nonetheless shows how the operating voltage can be customised.

**Listing 3.12:** Changing the lower- and upper cut-off voltage

```

1 # lowering the cut-off voltage to an
2 # arbitrary value of 1.8 V
3 params.update({"Lower voltage cut-off [V]" : 1.8})
4
5 # increasing the cut-off voltage to an
6 # arbitrary value of 4.9 V
7 params.update({"Upper voltage cut-off [V]" : 4.9})

```

### 3.4.4 IDACalcIC Unable to Recover

The group experienced that higher spatial resolutions, which were obtained by increasing the mesh resolution in both the radial and in the x-direction curbed several solver errors such

as "The residual routine or the linear setup or solve routine had a recoverable error, but IDACalcIC was unable to recover" and "IDA\_NO\_RECOVERY". After correspondence with one of the developers, it was suggested that the condition number of the problem gets slightly reduced by increasing the mesh resolution. In essence this means that the degree to which errors in the input of a function affect errors in the output is possibly slightly reduced by increasing the mesh resolution. Listing 3.8 shows how mesh resolution can be increased. Its effectiveness as a technique for solving solver issues cannot be understated, and whilst utilising this solution comes at the benefit of increased result accuracy, it can also drastically increase solve time.

### 3.5 Liionpack

Liionpack is similar to PyBaMM an open-source Python package. The package can be leveraged to scale up PyBaMM simulations to the battery pack level. Liionpack allows its users to specify battery pack configurations through netlists or by specifying the number of cells which are connected in series and parallel. It is of particular interest to note that the parameters are directly inherited from PyBaMM, and that experiments are conducted in the same manner. Liionpack accounts for the cell position within the pack, and a built-in function can be used to easily visualise the differences between each cell in the pack.

PyBaMM is, as previously mentioned, still very much in active development. This is also the case for Liionpack, which at the time of writing this report is still in beta testing. Listings with examples on how to use Liionpack have therefore not been constructed as these more than likely would be outdated quickly, and would consequently be of no help to beginners, as the PyBaMM listings aspires to. Therefore, the authors of this report would like to refer to the Liionpack GitHub repository for installation guidance, access to the source code, code documentation, and likely more up-to-date examples than what the listings could have provided for [116].

## 4 Results and Discussion

The results and discussion start off by following the PyBaMM pipeline, which is shown as five distinct steps in Figure 3.1. The first step in every PyBaMM simulation is defining the base model, and its submodel options. Next parameter values are defined. Thereafter, the discretization of the model is set, which leaves the last step of interest, i.e., choosing a solver and its solver options. To determine simulation settings for each step, quantitative methods, such as refinement studies, were applied. It was however not practical to employ these techniques for every step in Figure 3.1. Thus, reasoned, and qualitative, assessments and decision making had to be employed. After the model is constructed result oriented simulations are conducted, the model is iterated upon through troubleshooting to curb unexpected behaviour, subsequently a set of final results are obtained.

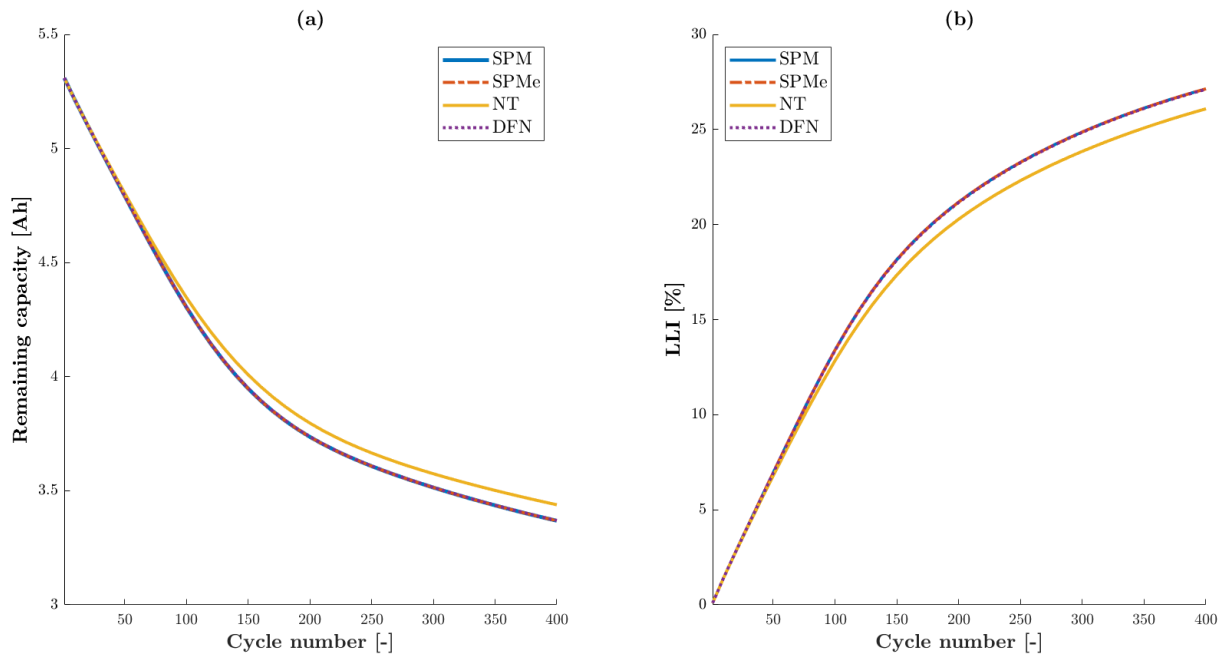
### 4.1 Base Lithium-ion Model Study

A comparison of the relevant battery models implemented in PyBaMM were conducted to explore trends in differences of results as a function of base model. The results of the most significance to this report were examined. Making a table which shows the average result differences, as found in Section 4.4 and Section 4.5, is not the most effective way to determine the practical differences between base models. The differences between models were therefore chosen to be visualized in Figure 4.1, Figure 4.2, and Figure 4.3, such that the solutions produced by each model could be easily broken down. Table 4.1 was produced to highlight differences in solve time. Note that 400 cycles were conducted in this study. This was done to examine if the eventual differences grew over time. Optimally thousands of cycles, and multiple simulation types, should have been computed, this was however unfeasible as the authors were limited by both memory capacity and computing power.

**Table 4.1:** Base battery model solve time comparison

Base model	Solve time [s]	Time increase [%]
SPM	2909	-
NT	2993	2.888
SPMe	3222	7.651
DFN	60 921	1790

The solve times from Table 4.1 are not that interesting in of themselves, as this time varies with the hardware which is utilized to run the simulation. The differences in solve time between models, are however of interest. From Table 4.1 one can, as expected, observe that the least complex models are the fastest. The Single Particle Model with Electrolyte, is also as expected, slower than the less complex standard SPM model. The most popular model, in the battery modelling community, the Doyle-Fuller-Newman model, is the most computationally heavy base model which is implemented in PyBaMM, and it is consequently, by far the slowest. For this simulation it took approximately 17 hours, which was approximately 21 times longer than the fastest model. The most important results from the base model study are plotted in the following figures.

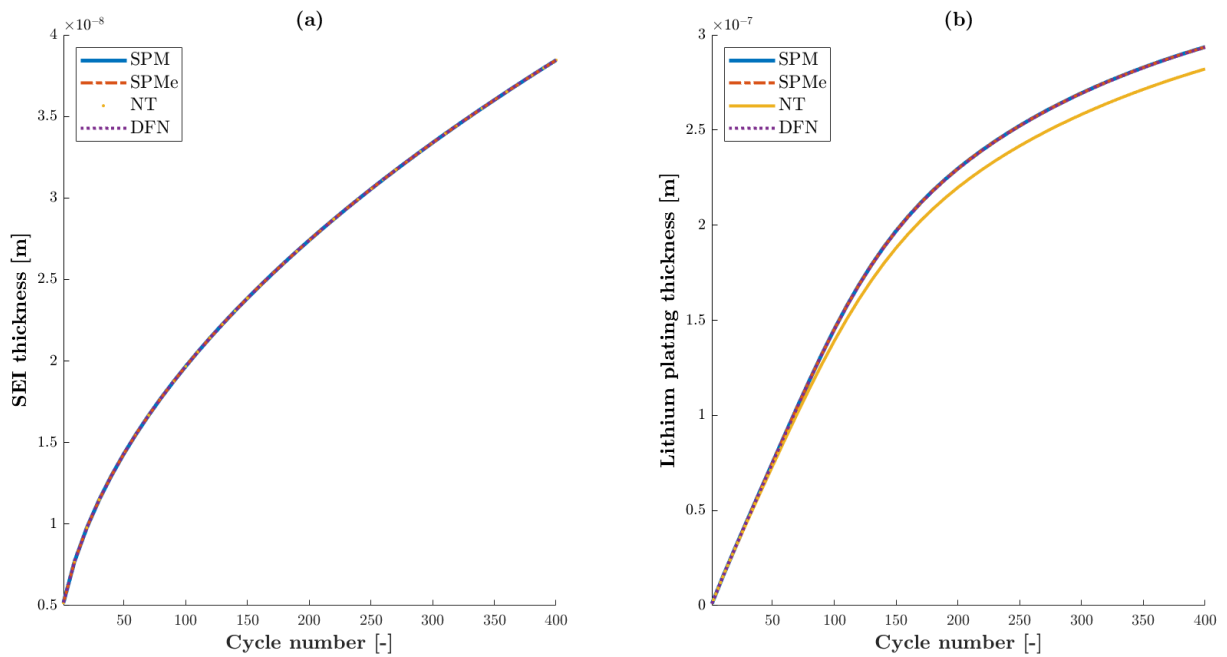


**Figure 4.1:** (a) shows calculated reduction in cell capacity over 400 cycles by base model. (b) shows the calculated development of LLI over 400 cycles by base model.

Figure 4.1 shows the simulated results on capacity fade and loss of lithium inventory in a LiB cell. All the paths which the capacity follows in Figure 4.1a seem to have unnaturally large slopes for the first 200 cycles, before it comparatively levels off. Why these slopes are found, is discussed at length in Section 4.7. The SPM and SPMc models agree with the computationally expensive DFN model. For this report, this somewhat undermines the legitimacy of choosing to simulate with the DFN base model, as this simulation approximately took 16 hours longer than the SPM and SPMc simulations, which only needed an hour to largely get the same results. The Newman-Tobias base model overestimates capacity retention in Figure 4.1a, and accordingly underestimates LLI in Figure 4.1b. This is, by itself, a rational result since more lithium should be cyclable at higher capacities. Considering the agreement of the other base models, and the extensive simulation time of the DFN model, the overall NT results are however deemed, to most likely, be of the least accuracy. Figure 4.2 shows the individual base model results on average SEI and lithium plating growth.

Figure 4.2 shows that the results on average SEI growth coincides for all the base models. The average lithium plating results are also similar, considering the exponent of the y-axis in Figure 4.2b. NT, is however once again the outlier in Figure 4.2b. The NT result presented in Figure 4.2b implicates that less lithium is exploited by lithium plating. The amount of lithium which is freed up from this could explain the discrepancy between NT and the other base models in capacity and LLI, as shown in Figure 4.1. The coinciding findings presented in Figure 4.2 could further be explained with how submodel options and parameter values were chosen. As they, in this study were left unchanged between base models. This was purposefully done to convey the differences between base models, and not the differences between different submodel options and parameter values. It should however be noted that such studies could provide valuable information. Due to computational constraints, a more practical approach, elaborated on in Section 4.2 and Section 4.3, was chosen to determine the submodel options and parameter





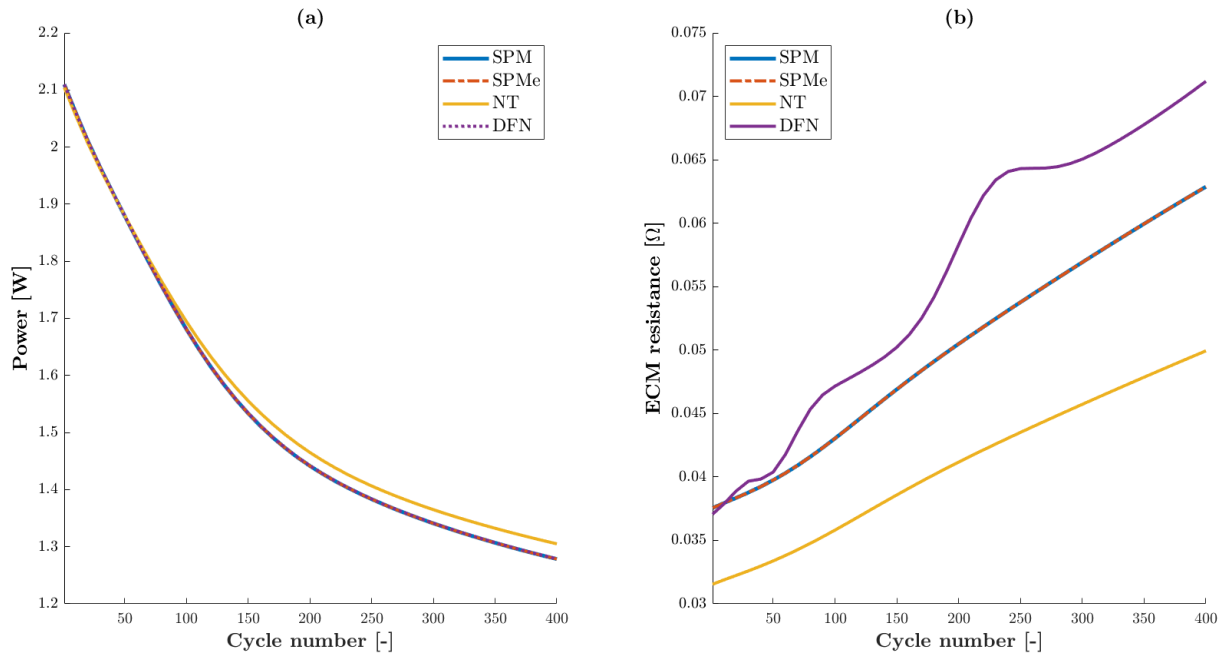
**Figure 4.2:** (a) shows calculated SEI growth by base model for 400 cycles, and (b) shows the calculated development of lithium plating over 400 cycles by base model.

values which were used in this report. For Figure 4.2 it is of particular interest to note that the same SEI and lithium plating parameter values- and submodel options were set, which inevitably produced similar results. Yet, NT produced a dissimilar result in Figure 4.2b. Figure 4.1 supports this deviation, and points towards the base models interacting differently with the customized submodels. In other words, the uniqueness of each base model could provide the common SEI and lithium plating submodels with unique inputs such that different results are observed. The outputs from these models are also treated differently by the base models. Figure 4.3 shows a result, wherein the complexity of the DFN develops a more dynamic result path, than the other base models throughout the simulation.

In Figure 4.3a results on cell power generation coincides with previous results from Figure 4.1 and Figure 4.2, in which similar deviations were observed for the NT base model. Figure 4.3a show a larger power result for the NT base model, which is to be expected due to its previous results which showed lessened ageing. Figure 4.3b however shows a most interesting result on the equivalent circuit model resistance of the LiB cell. This results highlights some of the difference between the fast base models and the DFN base model. A linear relationship is observed between ECM resistance and number of cycles for the fast models. The DFN model result however, has a more cyclic development, wherein the gradient switches between being larger and smaller than the other base model results. It would be of great interest to see if the simpler and faster models, at one point meaningfully starts to deviate from the DFN calculated resistance. This could however not be determined by the authors, owing to the vast computational power this would require.

When observed independently all the NT results, have thus far been coherent, but they have also often been observed to not completely agree with the results generated from the other base models. It is nonetheless considered appropriate for most of the simulations in this report, since its results are accurate enough to convey the same general trends as the other base models.





**Figure 4.3:** (a) shows the cell power over 400 cycles by base model (b) shows the equivalent circuit model resistance for a LiB cell over 400 cycles by base model.

The NT base model solves quickly, its biggest advantage however, is that the least number of errors and issues have followed when using this base model at large C-rates. On the downside the particle mechanics and loss of active material submodels are only stated to work with the SPM, SPM<sub>e</sub> and DFN base models [117]. Both SPM models are also considered appropriate for the purpose of this report, as the results generated from these models largely produced the same results as the DFN model, in this study. The SPM<sub>e</sub> is however preferred over the SPM, as it takes account for changes in the electrolyte. Preferably the authors of this report would have liked to use the DFN model. This was however deemed too computationally expensive for continued use. It should however not be completely ruled out based on this, as it should reign supreme in result accuracy. To not give the wrong impression, it should also be noted that the DFN model is implemented well, and normally solves very quickly in PyBaMM. This is however not the case for the simulations in this report, as the chosen submodels drastically increased complexity and compute time. To show this one cycle of a simple thirty minute charge and discharge experiment were solved with the DFN base model. The first experiment was solved with the base model, the second expanded the DFN with chosen parameter values, and the third experiment included both parameter values and chosen submodel options. The results which were obtained from the experiments are presented in Table 4.2.

**Table 4.2:** The effect of adding custom parameters- and submodels on solve time

	Standard	Selected parameters	Selected submodels
Solve time [s]	0.8	1.2	38.9
Solve time increase [%]	-	50	3141

Table 4.2 shows how complexity increases solve time. This effect has also been observed to grow as the number of cycles increases, such that the solve time for late cycles grows from seconds to multiple minutes. Even though the simulation problem for this study was chosen such that it was as similar as possible to all the simulations which were conducted in this report, a best practice method would be to conduct simulations with more than one base model for every new problem, such that discrepancies can be uncovered. Another alternative is flat out using the DFN base model, this is however very computationally intensive, as asserted in both Table 4.1 and Table 4.2.

## 4.2 Selected Submodels

In the interest of creating simulation conditions which were as close as possible to the physics which are found in the real world, an all-inclusive approach was taken. Additional submodels were therefore included into the base lithium-ion models, which were discussed in Section 4.1. It can be a bit unclear how to differentiate between base- and submodels. Therefore, to recap, a base model is a complete battery model. Base models are constructed by a collection of submodels. A submodel describes one part of the physics, such as the thermal behavior. A large quantity of submodels are implemented in PyBaMM, such that the desired physical phenomena can be simulated and studied. Submodel options can, for a submodel, be set such that a specified physical model is used to model a given part of the physics. Submodel options are therefore meticulously chosen to simulate with desired physical models.

Submodels were also chosen with regards to the chosen parameters. Revealing the fact that the simple linear process, which is presented in Figure 3.1, is only meant to describe how a battery simulation is set up in PyBaMM. Figure 3.1 does however not describe how to decide on the intricate simulation setting, connected with each consecutive step. This can be done through multiple iterations of the linear process, creating a circular process. Settings can also be chosen through a holistic approach in which each choice inside each step is taken with regards to the other steps. Even though this technique was employed by the authors, reiterations were needed, in large quantities, thus closing the loop and making a circular process. The submodels which were used in this report are, if not stated otherwise, as presented in Listing 4.1.

**Listing 4.1:** The customized submodel options which were used in the simulations in this report

```
1 chosen_submodel_options = {  
2     "SEI" : "solvent-diffusion limited",  
3     "SEI film resistance" : "distributed",  
4     "SEI porosity change" : "true",  
5     "lithium plating" : "irreversible",  
6     "lithium plating porosity change" : "true",  
7     "particle mechanics": "swelling and cracking",  
8     "loss of active material" : "stress-driven",  
9     "thermal" : "lumped"  
10 }
```

Some effects are only visible when all submodels in Listing 4.1 are activated. This is why they all were chosen to work in tandem with one another. Section 4.2.1 through Section 4.2.5 are dedicated towards discussing why the specific submodels which are found in Listing 4.1, were either added to or customized within the base model.

### 4.2.1 SEI Submodels

As discussed in Section 2.6.1 SEI growth is one of the major degradation mechanisms for lithium-ion cells, it is also involved in multiple mechanisms, as shown in Figure 2.18. Which shows why it is important to include in the battery model, when studying ageing. There exists submodel options which can be used to model a constant SEI thickness, no SEI, which is the default mode, or SEI growth. The last category contains the submodel options, which are of interest to this report. This is because the changes inside a cell, has to be modeled accurately, in order to predict its capacity fade well.

The `solvent-diffusion limited` submodel option were chosen as the main SEI submodel. It's formulation, assumptions, transport equations and analytical solutions can be found in Solvent Diffusion Model for Ageing of Lithium-Ion Battery Cells, by Ploehn et al. [118]. For the scope of this report, the submodel's solutions of interest, are information on capacity loss due to SEI, and SEI thickness.

The SEI film resistance option is set to the `distributed` submodel [119]. This option is chosen since it is the only method to properly model the additional resistance term, which is created after the initial SEI formation. In 2004 Ploehn et al. commented that electrochemical impedance spectroscopy implied that SEI porosity changed with time. This was later also asserted by Yang et al., in newer research from 2017, which is why porosity change were set to `true`, and thus enabled [118, 119].

### 4.2.2 Lithium Plating Submodels

In Section 2.6.4, dendrite formation due to irreversible lithium plating were presented. Two lithium plating submodels are implemented in PyBaMM. `Reversible` and `irreversible` lithium plating. These submodels are derived from O'Kane et al.'s paper, which iterated on Yang et al.'s initial lithium plating model [113, 120].

The distinction between reversible and irreversible lithium plating is that in the reversible case; lithium plates are able to re-intercalate into the anode. Whilst for irreversible plating, plates are electrically isolated from the anode. This isolation, leaves lithium unable to return to the anode, resulting in irreversible capacity loss, through LLI. Thus, since capacity fade is investigated in this report, the `irreversible` lithium plating submodel has to be chosen [121]. Lithium plating has an effect on the porosity over time, which is why the `lithium plating porosity` option were enabled. Porosity change is thus accounted for by Sikha et al.'s method [122].

### 4.2.3 Particle Mechanics Submodel

As discussed in Section 2.4.5, large volume changes are very strenuous, and can, if large enough, quickly lead to irreversible capacity fade. Particle mechanics, which accounts for the mechanical effects of swelling and cracking, are thus paramount to include in the LiB model when studying ageing. Due to this the `swelling and cracking` submodel option were chosen for the particle mechanics submodel. The two other options are; the default option, i.e., `none` particle mechanics, and the `swelling only` submodel [123, 124].

The pouch cell geometry is applied in the simulations of this report. This cell type is prone to swelling, as discussed in Section 2.3.2. Which makes it appropriate to include the effects of

swelling. A **cracking only** submodel does not exist in PyBaMM, therefore, if cracking is to be applied, swelling must also be applied. In Section 2.6.3, the ageing phenomena of particle cracking were presented. Wherein cracks continuously emerges on the electrode surfaces, which allows for additional SEI growth in the newly exposed regions. This is shown in Figure 2.14 and Figure 2.16. It was therefore a natural choice to use the most comprehensive submodel, for both electrodes, such that accurate capacity fade could be modeled. How the **swelling and cracking** submodel models particle mechanics are out of the scope of this report. The reader is however encouraged to explore this for themselves, by reading the scientific papers which were utilized to implement these submodels in PyBaMM [123, 124].

It should also be noted that when the particle mechanics submodel is changed from the default option, **none**, the **stress-induced diffusion** submodel, is enabled. This is important for the modelling of particle cracking. Discussion on this is however also out of the scope of this report. The authors therefore refers to a description of this by Zhu et al. [125].

#### 4.2.4 Loss of Active Material Submodel

Loss of active material in the positive- and negative electrodes is attributed as two of three main degradation modes in Section 2.7. This is also highlighted by Figure 2.14, found at the bottom of Section 2.7. It was therefore apparent that a submodel which covered this, had to be added to the base model. Three submodels have been implemented to model the LAM in PyBaMM. By default the LAM submodel is set to **none**. It can however be changed to either **stress-driven**, **reaction-driven** or **stress and reaction-driven** [124, 126].

In this report an extra emphasis has been put on the NMC LiB, since NMCs are suitable for use in ferries. There are however, at the time of writing this report, no predefined parameters which describes the mechanical, crack, and LAM properties of the NMC chemistry in PyBaMM. Because of this parameters had to be imported from Ai2020, which is the only parameter set that includes these parameters in PyBaMM [124, 127]. The Ai2020 parameters are however based on the LCO chemistry.

To model LAM in the cathode, it was therefore assumed that the mechanical LCO parameters, still are representative for an NMC cathode. The cathode reactions of the LCO and NMC batteries are inherently different, which made it easy to exclude the **reaction-driven** submodel alternatives. This left one viable choice; the **stress-driven** submodel. The details on how the different LAM submodels function, can be explored in the scientific papers which were utilized to define the LAM submodels in PyBaMM [124, 126].

PyBaMM also allows for individual anode and cathode definitions of the LAM submodel. Further, the anode in the Ai2020 parameter set is, similarly to the base parameter set, made of graphite. The **stress and reaction-driven** submodel could therefore be applied to the anode, whilst the **stress-driven**, or **none** option could be applied to the cathode. This approach was however not applied in this report, observing the differences between these alternatives, could however yield interesting results.

### 4.2.5 Thermal Submodel

There are, as of writing this report, four different thermal submodels. The default thermal submodel is `isothermal`. In the `isothermal` assumption there is no heating of the cell. Thus giving a uniform, constant cell temperature. In order to use a more accurate thermal model, where the cell temperature develops through time, and the cell itself has a temperature gradient across its cell geometry, this submodel had to be swapped with one of the other thermal submodels [128]. Note that this choice, as with the previous submodel choices, contributes towards making simulations computationally intensive, as seen in Table 4.2.

The first submodel which were considered is called `x-full`. It only models heat conduction through the two large large pouch surfaces. The two remaining submodels are essentially the same thermal `lumped` submodel, but they work with slightly different parameter inputs. For an overview of the governing equations of the `lumped` and `x-full` thermal submodels, consult the PyBaMM GitHub examples [128].

The first variation of the `lumped` submodel is called the `x-lumped` submodel. What is important to know about this submodel is that PyBaMM lets the user specify the heat transfer coefficients which are used for the different surfaces of the cell. Whilst in the `lumped` submodel an average heat transfer coefficient is defined instead, such that the cell is cooled uniformly from all directions. It should be noted that in addition to controlling heat transfer coefficients, both cell cooling surface and cell volume parameters can be set. Which are of similar importance to the heat transfer coefficient, when Newton's cooling law is applied [128]. For the purpose of this report the `lumped` thermal submodel provides both adequate control of cooling properties, as well as desired complexity, and is thus used if not stated otherwise.

## 4.3 Choosing Parameter Values

Since an all-inclusive approach was taken with the choice of submodels in Section 4.2, great care had to be taken with the choice of parameters, which the submodels relies on in the simulations of this report. This was done to not undo the previous work. In the following sections the choice of base parameter set is shown. Thereafter, the choice of additional parameters, required by the customized battery model is shown.

### 4.3.1 Base Parameter Set

As shown in Listing 3.3, there exists a range of predefined parameter sets in PyBaMM. All sets can be explored in the GitHub repository, or in the documentation [111]. Since this report focuses on the NMC battery chemistry, there were four applicable base parameter sets. Refer to Section 2.4.4 for a discussion on the NMC chemistry. The four sets of interest are listed in Table 4.3.

**Table 4.3:** Summary of NMC base parameter sets, their chemistries, and individual comments on them [112, 129–131]

Parameter set	Chemistry	Chosen	Discarded	Comment
Chen2020	NMC-811	✓		Best alternative
Chen2020_plating	NMC-811		✓	Undesired plating
Mohtat2020	NMC UMBL*		✓	Unclear chemistry
ORegan2021	NMC-811		✓	Mesh complications
Xu2019	NMC-532		✓	Implementation issues

\* University of Michigan Battery Lab

Table 4.3 illustrates why the different base parameter sets were eliminated. It is also shown that the `Chen2020` base parameter set is chosen for the simulations in this report [112]. This is because the `Chen2020` set has the most aligned properties to what was desired. It is however a bit questionable if the NMC-811 is utilized in ferries, being nickel-rich, it is energy dense, which is a required property in portable energy storage systems. Its cycle lifetime might however be shorter than desired. Which imposes a limitation on the results in this report. In the future, results with new NMC-811 parameters might become more applicable as the technology matures.

The `Chen2020_plating` set uses undesired plating parameters, which is discussed in Section 4.3.3 [112, 114]. The `Mohtat2020` parameters are based on an in-house NMC chemistry from UMBL, its chemical composition were unclear to the authors, which is why it got excluded [129]. Readers of this report should not be discouraged from using the `ORegan2021` parameter set, as the PyBaMM creators have made a guide on how its mesh should be customised to get the solver to converge [130, 132]. Lastly, the `Xu2019` set describes an interesting chemistry. There were however issues with its implementation, it is unclear if the issue lies with the authors' implementation of the set, or if there are any issues within the `Xu2019` set [131].

### 4.3.2 SEI Parameters

There are three sets of SEI parameters implemented in PyBaMM to chose from. These are the `example`, `ramadass2004` and `yang2017` parameters [114, 118, 133–135]. An overview over the SEI parameter sets are given in Table 4.4.

**Table 4.4:** Summary of SEI parameter sets which are implemented in PyBaMM [112, 129–131]

SEI parameter set	Parameters [-]	Chosen	Discarded
<code>example</code>	19	✓	
<code>ramadass2004</code>	19		✓
<code>yang2017_sei</code>	9		✓

In Table 4.4, it is shown that `yang2017_sei` includes 9 parameters [114, 135]. When using

the submodels which were defined in Section 4.2, this parameter set is not sufficient. Thus, if `yang2017_sei` is used, an exception gets raised, since the code expects to find a concentration value which has not been defined for instance. Both the `example` [114, 118, 133–135] and `ramadass2004` [133, 135] SEI parameters could be utilized, as they have the required amount of parameters defined. The `example` SEI parameters are however chosen since they are based on a review, in which the parameter values were used to fit the SEI submodels to experimental data.

### 4.3.3 Lithium Plating Parameters

There are two sets of lithium plating parameters in PyBaMM; `okane2020_Li_plating` and `yang2017_Li_plating` [113, 114]. An overview over these lithium parameter sets are found in Table 4.5.

**Table 4.5:** Summary of lithium plating parameter sets which are implemented in PyBaMM [113, 114]

Lithium plating parameter set	Parameters [-]	Parameters defined by functions [-]	Chosen	Discarded
<code>okane2020_Li_plating</code>	6	2	✓	
<code>yang2017_Li_plating</code>	4	0		✓

From Table 4.5 it can be observed that the `okane2020_Li_plating` includes more parameters than the `yang2017_Li_plating` [113, 114]. It also has 2 parameters which are defined by functions. Thus, by choosing `okane2020_Li_plating`, two parameters becomes a function of temperature, and lithium plating and electrolyte concentration. This is more accurate than attributing a constant value for every combination of temperature and concentration, which is what is done in `yang2017_Li_plating`. Since `yang2017_Li_plating` also includes less parameters the `reversible` lithium plating submodel cannot be ran. The simulation can however still run without errors if the `irreversible` lithium plating submodel is used [114].

### 4.3.4 Particle Mechanics and Loss of Active Material Parameters

Only one parameter set can be used to model particle mechanics and LAM, as of the time of writing this report. The `Ai2020` parameter set, either has to be used, or be extracted from, as shown at the bottom of Listing 3.3 [124, 127]. Since the `Chen2020` base parameter set were chosen in Section 4.3.1, parameters has to be extracted from `Ai2020` to `Chen2020` [112, 124, 127]. The particle mechanics, and LAM parameters are separately defined for the anode and cathode, an overview of the `Ai2020` particle mechanics- and LAM parameters are found in Table 4.6 [124, 127].



**Table 4.6:** Summary of particle mechanics and LAM parameters which are implemented in PyBaMM [124, 127]

Particle mechanics- and LAM parameter set		lico2_Ai_2020	graphite_Ai_2020
Parameters [-]	Mechanical	5	5
	Cracking	7	7
	LAM	3	3
Parameters defined by functions [-]	Mechanical	1	1
	Cracking	1	1
	LAM	0	0

The Ai2020 parameters were used since the authors wanted to model as many ageing mechanisms as possible. Also, considerations which should limit errors were made with the coupled submodels in Section 4.2.3 and Section 4.2.4. Thus, both the cathode parameters, `lico2_Ai_2020`, and anode parameters, `graphite_Ai_2020`, were imported and used in the simulations of this report.

#### 4.4 Mesh Refinement Study

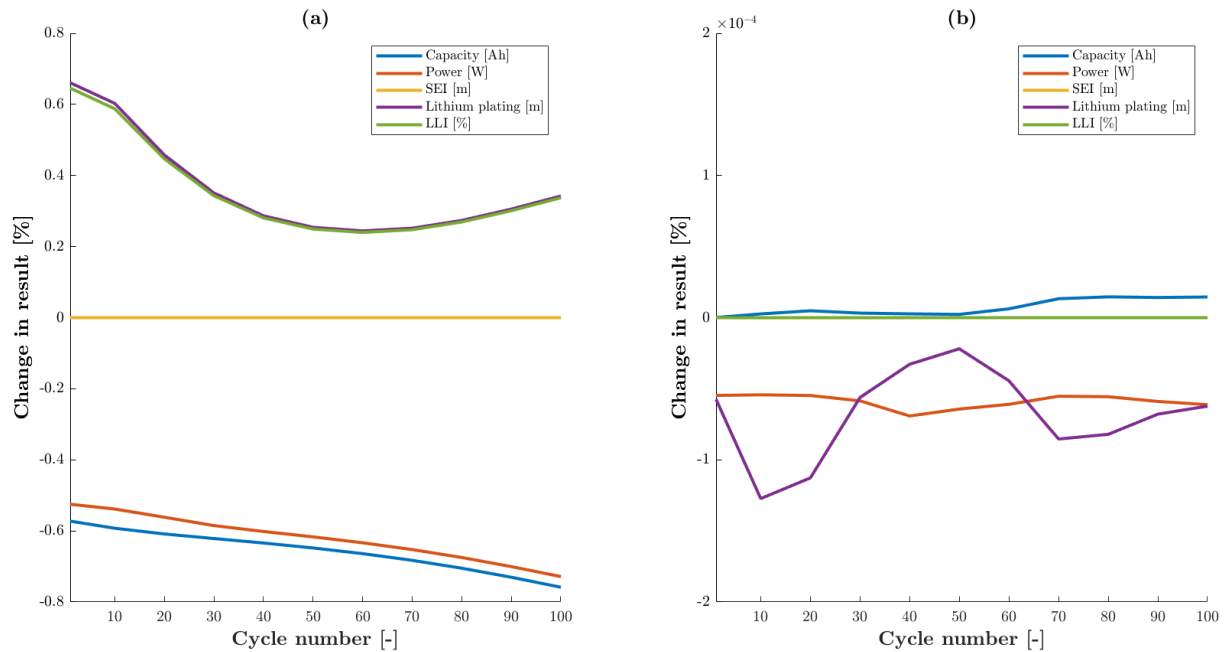
To find a suitable mesh resolution a mesh refinement study was conducted. In this mesh refinement study the number of volumes per particle, and the number of grid points along the x-axis were changed. Refer to `var_pts` in Listing 3.8 to see all the mesh resolution variables. The number of grid points in the y- and z-direction were kept at their original value of 10, the particle radii were also left unchanged for both the negative and positive particles. The base simulation which was used in the refinement studies were built to be a best possible representation for all the simulations conducted in this report. To conduct a meaningful mesh refinement study, the simulation problem must be simulated multiple times whilst gradually increasing the grid points, so that the mesh is refined from a coarser to a finer mesh. In the end the observed change in results, can be weighed against increase in computational expense. The standard number of `x_n`, `x_s` and `x_p` points, and `r_n` and `r_p` volumes, in `var_pts` is 20. This number were therefore chosen to define the initial mesh resolution. This was however found to be an impossible starting point, as the error which were discussed in Section 3.4.4, persisted up until 45 points and volumes. Table 4.7 summarizes the results of the mesh refinement study that followed from this starting point.

**Table 4.7:** Mesh refinement study results

Points [-]	Result difference [%]	Solve time [s]	Time increase [%]
45	-	4 590	-
50	0.3825	5 133	11.83
55	0.0015	8 299	61.68
60	0.0013	18 662	124.9



To be able to solve the simulations, the mesh is already made very fine, because of this the results in Table 4.7 show convergence within an acceptable margin of error, even for the very first step between 45 and 50. To get additional leeway from solver issues, 50, were chosen as the starting point for the simulations in this report. The finest mesh resolutions should only be used when absolutely required by the solver, since the results which are generated from these settings are approximately the same as those found at 45. From Table 4.7 it also apparent that it is very inefficient to use 55 or 60 since there is only marginal differences in results compared to 45. In addition to this, the computational time is quadrupled when increasing from 45 to 60. Figure 4.4 shows some results from the study which are of special interest to this report.



**Figure 4.4:** Selected results from the mesh refinement study. (a) Shows the difference in results between 45 and 50. (b) Reflects the observed change between 50 and 55.

In the last two columns of Table 4.7 it is shown that the simulations are time consuming, because of this only 100 cycles were chosen to be examined during this refinement study. The results from the study did nonetheless show a large degree of convergence and stability. In spite of this, some results are suspected to diverge, whenever 50 is used instead of 60, especially as the number of cycles are greatly increased, and the initial errors are allowed to propagate forward. It is however assumed that this propagation will not, for the limited number of cycles which are conducted in the simulations of this report, detrimentally effect the results. Other issues, assumptions, and uncertainties such as choice of base model and parameter values are expected to have a much larger impact on result accuracy.

#### 4.5 Solver Tolerance Refinement Study

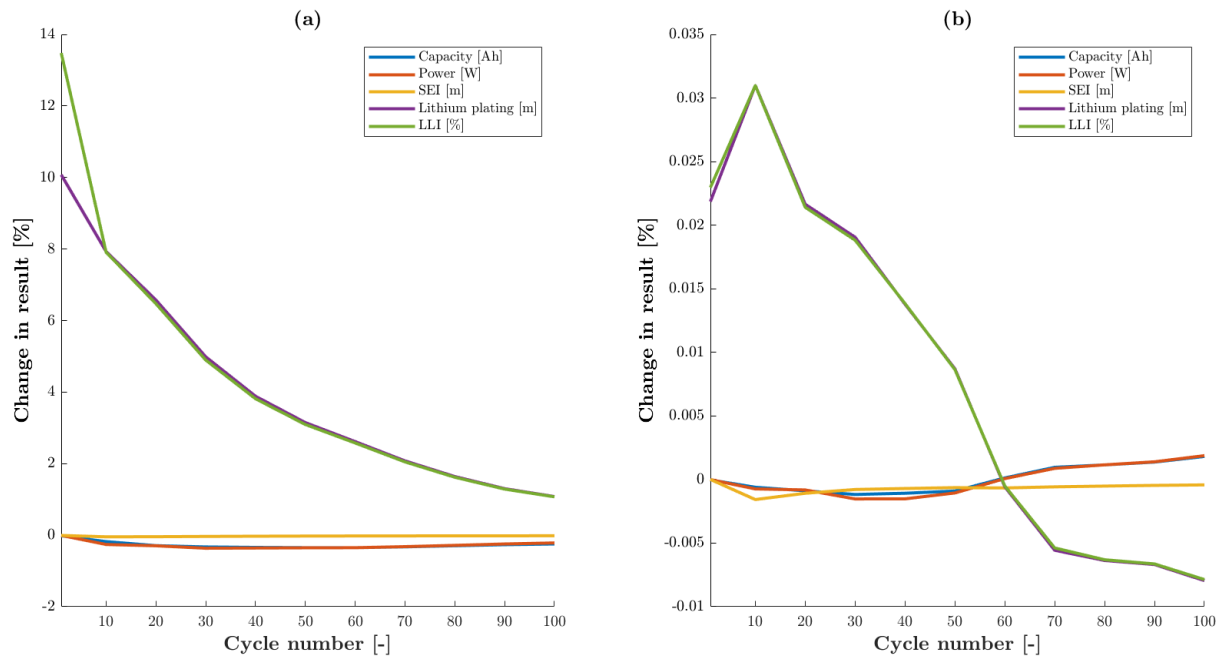
A study on CasADi solver tolerances was conducted to determine viable tolerance settings. In this study the same basic simulation was conducted four times. The mode of the CasADi solver was set to "Fast with events" since this setting is faster than the "Safe" mode. The CasADi solver was also chosen since it was the fastest. Both the absolute, and relative tolerances were set to the values reflected by Table 4.8. The extrapolation and root finding tolerance were

however set to their respective default values of 0 and  $10 \cdot 10^{-6}$ . It is also worth noting that the optimal value of 50 from Section 4.4 was employed in this study. The overarching goal of the tolerance study was to find a tolerance range which could be utilized without excessively impacting result accuracy. Especially considering Section 3.4.2 and the solution found to counter linesearch algorithm issues, i.e., loosening solver tolerances. Table 4.8 shows the results obtained from the solver tolerance refinement study.

**Table 4.8:** Solver tolerance refinement study results

Tolerance [-]	Result difference [%]	Solve time [s]	Time increase [%]
$5 \cdot 10^{-1}$	-	3391	-
$10 \cdot 10^{-2}$	1.8071	3625	6.901
$10 \cdot 10^{-4}$	0.4260	4112	13.43
$10 \cdot 10^{-6}$	0.0060	5133	24.83

Table 4.8 and Table 4.7 reflects that tightening the solver tolerances are not as strenuous as increasing the mesh resolution.  $10 \cdot 10^{-6}$  was chosen to be the tightest tolerance in this study since this is at the level where the linesearch algorithm has been found to fail. The loosest setting was set to  $5 \cdot 10^{-1}$  in this study. Figure 4.5 shows some values of interest from the first and last step in Table 4.8.



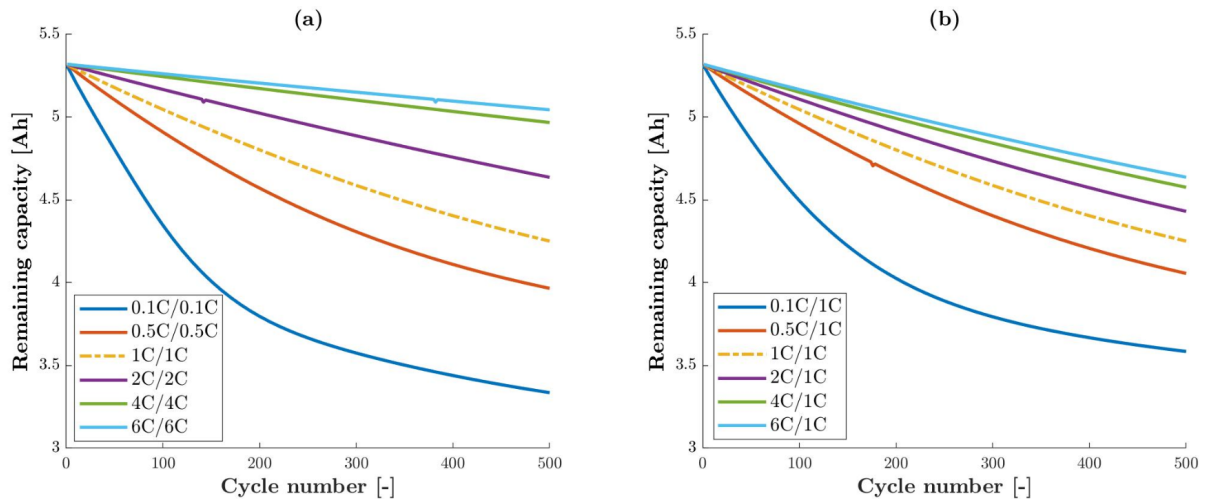
**Figure 4.5:** Selected results from the solver tolerance refinement study. (a) Shows the difference in results between  $5 \cdot 10^{-1}$  and  $10 \cdot 10^{-2}$  solver tolerances. (b) Reflects the observed change between solver tolerances of  $10 \cdot 10^{-4}$  and  $10 \cdot 10^{-6}$

In Section 4.4 it was discussed that some results were expected to diverge as the number of cycles increase. The opposite trend seems to be observed in Figure 4.5, but as previously asserted this is not the expected outcome in the long run. Due to the computing resources available to the

authors, this solver study is limited to 100 cycles. Which makes it hard to confidently project trends and result accuracy. From the results presented in Table 4.8 it is however suggested that loosening the solver tolerance from the default value of  $10 \cdot 10^{-6}$  by a factor of up to 100 is a viable solution when facing linesearch algorithm issues. This is fortunate, since it is the only method known to the authors that can deal with this type of error. Using  $10 \cdot 10^{-6}$  as a starting point and moving towards tolerances in the upper  $10 \cdot 10^{-3}$  to  $10 \cdot 10^{-6}$  range, therefore seemed like an apt way forward. This method should both be time efficient, seemingly give accurate results and curb linesearch errors.

## 4.6 Initial Results

The batteries inside ferries are often charged at around 2C since they lie by land only for a short period of time. The discharge C-rate varies with the load profile, as seen in Figure 2.5. In the following simulations, large and small C-rates from 0.1C to 6C were tested. The charging and discharging C-rates were set to the same value in Figure 4.6a, whilst different charging rates were tested in Figure 4.6. The simulations were devised to determine the impact of different C-rate magnitudes on capacity, at a SoC range between 40-70%. With the Newman-Tobias base model applied the following results presented in Figure 4.6, were acquired.



**Figure 4.6:** Capacity fade as a function of C-rates. Curves in (a) has the same charge and discharge C-rate. (b) shows various charging rates coupled with a discharging rate of 1C. The SoC window is between 40% and 70% for both figures.

The graphs in Figure 4.6 represents the remaining capacity as a function of the number of cycles. Figure 4.6a shows that the greater the C-rate is, the greater the capacity retention becomes. In general, it is however a known result that increasing the C-rate, will decrease the capacity of a LiB faster. Hence, decreasing the C-rate should increase capacity retention [87]. Surprisingly, the results presented in Figure 4.6, does not indicate what has been shown to be the case in theory and practice. Simulation results misleadingly indicates that higher C-rates provides better battery health than low C-rates.

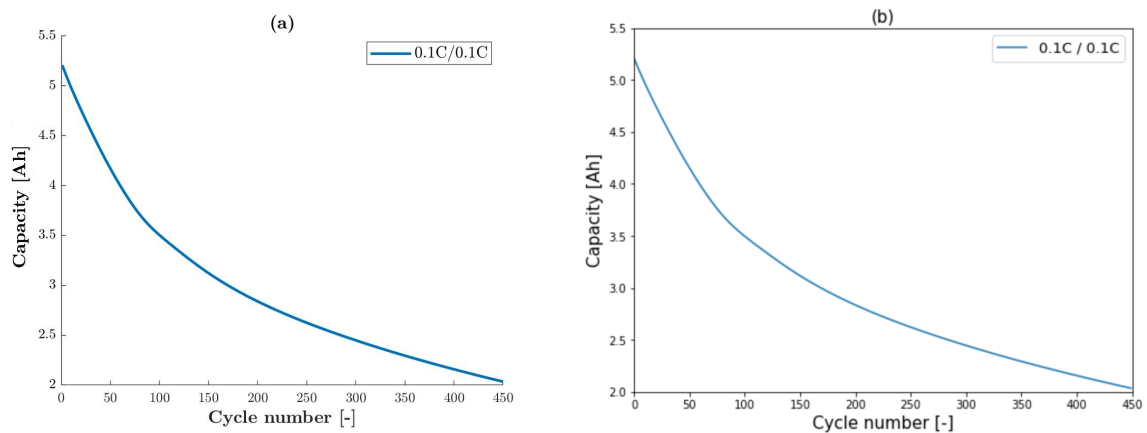
A battery reacts differently to fast charging and discharging. Varying the charging C-rate was therefore explored in Figure 4.6b, such that it could be compared with the results from a similar discharge study. A 1C/1C, discharge and charge C-rate, were defined as a reference point for

the cell which were simulated at different charging C-rates in Figure 4.6b. The reference is indicated by the yellow dotted line. The results presented in Figure 4.6b also contrasts those which are described in the literature. Qu et al. asserts that larger C-rates increases stress in the electrode and induces cracks, which creates new locations for more SEI formation on the electrode particles. Qu et al. also presents that large charging C-rates provokes lithium plating on the anode, due to lithium not being able to intercalate quickly enough into the anode. Qu et al. also points out that SEI growth is aggravated at higher temperatures, which are observed due to the increased heat generation at larger C-rates [136].

The results presented in this section thus looks incorrect. In addition to the results presented in Figure 4.6, the authors obtained and analyzed data from a range of simulations which varied both the SoC window, testing both deep and shallow cycles, and the charging and discharging rates. The results were all similar to those found in Figure 4.6, in that they all displayed the similar contrary C-rate pattern. Five corresponding simulations to Figure 4.6b, were for instance conducted for the aforementioned comparison between the effects of increasing charge versus increasing discharge. In the base model study of Section 4.1, it was proposed that different base models should be tested for every simulation problem, because of this the SPM<sub>e</sub> model was also applied. The incorrect result trends presented in Figure 4.6, produced by the Newman-Tobias base model, were mirrored by the SPM<sub>e</sub> results. As a consequence of this the author's of this report were sent down a spiral of troubleshooting. This was done to rule out possible errors, and to grasp why the results presented in Figure 4.6, and in consequent simulations, presented contrary results.

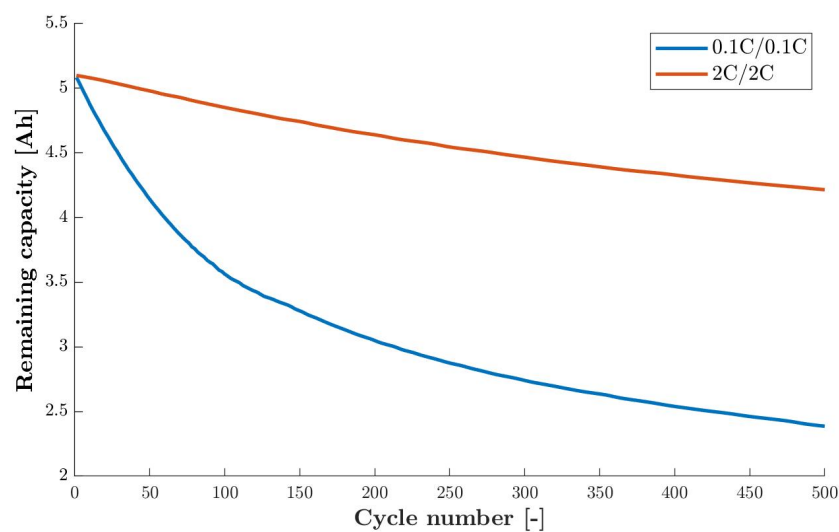
#### 4.7 Troubleshooting Initial Results

To rule out a possible, simple data processing mistake, and to ensure that the correct data were obtained for further visualization in MATLAB. Figures were both produced in MATLAB and in Python. This resulted in Figure 4.7. Figure 4.7a and Figure 4.7b shows two identical plots in terms of data. Figure 4.7a were constructed in MATLAB, and Figure 4.7b were made with Python. This proves that the `save_data_to_csv` function, as shown in Listing A.2, works as intended. Data processing was thus ruled out as an issue. Figure 4.7 also shows that figures made with MATLAB are more intelligible than those created in Python, which is why MATLAB were preferred for visualization.



**Figure 4.7:** A comparison of MATLAB and Python figures. (a) was constructed in MATLAB. (b) shows a Python plot of the same values, but they are directly read from the simulation output.

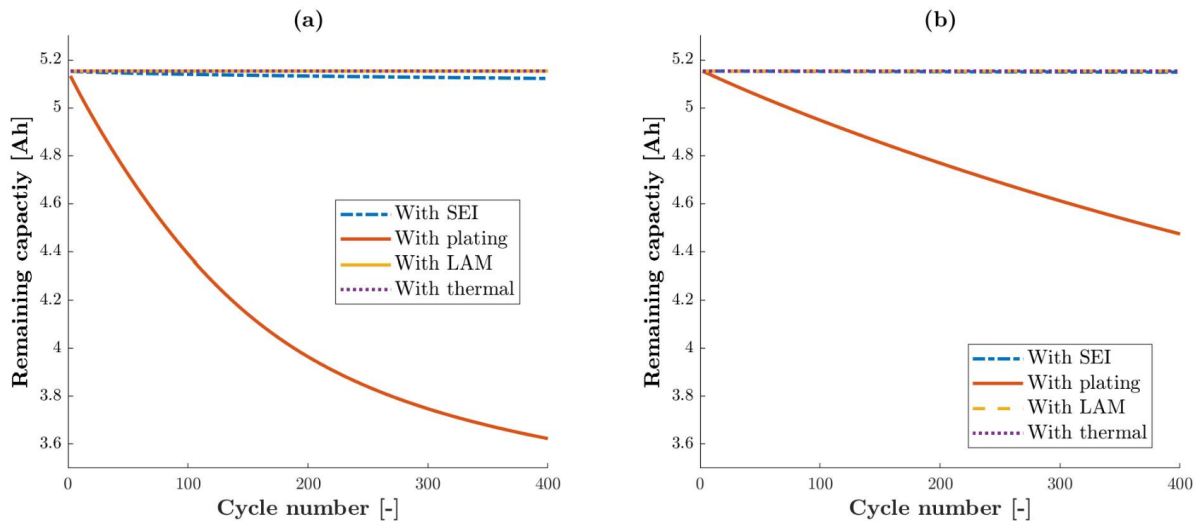
To continue ruling out possible simple mistakes, the group tried a basic simulation, in which no customisation was made to the parameters. Hence, the `chen2020_plating` base parameter set, from Section 4.3.1 were chosen [112]. It was chosen since it is a NMC parameter set which includes parameters on lithium plating and SEI by default. This eliminates the need for any parameter imports. LAM and particle mechanics could not be tested in these simulations, since this requires parameter imports. The `Ai2020` parameter set, which includes LAM and particle mechanics, was not tested because it is a LCO, and not a set of NMC parameters [124, 127]. A test of the `chen2020_plating` base parameter set should be able to eliminate or uncover parameter imports as an issue. In the simulations which ensued, a minimal amount of custom submodels were employed to simplify the simulations as much as possible. Only two custom submodels, on SEI and lithium plating, were applied to acquire the simulation results presented in Figure 4.8.



**Figure 4.8:** Simulation results generated by using the `chen2020_plating` base parameter set, without any imports from other parameter sets.

Figure 4.8 shows that the C-rate issue is not corrected by using a set of parameters that does not require any imports. This likely excludes parameter imports as an issue, since the same error is seen with and without external parameter imports.

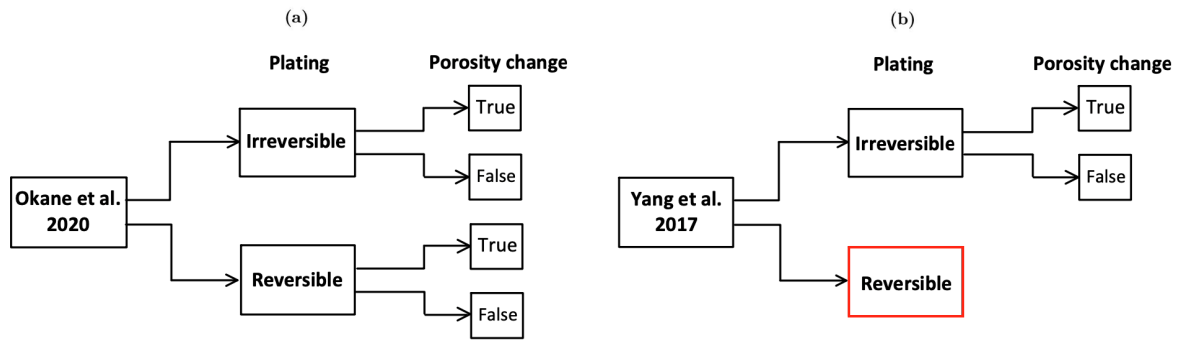
The submodels governs which, and how, the physics inside a battery are modeled in a PyBaMM simulation. Due to the nature of the results, it became a natural next step in the troubleshooting process. The customized submodel options which were utilized, and defined in Section 4.6, therefore had to be reviewed. The approach taken in this review involved running one simulation with each customised submodel. Thus, in the end, results produced by the individual submodels could illuminate if any particular submodel was misbehaving. Each customised submodel in Section 4.2 was run independently, that is, with the other customised submodels disabled. This review lead to the results which are found in Figure 4.9.



**Figure 4.9:** Results from the submodel review. The results in (a) were obtained with a C-rate of 0.1C, whilst the results in (b) were obtained with a C-rate of 2C. Both were cycled in a SoC window between 30% and 70%.

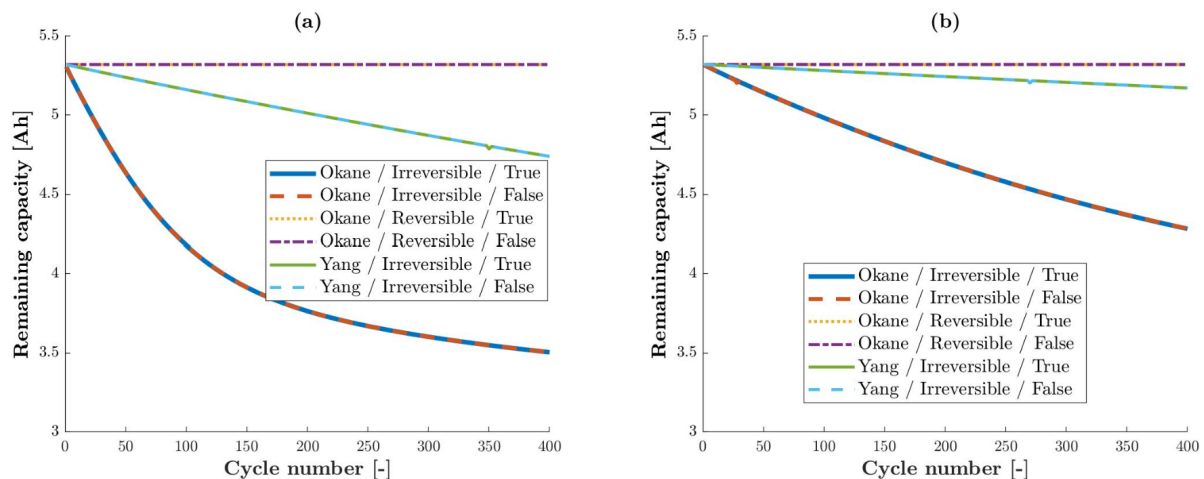
According to Figure 4.9, lithium plating dominates cell ageing at 0.1C and 2C, therefore, it is the main ageing contributor observed in Figure 4.1a and Figure 4.6a. This is concluded with relatively high confidence, since the capacity curves of Figure 4.1a and Figure 4.6a match the shapes produced by the lithium plating submodels in Figure 4.9. The amount of capacity loss, which is observed from lithium plating in these figures should not be found, as the simulation were ran at a modest ambient temperature of 25°C. In contrast, lithium plating is known to be caused by low temperatures [87]. There should also be more lithium plating at 2C than at 0.1C, which is not observed in Figure 4.9 [87]. Lessened ageing is still observed at higher C-rates after this submodel review. Thus, the initial issue of Section 4.6 still remains. However, it can be observed that the lithium plating and SEI submodels stand out, as they are the only submodels that individually produce ageing in Figure 4.9. No ageing is observed, as expected from the thermal submodel. The same is also observed for the LAM submodels. This could possibly be explained by the fact that the customised submodels were run together with the default submodels. Where the default thermal submodel is set to `isothermal`. For more on this refer to Section 4.2.5. The submodel study did not uncover the issue, but it narrowed down the continuation of the search to the SEI and lithium plating submodels.

Lithium plating appears to be especially contradictory in Figure 4.9. There exists two sets of lithium plating parameters, as illustrated in Section 4.3.3. And there are two settings for both lithium plating submodels, as discussed in Section 4.2.2. Thus, eight combinations exist. The `yang2017_Li_plating` parameters are however not compatible with the `reversible` lithium plating setting, and the number of combinations is due to this reduced to six, as shown in Figure 4.10.



**Figure 4.10:** Flowchart of every lithium plating setting to troubleshoot. (a) shows the submodels which are available for `okane2020_Li_plating`. (b) shows the submodels which are available for `yang2017_Li_plating`.

The purpose of examining all configurations is to uncover whether there are combinations that yield better results. The initial simulations were conducted at 0.1C since this is the C-rate the most contradictory results have been produced. The results of this are shown in Figure 4.11a. Thereafter the behaviour at 2C were simulated and summarized in Figure 4.11b. This was done so that Figure 4.11a and Figure 4.11b could be compared with the misleading trends from earlier simulations.



**Figure 4.11:** Results from the lithium plating review. Results were obtained with a C-rate of 0.1C in (a), and with 2C in (b). Both were simulated within a SoC of 30%-70%.

Figure 4.11 shows that using `okane2020_Li_plating` leads to a significant capacity loss whenever lithium plating porosity change is set to true. The `yang2017_Li_plating` is more accurate in this study, considering the modest temperature which were applied in this simulation [87]. Thus, it seems that the `yang2017_Li_plating` can be utilized to get more accurate results, at least with the error from Section 4.6 still present. Figure 4.11 still shows that both `okane2020_Li_plating` and `yang2017_Li_plating` display similar trends, wherein capacity loss mistakenly is reduced with increasing C-rate. Figure 4.11 also shows that all combinations of lithium plating, which are implemented in PyBaMM, were tested. The main issue were however not uncovered.

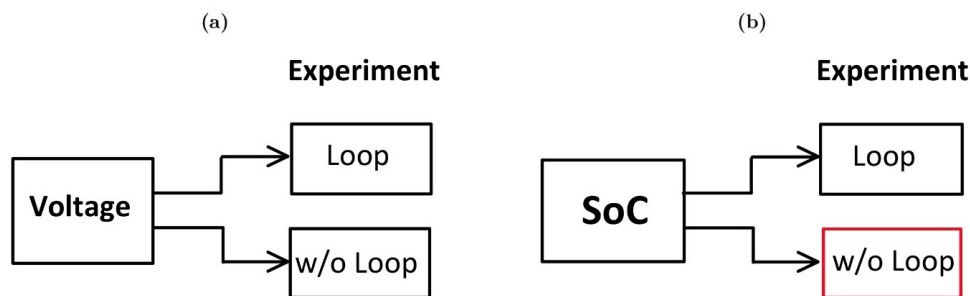
SEI were also identified as an issue in Figure 4.9. Therefore, a similar study on combinations of



SEI submodels and parameter sets was conducted for the SEI. The study yielded similar results, in which 0.1C still produced more ageing than 2C. The issue from Section 4.6, has thus not been resolved by either study.

The simulations from Section 4.6 were based on specified SoC windows, as defined in Section 2.5.3, rather than the traditional cycling method, which involves cycling between two fixed potentials. A complication with the SoC cycling method is that the maximum usable cell capacity decreases with time. This is important, since, for example, the capacity of an unaged cell at 70% SoC is larger than the capacity at 70% SoC for the same aged cell. Thus, when cycling with constant C-rates inside a specific SoC window, the charging and discharging currents have to decrease as the capacity fades. This relation is made apparent from Equation 2.3. However, if the current is not corrected, the SoC window could gradually expand, thus allowing the cell to cycle at undesired states of charge. This makes intuitive sense, since if the cell has less capacity, but is charged as if no capacity has faded, it will evidently charge towards larger states of charge. The currents can however be corrected for in simulations, as the remaining capacity is readily available in between each cycle.

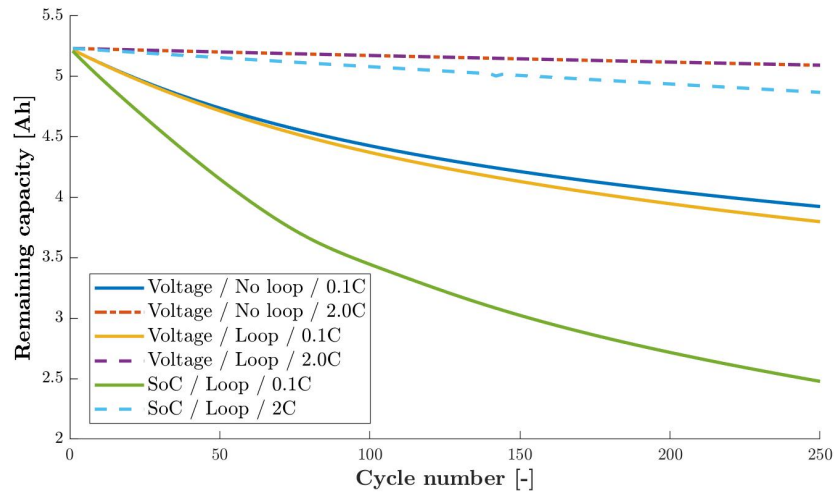
Unexpected results were obtained with the SoC cycling scheme applied in Section 4.6. Since it is also a less conventional method than using two voltage criteria, it became natural to troubleshoot. To either eliminate, or determine that the SoC cycling method were the root cause of the issue, it had to be compared with the voltage method. Figure 4.12 shows three cycling methods in black, all of which were used in the cycling study.



**Figure 4.12:** Flowchart of the voltage and the SoC based cycling schemes.

The flowchart on the left shows the two methods in which voltage were used as a cycling reference. In the first procedure, without (w/o) using a `for` loop, the simulation relies on the `experiment` class alone, wherein the number of cycles and operating conditions, such as: "Charge at 1C until 4 V", of the experiment are set. "Charge at 1C until 80% SoC" is yet to be implemented in PyBaMM, because of this a `for` loop had to be constructed and combined with the `experiment` class to simulate in specified SoC windows. This was done such that the required current to charge and discharge, at set C-rates, could be recalculated in between each cycle. This gives one possible SoC cycling scheme, which is shown in black in Figure 4.12, and in Appendix B. Furthermore, the voltage method could be expanded to also utilise a `for` loop. Thus, two voltage cycling methods could be tested. This approach should highlight any errors with the SoC cycling scheme in general or with the `for` loop on which it relied. With the completion of the outlined simulations from Figure 4.12, the results presented in Figure 4.13 were obtained.





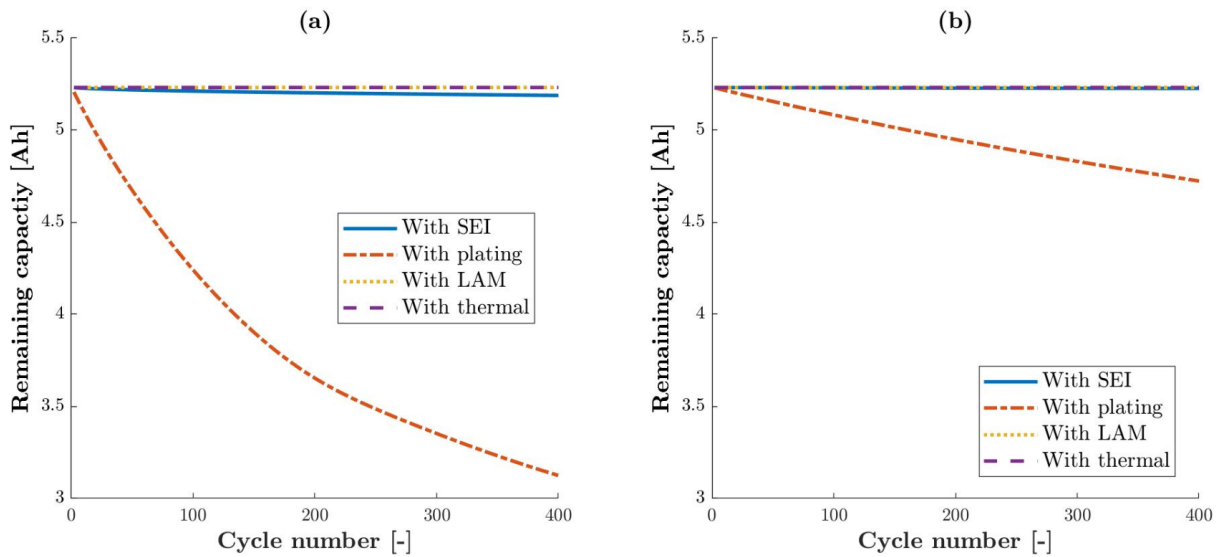
**Figure 4.13:** Results from the troubleshooting of the SoC cycling scheme.

Figure 4.13 shows that the `for` loop and the `experiment` only, or no loop method, yields different results. This indicates that the calculations in the loop do not correct the current in a manner similar to what is done with the `experiment` only method. By communication with the PyBaMM community it was confirmed that this is correct and that the C-rate in the `experiment` only method is calculated with the "Nominal cell capacity [A.h]" variable, so that the current stays constant throughout the experiment. The current in the `for` loop is however recalculated based on the remaining capacity, which is determined by the "Capacity [A.h]", and not the "Nominal cell capacity [A.h]", as seen in Listing B.2. As the currents diverge with time, it is expected that the ageing observed for loop and loop are different in Figure 4.13. It is also expected that the loop method ages the cell faster since it calculates smaller currents over time as the "Capacity [A.h]" decreases. Note that this is only expected due to the C-rate issue still persisting in these simulations. Normally lesser currents gives better SoH over time. The SoC cycling method ages faster than the voltage method in Figure 4.13. Differences in observed ageing between SoC cycling and voltage cycling can be explained by the choice of upper and lower voltages and states of charge. These values were not set up to exactly match one another, the results are hence expected to differ, like they do in Figure 4.13. It should be noted that the most defining feature of Figure 4.13 is still the C-rate, where the 2C and 0.1C graphs are grouped together. The 2C voltage cycling results overlaps since ageing is small at this setting, given enough cycles they too would diverge, like the 0.1C voltage results. This is the case since the ageing is mistakenly larger at 0.1C in these and previous simulations.

To summarize, it has been argued that the SoC cycling method is not the root cause behind the inverse results. This is concluded since both the voltage and SoC cycling methods produce equally contrary results with regards to C-rate. This can be concluded since differences between loop and no loop with the voltage method have been given a satisfactory explanation, and because there has also been found a likely explanation for the differences which are observed in ageing between the SoC and voltage cycling methods. Still, it is more secure to use the normal voltage cycling method. The authors were able to obtain "Negative electrode SoC" and "Positive electrode SoC" variables, yet a "Cell SoC" variable does not exist in PyBaMM. It could not be inferred from the other available variables in PyBaMM by the authors, because

of this it cannot be confirmed whether the SoC cycling method cycles inside or outside the desired SoC window. To be confident in this, another approach or additional functionality is needed in PyBaMM. Furthermore, the authors assumed that the default Coulomb efficiency term is set to 1 in PyBaMM. The authors has conducted simple simulations, with no customized submodels, which supports this assumption. However, the Coulomb efficiency is assumed to be lowered by side reactions, such as those provided by the SEI and lithium plating submodels. These side reactions should adsorb some current, and lower the Coulomb efficiency [137]. This was confirmed after communication with two PyBaMM developers. However, the authors were unable to correct for this in the SoC cycling method. Results produced by this means of cycling should therefore be presented with an asterisk that points out that the SoC window might not be as expected. Hence, the SoC method devised in this thesis must be improved, preferably by using built in SoC cycling functionality in PyBaMM, if it is released in the future.

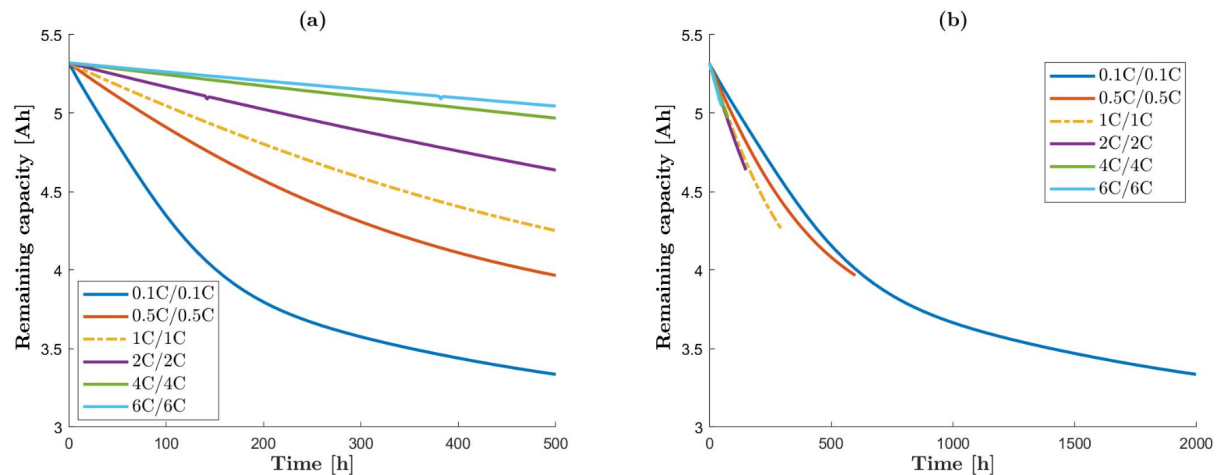
To further confirm that the SoC cycling method, was not the root cause of the issue, which first presented itself in Section 4.6, a similar submodel study to the one found in Figure 4.9 were conducted with the traditional voltage cycling method. Note that in this method, the cycling is carried out between two set voltages and that no loop was used, so that the cycling is based only on the `experiment` class. Figure 4.14 shows the results of the voltage cycling submodel study. Figure 4.14a were obtained by using 0.1C, whilst Figure 4.14b were obtained by using 2C.



**Figure 4.14:** Results from the submodel review when cycling between two set voltages. The results in (a) were obtained with a C-rate of 0.1C, whilst the results in (b) were obtained with a C-rate of 2C. Both were cycled between the same upper and lower voltages.

The results of the voltage cycling submodel study are remarkably similar to those of the SoC cycling submodel study found in Figure 4.11. The same inverse results as a function of C-rate is seen in both figures. In addition to this the SEI and lithium plating curves are very reminiscent of each other in the two submodel review figures, i.e., Figure 4.11 and Figure 4.14. Which again, indicates that the main issue lies elsewhere, and not with the cycling method which were utilized. Further studies were also conducted, as with the SoC cycling method, on SEI and lithium plating. However, the analyses did not provide any new information on how to correct the issue which has persisted from Section 4.6.

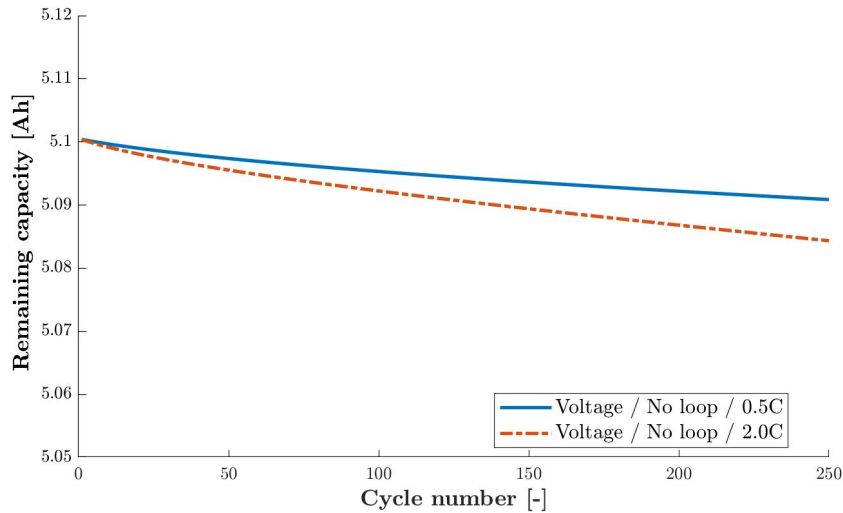
To obtain another view of the data, it was plotted against time. Initially, the simulation results were presented as a function of cycle number, as shown in Section 4.6. In Figure 4.15a and Figure 4.15b, the capacity has, however, been plotted with two different x-axes: one based on cycle number and the other based on time. In Figure 4.15a the rate of degradation is unchanged from the previous section. When plotted with respect to time however, it becomes evident that with higher C-rates, the degradation is higher, and thus capacity fades more rapidly.



**Figure 4.15:** Capacity fade as a function of constant charging and discharging C-rates. (a) shows the data plotted against number of cycles, and (b) shows the same simulation data plotted against time.

While the results shown in Figure 4.15b are more sensible, it is, with the exception of calendar aging, not common to display results on ageing based on time. The results obtained from the cycling are normally presented with respect to the number of cycles. Thus, the results in Figure 4.15b are still considered to be inconclusive, since the results in Figure 4.15a still provides opposite results from what is expected. Note that the lower C-rates in Figure 4.15b produce longer graphs when the same amounts of cycles are simulated. This is explained by the fact that a 0.1C cycle takes ten times longer than a 1C cycle.

Although the results of Figure 4.15, did not provide a solution to the C-rate issue, they gave the authors an idea of a possible problematic factor to troubleshoot. In Section 4.2.1 the `solvent-diffusion limited SEI` submodel were chosen. With the simulation setup which have been used in this report, it has been shown that the `solvent-diffusion limited` submodel has produced contrary C-rate results. In a research paper by Single et al. most of the SEI submodels which are implemented in PyBaMM were studied. Single et al. found that the `solvent-diffusion limited` submodel did not reproduce experimental results. However, in the article it is stated that the `interstitial-diffusion limited` submodel results in the most promising agreement with experimental data [118, 138]. Thus two simulations with the `interstitial-diffusion limited` submodel were conducted. The results are found in Figure 4.16.



**Figure 4.16:** Comparison of the `interstitial-diffusion` limited SEI submodel, at 0.5C and 2C.

Figure 4.16 shows that ageing due to SEI is in accordance with theory, as larger C-rates are shown to accelerate capacity fade when applying the `interstitial-diffusion` limited SEI submodel. Thus, half of the issue from Section 4.6 has been resolved. Hence, it is concluded that the `interstitial-diffusion` limited submodel should be applied in simulations, and with hindsight it should have been chosen in Section 4.2.1.

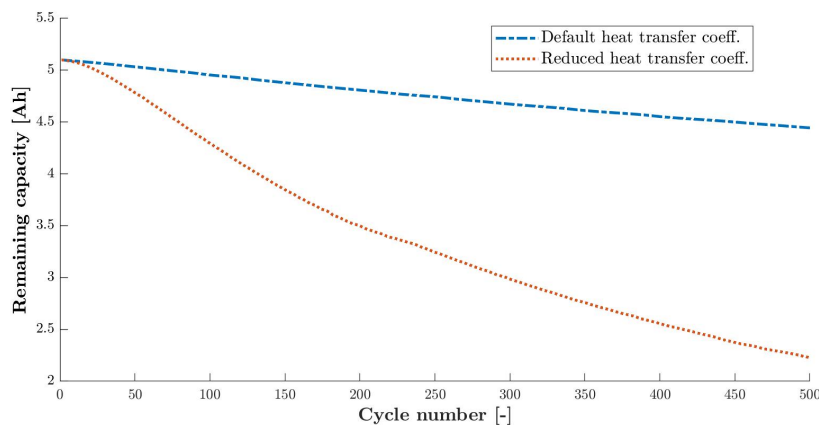
Within the time frame of this report, the authors have exhausted as much troubleshooting on lithium plating as possible. All 6 lithium plating alternatives have been tested at two C-rates for both SoC cycling and voltage cycling. This has been conducted with both the NT and SPMe base models. The issue with the SEI got resolved after change of submodel, which indicates that there are no issues with the discretization, choice of solver, and solver options, which were discussed in Section 4.4 and Section 4.5 respectively. In summary, all lithium plating parameters and submodels have been tested. Furthermore, the discretization and solver seem to be functioning as intended, as the SEI issue was resolved without touching these simulation settings. This leaves the possible issue to either be with the base parameter set or the base battery model. The base NMC parameter sets were summarized in Table 4.3. The authors had trouble getting the `Xu2019` simulations to finish, whilst the `Chen2020` were the most appropriate parameter set for the simulations in this report. The `Chen2020_plating` base parameter set were also tested, the results of which can be found in Figure 4.8. The `Mohtat2020` base parameter set have also been tested. This leaves only the `ORegan2021` base parameter set, which should be tested together with the lithium plating submodels to uncover whether this is a solution to the C-rate issue or not. In addition to this, the DFN base model should be applied to uncover whether or not this leads to different lithium plating results. This might be the case, as results between base models were shown to differ in Figure 4.3.

The troubleshooting process resolved 1 of 2 known simulation issues from Section 4.6. The new SEI submodel ensures that the SEI is modeled closer to how it develops in practice. Importantly the SEI grew faster at higher C-rates, and slower at lower C-rates. With respect to lithium plating, it is unclear why it reacts in the way it does. Numerous troubleshooting tests have been conducted to determine the root cause of the contradictory lithium plating results. The tests

carried out in this section leaves the ORegan2021 base parameter set, and the DFN base model to be tested. However, this could not be performed within the time frame of this report, but seem like the natural next steps in the troubleshooting process. The troubleshooting process took a considerable amount of time due to the sheer number of settings to test, and the simulations taking a long time to complete. The authors therefore recommends conducting a greater amount of tests when adding more complexity to a customized battery model, such that issues are uncovered, and subsequently narrowed down at an earlier stage.

## 4.8 Final Results

Simulations where the C-rate stays constant and the other settings are varied to obtain results have been observed to be in accordance with theory and practise. Thus, simulations were conducted by varying other parameters. Figure 4.17 highlights the importance of adequate cooling. These simulations were constructed by changing the definitions of heat transfer coefficients, in which a poorly cooled cell is assumed to be analogous to a cell with reduced heat transfer coefficients. The test was set up so that the two large surfaces of a pouch cell were completely insulated, i.e., not cooled. Thus, heat transfer is limited to only go through the smaller side surfaces of the pouch cell. This was then compared to a cell with normal, uniform cooling over the whole cell surface. Default heat transfer coefficients were used for all but the two large pouch surfaces in the limited heat transfer cell.



**Figure 4.17:** Simulation results from differently cooled cells. The default cell uses the default heat transfer coefficients. The reduced heat transfer cell, has its cooling limited by the two largest pouch cell surfaces having heat transfer coefficients of  $0 \text{ W}/(\text{m}^2\text{K})$ .

The simulations presented in Figure 4.17 shows that when a cell has to sustain higher than optimal operating temperatures, its ageing is accelerated [87]. A heat transfer coefficient refinement study was planned to identify the benefits of improved cooling in terms of capacity retention. Figure 4.17 shows that this type of study can be done. For the thermal category, a study was also planned to find the optimal ambient temperature at rest and while cycling. This can be done for any parameter of interest, and it is simply done by changing three lines of code for each parameter, as outlined in Listing B.1. Time consuming simulations have to be conducted to find the optimal value for a given parameter in a given situation, which is why only its potential, and not its execution is shown in Figure 4.17.

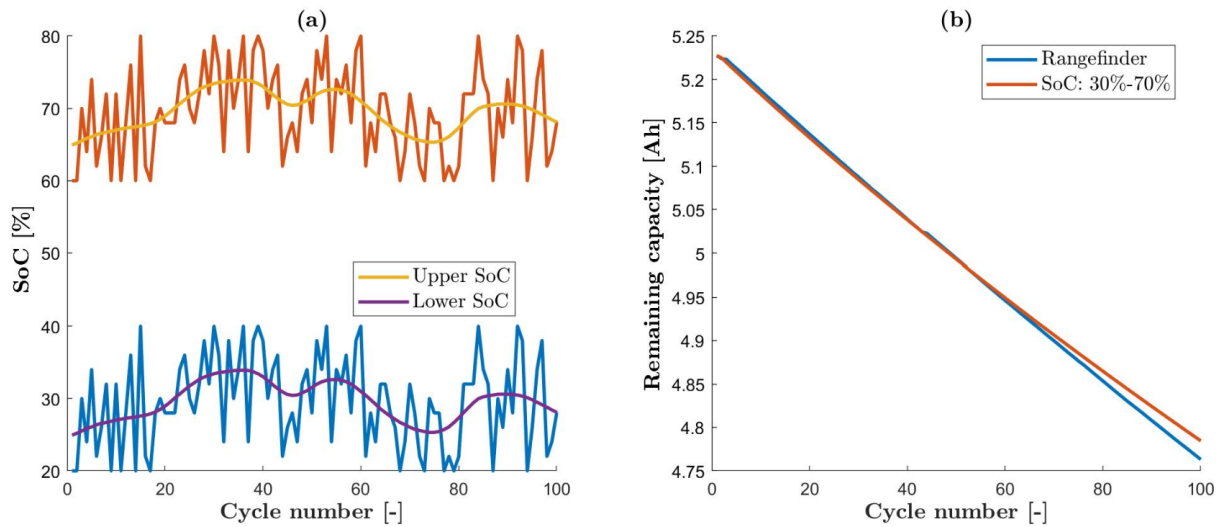
Ferries have relatively strict operating conditions, and because of this many parameters such as the charging C-rate cannot be changed during the day. However, there are some measures that can be taken to prolong cycle life. Tuning the cooling system, and ambient temperature during day, or cycling, and during nighttime, or rest, finding the optimal SoC which the batteries should rest at every night, charging slowly at the end of the day, as opposed to the typical 2C charging rate, and finding the SoC range which is most optimal to operate in. How these factors change over time is also of interest to model so that battery health can be preserved. A motivation for the implementation of the SoC cycling scheme was to be able to compare different SoC ranges and their respective impacts on ageing. With the SoC cycling method implemented, PyBaMM allows for the test of every possible SoC range.

Testing and comparing ranges of equal width, but at different states of charge, then becomes an interesting proposition, as this could be done to find the optimal upper and lower SoC for every cycle. This is interesting for ferries, since they approximately have the same energy requirement per trip, which can be directly translated to the width of an SoC range. This means that the SoC range which reduces ageing the most can be found, and be applied on a per cycle basis. The SoC width for a given amount of energy grows larger as a function of time, due to capacity fade. This can however be accounted for such that the optimal SoC values are found, even as the SoC range grows wider.

There are a reduced amount of battery parameters which can be optimized in ferries with operational battery energy storage systems. This led to the idea of an optimal SoC rangefinder algorithm. To the knowledge of the authors, this has not yet been done by other PyBaMM users, and it has proven difficult to find similar algorithms in the literature. Therefore, the authors have proposed an optimal SoC finder algorithm. It tests SoC values in a predefined range. The algorithm were in this report instructed to exclude the bottom and upper 20% from the optimal SoC search, as these areas are undesirable to cycle in. The algorithm functions by comparing the capacity retention of SoC ranges with equal width, such that only the best SoC values are saved for each cycle. The best SoC values are defined as those that ensure that the cell retains the most amount of capacity.

The best lower SoC is used as the starting point for the next cycle. The cycles are simulated by first charging to an upper SoC, the next step in the cycle is a discharge to a lower SoC, which is defined such that the specified SoC range width is cycled. The step length were set to 2% in the algorithm to reduce the amount of cycles which are simulated per cycle. Therefore, the lower SoC, which controls how deep a battery is discharged in the second step, goes to 20% in the first test and proceeds to 22% in the subsequent test. When the upper SoC reaches 80%, all defined cycles of equal width have been tested for a single cycle. The algorithm then proceeds by using the cycle with the best SoC values as the starting point for the next cycle. The initial version of the algorithm was used to find the optimal SoC for 100 cycles, it produced Figure 4.18.



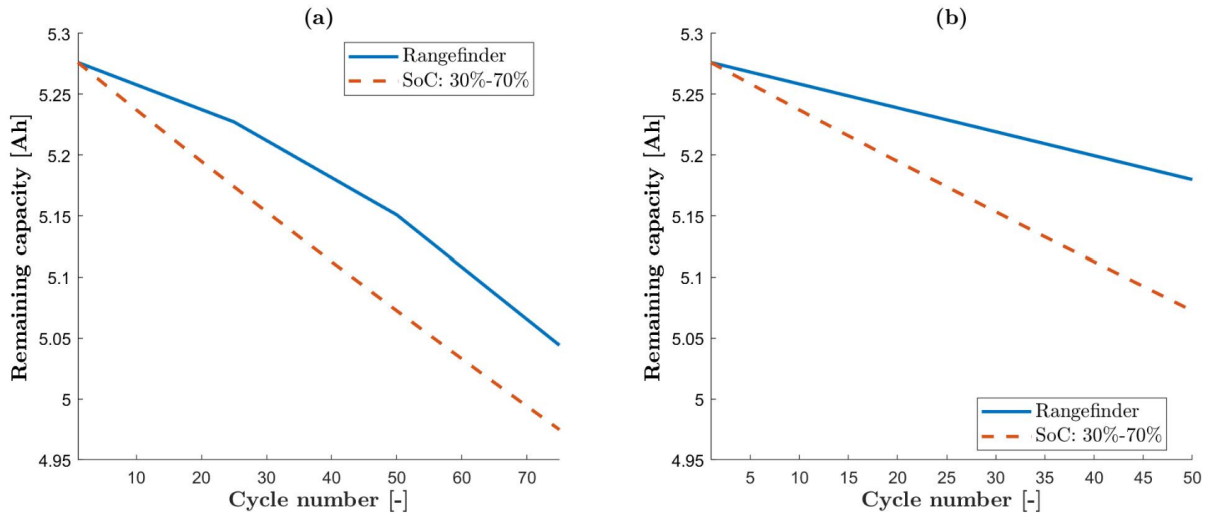


**Figure 4.18:** Results from the initial version of the proposed optimal SoC pathfinder algorithm. (a) shows the optimal SoC values which were found. Two smoothed SoC lines are also presented. (b) shows the capacity retention of a static SoC window between 30% and 70% SoC. A SoC range width of 40% were applied in both simulation schemes.

Figure 4.18a shows that the actual optimal SoC, which is indicated by the red and blue graphs, is very volatile. It spikes up and down throughout the 100 cycles that were simulated. This might indicate that the reversal of some ageing mechanisms, such as the stripping of lithium plates, takes place by applying different conditions to the electrodes in subsequent simulations. A more likely explanation is however that the differences in capacity retention are so small per cycle, that the results become noisy. Two smoothed lines have been produced in Figure 4.18a to show any trends in SoC over time, this is however most useful when thousands of cycles are simulated. However, this was impractical, since this simulation problem was the problem which required the most amount of computing power and memory in this report. It also should be noted that Figure 4.18b shows that the optimal SoC algorithm is beaten by the static SoC range, as it retains the most amount of capacity over 100 cycles. The authors considers that a likely explanation to this, is that the previous cycle gives the current cycle an ineffective lower starting SoC, the noisy SoC over time is also suggested to contribute towards accelerated ageing. With these observations in mind a second version of the proposed algorithm were developed.

The first version of the algorithm was rewritten to address the possible problems identified in Figure 4.18. Hence, the second version improves on the first by making the optimal SoC less volatile over time and by making the starting point of the previous cycle less impactful on the current optimal SoC search. This was done through the common solution of finding the optimal SoC values for multiple cycles at a time, as apposed to finding the best SoC for each cycle, the algorithm has been constructed such that any number of cycles can be run before new optimal SoC values are found. This is done by simulating a set number of cycles with a given upper and lower SoC values, and then saving the capacity retention after the cycles are simulated. The algorithm then backtracks the set number of cycles and starts over from this point by simulating the same cycles with two incrementally larger SoC values, as described in the previous paragraph. When all desired SoC ranges have been simulated, their capacities are compared, and the best solution is saved.

This solution is continued on for the next amount of cycles. This should be able to negate most of the negative consequences which originates from changing optimal SoC too often, as seen in Figure 4.18. Quantifying the optimal number of cycles before new optimal SoC values are found is out of the scope of this report. However, the algorithm has been tested with cycle amounts of 1, 25 and 50, as seen in Figure 4.18a, Figure 4.19a and Figure 4.19b respectively. However, it could not be determined whether 25 or 50 cycles were the best approach, since too few cycles could be conducted with the available computing power. Figure 4.19 shows how the proposed second generation algorithm performs against the same static SoC range of 30% to 70% from Figure 4.18.



**Figure 4.19:** Results from the final version of the proposed optimal SoC pathfinder algorithm. (a) shows the capacity retention when optimal SoC is found for 50 cycles at a time. (b) shows the capacity retention of the LiB cell when optimal SoC is recalculated every 25 cycles. Both alternatives are compared to the static SoC window simulation, which were cycles between a SoC range of 30% and 70% SoC. All simulations used a SoC range width of 40%.

Figure 4.19 shows that capacity retention can be improved by periodically finding new optimal states of charge. Whilst at the same time providing the same amount of energy, as a static SoC range with equal range width. Note that the simulation results presented in Figure 4.19a, only shows 50 cycles, whilst the results in Figure 4.19b, completed 3 searches, resulting in 75 cycles. The results are promising, but are as known limited by the fact that the authors have no way of checking, or inferring that the the "Cell SoC" is as intended, as discussed in Section 4.7. The small number of cycles also limits the credibility of the results presented in Figure 4.19, which can be easily improved by obtaining access to more computing power and memory. The conjectured algorithm has, however, been shown to increase capacity retention in the figures, and could thus be used as a framework for a SoC cycling scheme, which has been verified to function as intended. The algorithm could also be expanded to account for different SoC range widths over time. The simulations are computationally heavy, and because of this, the algorithm could benefit from being optimised to search in a more efficient manner than the simple brute force approach which was applied and is shown in Listing C.1.

After the SEI issue were resolved in Section 4.7 simulations with differing C-rates could be conducted with success, given that the effects of lithium plating were left out of the simulation. The solution for the SEI submodel was however found at a prohibitively late stage, such that no simulations could be conducted to show this, with the exception of Figure 4.16.



## 4.9 Limitations

Throughout the thesis, the group faced several challenges. Initially, the objective of the thesis was to simulate at the battery pack level by utilising PyBaMM and Liionpack. A beta version of Liionpack was released during the early stages of this semester. Since Liionpack was still in its early stages of development, the group decided to focus on using PyBaMM and simulate at the unit cell level. Familiarizing oneself with PyBaMM first, is nonetheless an advantage when using its extension, Liionpack, to simulate larger units. The initial objective of using PyBaMM was to be able to construct a realistic battery model that could accurately predict the ageing of LiB cells that are used in the industry.

PyBaMM offers a unique degree of flexibility and customisation. However, choosing models, submodels, parameters, geometries, discretizations, and solvers can be time consuming and challenging for those who are unfamiliar with battery modelling. Simulations can be made as simple or as complex as desired when using PyBaMM. It should be noted that this is seen as a strength, rather than a weakness. The authors were, however, not interested in constructing simple models out of ease. Rather than this, the authors were interested in building a complex battery model and learning as much as possible about every simulation step. In retrospect, it might not have been feasible to finish this in one semester, where this report only accounted for two-thirds of the semester's content. However, the approach has enabled the authors to gain a much greater appreciation and understanding of battery modelling as a whole, as opposed to choosing simple solutions.

Since PyBaMM is still a relatively young program, some issues have proved difficult to solve. For instance, the source of inspiration on the solution to the IDACalcIC issue, discussed in Section 3.4.4, was found in a COMSOL forum post from 2011 [139]. In the previous experience of the authors, numerous solutions have been easy to locate for the exact same programming issue. However, this was not the case with PyBaMM as it is still young. This challenge will inevitably be resolved as more people encounter issues, makes posts online, and PyBaMM grows.

The group has also been lucky to both participate providing and receiving help from the PyBaMM community. Through their Slack channel and GitHub repository. In the later stages of the semester the group were able to pay this forward by helping newcomers in the Slack channel, as well as uncovering some minor mistakes in the PyBaMM GitHub repository. Throughout this semester multiple individuals have asked about an SoC cycling feature in the Slack PyBaMM channel. The feature has not been implemented, as the developers are involved in the implementation of other features. This resulted in the authors trying to implement this feature on their own. A cycling method that was not as trivial to implement as initially anticipated. The SoC cycling method, which the authors devised, was also not compatible with every PyBaMM functionality, like the `save_at_cycles` argument for instance, which is used to reduce memory usage. Hence, a memory clearing function which aimed to copy this functionality were constructed.

After the battery model was sufficiently customised, it was made evident that memory utilisation would become an issue. Each cycle resulted in 0.1 GiB of memory usage, which meant that 100 GiB of RAM was required to simulate 1 000 cycles. Note that this issue were made more severe due to the authors desire to make a comprehensive battery model. Due to this complexity the DFN model also took 17 hours to solve a simulation of 400 cycles. Compared to the

SPM, the DFN base model took 21 times longer to solve. Therefore, additional computing resources and time were required for the authors to run the most accurate base model, i.e., DFN. Consequently, the SPM<sub>e</sub> and NT base models were applied, and accuracy was sacrificed for the sake of completing more simulations. The simulations were still time consuming, making it difficult to iterate and troubleshoot.

## 5 Conclusion

Following the PyBaMM pipeline, the base battery model was chosen first. Consequently, a base model study was conducted to determine which battery models were the most applicable for the purpose of this report. In the base model study, it was found that the base models produced similar results. While the DFN model is likely to offer the best results, it could not be incorporated because it solved too slowly with the select submodels applied in Table 4.2. Because of this, the NT and SPM<sub>e</sub> base models, which yielded similar results, were chosen for further usage in this report. Using PyBaMM, all ageing submodels were integrated into a single battery model. The initial choice of SEI submodel, the `solvent-diffusion limited` submodel, was shown to produce contrary results. Through an extensive troubleshooting process, described in Section 4.7, it was determined that the `interstitial-diffusion limited` submodel should be applied instead as it corrected the SEI results. SEI film resistance and porosity change were also accounted for by the choice of submodels so that the ageing could be modelled more accurately.

To account for ageing due to lithium plating, its submodel was set to `irreversible`. The authors conducted numerous tests to uncover why contrary results were produced with lithium plating enabled. The authors were able to rule out a range of possible lithium plating issues in Section 4.3.3. Possible next steps in the troubleshooting process, of which there was no time left to conduct, are defined in the subsequent further work section. The `stress-driven LAM` submodel were chosen since no coupled parameters existed for the NMC chemistry. The `Chen2020 NMC-811` base parameter set were considered the most appropriate for the simulations of this report. An analytical approach was taken to determine which additional parameters should be used by the battery model. The most comprehensive set of parameters was chosen and imported. Due to `linesearch` and `IDACalcIC` issues the group had to increase spatial resolution and loosen solver tolerances. Both mesh and solver tolerance refinement studies were conducted to find the most suitable simulation settings. The results of these studies were ideally aligned with the measures that had to be taken to avoid both `linesearch` and `IDACalcIC` errors.

The initial simulation results did not align with published LiB degradation research. Subsequently, 1 out of 2 known simulation issues were uncovered in the troubleshooting process. With lithium plating disabled, final results could be obtained from simulations that varied heat transfer coefficients, SoC range, temperature, and C-rates. A range of analysis were planned on all these parameters to find the operating conditions that prolong battery life the most. Time constraints left the authors to focus on heat transfer coefficients and SoC. A promising proposal for an optimal SoC rangefinder algorithm was constructed. The algorithm was shown to increase capacity retention over simulations with a static SoC window. These results need to be validated by conducting simulations with more cycles. The custom SoC cycling scheme, which was devised by the authors, also needs to be validated, either by reading or inferring `Cell SoC`.

As far as the authors know, no previous NTNU theses have used PyBaMM to model lithium-ion batteries. Making this project a potentially useful tool for future students of PyBaMM. The methods were constructed by condensing multiple PyBaMM resources into a single unit, and included solutions to a range of initial issues the authors faced. Throughout the troubleshooting process, adjustments were made to the initial customised battery model to form a more coherent battery model. Hopefully, the outline of how it was created in the methods, discussion, results, and appendices could accelerate future battery modelling projects by serving as an initial model to iterate from.

## 6 Further Work

This project report is hopefully the start of a series of NTNU PyBaMM theses. There exists few limitations on how can be applied to simulate at the unit cell level. Thus, this report only explores some of the functionality provided by PyBaMM, and a large amount of work is still available for those interested in battery modelling. In addition to this extensive troubleshooting left ambitious project objectives unanswered. The work presented in this report can be iterated on to provide further answers. The following list outlines recommendations from the authors for further work.

- When facing various PyBaMM issues, refer to the solutions presented in the methods from Section 3.4 through Section 3.4.4.
- Use the `interstitial-diffusion limited` SEI submodel, as this has been shown to best match the experimental data. [138]. It has also produced the most coherent SEI simulation results when applied in this report.
- Given that NMC cathode particle mechanics and LAM parameters are not implemented in PyBaMM, individual anode and cathode definitions could be used for the LAM submodel. This is discussed in Section 4.2.4. Alternatively, one could seek to acquire these parameters from the literature, and not directly from PyBaMM itself.
- Try using the spectral volume discretisation to decrease solve time. This might allow for the usage of the DFN model if the solve time is drastically reduced. An added benefit could be an increase in the result accuracy due to the higher spatial resolution. The authors were recommended to try this through personal communication with PhD student Yannick Kuhn [140].
- Uncover the issue with lithium plating such that every ageing submodel which is implemented in PyBaMM can be combined into a single coherent battery model. The next recommended steps to possibly achieve this is to use the `ORegan2021` base parameter set, and the DFN model, if allowed by the solve time [130, 132].
- Further develop the `save_data_to_csv` function, found in Listing A.2, to save complete cycle data for specified cycles.
- Further develop the SoC cycling method to be able to verify that the correct SoC window is cycled in. Alternatively, wait for additional PyBaMM functionality to be able to validate the SoC cycling scheme.
- Further optimize and test the proposed optimal SoC pathfinder algorithm which is found in Listing C.1.
- Continue work on life span extension strategies to more optimally utilize batteries, no matter the application. Whether through optimised operational parameters, or through the expansion of battery packs through the combination of aged and unaged batteries.

## Bibliography

- [1] Roberto Lacal Arantegui and Arnulf Jäger-Waldau. “Photovoltaics and wind status in the European Union after the Paris Agreement”. In: *Renewable and Sustainable Energy Reviews* 81 (2018), pp. 2460–2471. ISSN: 1364-0321. DOI: <https://doi.org/10.1016/j.rser.2017.06.052>.
- [2] Sergio Vazquez et al. “Energy Storage Systems for Transport and Grid Applications”. In: *IEEE Transactions on Industrial Electronics* 57.12 (2010), pp. 3881–3895. DOI: [10.1109/TIE.2010.2076414](https://doi.org/10.1109/TIE.2010.2076414).
- [3] EU. *Energy storage*. URL: [https://energy.ec.europa.eu/topics/research-and-technology/energy-storage\\_en](https://energy.ec.europa.eu/topics/research-and-technology/energy-storage_en) (visited on 04/11/2022).
- [4] Jiangong Zhu et al. “Investigation of lithium-ion battery degradation mechanisms by combining differential voltage analysis and alternating current impedance”. In: *Journal of Power Sources* 448 (2020), p. 227575. ISSN: 0378-7753. DOI: <https://doi.org/10.1016/j.jpowsour.2019.227575>.
- [5] Klima-og miljødepartementet. *Handlingsplan for grønn skipsfart*. Regjeringen.no. June 20, 2019. URL: <https://www.regjeringen.no/no/dokumenter/handlingsplan-for-gronn-skipsfart/id2660877/> (visited on 04/23/2022).
- [6] Henrik Eli Almaas. *Økning i antall ladestasjoner for ferjer og anlegg for landstrøm*. 2021. URL: <https://www.ssb.no/transport-og-reiseliv/sjotransport/artikler/okning-i-antall-ladestasjoner-for-ferjer-og-anlegg-for-landstrom> (visited on 03/29/2022).
- [7] Øivind Fandrem Skotland Christer Heen & Høivik. *Premissanalyser – tiltaksanalyse for utvikling av ferjemarkedet på lang sikt- Rapportnummer 2016-23*. 2016. URL: <https://www.vegvesen.no/globalassets/fag/trafikk/ferje/oslo-economics-rapport-2016-23-premissanalyser.pdf> (visited on 03/30/2022).
- [8] Royal Academy of Engineering. “Future Ship Powering Options - Exploring alternative methods of ship propulsion”. In: (2013), p. 51. URL: <https://www.raeng.org.uk/publications/reports/future-ship-powering-options>.
- [9] Jonas Kristiansen Nøland. *Kartlegging av potensialet for batteridrift på ferger i Norge*. 2016. URL: [https://www.zero.no/wp-content/uploads/2016/05/kartlegging\\_av\\_potensialet\\_for\\_batteridrift\\_paa\\_ferger\\_i\\_norge.pdf](https://www.zero.no/wp-content/uploads/2016/05/kartlegging_av_potensialet_for_batteridrift_paa_ferger_i_norge.pdf) (visited on 03/30/2022).
- [10] Masaki Yoshio. “Lithium-Ion Batteries”. In: Springer, 2009. ISBN: 978-0-387-34444-7. URL: <https://link.springer.com/book/10.1007/978-0-387-34445-4>.
- [11] Da Deng. “Li-ion batteries: basics, progress, and challenges”. In: *Energy Science & Engineering* 3.5 (2015), pp. 385–418. DOI: <https://doi.org/10.1002/ese3.95>.
- [12] Kongsberg Maritime. *Kongsberg integrated hybrid power propulsion system - The full picture*. URL: <https://www.kongsberg.com/maritime/campaign/Hybrid-Solutions/kongsberg-integrated-hybrid-power-propulsion-system> (visited on 03/25/2022).
- [13] DNV - Alternative Fuels Insight. *Overview*. URL: <https://afi.dnv.com/Statistics?repId=0> (visited on 05/03/2022).

- [14] Thomas Imre Cyrille Buiding and Florin Mariasiu. “Battery Thermal Management Systems: Current Status and Design Approach of Cooling Technologies”. In: (2021), p. 32. URL: <https://www.mdpi.com/1996-1073/14/16/4879/htm#> (visited on 05/02/2022).
- [15] Taya Andrew. A. O. L.H. Sawa and L. Winston Zhangb. “Thermal Management of Lithium-ion Battery Pack with Liquid Cooling”. In: (2015), p. 5. URL: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7100176&tag=1> (visited on 05/02/2022).
- [16] Bjarte Hoff. *Ladeløsninger for elektriske ferger*. URL: <https://site.uit.no/ladeteknologi/2019/11/06/ladelosninger-for-elektriske-ferger/> (visited on 03/25/2022).
- [17] Nord Pool AS. *Day-ahead prices*. URL: <https://www.nordpoolgroup.com/en/Market-data1/Dayahead/Area-Prices/NO/Daily1/?view=chart> (visited on 05/03/2022).
- [18] DNV - Alternative Fuels Insight. *Fuel price statistics*. URL: <https://afi.dnv.com/Statistics?repId=4> (visited on 05/03/2022).
- [19] DNV - Alternative Fuels Insight. *Battery Statistics*. URL: <https://afi.dnv.com/Statistics?repId=3> (visited on 05/03/2022).
- [20] Balasubramanian Viswanathan. “Energy Source”. In: (2016), p. 394. DOI: <https://doi.org/10.1016/C2011-0-05048-2>. (Visited on 04/03/2022).
- [21] Nikita Pavlenko et al. “The climate implications of using LNG as a marine fuel”. In: (2020), p. 40. URL: <https://theicct.org/publication/the-climate-implications-of-using-lng-as-a-marine-fuel/>.
- [22] Odne Stokke Burheim. “Engineering Energy Storage”. In: Elsevier, 2017, p. 1. ISBN: 978-0-12-814100-7. URL: <https://www.elsevier.com/books/engineering-energy-storage/burheim/978-0-12-814100-7>.
- [23] Arslan Habib and Chan Sou. “Analytical Review on the Trends and Present Situation of Large-scale Sustainable Energy Storage Technology”. In: *European Journal of Sustainable Development Research* 2.3 (July 9, 2018). ISSN: 25424742. DOI: [10.20897/ejosdr/86200](https://doi.org/10.20897/ejosdr/86200). (Visited on 04/13/2022).
- [24] Yuqing Yang et al. “Battery energy storage system size determination in renewable energy systems: A review”. In: *Renewable and Sustainable Energy Reviews* 91 (2018), pp. 109–125. ISSN: 1364-0321. DOI: <https://doi.org/10.1016/j.rser.2018.03.047>.
- [25] David Connolly. “The Integration of Fluctuating Renewable Energy Using Energy Storage”. English. PhD thesis. 2010.
- [26] Olje- og energidepartementet. *Leveringskvalitetsforskriften*. URL: <https://lovdata.no/dokument/SF/forskrift/2004-11-30-1557> (visited on 04/11/2022).
- [27] Kein Huat Chua, Yun Seng Lim, and Stella Morris. “Energy storage system for peak shaving”. In: *International Journal of Energy Sector Management* 10.1 (Jan. 1, 2016). Publisher: Emerald Group Publishing Limited, pp. 3–18. ISSN: 1750-6220. DOI: [10.1108/IJESM-01-2015-0003](https://doi.org/10.1108/IJESM-01-2015-0003). (Visited on 04/11/2022).

- [28] I. Serban and C. Marinescu. “Battery energy storage system for frequency support in microgrids and with enhanced control features for uninterruptible supply of local loads”. In: *International Journal of Electrical Power & Energy Systems* 54 (2014), pp. 432–441. ISSN: 0142-0615. DOI: <https://doi.org/10.1016/j.ijepes.2013.07.004>.
- [29] K. H. Chua et al. “Energy Storage System for Mitigating Voltage Unbalance on Low-Voltage Networks With Photovoltaic Systems”. In: *IEEE Transactions on Power Delivery* 27.4 (2012), pp. 1783–1790. DOI: [10.1109/TPWRD.2012.2195035](https://doi.org/10.1109/TPWRD.2012.2195035).
- [30] M. A. Kashem and G. Ledwich. “Energy requirement for distributed energy resources with battery energy storage for voltage support in three-phase distribution lines”. In: *Electric Power Systems Research* 77.1 (2007), pp. 10–23. ISSN: 0378-7796. DOI: <https://doi.org/10.1016/j.epsr.2006.01.008>.
- [31] Kristina Hamachi LaCommare and Joseph H. Eto. *Understanding the cost of power interruptions to U.S. electricity consumers*. LBNL–55718, 834270. Sept. 1, 2004, LBNL–55718, 834270. DOI: [10.2172/834270](https://doi.org/10.2172/834270). (Visited on 04/11/2022).
- [32] Alexis Lagrange et al. “Sustainable microgrids with energy storage as a means to increase power resilience in critical facilities: An application to a hospital”. In: *International Journal of Electrical Power & Energy Systems* 119 (2020), p. 105865. ISSN: 0142-0615. DOI: <https://doi.org/10.1016/j.ijepes.2020.105865>.
- [33] Maryam Arbabzadeh et al. “The role of energy storage in deep decarbonization of electricity production”. In: *Nature Communications* 10.1 (July 30, 2019), p. 3413. ISSN: 2041-1723. DOI: [10.1038/s41467-019-11161-5](https://doi.org/10.1038/s41467-019-11161-5).
- [34] Abraham Alem Kebede et al. “A comprehensive review of stationary energy storage devices for large scale renewable energy sources grid integration”. In: *Renewable and Sustainable Energy Reviews* 159 (2022), p. 112213. ISSN: 1364-0321. DOI: <https://doi.org/10.1016/j.rser.2022.112213>.
- [35] A. G. Olabi et al. “Critical review of energy storage systems”. In: *Energy* 214 (2021), p. 118987. ISSN: 0360-5442. DOI: <https://doi.org/10.1016/j.energy.2020.118987>.
- [36] Md Mustafizur Rahman et al. “Assessment of energy storage technologies: A review”. In: *Energy Conversion and Management* 223 (2020), p. 113295. ISSN: 0196-8904. DOI: <https://doi.org/10.1016/j.enconman.2020.113295>.
- [37] Henok Ayele Behabtu et al. “A Review of Energy Storage Technologies’ Application Potentials in Renewable Energy Sources Grid Integration”. In: *Sustainability* 12.24 (2020). ISSN: 2071-1050. DOI: [10.3390/su122410511](https://doi.org/10.3390/su122410511).
- [38] Xing Luo et al. “Overview of current development in electrical energy storage technologies and the application potential in power system operation”. In: *Applied Energy* 137 (2015), pp. 511–536. ISSN: 0306-2619. DOI: <https://doi.org/10.1016/j.apenergy.2014.09.081>.
- [39] Maria C. Argyrou, Paul Christodoulides, and Soteris A. Kalogirou. “Energy storage for electricity generation and related processes: Technologies appraisal and grid scale applications”. In: *Renewable and Sustainable Energy Reviews* 94 (2018), pp. 804–821. ISSN: 1364-0321. DOI: <https://doi.org/10.1016/j.rser.2018.06.044>.



- [40] Jia Liu et al. “Overview on hybrid solar photovoltaic-electrical energy storage technologies for power supply to buildings”. In: *Energy Conversion and Management* 187 (2019), pp. 103–121. ISSN: 0196-8904. DOI: <https://doi.org/10.1016/j.enconman.2019.02.080>.
- [41] Henrik Zsiborács et al. “Intermittent Renewable Energy Sources: The Role of Energy Storage in the European Power System of 2040”. In: *Electronics* 8.7 (2019). ISSN: 2079-9292. DOI: [10.3390/electronics8070729](https://doi.org/10.3390/electronics8070729).
- [42] REN21. *Renewables 2021 Global Status Report*. 2021. URL: [https://www.ren21.net/wp-content/uploads/2019/05/GSR2021\\_Full\\_Report.pdf](https://www.ren21.net/wp-content/uploads/2019/05/GSR2021_Full_Report.pdf) (visited on 04/11/2022).
- [43] IEA. “Renewables 2020 - Analysis and forecast to 2025”. In: (2020), p. 172. URL: [https://iea.blob.core.windows.net/assets/1a24f1fe-c971-4c25-964a-57d0f31eb97b/Renewables\\_2020-PDF.pdf](https://iea.blob.core.windows.net/assets/1a24f1fe-c971-4c25-964a-57d0f31eb97b/Renewables_2020-PDF.pdf) (visited on 04/11/2022).
- [44] IEA. *How rapidly will the global electricity storage market grow by 2026?* URL: <https://www.iea.org/articles/how-rapidly-will-the-global-electricity-storage-market-grow-by-2026> (visited on 04/11/2022).
- [45] Wei He et al. “Technologies and economics of electric energy storages in power systems: Review and perspective”. In: *Advances in Applied Energy* 4 (2021), p. 100060. ISSN: 2666-7924. DOI: <https://doi.org/10.1016/j.adapen.2021.100060>.
- [46] Richard J.D. Tilley. “Understanding Solids The Science of Materials”. In: Wiley, 2004. ISBN: 0-470-85275-5. DOI: [10.1002/0470020849](https://doi.org/10.1002/0470020849).
- [47] Keith B. Oldham. “Electrochemical Science and Technology Fundamentals and Application”. In: Wiley, 2013. ISBN: 9780470710852.
- [48] Vincenzo Caramia and Benedetto Bozzini. “Materials science aspects of zinc–air batteries: a review”. In: *Materials for Renewable and Sustainable Energy* 3.2 (Apr. 3, 2014), p. 28. ISSN: 2194-1467. DOI: <https://doi.org/10.1007/s40243-014-0028-3>.
- [49] Yeru Liang et al. “A review of rechargeable batteries for portable electronic devices”. In: *InfoMat* 1 (Mar. 2019). DOI: [10.1002/inf2.12000](https://doi.org/10.1002/inf2.12000).
- [50] *BU-301a: Types of Battery Cells*. Battery University. Sept. 10, 2010. URL: <https://batteryuniversity.com/article/bu-301a-types-of-battery-cells> (visited on 05/02/2022).
- [51] Markus S. Wahl et al. “The Importance of Optical Fibres for Internal Temperature Sensing in Lithium-ion Batteries during Operation”. In: *Energies* 14.12 (2021). ISSN: 1996-1073. DOI: [10.3390/en14123617](https://doi.org/10.3390/en14123617).
- [52] Robert Bock et al. “Thermal Gradients with Sintered Solid State Electrolytes in Lithium-Ion Batteries”. In: *Energies* 13.1 (2020). ISSN: 1996-1073. DOI: [10.3390/en13010253](https://doi.org/10.3390/en13010253).
- [53] Ingvild Brekke Espedal. “State of Charge Estimation of Lithium-ion Batteries by Neural Networks”. MA thesis. Department of Energy and Process Engineering, 2021, p. 197.
- [54] Phillip Weicker. *A systems Approach to Lithium-Ion Battery Management*. Artech House, 2014. ISBN: 978-1-60807-659-8.
- [55] *BU-204: How do Lithium Batteries Work?* en. Sept. 2010. URL: <https://batteryuniversity.com/article/bu-204-how-do-lithium-batteries-work> (visited on 05/16/2022).



- [56] Reiner Korthauer. “Lithium-Ion Batteries: Basics and Applications”. In: Springer, 2019, p. 1. ISBN: 978-3-662-53069-6. DOI: <https://doi.org/10.1007/978-3-662-53071-9>.
- [57] Manh-Kien Tran et al. “Comparative Study of Equivalent Circuit Models Performance in Four Common Lithium-Ion Batteries: LFP, NMC, LMO, NCA”. In: *Batteries* 7.3 (2021). ISSN: 2313-0105. DOI: [10.3390/batteries7030051](https://doi.org/10.3390/batteries7030051).
- [58] Bruno Scrosati and Jürgen Garche. “Lithium batteries: Status, prospects and future”. In: *Journal of Power Sources* 195.9 (2010), pp. 2419–2430. ISSN: 0378-7753. DOI: <https://doi.org/10.1016/j.jpowsour.2009.11.048>.
- [59] Elixabete Ayerbe et al. “Digitalization of Battery Manufacturing: Current Status, Challenges, and Opportunities”. In: *Advanced Energy Materials* 12.17 (2022), p. 2102696. DOI: <https://doi.org/10.1002/aenm.202102696>.
- [60] Evan Foreman et al. “A Review of Inactive Materials and Components of Flexible Lithium-Ion Batteries”. In: *Advanced Sustainable Systems* 1 (Sept. 2017), p. 1700061. DOI: [10.1002/adsu.201700061](https://doi.org/10.1002/adsu.201700061).
- [61] Silje N. Bryntesen. Personal communication. Mar. 4, 2022.
- [62] Weixiao Ji et al. “Electrode Architecture Design to Promote Charge-Transport Kinetics in High-Loading and High-Energy Lithium-Based Batteries”. In: *Small Methods* 5 (Sept. 2021). DOI: [10.1002/smt.202100518](https://doi.org/10.1002/smt.202100518).
- [63] Seunghun Jung. “Computational study about the effect of electrode morphology on the performance of lithium-ion batteries”. In: *International Journal of Energy Research* 40.8 (2016), pp. 1073–1084. DOI: <https://doi.org/10.1002/er.3501>.
- [64] Benjamin K. Sovacool. “The precarious political economy of cobalt: Balancing prosperity, poverty, and brutality in artisanal and industrial mining in the Democratic Republic of the Congo”. In: *The Extractive Industries and Society* 6.3 (2019), pp. 915–939. ISSN: 2214-790X. DOI: <https://doi.org/10.1016/j.exis.2019.05.018>.
- [65] Matthew Li and Jun Lu. “Cobalt in lithium-ion batteries”. In: *Science* 367.6481 (2020), pp. 979–980. DOI: [10.1126/science.aba9168](https://doi.org/10.1126/science.aba9168).
- [66] Naoki Nitta et al. “Li-ion battery materials: present and future”. In: *Materials Today* 18.5 (2015), pp. 252–264. ISSN: 1369-7021. DOI: <https://doi.org/10.1016/j.mattod.2014.10.040>.
- [67] Chixia Tian, Feng Lin, and Marca M. Doeff. “Electrochemical Characteristics of Layered Transition Metal Oxide Cathode Materials for Lithium Ion Batteries: Surface, Bulk Behavior, and Thermal Properties”. In: *Accounts of Chemical Research* 51.1 (Jan. 16, 2018), pp. 89–96. ISSN: 0001-4842. DOI: [10.1021/acs.accounts.7b00520](https://doi.org/10.1021/acs.accounts.7b00520).
- [68] Ghassan Zubi et al. “The lithium-ion battery: State of the art and future perspectives”. In: *Renewable and Sustainable Energy Reviews* 89 (2018), pp. 292–308. ISSN: 1364-0321. DOI: <https://doi.org/10.1016/j.rser.2018.03.002>.
- [69] Zheng Li et al. “Comparative Study of the Capacity and Rate Capability of  $\text{LiNiyMnyCo}_{1-2y}\text{O}_2$  ( $y = 0.5, 0.45, 0.4, 0.33$ )”. In: *Journal of The Electrochemical Society* 158.5 (2011), A516. DOI: [10.1149/1.3562212](https://doi.org/10.1149/1.3562212).

- [70] Ruifeng Zhang et al. “State of the Art of Lithium-Ion Battery SOC Estimation for Electrical Vehicles”. In: *Energies* 11.7 (2018). ISSN: 1996-1073. DOI: [10.3390/en11071820](https://doi.org/10.3390/en11071820).
- [71] *BU-205: Types of Lithium-ion*. Battery University. Sept. 18, 2010. URL: <https://batteryuniversity.com/article/bu-205-types-of-lithium-ion> (visited on 05/13/2022).
- [72] Petros Selinis and Filippos Farmakis. “Review—A Review on the Anode and Cathode Materials for Lithium-Ion Batteries with Improved Subzero Temperature Performance”. In: *Journal of The Electrochemical Society* 169.1 (Jan. 2022), p. 010526. DOI: [10.1149/1945-7111/ac49cc](https://doi.org/10.1149/1945-7111/ac49cc).
- [73] Gert Berckmans et al. “Analysis of the effect of applying external mechanical pressure on next generation silicon alloy lithium-ion cells”. In: *Electrochimica Acta* 306 (2019), pp. 387–395. ISSN: 0013-4686. DOI: <https://doi.org/10.1016/j.electacta.2019.03.138>.
- [74] Anix Casimir et al. “Silicon-based anodes for lithium-ion batteries: Effectiveness of materials synthesis and electrode preparation”. In: *Nano Energy* 27 (2016), pp. 359–376. ISSN: 2211-2855. DOI: <https://doi.org/10.1016/j.nanoen.2016.07.023>.
- [75] Wei-Jun Zhang. “Lithium insertion/extraction mechanism in alloy anodes for lithium-ion batteries”. In: *Journal of Power Sources* 196.3 (2011), pp. 877–885. ISSN: 0378-7753. DOI: <https://doi.org/10.1016/j.jpowsour.2010.08.114>.
- [76] John B. Goodenough and Youngsik Kim. “Challenges for Rechargeable Li Batteries”. In: *Chemistry of Materials* 22.3 (Feb. 9, 2010). Publisher: American Chemical Society, pp. 587–603. ISSN: 0897-4756. DOI: [10.1021/cm901452z](https://doi.org/10.1021/cm901452z).
- [77] Kang Xu. “Electrolytes: Overview”. In: *Encyclopedia of Electrochemical Power Sources* 5 (Jan. 2009), pp. 51–70.
- [78] John T. Warner. “Chapter 7 - Inactive materials”. In: *Lithium-Ion Battery Chemistries*. Ed. by John T. Warner. Elsevier, 2019, pp. 139–170. ISBN: 978-0-12-814778-8. DOI: <https://doi.org/10.1016/B978-0-12-814778-8.00007-7>.
- [79] Xianke Lin et al. “Lithium Plating Mechanism, Detection, and Mitigation in Lithium-Ion Batteries”. In: *Progress in Energy and Combustion Science* 87 (2021), p. 100953. ISSN: 0360-1285. DOI: <https://doi.org/10.1016/j.pecs.2021.100953>.
- [80] Pengcheng Zhu et al. “A review of current collectors for lithium-ion batteries”. In: *Journal of Power Sources* 485 (2021), p. 229321. ISSN: 0378-7753. DOI: <https://doi.org/10.1016/j.jpowsour.2020.229321>.
- [81] Kyusang Cho et al. “Corrosion study of nickel-coated copper and chromate-coated aluminum for corrosion-resistant lithium-ion battery lead-tab”. In: *Journal of Industrial and Engineering Chemistry* 106 (2022), pp. 537–545. ISSN: 1226-086X. DOI: <https://doi.org/10.1016/j.jiec.2021.11.028>.
- [82] Wen-Yeau Chang. “The state of charge estimating methods for battery: A review”. In: *International Scholarly Research Notices* 2013 (2013).

- [83] Ingvild B. Espedal et al. “Current Trends for State-of-Charge (SoC) Estimation in Lithium-Ion Battery Electric Vehicles”. In: *Energies* 14.11 (2021). ISSN: 1996-1073. DOI: [10.3390/en14113284](https://doi.org/10.3390/en14113284).
- [84] Jacqueline S. Edge et al. “Lithium ion battery degradation: what you need to know”. In: *Phys. Chem. Chem. Phys.* 23.14 (2021), pp. 8200–8221. DOI: [10.1039/D1CP00359C](https://doi.org/10.1039/D1CP00359C).
- [85] Bharat Balagopal and Mo-Yuen Chow. “The state of the art approaches to estimate the state of health (SOH) and state of function (SOF) of lithium Ion batteries”. In: *2015 IEEE 13th International Conference on Industrial Informatics (INDIN)*. 2015, pp. 1302–1307. DOI: [10.1109/INDIN.2015.7281923](https://doi.org/10.1109/INDIN.2015.7281923).
- [86] Christoph R. Birkl et al. “Degradation diagnostics for lithium ion cells”. In: *Journal of Power Sources* 341 (2017), pp. 373–386. ISSN: 0378-7753. DOI: <https://doi.org/10.1016/j.jpowsour.2016.12.011>.
- [87] Anthony Barré et al. “A review on lithium-ion battery ageing mechanisms and estimations for automotive applications”. In: *Journal of Power Sources* 241 (2013), pp. 680–689. ISSN: 0378-7753. DOI: <https://doi.org/10.1016/j.jpowsour.2013.05.040>.
- [88] Peyman Taheri, Scott Hsieh, and Majid Bahrami. “Investigating electrical contact resistance losses in lithium-ion battery assemblies for hybrid and electric vehicles”. In: *Journal of Power Sources* 196.15 (2011), pp. 6525–6533. ISSN: 0378-7753. DOI: <https://doi.org/10.1016/j.jpowsour.2011.03.056>.
- [89] S. Barcellona and L. Piegari. “Effect of current on cycle aging of lithium ion batteries”. en. In: *Journal of Energy Storage* 29 (June 2020), p. 101310. ISSN: 2352-152X. DOI: [10.1016/j.est.2020.101310](https://doi.org/10.1016/j.est.2020.101310). (Visited on 05/01/2022).
- [90] Samuel Pelletier et al. “Battery degradation and behaviour for electric vehicles: Review and numerical analyses of several models”. In: *Transportation Research Part B: Methodological* 103 (2017), pp. 158–187. ISSN: 0191-2615. DOI: <https://doi.org/10.1016/j.trb.2017.01.020>.
- [91] Matthew Pinson and Martin Bazant. “Theory of SEI Formation in Rechargeable Batteries: Capacity Fade, Accelerated Aging and Lifetime Prediction”. In: *Journal of The Electrochemical Society* 160 (Oct. 2012). DOI: [10.1149/2.044302jes](https://doi.org/10.1149/2.044302jes).
- [92] Seong Jin An et al. “The state of understanding of the lithium-ion-battery graphite solid electrolyte interphase (SEI) and its relationship to formation cycling”. In: *Carbon* 105 (2016), pp. 52–76. ISSN: 0008-6223. DOI: <https://doi.org/10.1016/j.carbon.2016.04.008>.
- [93] Laurence Hardwick et al. “An Investigation of the Effect of Graphite Degradation on Irreversible Capacity in Lithium-ion Cells”. In: *Journal of The Electrochemical Society* 155 (Jan. 2008), A442. DOI: [10.1149/1.2903882](https://doi.org/10.1149/1.2903882).
- [94] Rutooj Deshpande et al. “Battery Cycle Life Prediction with Coupled Chemical Degradation and Fatigue Mechanics”. In: *Journal of the Electrochemical Society* 159 (Aug. 2012), A1730–A1738. DOI: [10.1149/2.049210jes](https://doi.org/10.1149/2.049210jes).
- [95] Kun Feng et al. “Silicon-Based Anodes for Lithium-Ion Batteries: From Fundamentals to Practical Applications.” In: *Small* 14 8 (2018). DOI: [10.1002/smll.201702737](https://doi.org/10.1002/smll.201702737).

- [96] Venkat Srinivasan and Kenji Takahashi. “Examination of Graphite Particle Cracking as a Failure Mode in Lithium-Ion Batteries: A Model-Experimental Study - IOPscience”. In: *ournal of The Electrochemical Society* 162.4 (Jan. 2015). URL: <https://iopscience.iop.org/article/10.1149/2.0281504jes/meta> (visited on 04/21/2022).
- [97] Yi Cui. *Study Finds a Way to Prevent Fires in Next-Generation Lithium Batteries*. en. 2015. URL: <https://www6.slac.stanford.edu/news/2015-06-17-study-finds-way-prevent-fires-next-generation-lithium-batteries.aspx> (visited on 05/05/2022).
- [98] Thomas Waldmann et al. “Temperature dependent ageing mechanisms in Lithium-ion batteries – A Post-Mortem study”. en. In: *Journal of Power Sources* 262 (Sept. 2014), pp. 129–135. ISSN: 0378-7753. DOI: 10.1016/j.jpowsour.2014.03.112. (Visited on 04/21/2022).
- [99] J. Vetter et al. “Ageing mechanisms in lithium-ion batteries”. In: *Journal of Power Sources* 147.1 (2005), pp. 269–281. ISSN: 0378-7753. DOI: <https://doi.org/10.1016/j.jpowsour.2005.01.006>.
- [100] Zachary Ruff, Chao Xu, and Clare P. Grey. “Transition Metal Dissolution and Degradation in NMC811-Graphite Electrochemical Cells”. en. In: *Journal of The Electrochemical Society* 168.6 (June 2021). Publisher: The Electrochemical Society, p. 060518. ISSN: 1945-7111. DOI: 10.1149/1945-7111/ac0359. URL: <https://doi.org/10.1149/1945-7111/ac0359> (visited on 05/02/2022).
- [101] Mehdi Jafari, Khalid Khan, and Lucia Gauchia. “Deterministic models of Li-ion battery aging: It is a matter of scale”. In: *Journal of Energy Storage* 20 (2018), pp. 67–77. ISSN: 2352-152X. DOI: <https://doi.org/10.1016/j.est.2018.09.002>.
- [102] Johannes Schmalstieg et al. “A holistic aging model for Li(NiMnCo)O<sub>2</sub> based 18650 lithium-ion batteries”. In: *Journal of Power Sources* 257 (2014), pp. 325–334. ISSN: 0378-7753. DOI: <https://doi.org/10.1016/j.jpowsour.2014.02.012>.
- [103] Shunli Wang et al. “Online dynamic equalization adjustment of high-power lithium-ion battery packs based on the state of balance estimation”. In: *Applied Energy* 166 (2016), pp. 44–58. ISSN: 0306-2619. DOI: <https://doi.org/10.1016/j.apenergy.2016.01.013>.
- [104] Sebastian Paul et al. “Analysis of ageing inhomogeneities in lithium-ion battery systems”. en. In: *Journal of Power Sources* 239 (Oct. 2013), pp. 642–650. ISSN: 0378-7753. DOI: 10.1016/j.jpowsour.2013.01.068. (Visited on 05/14/2022).
- [105] Xianzhi Gong, Rui Xiong, and Chunting Chris Mi. “Study of the Characteristics of Battery Packs in Electric Vehicles With Parallel-Connected Lithium-Ion Battery Cells”. In: *IEEE Transactions on Industry Applications* 51.2 (2015), pp. 1872–1879. DOI: 10.1109/TIA.2014.2345951.
- [106] Thomas Bruen and James Marco. “Modelling and experimental evaluation of parallel connected lithium ion cells for an electric vehicle battery system”. In: *Journal of Power Sources* 310 (2016), pp. 91–101. ISSN: 0378-7753. DOI: <https://doi.org/10.1016/j.jpowsour.2016.01.001>.
- [107] Andrea Cordoba-Arenas et al. “Aging Propagation in Advanced Battery Systems: Preliminary Results.” In: *IFAC Proceedings Volumes* 46.21 (2013), pp. 313–318. ISSN: 1474-6670. DOI: <https://doi.org/10.3182/20130904-4-JP-2042.00122>.

- [108] V. Sulzer et al. “Python Battery Mathematical Modelling (PyBaMM)”. In: *Journal of Open Research Software* 9.1 (2021), p. 14. DOI: <http://doi.org/10.5334/jors.309>.
- [109] SG Marquis. “Long-Term Degradation of Lithium-ion Batteries”. PhD thesis. University of Oxford, 2022. URL: <https://ora.ouls.ox.ac.uk/objects/uuid:8afdcc34-cc42-48ba-b316-96a6d0f33a45>.
- [110] *PyBaMM*. URL: <https://www.pybamm.org/pybamm> (visited on 04/25/2022).
- [111] *PyBaMM Team*. GitHub. URL: <https://github.com/pybamm-team> (visited on 04/25/2022).
- [112] 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 (Jan. 2020), p. 080534. DOI: [10.1149/1945-7111/ab9050](https://doi.org/10.1149/1945-7111/ab9050).
- [113] Simon E. J. O’Kane et al. “Physical Origin of the Differential Voltage Minimum Associated with Lithium Plating in Li-Ion Batteries”. In: *Journal of The Electrochemical Society* 167.9 (Jan. 2020), p. 090540. DOI: [10.1149/1945-7111/ab90ac](https://doi.org/10.1149/1945-7111/ab90ac).
- [114] Valentin Sulzer et al. *Python Battery Mathematical Modelling (PyBaMM)*. Version 22.4. June 2021. DOI: [10.5334/jors.309](https://doi.org/10.5334/jors.309). (Visited on 05/16/2022).
- [115] Felipe Andrés Torres Quintero. “Development of an Evaluation Module to Support the Application of Multi-Stage NMPC in Chemical Processes”. In: (2018), p. 93. URL: <https://www.cervantesvirtual.com/obra/development-of-an-evaluation-module-to-support-the-application-of-multi-stage-nmpc-in-chemical-processes-desarrollo-de-un-modulo-de-evaluacion-para-soportar-la-aplicacion-de-nmpc-1078179/?msclkid=10c62c49d07811ec98fb6943aadf87cc>.
- [116] *pybamm-team/liionpack*. Apr. 10, 2022. URL: <https://github.com/pybamm-team/liionpack> (visited on 05/02/2022).
- [117] Valentin Sulzer et al. *Using crack submodels in PyBaMM*. June 2021. URL: [https://colab.research.google.com/github/pybamm-team/PyBaMM/blob/develop/examples/notebooks/models/submodel\\_cracking\\_DFN\\_or\\_SPM.ipynb](https://colab.research.google.com/github/pybamm-team/PyBaMM/blob/develop/examples/notebooks/models/submodel_cracking_DFN_or_SPM.ipynb) (visited on 05/16/2022).
- [118] Harry J. Ploehn, Premanand Ramadass, and Ralph E. White. “Solvent Diffusion Model for Aging of Lithium-Ion Battery Cells”. In: *Journal of The Electrochemical Society* 151.3 (2004), A456. DOI: [10.1149/1.1644601](https://doi.org/10.1149/1.1644601).
- [119] Xiao-Guang Yang et al. “Modeling of lithium plating induced aging of lithium-ion batteries: Transition from linear to nonlinear aging”. In: *Journal of Power Sources* 360 (2017), pp. 28–40. ISSN: 0378-7753. DOI: <https://doi.org/10.1016/j.jpowsour.2017.05.110>.
- [120] Xiao-Guang Yang et al. “A look into the voltage plateau signal for detection and quantification of lithium plating in lithium-ion cells”. In: *Journal of Power Sources* 395 (Aug. 15, 2018), pp. 251–261. ISSN: 0378-7753. DOI: [10.1016/j.jpowsour.2018.05.073](https://doi.org/10.1016/j.jpowsour.2018.05.073).
- [121] Ralph Bednorz and Tanja Gewald. “Investigation of the Effects of Charging Processes on Lithium-Ion Cells with SiC Anodes at Low Temperatures”. In: *Batteries* 6.2 (2020). ISSN: 2313-0105. DOI: [10.3390/batteries6020034](https://doi.org/10.3390/batteries6020034).



- [122] Godfrey Sikha, Branko N. Popov, and Ralph E. White. “Effect of Porosity on the Capacity Fade of a Lithium-Ion Battery”. In: *Journal of The Electrochemical Society* 151.7 (2004), A1104. DOI: [10.1149/1.1759972](https://doi.org/10.1149/1.1759972).
- [123] Rutooj Deshpande et al. “Battery Cycle Life Prediction with Coupled Chemical Degradation and Fatigue Mechanics”. In: *Journal of The Electrochemical Society* 159.10 (2012), A1730–A1738. DOI: [10.1149/2.049210jes](https://doi.org/10.1149/2.049210jes).
- [124] Weilong Ai et al. “Electrochemical Thermal-Mechanical Modelling of Stress Inhomogeneity in Lithium-Ion Pouch Cells”. In: *Journal of The Electrochemical Society* 167.1 (Oct. 2019), p. 013512. DOI: [10.1149/2.0122001jes](https://doi.org/10.1149/2.0122001jes).
- [125] Xuanchen Zhu et al. “The diffusion induced stress and cracking behaviour of primary particle for Li-ion battery electrode”. In: *International Journal of Mechanical Sciences* 178 (2020), p. 105608. ISSN: 0020-7403. DOI: <https://doi.org/10.1016/j.ijmecsci.2020.105608>.
- [126] Jorn M. Reniers, Grietus Mulder, and David A. Howey. “Review and Performance Comparison of Mechanical-Chemical Degradation Models for Lithium-Ion Batteries”. In: *Journal of The Electrochemical Society* 166.14 (2019), A3189–A3200. DOI: [10.1149/2.0281914jes](https://doi.org/10.1149/2.0281914jes).
- [127] Bernhard Rieger et al. “A New Method to Model the Thickness Change of a Commercial Pouch Cell during Discharge”. In: *Journal of The Electrochemical Society* 163.8 (2016), A1566–A1575. DOI: [10.1149/2.0441608jes](https://doi.org/10.1149/2.0441608jes).
- [128] Valentin Sulzer et al. *Thermal models(PyBaMM)*. Version 22.4. June 2021. URL: <https://github.com/pybamm-team/PyBaMM/blob/develop/examples/notebooks/models/thermal-models.ipynb> (visited on 05/16/2022).
- [129] Peyman Mohtat et al. “Differential Expansion and Voltage Model for Li-ion Batteries at Practical Charging Rates”. In: *Journal of The Electrochemical Society* 167.11 (Jan. 2020), p. 110561. DOI: [10.1149/1945-7111/aba5d1](https://doi.org/10.1149/1945-7111/aba5d1).
- [130] Kieran O’Regan et al. “Thermal-electrochemical parameters of a high energy lithium-ion cylindrical battery”. In: *ChemRxiv* (Jan. 2022). DOI: [10.26434/chemrxiv-2022-d2q4n](https://doi.org/10.26434/chemrxiv-2022-d2q4n).
- [131] Shanshan Xu et al. “Evolution of Dead Lithium Growth in Lithium Metal Batteries: Experimentally Validated Model of the Apparent Capacity Loss”. In: *Journal of The Electrochemical Society* 166.14 (2019), A3456–A3463. DOI: [10.1149/2.0991914jes](https://doi.org/10.1149/2.0991914jes).
- [132] Valentin Sulzer. “Run simulations with O’Regan 2021 parameter set (LG M50)”. In: (June 2021). URL: <https://github.com/pybamm-team/PyBaMM> (visited on 05/18/2022).
- [133] P. Ramadass et al. “Development of First Principles Capacity Fade Model for Li-Ion Cells”. In: 151.2 (2004), A196. DOI: [10.1149/1.1634273](https://doi.org/10.1149/1.1634273).
- [134] Fabian Single, Arnulf Latz, and Birger Horstmann. “Identifying the Mechanism of Continued Growth of the Solid–Electrolyte Interphase”. In: *ChemSusChem* 11.12 (2018), pp. 1950–1955. DOI: [10.1002/cssc.201800077](https://doi.org/10.1002/cssc.201800077).
- [135] M. Safari et al. “Multimodal Physics-Based Aging Model for Life Prediction of Li-Ion Batteries”. In: *Journal of The Electrochemical Society* 156.3 (2009), A145. DOI: [10.1149/1.3043429](https://doi.org/10.1149/1.3043429).

- [136] J. G. Qu, Z. Y. Jiang, and J. F. Zhang. “Investigation on lithium-ion battery degradation induced by combined effect of current rate and operating temperature during fast charging”. en. In: *Journal of Energy Storage* 52 (Aug. 2022), p. 104811. ISSN: 2352-152X. DOI: [10.1016/j.est.2022.104811](https://doi.org/10.1016/j.est.2022.104811). (Visited on 05/15/2022).
- [137] Jiagang Xu et al. “Electrode Side Reactions, Capacity Loss and Mechanical Degradation in Lithium-Ion Batteries”. In: *Journal of The Electrochemical Society* 162.10 (2015), A2026–A2035. DOI: [10.1149/2.0291510jes](https://doi.org/10.1149/2.0291510jes).
- [138] Prof.Dr. Latz Arnulf Single Fabian and Dr. Horstmann Birger. “Identifying the Mechanism of Continued Growth of the Solid–Electrolyte Interphase”. In: (2018). DOI: <https://doi.org/10.1002/cssc.201800077>.
- [139] *Error message: Nonlinear solver did not converge*. URL: <https://www.comsol.com/forum/thread/15647/Error-message-Nonlinear-solver-did-not-converge> (visited on 05/22/2022).
- [140] Valentin Sulzer et al. *compare\_spectral\_volume.py*. June 2021. URL: [https://github.com/pybamm-team/PyBaMM/blob/develop/examples/scripts/compare\\_spectral\\_volume.py](https://github.com/pybamm-team/PyBaMM/blob/develop/examples/scripts/compare_spectral_volume.py) (visited on 05/17/2022).
- [141] *tracemalloc — Trace memory allocations — Python 3.10.4 documentation*. URL: <https://docs.python.org/3/library/tracemalloc.html> (visited on 05/20/2022).

