Lifetime estimation of lithium-ion batteries for stationary energy storage systems
<table>
<thead>
<tr>
<th>Abbreviations</th>
<th>Full Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>AFM</td>
<td>Atomic force microscopy</td>
</tr>
<tr>
<td>ANN</td>
<td>Artificial neural network</td>
</tr>
<tr>
<td>ARIMA</td>
<td>Autoregressive integrated moving average</td>
</tr>
<tr>
<td>BMS</td>
<td>Battery management system</td>
</tr>
<tr>
<td>CC-CV</td>
<td>Constant charge-constant voltage</td>
</tr>
<tr>
<td>CDKF</td>
<td>Central difference Kalman filter</td>
</tr>
<tr>
<td>DEC</td>
<td>Diethyl carbonate</td>
</tr>
<tr>
<td>DEKF</td>
<td>Duel extended Kalman filter</td>
</tr>
<tr>
<td>DMC</td>
<td>Dimethyl carbonate</td>
</tr>
<tr>
<td>DoD</td>
<td>Depth of discharge</td>
</tr>
<tr>
<td>EC</td>
<td>Ethylene carbonate</td>
</tr>
<tr>
<td>EKF</td>
<td>Extended Kalman filter</td>
</tr>
<tr>
<td>EoL</td>
<td>End of life</td>
</tr>
<tr>
<td>ESS</td>
<td>Energy storage system</td>
</tr>
<tr>
<td>ESVEKF</td>
<td>Enhanced state vector extended Kalman filter</td>
</tr>
<tr>
<td>EV</td>
<td>Electric vehicle</td>
</tr>
<tr>
<td>HEV</td>
<td>Hybrid electric vehicle</td>
</tr>
<tr>
<td>ICA</td>
<td>Incremental capacity analysis</td>
</tr>
<tr>
<td>LCO</td>
<td>Lithium cobalt oxide</td>
</tr>
<tr>
<td>LFP</td>
<td>Lithium iron phosphate</td>
</tr>
<tr>
<td>Li-ion</td>
<td>Lithium-ion</td>
</tr>
<tr>
<td>LMO</td>
<td>Lithium manganese oxide</td>
</tr>
<tr>
<td>LTO</td>
<td>Lithium titanate oxide</td>
</tr>
<tr>
<td>NCA</td>
<td>Lithium nickel cobalt aluminum oxide</td>
</tr>
<tr>
<td>NMC</td>
<td>Lithium nickel manganese cobalt oxide</td>
</tr>
<tr>
<td>OCV</td>
<td>Open circuit voltage</td>
</tr>
<tr>
<td>P2D</td>
<td>Pseudo-2D</td>
</tr>
<tr>
<td>PC</td>
<td>Propylene carbonate</td>
</tr>
<tr>
<td>PDE</td>
<td>Partial differential equation</td>
</tr>
<tr>
<td>RC</td>
<td>Resistor-capacitor</td>
</tr>
<tr>
<td>RLS</td>
<td>Recursive least squares</td>
</tr>
<tr>
<td>RuL</td>
<td>Remaining useful life</td>
</tr>
<tr>
<td>RW</td>
<td>Random walk</td>
</tr>
<tr>
<td>SEI</td>
<td>Solid electrolyte interface</td>
</tr>
<tr>
<td>SNN</td>
<td>Structured neural network</td>
</tr>
<tr>
<td>SoC</td>
<td>State of charge</td>
</tr>
<tr>
<td>SOH</td>
<td>State of health</td>
</tr>
<tr>
<td>SPKF</td>
<td>Sigma point Kalman filter</td>
</tr>
<tr>
<td>SPM</td>
<td>Single particle model</td>
</tr>
<tr>
<td>STEM</td>
<td>Scanning transmission electron microscopy</td>
</tr>
<tr>
<td>SWDEKF</td>
<td>Single weight dual extended Kalman filter</td>
</tr>
<tr>
<td>UKF</td>
<td>Unscented Kalman filter</td>
</tr>
<tr>
<td>VC</td>
<td>Vinylene carbonate</td>
</tr>
</tbody>
</table>
With the continuing transition to renewable inherently intermittent energy sources like solar- and wind
power, electrical energy storage will become progressively more important to manage energy
production and demand. A key technology in this area is Li-ion batteries. To operate these batteries
efficiently, there is a need for monitoring of the current battery state, including parameters such as
state of charge and state of health, to ensure that adequate safety and performance is maintained.
Furthermore, such monitoring is a step towards the possibility of the optimization of battery usage
such as to maximize battery lifetime and/or return on investment. Unfortunately, possible online
measurements during actual operation of a lithium-ion battery are typically limited to current, voltage
and possibly temperature, meaning that direct measurement of battery status is not feasible. To
overcome this, battery modeling and various regression methods may be used. Several of the most
common regression algorithms suggested for estimation of battery state of charge and state of health
are based on Kalman filtering. While these methods have shown great promise, there currently exist
no thorough analysis of the impact of so-called filter tuning on the effectiveness of these algorithms in
Li-ion battery monitoring applications, particularly for state of health estimation. In addition, the
effects of only adjusting the cell capacity model parameter for aging effects, a relatively common
approach in the literature, on overall state of health estimation accuracy is also in need of
investigation.

In this work, two different Kalman filtering methods intended for state of charge estimation: the
extended Kalman filter and the extended adaptive Kalman filter, as well as three intended for state of
health estimation: the dual extended Kalman filter, the enhanced state vector extended Kalman filer,
and the single weight dual extended Kalman filer, are compared from accuracy, performance, filter
tuning and practical usability standpoints. All algorithms were used with the same simple one resistor-
capacitor equivalent circuit battery model. The Li-ion battery data used for battery model development
and simulations of filtering algorithm performance was the “Randomized Battery Usage Data Set”
obtained from the NASA Prognostics Center of Excellence.

It is found that both state of charge estimators perform similarly in terms of accuracy of state of charge
estimation with regards to reference values, easily outperforming the common Coulomb counting
approach in terms of precision, robustness and flexibility. The adaptive filter, while computationally
more demanding, required less tuning of filter parameters relative to the extended Kalman filter to
achieve comparable performance and might therefore be advantageous from a robustness and
usability perspective. Amongst the state of health estimators, the enhanced state vector approach was
found to be most robust to initialization and was also least taxing computationally. The single weight
filter could be made to achieve comparable results with careful, if time consuming, filter tuning. The
full dual extended Kalman filter has the advantage of estimating not only the cell capacity but also the
internal resistance parameters. This comes at the price of slow performance and time consuming filter
tuning, involving 17 parameters. It is however shown that long-term state of health estimation is
superior using this approach, likely due to the online adjustment of internal resistance parameters.
This allows the dual extended Kalman filter to accurately estimate the SoH over a full test representing
more than a full conventional battery lifetime. The viability of only adjusting the capacity in online
monitoring approaches therefore appears questionable. Overall the importance of filter tuning is
found to be substantial, especially for cases of very uncertain starting battery states and
characteristics.
ACKNOWLEDGEMENTS

I would like to give my deepest gratitude to all those who made this thesis a reality. To my supervisors Longcheng Liu at the department of Chemical Engineering, KTH and Jinying Yan at Vattenfall R&D for their unrelenting support throughout the work and for the opportunity to take part in the project in the first place. Thank you for all the insightful input and fruitful discussions.

My thanks go out to my examiner Matthäus Bäbler for all the help and assistance I ever needed.

I would also like to thank everybody at Vattenfall R&D and KTH who helped and supported me in case of need.

Thanks to Jonas Ricknell for insightful comments on the work.

Finally I would like to thank my mother and my father for always supporting me and for providing motivation during rough times. I could never have made it without you.

Joakim Andersson, Stockholm, June 2017.
# TABLE OF CONTENTS

1  Introduction...................................................................................................................... 1

   1.1  Background.................................................................................................................... 2

      1.1.1  Historical perspective and outlook on lithium-ion batteries .................................. 2

      1.1.2  Lithium-ion batteries in stationary applications ...................................................... 5

      1.1.3  Main components of Li-ion batteries ....................................................................... 6

      1.1.4  Chemistry of Li-ion batteries .................................................................................. 7

         1.1.4.1  Cathode materials ......................................................................................... 8

         1.1.4.2  Anode materials ........................................................................................... 12

         1.1.4.3  Additives ........................................................................................................ 13

      1.1.5  Battery management systems ............................................................................... 13

   1.2  Lithium-ion battery aging ............................................................................................ 14

      1.2.1  Aging of electrolyte ................................................................................................ 16

      1.2.2  Aging mechanisms at anode.................................................................................. 16

      1.2.3  Aging mechanisms at cathode............................................................................... 19

         1.2.3.1  Chemistry-specific cathode aging processes .................................................... 20

      1.2.4  Main aging impact factors ...................................................................................... 22

         1.2.4.1  Calendar aging ............................................................................................... 23

         1.2.4.2  Cycle aging .................................................................................................... 25

      1.2.5  Summary of Li-ion battery aging impact factors ...................................................... 26

   1.3  Lithium-ion battery modeling ....................................................................................... 28

      1.3.1  Empirical models .................................................................................................. 29

      1.3.2  Electrochemical models ....................................................................................... 29

      1.3.3  Equivalent circuit models ..................................................................................... 30

   1.4  State of charge estimation ............................................................................................ 32

   1.5  State of health and remaining useful life estimation ..................................................... 34

2  Scope and aim of thesis ...................................................................................................... 38

3  Methodology ...................................................................................................................... 39

   3.1  Equivalent circuit battery model development ............................................................ 39

      3.1.1  Capacity determination ......................................................................................... 40

      3.1.2  Open circuit voltage – state of charge expression .................................................. 42

      3.1.3  Identification of equivalent circuit model parameters .......................................... 46

   3.2  State of health estimation validation and considerations ............................................... 50

   3.3  Kalman filtering ............................................................................................................ 53

      3.3.1  Extended Kalman filter ........................................................................................ 57
3.3.2 Adaptive extended Kalman filter .................................................. 59
3.3.3 Dual extended Kalman filter .......................................................... 59
3.3.4 Enhanced state vector extended Kalman filter .............................. 63
3.3.5 Single weight dual extended Kalman filter ................................... 64

4 Results and discussion ........................................................................ 65
  4.1 Extended Kalman filter .................................................................... 65
  4.2 Adaptive extended Kalman filter ...................................................... 76
  4.3 Dual extended Kalman filter ............................................................ 79
  4.4 Enhanced state vector extended Kalman filter .............................. 88
  4.5 Single weight dual extended Kalman filter ................................... 92
  4.6 Comparison of algorithms and summary ...................................... 95
    4.6.1 State of charge algorithms ..................................................... 95
    4.6.2 State of health estimation algorithms .................................... 96

5 Conclusions and future work ............................................................... 103

6 References ......................................................................................... 104

7 Appendices ........................................................................................ 111
  Appendix 1: Code for plotting and organization of battery diagnostic data ........................................ 111
  Appendix 2: Code for extended Kalman filter ....................................... 115
  Appendix 3: Code for adaptive extended Kalman filter .......................... 117
  Appendix 4: Code for dual extended Kalman filter ............................ 119
  Appendix 5: Code for enhanced state vector extended Kalman filter ........ 122
  Appendix 6: Code for single weight dual extended Kalman filter ........... 123
List of figures

Figure 1: Price development of Li-ion batteries 2005-2030 [21] ......................................................... 4
Figure 2: Gravimetric energy density and specific power of different available battery technologies [2]. ......................................................... 4
Figure 3: Simple illustration of the original LiCoO$_2$-based Li-ion cell. Electrodes, current collectors and separator are all submerged in the electrolyte solution [36]. ......................................................... 8
Figure 4: The three crystal structures of common Li-ion battery cathode materials. Green dots represent Li-ions or Li-ion intercalation sites [37] ......................................................... 9
Figure 5: Comparison of market share and size of Li-ion battery cathode materials in 1995 and 2010 [30]. ......................................................... 9
Figure 6: Composition dependence for performance characteristics of NMC cathodes [46]. .......... 11
Figure 7: Graphical summary of the most important Li-ion battery degradation mechanisms [68] .... 15
Figure 8: Example of anode side reaction leading to SEI formation via formation of DMDOHC [73]. . 17
Figure 9: Main degradation mechanisms at graphite-based anodes [40] .................................................. 18
Figure 10: Main Li-ion battery cathode aging processes [40] ................................................................. 20
Figure 11: Mechanism of the dissolution of manganese from LMO cathode [40] .................................. 21
Figure 12: Aging mechanisms of layered transition metal oxides [40] ..................................................... 22
Figure 13: Effect of storage temperature on aging in terms of percentage of capacity loss. All batteries were stored at 50 % SoC during testing [85] ......................................................... 23
Figure 14: Arrhenius plot for high and low temperature behavior. r is the rate of reaction [45] .......... 23
Figure 15: Calendar aging of LTO anode compared to graphite anode. ASI=area specific impedance. The impedance can be seen as the equivalent of resistance for alternating current [82] .......... 25
Figure 16: A simple equivalent circuit model consisting of a voltage source, dependent on the SoC, and a resistor in series ......................................................... 30
Figure 17: n resistor-capacitor equivalent circuit model ................................................................. 31
Figure 18: A 1 resistor-capacitor equivalent circuit model ................................................................. 32
Figure 19: Determining RuL by extrapolating SoH measurements [64] .................................................. 36
Figure 20: Reference discharge voltage profiles for determination of cell capacity changes over the course of battery testing for RW9 ......................................................... 40
Figure 21: Capacity degradation during testing of RW9. Note the total timespan of around half a year. ........................................................................................................... 42
Figure 22: OCV-time relationship for low current discharge test of RW9 ................................................ 42
Figure 23: OCV-SoC for RW9 battery .................................................................................................. 43
Figure 24: Comparison of polynomial and combined model for OCV-SoC curve fitting ................. 45
Figure 25: Fitted OCV-SoC polynomial is clearly stable while the combined model "spikes". ......... 45
Figure 26: All pulsed discharge tests for determination of ECM parameter values for RW9 .......... 46
Figure 27: Ballpark estimation of ECM parameters from pulsed discharge ........................................ 47
Figure 28: 1 RC ECM model fitting results for first pulsed discharge test of RW9 ......................... 48
Figure 29: Modeling error in first pulsed discharge test of RW9 using 1 RC ECM ......................... 49
Figure 30: Difference in modelling error when not adjusting for changing capacitance ............... 50
Figure 31: Voltage, current and temperature over the first 100 RW steps for RW9 ....................... 52
Figure 32: Negative SoC by Coulomb counting in RW phase 10 for battery RW9 ......................... 52
Figure 33: The Kalman filtering principle [114] .................................................................................. 54
Figure 34: The DEKF algorithm. Solid lines represent the paths of state- and weight vectors while dashed lines are the flows of the covariance matrices [123] ................................. 63
Figure 35: SoC estimated by EKF over first 100 RW steps of battery RW9 ....................................... 66
Figure 36: SoC estimated EKF during end of first RW phase of battery RW9 ................................. 67
Figure 37: SoC estimation by EKF with unadjusted ECM parameters for RW phase 5, battery RW9.
Figure 38: EKF converging to correct SoC value despite incorrect starting values.
Figure 39: Differences in convergence rate to Coulomb counting reference value for varying starting
SoC covariance.
Figure 40: Eventual convergence of EKF SoC estimate despite poor choice of initial SoC covariance.
Figure 41: EKF SoC estimation with different measurement noise covariances.
Figure 42: Overall trend of rms SoC error versus logarithm of measurement noise covariance for cases
of noisy signals and correct starting SoC for the EKF.
Figure 43: Cell voltage predictions of the EKF for the case of a noisy voltage signal using different
measurement noise covariance matrices.
Figure 44: Error bounds per (20) of SoC estimate using EKF.
Figure 45: The stabilization of SoC covariance in the EKF, indicating estimation convergence.
Figure 46: SoC estimation for 100 first RW steps using AEKF with Coulomb counting reference.
Figure 47: Rapid convergence of AEKF SoC estimate for various initial process noise covariance
matrices.
Figure 48: The quick converging of the AEKF for various starting measurement covariances.
Figure 49: Capacity estimation over first full RW phase of RW9.
Figure 50: SoC- (left) and capacity (right) estimates by DEKF when initial guesses of both SoC and
capacity are poor for 100 first RW steps of first RW phase of RW9.
Figure 51: Rapid convergence of both internal resistance parameter estimates using the DEKF despite
poor initial guesses.
Figure 52: Measured cell voltage for RW cycling using CC-CV mode charging for RW2. Note the periods
of constant voltage at 4.2 V.
Figure 53: Current over time for cycling of battery RW2.
Figure 54: SoC by Coulomb counting and DEKF for CC-CV mode charge cycling of RW2.
Figure 55: Convergence of capacity estimates despite large disparity in starting values using DEKF and
data from battery RW2.
Figure 56: Quick convergence of capacity estimates with different initial guesses of capacity for DEKF
using battery data from RW2.
Figure 57: Capacity estimation using the DEKF over the first RW phase of RW9 with the capacity process
noise covariance set too high. Internal filter parameters are like in table 12 except for capacity process
noise covariance=1∙10^{-8}.
Figure 58: Convergence of estimates of internal resistance parameters for varying values of process
noise covariance in the DEKF.
Figure 59: Capacity estimation over entire first RW phase of battery RW9 using ESVEKF.
Figure 60: SoC estimation using the ESVEKF for a poor initial guess of cell capacity.
Figure 61: Capacity estimation using the ESVEKF converging to measured value independently of
starting guess.
Figure 62: Convergence of SoC using the ESVEKF despite poor guesses of both SoC and capacity.
Figure 63: Rapid convergence of capacity estimates with three different initial guesses using ESVEKF
and data from battery RW2.
Figure 64: Convergence of capacity estimate for SWDEKF despite poor initial guess.
Figure 65: Convergence of capacity estimates to measured value for RW9 for different starting guesses
in SWDEKF.
Figure 66: Capacity estimates for battery RW2 using the SWDEKF for different starting guesses of
capacity.
Figure 67: Capacity estimates using SWDEKF for RW2 for high initial capacity covariance.
Figure 68: Comparison of long-term capacity estimation of SoH algorithms.
Figure 69: Principle of "accidental" convergence of SoC estimation for faulty concurrent estimation of capacity. The red line represents an SoC estimate with a low capacity, the green line high capacity, and the blue line is the in-between case.

**List of tables**

Table 1: Comparison of current battery technologies [4]. ................................................................. 5  
Table 2: Common components of commercial Li-ion batteries ............................................................ 7  
Table 3: Qualitative comparison of common cathode materials [47-49]. ........................................ 11  
Table 4: Advantages, disadvantages and applications of common Li-ion battery cathode materials [9]. ......................................................................................................................... 12  
Table 5: Factors affecting calendar and cycle aging ......................................................................... 26  
Table 6: Impact factors and effects of aging mechanisms [68]. ........................................................... 27  
Table 7: Measured cell capacities for RW9. ....................................................................................... 41  
Table 8: Development of ECM parameters with aging for battery RW9. ........................................... 49  
Table 9: Internal EKF parameters for the results in figures 35 and 36. ............................................. 66  
Table 10: Internal filter parameters for AEKF for figure 46. ............................................................... 76  
Table 11: The effect of moving estimation window size on AEKF SoC estimation accuracy. ............ 78  
Table 12: Internal filter parameter values for initialization of DEKF in figure 49. .......................... 80  
Table 13: Internal filter parameters of DEKF for results in figure 51. .............................................. 81  
Table 14: Differences in temperature between pulsed discharge tests and RW phases for RW9. ...... 82  
Table 15: Internal filter parameters of the ESEKF for the results in figures 59 and 60. ................. 90  
Table 16: Initial ECM parameters of battery RW2. ........................................................................ 91  
Table 17: Internal filter parameters for SWDEKF results in figure 64. ............................................ 93  
Table 18: Internal filter parameters for quantitative comparison of EKF and AEKF algorithms ....... 95  
Table 19: Results of comparison of EKF and AEKF for SoC estimation .......................................... 95  
Table 20: Internal filter parameters for comparison of long-term capacity estimation performance of SoH algorithms. ................................................................. 97  
Table 21: Root-mean-square capacity estimation errors for results in figure 65 for investigated SoH estimation algorithms. ................................................................................................................. 98  
Table 22: Root-mean-square error over first four measurement points of long-term capacity estimation test ................................................................................................................. 98  
Table 23: Execution times of SoH estimation algorithms for simulations of 300 RW cycles. ......... 100
Table 14: Differences in temperature between pulsed discharge tests and RW phases for RW9. ...... 82
Table 15: Internal filter parameters of the ESVEKF for the results in figures 59 and 60. .................. 90
Table 16: Initial ECM parameters of battery RW2. .......................................................... 91
Table 17: Internal filter parameters for SWDEKF results in figure 64. ..................................... 93
Table 18: Internal filter parameters for quantitative comparison of EKF and AEKF algorithms........ 95
Table 19: Results of comparison of EKF and AEKF for SoC estimation .................................... 95
Table 20: Internal filter parameters for comparison of long-term capacity estimation performance of
SoH algorithms. ......................................................................................................................... 97
Table 21: Root-mean-square capacity estimation errors for results in figure 65 for investigated SoH
estimation algorithms. ............................................................................................................... 98
Table 22: Root-mean-square error over first four measurement points of long-term capacity estimation
test.................................................................................................................................................. 98
Table 23: Execution times of SoH estimation algorithms for simulations of 300 RW cycles............ 100
1 INTRODUCTION

Due to increasing concerns of climate change there currently exists a growing interest within the energy sector to transfer from fossil fuels such as oil, gas and coal to renewable sources of energy like solar- and wind power. While there are several upsides to this transition, such as a lowered environmental impact and an increased energy security, there are also several challenges that still must be faced. One such challenge is the question of electrical energy storage. Unlike traditional energy sources, the production of renewable energy varies depending on time of day, climate and weather. Energy storage therefore becomes critical to manage the variable demand for electricity in an efficient manner. Simply put, when there is an excess of energy production this should be stored and when there is more demand than production the previously stored energy should be tapped into to reduce the strain on the energy system. There exists a multitude of suggested technologies, so-called energy storage systems (ESSs), intended for this purpose. These can in principle be divided into the following categories based on the form in which the energy is stored [1]:

- Electrical: capacitors, super capacitors.
- Mechanical: flywheels, pumped hydroelectric systems, compressed air.
- Chemical: hydrogen or other chemical storage.
- Thermal: hot water, molten salts.
- Electrochemical: batteries.

Battery energy storage systems, the focus of this work, possess several key advantages amongst these, including efficiency, low pollution, rapid response time, flexibility of siting and low need for maintenance. The modular nature of battery technologies also allows for great flexibility and adaptability [2]. Battery power represents a very attractive option for renewable energy storage, peak-shaving during intensive grid loads and furthermore as a back-up system for controlling voltage drops in the energy grid, phenomena which will be of increasing importance as the transition to renewable energy continues with the associated intrinsic climate and weather dependence [3-5].

Within the segment of battery energy storage, Li-ion batteries are of special interest, and is currently the market-leading technology [3, 6, 7]. This is partly due to their suitability for mobile applications, especially electric vehicles (EVs) and hybrid electric vehicles (HEVs). These are quickly developing and growing markets, which has led to an increased interest in research within the field of Li-ion battery technology [8]. The most rapid developments have been within the areas of Li-ion battery materials, efficiency, reliability, safety and management systems to meet customers’ demands for vehicle performance. The requirements of batteries for electric vehicles and for stationary energy storage overlap in several regards. Both types of applications require long-term stability, safety, low costs and high energy density. The differences lie mainly in the emphasized type of energy density (volumetric density for stationary power, gravimetric density for mobile applications), and demands for flexibility of power deliverance [2]. It is therefore believed that a synergistic development is possible and highly advantageous. This is especially evident in the possible reuse of “worn-out” EV/HEV Li-ion batteries for stationary applications, thus allowing effective battery lifetime to be increased by as much as a factor of two [4].

This work aims to investigate methodologies with which to estimate the degradation of Li-ion battery performance over time, a phenomenon referred to as battery aging. To achieve this, a simple equivalent circuit battery model (ECM) is suggested and methods of identifying said models parameters are investigated. The suitability of the ECM is validated using real battery data. To properly assess battery aging an extended Kalman filter (EKF) is developed for accurate state of charge (SoC)
estimation and compared to the adaptive extended Kalman filter (AEKF) with updating process- and measurement noise covariance matrices. The estimation of battery state of health (SoH), here defined as the relative loss of battery capacity, is then examined using three different extended- or dual Kalman filters, each integrated with the previously mentioned EKF SoC estimator. First the standard dual extended Kalman filter (DEKF), capable of also estimating the internal resistance of the cell, is investigated. Then two modified Kalman filter methods aiming for lower computational strain are introduced. The first of these applies the EKF methodology but uses a state vector that has been extended to include the cell capacity directly as a state variable. The second is based on the DEKF methodology but reduces the weight filter to only incorporate the cell capacity. All Li-ion battery SoH estimators are finally evaluated and compared based on accuracy, computational performance, and stability with special regards to tuning of internal filter parameters. All data used for model- and SoC/SoH estimation algorithm validation was obtained from the NASA Prognostics Center of Excellence [9].

The structure of the thesis will be as follows: Below in the background section necessary theory and earlier work within the fields of Li-ion cell construction, chemistry, aging, and modelling, with special regard to Li-ion battery SoC and SoH monitoring will be introduced and discussed. The objective and scope of this current work will then be outlined in chapter two. Chapter three will contain a thorough description of the applied methodology, including Li-ion battery model development and details of the used Kalman filtering algorithms. The obtained results using said methods are presented and discussed in the fourth chapter and, finally, conclusions are then drawn in chapter five, along with suggestions for future work.

1.1 BACKGROUND

Lithium-ion batteries have in a relatively short time span become essential technology. With new innovations in safety, cost, and energy density, it is predicted to remain a key competitor in the battery sector for many years to come [10]. In this background section the goal is to provide an introduction to Li-ion battery technology at large by considering historical, economical, physical and chemical aspects.

1.1.1 Historical perspective and outlook on lithium-ion batteries

The Li-ion battery technology know today was developed by Asahi Chemicals and Sony during the 1980s and first became commercially available in 1991. The main goal was to come up with a novel battery technology capable of achieving increased energy densities compared to existing solutions for the then quickly growing market of mobile applications, mainly small portable electronics such as mobile phones, video cameras and laptop computers [11]. The idea was to use lithium due to its low molecular weight, leading to high energy density and fast diffusion [2, 10]. The main competing technologies at the time were nickel-cadmium (Ni-Cd) and nickel-metal hydride (Ni-MH), both lacking in terms of gravimetric and volumetric energy density compared to the new Li-ion technology. The memory effect (permanent loss of capacity when not fully charging the battery) is also lower for Li-ion batteries [12]. In fact, for a long time it was believed that Li-ion batteries suffered from no memory effect at all but recently it has been proven to occur, at least to a minimal extent [13].

One price to pay for this increased performance was the need to use organic solvents instead of an aqueous solution as the electrolyte for these batteries. The cells also must be pressurized. This causes some concerns regarding safety, and special requirements for the construction of the cells. Li-ion batteries can, when abused or improperly assembled or designed, infamously explode. One recent example is the widely published Samsung Note scandal. Due to risks like these, testing protocols for Li-
ion batteries must be especially rigorous, which naturally is an added cost over other more inherently safe battery technologies [14].

Another persisting issue has been the relatively high manufacturing cost of Li-ion batteries, mainly due to the comparatively expensive transition metal based cathode materials [15]. Long-term there also exists some concern regarding the use of lithium metal in the manufacturing process considering the quite limited world inventory. Some work has therefore gone into trying to produce batteries based on Na-ion intercalation compounds due the similarity in chemistry, sodium being an alkali metal like lithium, as the price and the worldwide available supply of sodium is overall much healthier than for lithium [2]. Some authors have however claimed that the price of lithium is in fact not a major driver of battery prices, or will at least not be so long term [16].

The development of Li-ion battery technology and its market share has been drastic since its introduction in 1991. Between 1995 and 2005 prices halved and energy density doubled. Today prices are one tenth of what they were in 1991 and sales have increased dramatically as well. In 2013 five billion Li-ion cells were sold just for powering of portable electronics. There is however a growing concern within the industry that Li-ion battery performance will soon reach its peak after continual improvements for over 25 years. Researchers currently believe that the limit is a further increase of gravimetric density by approximately 30 percent [17].

Almost all Li-ion batteries have historically been manufactured in Japan, Korea or China, with very little production in Europe or North America. This is also the case currently. As EVs and HEVs viable and widely available during the early 2010s these also became a key usage areas for Li-ion batteries. Electric- and hybrid vehicles are of course quickly developing sectors with some forecasts predicting that 50 percent of all sold cars will use a battery for propulsion by 2020 [4, 15]. Currently Ni-MH batteries are dominating this market, but this is expected to change rapidly. It is projected that Li-ion batteries will be the leading format by as soon as 2020 [18]. Note however that when one only considers fully electric vehicles, such as the Tesla Model S and X, BMW i3, Nissan Leaf and Chevrolet Spark, Li-ion technology is already the front running technology [10, 19, 20]. It is predicted that the increased production of Li-ion batteries for these automotive applications, along with the growing portable electronics market, will drive down prices to the point where it becomes profitable to utilize these in stationary battery ESS applications as well, where lead-acid batteries are currently dominating [2, 21].

Presently Li-ion batteries are four to eight times more expensive than equivalent lead-acid alternatives and one to four times more expensive than Ni-MH [4]. It is required that the average price of Li-ion batteries drops further to around $125/kWh, the goal set for 2022 by the American Department of Energy, down from $190/kWh in 2016, in order to achieve market-wide penetration in the stationary application market [16]. Prices of first generation Li-ion batteries are expected to keep on decreasing, despite the already dramatic lowering these past years, with an estimated 60 to 70 percent total or approximately 7 percent annual decrease in cost per watt-hour between 2007 and 2014, with purchasing costs for EV manufacturers dropping even more [16, 22]. Nykvist and Nilsson project a cost reduction rate of around the same magnitude going forward as well, noting also that there currently exists a large price discrepancy amongst suppliers. This discrepancy is also projected to decrease per figure 1 below.
For Li-ion battery powered electric vehicles to become cost competitive with traditional internal combustion varieties it is commonly agreed that prices of Li-ion batteries need to fall to around $150/kWh. This means that the paradigm shift to Li-ion technology is expected to occur within the transport sector first, followed by the stationary applications at a later stage [22]. Synergistic aspects exist between these at first seemingly quite different applications however, as shall be discussed in the section below.

Gravimetric energy densities and specific power of different battery technologies are compared in figure 2. The current state of the most common battery technologies is summarized in table 1.
As can be seen in figure 2, Li-ion technology has the advantage in terms of both gravimetric- and volumetric energy density over other common battery types. Also notable is the large difference energy density within the category of Li-ion batteries itself. As shall be seen, Li-ion battery technology encompasses a diverse group of chemistries.

<table>
<thead>
<tr>
<th>Table 1: Comparison of current battery technologies [4].</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
<tr>
<td>Gravimetric energy density (Wh/kg)</td>
</tr>
<tr>
<td>Cycle life ($\Delta$SoC = 80 % each cycle)</td>
</tr>
<tr>
<td>Fast charge time (h)</td>
</tr>
<tr>
<td>Self-discharge per month at room temperature</td>
</tr>
<tr>
<td>Nominal cell voltage (V)</td>
</tr>
<tr>
<td>Maintenance requirements</td>
</tr>
<tr>
<td>Toxicity</td>
</tr>
<tr>
<td>Peak C-rate (Ah/h)</td>
</tr>
</tbody>
</table>

As can be gleaned from table 1 and figure 2, the main advantages of Li-ion battery technology over its common counterparts are its long cycle life, high nominal voltage, low need for maintenance and high energy density. In the coming sections the usage of Li-ion batteries in stationary applications will be considered in more detail.

1.1.2 Lithium-ion batteries in stationary applications

Usage of batteries for storage of most importantly renewable energy from solar- and wind power has been investigated for several years. The main competing technology has generally been pumped hydroelectric energy storage, which currently makes up 99% of all worldwide electrical grid storage capacity [21, 23]. The main driving force in the integration of ESSs in the electrical grid has mostly been government regulations to incentivize an accelerated transition to renewable energy generation [10].

Stationary battery applications can be split into two broad categories: energy applications and power applications. The main difference between these is the C-rate$^1$ of discharge, energy applications having discharge times in hours and power applications in the range of seconds to minutes. Energy applications such as peak shaving and load leveling involve reducing the maximum grid load at, or close to, maximum energy demand. Grid frequency regulation and power quality control are typical modes of operation within the power applications category [3, 21]. It is widely accepted that no currently existing ESS is perfect in all regards. Because of this a mixture of storage systems is expected to be the optimal solution going forward. Li-ion battery technology is expected to play a key role, particularly for short- and medium-term storage solutions. Smaller scale applications are considered especially suitable for Li-ion battery ESSs due to the high both volumetric and gravimetric energy density [24].

---

$^1$ Charge/discharge rate is often written as C-rate, given as the amount of times the battery could theoretically be fully charged or discharged during one hour at the chosen current.
As mentioned above, currently around 99% of the worldwide installed EES consists of pumped hydroelectric systems. It is however expected that Li-ion battery technology will soon reach a point where cost is not inhibiting its use for these purposes, and several demonstration projects are already running with more planned [2].

As mentioned, one of the most attractive aspects of Li-ion batteries compared to other electrochemical ESSs is its growing use for EV applications. Not only is this driving the technological development, with an expected yearly increase in gravimetric energy density between 2 and 6 percent [19], but it is also driving down manufacturing costs through the economy-of-scale principle. Furthermore, the prospect of re-using batteries that have lost 20 percent of their maximum capacity through aging during automotive applications, and would otherwise be considered ready for recycling, for stationary ESS is promising [2, 25]. The potential residual lifetime for stationary ESS might be as long as ten years, representing a further loss of around 15 percent capacity. This not only dramatically lowers the lifecycle cost of Li-ion batteries as a whole, including purchasing cost for both the EV and ESS customers, but also lowers their overall environmental impact [25]. In fact, it is currently not considered profitable to recycle worn-out EV batteries [26]. The synergistic aspects between the applications are also obvious: EVs could be charged using 2nd generation reused Li-ion batteries which were in turn charged with renewable energy. The needed battery conversion to allow for stationary use would be rather simple. Spent EV batteries would have to be tested for capacity and safety aspects such as leakage and then repackaged with appropriate software and hardware. The main applications for these second generation Li-ion batteries are expected to be transmission support, area regulation, light commercial load following, load leveling, power reliability, residential load following, distributed node telecom backup and renewable energy firming [25]. These applications can essentially be divided into two different classes. Either large numbers of battery packs are assembled together for large scale applications, such as integration with renewable energy generation, or smaller numbers of packs are used for decentralized peak-shaving purposes in businesses and homes [27].

The so-called “smart grid” applications are of particular interest for second generation Li-ion batteries. This is partly due to the expected-to-be limited supply of these second-generation batteries considering the size of large scale ESS solutions. Smaller, distributed, decentralized ESSs for homes and businesses can help improve efficiency and flexibility by allowing customers to regulate their personal energy consumption. It is also considered to be safer from a financial standpoint to invest in a larger number of smaller units than vice versa [27].

Having discussed the current and historical state of Li-ion battery markets, the next sections aim to cover the basics of Li-ion battery construction and chemistry.

### 1.1.3 Main components of Li-ion batteries

The currently available Li-ion battery electrodes are based on Li-ion intercalation compounds. The goal is for these intercalation/de-intercalation processes to be as reversible and rapid as possible to ensure a long battery life and to have as high cell voltage as possible to increase the energy density and achievable power from the cell. This means that demands on material performance are very high. The first Li-ion batteries developed by Sony during the 1980’s were based on a graphite anode with a lithium cobalt oxide (LiCoO₂) cathode. Even though several improvements have been made to the technology since then, the basic principle and materials used still largely remain the same, the cathode material often being some transition metal oxide and the anode almost always some graphite-based compound [28]. The most diversified aspect of Li-ion batteries is easily the cathode material with many commercialized alternatives, each with its own strengths and weaknesses. Li-ion battery chemistries are therefore often categorized based on the cathode material. Li-ion batteries are manufactured in four different geometrical shapes: coin, cylindrical, flat and prismatic [29].
Lithium, being an alkali metal, is very sensitive to water and therefore carefully dried aprotic organic solvent electrolytes must be used to avoid excessive lithium consumption. These are usually mixtures of alkyl carbonate esters such as diethyl carbonate (DEC), dimethyl carbonate (DMC), propylene carbonate (PC) and/or ethylene carbonate (EC) with a lithium salt such as lithium hexafluorophosphate (LiPF$_6$), lithium perchlorate (LiClO$_4$) or lithium tetrafluoroborate (LiBF$_4$) [30]. LiPF$_6$ is the most commonly found amongst these due to safety and stability reasons [31]. The lithium salt is responsible for ionic transport through the separator and the electrolyte and should therefore preferably be as ionic as possible. In addition, the added salt should be thermally stable, able to form stable passivating films on both electrodes, and be as inert as possible towards all the components of the cell [32].

The solvent has three main requirements: it should have a high dielectric permittivity and a low viscosity to allow for rapid transport of Li-ions and it should be as inert as possible to reduce lithium consumption, thus allowing for longer battery lifetimes [33]. The goal when developing a solvent mixture to be used as an electrolyte is to combine the advantages of the individual compounds to achieve a good compromise of properties. It is often so that low viscosity solvents are mixed with ones with high ionic conductivity for instance. This is as a rule done empirically [30]. Cyclic carbonate esters, such as EC and PC, usually have the role of contributing a high conductivity while linear carbonate esters, mainly DEC and DMC, generally constitute the lower viscosity compounds [31, 32, 34].

There will also always be a set of additives added to the electrolyte. These are introduced for several reasons, for instance to minimize electrolyte decomposition or to improve safety. As an example, vinylene carbonate (VC) is regularly added to improve high temperature performance [35].

A polymeric separator, usually made of polypropylene or polyethylene, prevents transport of electrons through the electrolyte but still allows for Li-ion passage. The most popular Li-ion battery component materials are summarized in table 2 [6]:

**Table 2: Common components of commercial Li-ion batteries.**

<table>
<thead>
<tr>
<th>Component</th>
<th>Materials</th>
<th>Characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cathode</td>
<td>LiCoO$<em>2$ (LCO), LiNi$</em>{1-x}$Mn$_x$O$_2$ (NMC), LiMn$_2$O$_4$ (LMO), LiFePO$<em>4$ (LFP), LiNi$</em>{1-x}$Co$_x$Al$_x$O$_2$ (NCA)</td>
<td>High voltage</td>
</tr>
<tr>
<td>Anode</td>
<td>Carbonaceous material (natural/artificial graphite, hard/soft carbon)</td>
<td>Low voltage</td>
</tr>
<tr>
<td>Electrolyte</td>
<td>Non-aqueous solvents + Li salt + additives (DEC/DMC/PC/EC + LiPF$_6$/LiClO$_4$/LiBF$_4$ + VC)</td>
<td>Li-ion conductor</td>
</tr>
<tr>
<td>Separator</td>
<td>Porous polymer membrane (PP, PE)</td>
<td>Electrical insulator</td>
</tr>
</tbody>
</table>

In addition, current collectors, most often made of copper on the anode side and aluminum on the cathode side, help with the transport of electrons to/from the cell on charging/discharging.

In large scale applications, such as stationary energy storage or in EVs and HEVs, a single battery system may consist of thousands of connected Li-ion cells. The cells can be connected in series or in parallel configurations. More cells will be connected in parallel if a higher current is required and more in series for a higher voltage. Connecting the cells in parallel also allows for a higher total capacity [36].

### 1.1.4 Chemistry of Li-ion batteries

The basic chemical processes occurring in the Li-ion battery during operation, as for any electrochemical cell, are the redox reactions at the cathode and anode. During discharge of the battery, oxidation is taking place at the anode and reduction at the cathode. During this process, Li-ions are deintercalated from the anode, transported through the electrolyte, and finally intercalated into the cathode crystal lattice. At the same time electrons are also transferred in the same direction,
that is from anode to cathode, but through an external circuit to the load. The discharge reaction is spontaneous, meaning that energy can be extracted from the process. This is the basic principle which allows for electricity to be outputted from the Li-ion cell. The process is also exothermal, meaning that heat is produced. The driving force for the intercalation/deintercalation process is difference in potential between the cathode and anode, which is dependent on the cell chemistry, the current state of charge, and operation conditions but normally lies between 3 and 4 V for Li-ion cells. Upon charging the direction of this process is reversed and the anode is re-stocked with intercalated lithium ions. Due to this back-and-forth motion of the lithium ions on charging and discharging these batteries are frequently referred to as “rocking chair batteries”. For a generic metal oxide LiMO₂ cathode cell the partial redox reactions are the following on discharge:

Anode: \[ x\text{LiC}_6 \Rightarrow x\text{Li}^+ + xe^- + x\text{C}_6 \]  
Cathode: \[ x\text{Li}^+ + xe^- + \text{Li}_{1-x}\text{MO}_2 \Rightarrow \text{LiMO}_2 \]

The most common alternatives for the various Li-ion components will now be discussed, starting with cathode materials followed by anode materials, and then finally the important electrolyte additives. A simple sketch of a Li-ion cell illustrating its working principle is shown below in figure 3.

![Figure 3: Simple illustration of the original LiCoO₂-based Li-ion cell. Electrodes, current collectors and separator are all submerged in the electrolyte solution [37].](image-url)

### 1.1.4.1 Cathode materials

There currently exists a wide range of commercialized alternatives for Li-ion battery cathode materials. Generally speaking, the cathode material should be able to achieve a high energy density and have a high electric conductivity. A high energy density is effectively equivalent to being able to intercalate a large number of Li-ions. In addition, Li-ions should diffuse quickly through the material and the material should be as inert as possible towards common electrolytes to reduce aging effects [34]. The most common cathode materials can be divided into three groups based on their crystal structure. Lithiated layered transition metal oxides such as LiCoO₂ (LCO), LiNiO₂ (LNO), LiNi_{1-x}CoₓAlₓO₂ (NCA) and LiNi_{1-x-y}MnxCoₓO₂ (NMC); olivine structured lithium iron phosphate (LiFePO₄ (LFP)) and spinel structured lithium manganese oxide (LiMn₂O₄ (LMO)) [12]. There currently does not exist a universally optimal cathode material. All the discussed materials have distinct advantages and disadvantages as well as
key commercial applications for which they are most suitable. The three common cathode material crystal structures are seen below in figure 4.

![Cathode structures](image)

*Figure 4: The three crystal structures of common Li-ion battery cathode materials. Green dots represent Li-ions or Li-ion intercalation sites [38].*

The three structure types are sometimes referred to as 3D (olivine), 2D (layered), or 1D (spinel) based on the dimensionality of Li-ion motion through the materials [39]. Although LCO remains the most prominent cathode material overall, its market share has been decreasing continuously over time, mostly due to cost and safety aspects. A comparison of the cathode material market in 1995 and in 2010 is shown in figure 5 below. Note also the massive growth of the total market in that time: from 650 tons of cathode materials in 1995 to 45,000 tons in 2010, almost a 70-fold increase.

![Market comparison](image)

*Figure 5: Comparison of market share and size of Li-ion battery cathode materials in 1995 and 2010 [31].*

A more detailed look at the various common cathode materials will be given in the coming sections.

1.1.4.1.1 Lithium cobalt oxide

The first commercialized Li-ion batteries in 1991 utilized the LCO cathode. LCO remains the most common Li-ion battery cathode material currently, even though it has been losing ground to other chemistries. Cobalt metal itself is quite expensive, heavy, and the LCO based batteries have poorer thermal stability than most other commercialized alternatives. Aging in terms of capacity loss is fast during high current-, deep discharge-, cycling which might limit suitability for some applications [40]. Its most popular use is in portable energy devices due to the high C-rate capability and the high volumetric energy density [3]. Safety issues remain a concern for LCO-based batteries. Recently many incidents have been reported for two relatively new applications of LCO-based batteries: hoverboards and e-cigarettes, where the batteries would self-ignite, once again causing the inherent safety of these types of batteries to be put into question [10].
1.1.4.1.2 Lithium nickel oxide
LNO is very similar to LCO in terms of its physical structure and theoretical capacity. It was originally introduced as a cheaper alternative but still possesses a higher energy density (20 percent higher by weight). LNO is unfortunately quite difficult and expensive to produce in a pure enough form as Ni$^{2+}$ ions have a tendency to take the place of Li$^{+}$ ions during synthesis and cycling [40], especially at high temperatures, causing the reduction of Ni$^{3+}$ ions for charge balancing in the process [41]. LNO also displays issues of structural Jahn-Teller distortion, safety concerns and exothermic release of oxygen at higher temperatures [39]. The material is therefore not commonly used [3]. It was however a key development on the way to NCA- and NMC based cells [32, 39].

1.1.4.1.3 Lithium nickel cobalt aluminum oxide
NCA is a modified version of LNO that incorporates aluminum to produce a more stable material while still maintaining most of the advantages of LNO. The capacity of NCA is not quite as high as for LNO, mainly due to the fact that the Al$^{3+}$ ions do not partake in the electrochemistry of the cell, but its aging behavior is instead superior which has led to widespread usage [32]. Unfortunately, these batteries are especially sensitive to moisture during their assembly, leading to increased manufacturing costs and the need for strict environmental control. The upside is that these cells are suitable for both high energy and high power applications and are generally considered to be of high quality in terms of performance. The batteries supplied by Panasonic for Tesla EVs make use of this cathode material for instance [40].

1.1.4.1.4 Lithium manganese oxide
LMO was the second cathode material to be commercialized after LCO but had already been suggested by Thackeray et al. as early as 1983 [31, 42]. Manganese is approximately five times cheaper than cobalt and is also quite plentiful in nature by comparison. Manganese based materials also have the advantage of being more environmentally benign than cobalt based ones. LMO has a higher nominal voltage than LCO- or LNO-based chemistries. The energy density is however around 20 percent lower by weight than LCO and cycle stability is relatively poor. The fast battery aging on cycling is usually attributed to the dissolution of Mn$^{2+}$ into the electrolyte. This mechanism will be investigated further in section 1.2.3.1.1. On the other hand, LMO-based batteries have higher thermal stability and they are cheaper as they do not require expensive cobalt or nickel [3]. This battery chemistry is often used for applications requiring high power outputs, such as power tools, due to its high voltage [32]. Another promising route is to blend in LMO with NMC cathodes to reduce costs, increase capacity, and improve thermal stability [32].

1.1.4.1.5 Lithium iron phosphate
Another commercialized alternative is LFP which was first introduced in 1997 [43]. The main advantages of the iron phosphate-based electrode materials are the flat voltage profile, reduced cost, increased stability against common electrolytes, excellent high temperature performance, and practically non-existent toxicity. Electron conductivity, energy density and theoretical capacity are however lower than for other common alternatives. The lower potential does have the added benefit of reducing issues of electrolyte oxidation and the resulting battery aging thus improving lifetime and safety [3]. Some authors consider this to be the most promising cathode material for large scale energy storage for these reasons, especially when combined with a titanate anode to further reduce degradation rate and increase safety [2, 4, 32, 39, 44]. The issues of low ionic and electronic conductivity could be resolved by addition of dopants, metal dispersions or nanostructuring with the downside of increased manufacturing cost. Nanostructuring would also likely result in a lowered volumetric energy density [39, 45].
1.1.4.1.6 Lithium nickel manganese cobalt oxide

On the other end of the voltage spectrum from iron phosphates we find the so-called high voltage electrode materials. These materials do not have any advantages over conventional cathode alternatives in terms of capacity, but instead focus on achieving a high discharge voltage (close to 5 V is possible). One of these is NMC which is a promising replacement material for the traditional LCO due to increased cycling and thermal stability as well as higher capacity [39]. The ratio of nickel/manganese/cobalt in NMC batteries varies and can be adjusted as to fit the intended application. Roughly speaking, increasing relative manganese content increases safety, increasing relative nickel content improves capacity, and increasing relative cobalt content results in higher achievable C-rates [46]. This is summarized in figure 6 below. Recently NMC has been used extensively for automotive applications and is currently the cathode chemistry of choice for EVs made by BMW, Volkswagen, Fiat, Kia and Ford [10]. A newer development of NMC, LiNi\(_{1/2}\)Mn\(_{3/2}\)O\(_4\) belongs to the category of high voltage electrode materials but possesses a spinel structure much like LMO. Synthesis of these materials is challenging however and they are quite unstable with regards to the commonly used electrolytes at high voltages [3, 32].

![Figure 6: Composition dependence for performance characteristics of NMC cathodes [47].](image)

A simple comparison and summary of common cathode materials is presented in the table below [36].

**Table 3: Qualitative comparison of common cathode materials [48-50].**

<table>
<thead>
<tr>
<th></th>
<th>LCO</th>
<th>NMC</th>
<th>NCA</th>
<th>LMO</th>
<th>LFP</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Cell voltage</strong></td>
<td>3.7-3.9 V</td>
<td>3.8-4.0 V</td>
<td>3.65 V</td>
<td>4.0 V</td>
<td>3.3 V</td>
</tr>
<tr>
<td><strong>Energy</strong></td>
<td>++</td>
<td>+++</td>
<td>+++</td>
<td>+</td>
<td>++</td>
</tr>
<tr>
<td><strong>Power</strong></td>
<td>++</td>
<td>++</td>
<td>+++</td>
<td>+++</td>
<td>++</td>
</tr>
<tr>
<td><strong>Calendar life</strong></td>
<td>+</td>
<td>+</td>
<td>+++</td>
<td>-</td>
<td>++</td>
</tr>
<tr>
<td><strong>Cycle life</strong></td>
<td>+</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
</tr>
<tr>
<td><strong>Safety</strong></td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>++</td>
<td>+++</td>
</tr>
<tr>
<td><strong>Price estimate</strong></td>
<td>--</td>
<td>++</td>
<td>+</td>
<td>++</td>
<td>+</td>
</tr>
</tbody>
</table>

Due to the various strengths and weaknesses of the Li-ion battery cathode materials, the optimal choice of material is largely application-dependent, as seen in table 4 below [3, 36].
Table 4: Advantages, disadvantages and applications of common Li-ion battery cathode materials [10].

<table>
<thead>
<tr>
<th></th>
<th>LCO</th>
<th>NMC</th>
<th>NCA</th>
<th>LMO</th>
<th>LFP</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Advantages</strong></td>
<td>Cycle life, energy density</td>
<td>Excellent energy density</td>
<td>Cycle life, power</td>
<td>Thermal stability, price, energy density</td>
<td>Excellent cycle life</td>
</tr>
<tr>
<td><strong>Disadvantages</strong></td>
<td>Thermal stability</td>
<td>Patent issues</td>
<td>Sensitive to moisture</td>
<td>Cycle life</td>
<td>Energy density, power</td>
</tr>
<tr>
<td><strong>Common applications</strong></td>
<td>Portable electronics</td>
<td>Power tools, EVs</td>
<td>High quality electronics</td>
<td>Power tools</td>
<td>Power tools, stationary energy storage, e-bikes</td>
</tr>
</tbody>
</table>

### 1.1.4.2 Anode materials

Due to the relatively low capacity and stability of the commonly used carbon based electrode materials, researchers have been trying to find viable alternatives. These are all still lithium intercalation compounds. It has however seemed to be difficult to find materials that can compete with carbon in terms of overall compromise between performance and cost. The diversity of Li-ion battery anode materials is therefore much lower than that of cathode materials. Titanate, as mentioned below, is the only commercialized alternative [36, 51, 52].

Besides the advantage of low cost, carbon based materials are also abundantly available, display a low potential vs. Li/Li⁺, allow for fast Li⁺ diffusion, and show relatively little volume change on Li-ion intercalation/deintercalation, leading to low mechanical stress on cycling. Common carbon anodes are either based on graphite (soft carbon) or hard carbon. Graphite based materials have the advantage of high capacity, but lack compatibility with PC based electrolytes due to excessive side reactions. On the other hand, hard carbons generally have a lower capacity but are instead more stable over time with regards to aging [40].

Tin oxide anodes of composition SnMₓOᵧ (M=B, P, Al) were first suggested in 1997. The main advantage that these materials provide over traditional graphite is a higher energy density. Tin oxide anode materials will however degrade relatively quickly on cycling and due to the persistence of this issue they have not been commercialized as of yet. The practical energy density is also frequently substantially lower than predicted theoretical values. Composites of tin-oxide with various other compounds have therefore been investigated to alleviate these issues, for instance with carbon nanotubes, magnesium oxide, silicon oxide, zinc oxide, or cobalt oxide. This is very much an active research area [51].

The fact that LTO could be used as a material for reversible Li-ion intercalation/deintercalation was first reported in the 1980s [53]. LTO is unique as an anode material in that its potential is essentially independent of the degree of intercalation of Li-ions. This flat voltage curve allows for nearly constant power delivery over the entire SoC range of the battery. Also, the change in volume on intercalation of Li-ions is practically non-existent. This reduces aging due to mechanical stress dramatically. For traditional graphite electrode materials, the volume change on intercalation is around 10 percent which, long-term, leads to fissures, cracking and exfoliation of the anode material. These processes are investigated closer in section 1.2.2. The great disadvantages of LTO are the low electron conductivity along with the rather extensive release of gas by-products, predominately H₂, CO and CO₂, during operation at high temperatures [54]. These issues have limited the use of LTO cells for large-scale applications [55]. Researchers have been trying to improve the material properties in these regards through doping or protective coatings [51]. In addition, LTO-based batteries have lower energy
densities and capacities than graphite based ones due to the higher weight and also higher potential vs. Li/Li⁺ (about 1.5 V higher than graphite anodes) [56].

Silicon-based anode materials have also been suggested due to their potential very high capacity for intercalation of Li-ions. In fact, Silicon has the highest theoretical potential for Li-ion intercalation of all commonly found elements, around 4200 Ah/kg. The most common alternative is thin-film electrodes of amorphous silicon. Historically, the issue with these materials has been the large volume change on intercalation/de-intercalation, causing cracking and even pulverization of the material. Electron conductivity of Silicon is also generally quite poor. Widespread commercialization of Si-based anode materials has therefore not taken place [51, 57].

1.1.4.3 Additives
Additives are always added to Li-ion battery electrolytes with the general intentions of reducing aging effects and improving performance. More specifically, the main goals are to improve the anode SEI stability, to reduce LiPF₆ decomposition, to protect the cathode, to reduce or stabilize metallic Li deposition and to improve overall safety. Certain additives might also be added to reduce the corrosion of the aluminum current collector, to improve the wetting of the polymer separator, or to act as a fire retardant [58]. The most commonly added additive, as mentioned above, is VC. The purpose of VC is to limit the loss of active lithium due to SEI formation by forming a thin film on the anode. VC is reduced and polymerizes before SEI formation, thus improving battery life by avoiding a loss active cycling Li-ions. Other additives, such as lithium bis(oxalato)borate (LiBOB) will react with already formed SEI compounds to produce more stable ones, thus effectively suppressing further SEI growth and subsequent decomposition [32].

One of the major sources of side reactions in common Li-ion batteries leading to loss of capacity and increased internal resistance is the decomposition of the common LiPF₆ salt to form hydrofluoric acid (HF) via a series of reactions. This is especially evident at higher temperatures and needs to be suppressed as much as possible to ensure long term battery performance. The first step in this process is the following reaction:

\[ \text{LiPF}_6 \leftrightarrow \text{LiF} + \text{PF}_5 \]  

(C)

PF₅ is the compound that will go on the produce not only HF but also CO₂ and the toxic POF₃. The last two compounds are both in gas form and their formation will therefore lead to a pressure build-up in the cell, a serious safety concern. This equilibrium can be shifted towards the reactant side by simply adding LiF to the electrolyte which is frequently done. Another strategy is to reduce the reactivity of already formed PF₅ by adding complexing agents, often tris(2,2,2-trifluoroethyl) phosphite. Finally, HF or H₂O scavengers, such as butyl amine, are also frequently added as a last resort solution [32, 58]. The mechanisms involved in HF formation and the related Li-ion battery aging processes will be explored further in section 1.2.

1.1.5 Battery management systems
The specialized combination of hardware and software for real-time, online monitoring and control of battery performance is called the battery management system (BMS). A BMS is necessary for all Li-ion batteries due the high demands for strict control of operation parameters to ensure safe and efficient performance [59-61]. The first BMSs were developed in the 1990s by Texas Instruments, Ericsson and Motorola [8]. The tasks of the BMS can be divided into two broad categories [61, 62]:

13
- Optimization of operating parameters such as C-rate, ΔSoC, and cell voltage to ensure safe, efficient performance. This means keeping these parameters within certain battery-specific operation windows, often referred to as the safe operating area (SOA). These parameters also should be balanced evenly among all individual cells in the battery pack. This concept is called equalization.
- Continuous monitoring of vital battery parameters to determine battery SoC, SoH, remaining useful life (RuL) etc. via algorithms. This is called battery monitoring.

These two categories are of course closely interconnected. Operation outside of the SOA induces accelerated aging and can potentially damage the battery irreversibly. For instance, if the operation temperature drops below the recommended limit, lithium plating may deposit on the anode resulting in a loss of capacity and available power. Accurate estimation of say the SoH is likewise critical in determining the appropriate SOA for the optimization procedure as the optimal location of this window will certainly change as the battery ages [63].

The general issue of SoC-, SoH- and RuL determination is that the internal chemical processes of the Li-ion battery are not, at least directly, observable during operation. This means that one must identify critical variables that can be measured during operation that can somehow be used to indicate SoC, SoH and/or RuL via some functional relationship. It is often the case that the measurable parameters are limited to just current, voltage, and temperature [64]. Algorithms for online monitoring of SoC and SoH will be discussed in sections 1.4 and 1.5 respectively. It is often the case that the limited computational power and memory of the BMS are key factors to consider when developing a battery monitoring method, as is the accuracy of available measurement sensors. Measuring voltage with reasonable accuracy is most often not an issue. Accurate current measurement can however prove to be a challenge. In addition, the battery behavior will change over time due to various aging mechanisms, thus further complicating the work of the BMS. For these reasons the algorithms for SoC-, SoH-, and RuL estimation must strike a balance between accuracy, numerical robustness, flexibility and computational performance, which is not an easy task [62, 65]. To identify and indicate the extent of battery aging, one must first recognize the causes and symptoms of the involved physical and chemical processes, which is not elementary and understanding of these processes remains an active research area [66]. The most important aspects of Li-ion battery aging will be introduced in the subsequent section.

1.2 LITHIUM-ION BATTERY AGING

For the ideal battery, the redox reactions at the cathode and anode are completely reversible and the only chemical reactions occurring within the cell. For a real battery, this is not the case. The Li-ion battery will lose capacity and gain internal resistance over time due to various side reactions. These performance degradation processes are collectively referred to as battery aging. Battery aging reduces available power and energy of the battery and eventually leads to the battery having to be decommissioned, often as a safety precaution [67, 68].

Understanding of the chemical and physical processes that lead to battery aging is critical in optimizing battery usage, lifetime, chemistry and construction [69]. The issue is very challenging due to the large number of possible reactions and the interactions of these, along with the highly limited possibilities of collecting data during operation of the battery. As mentioned, most often only voltage, current and temperature can be measured during online operation. These problems become even more complex when one considers equalization aspects for entire battery modules and/or packs, consisting of several cells connected in series and/or parallel [63]. This work will focus on only the cell level aging phenomena.
The exact nature of the battery aging processes depends on current and historic operation conditions as well as the cell chemistry. While the differences in aging behavior between different cell chemistries, as will be discussed further in coming sections, can be quite substantial, some general trends of all Li-ion battery aging can be elucidated. The most important processes in Li-ion battery aging are [67]:

- Formation, decomposition and precipitation of a solid electrolyte interface (SEI) on the anode.
- Deposition at anode (dendrite formation, lithium plating).
- Metal dissolution from cathode.
- Loss of active electrode material (particle cracking, structural disordering, dissolution, exfoliation).

The growth of a passivating layer on the anode, commonly referred to as the SEI, is of particular interest. Many researchers consider this to be the most important aging process at the anode of Li-ion batteries, to the point where this is commonly the only mechanism considered in physics-based aging models [69]. The carbon anode is in fact not thermodynamically stable with respect to the common organic solvent electrolytes and chemical reactions between the electrolyte and the carbon will always occur to some extent. This is however not an issue during normal operation. The SEI actually protects the anode from corrosion and from degradation reactions with the electrolyte, essentially acting as a safety barrier [69, 70]. Figure 7 summarizes the most common aging effects in Li-ion batteries.

As can be seen in the figure above, the number of possible degradation reactions is very large. Determining the exact nature of these reactions, and furthermore their interactions, is often exceedingly difficult. It is also clear that several length scales must be considered simultaneously to get the complete picture of Li-ion battery aging; from the molecular to the macroscopic [107]. Of course, several time scales must be considered as well, from the near-instantaneous Li-ion intercalation/de-intercalation reactions to the slow growth of the SEI over the operating time of, potentially, months and years. This separation of time scales makes accurate modeling very challenging, especially considering the inherently slow rate of aging data acquisition, such as capacity fade.

In the coming sections component-specific aging phenomena of common electrolyte and electrode materials will be summarized and discussed.
1.2.1 Aging of electrolyte
While the electrolyte does not provide any chemical function in the battery bar ionic transport, it does play a key role in battery aging. Reactions between the electrolyte and the electrodes particularly so, as will be discussed in subsequent sections. It is however also important to consider the processes occurring within the confines of the electrolyte itself as these will often be the starting point of other reactions at the electrodes, the separator and/or current collectors.

Perhaps the most important of these electrolyte-internal processes is the formation of HF. After PF$_5$ has been formed via the decomposition of LiPF$_6$, see reaction (C) above, it can react further with traces of water, simple alcohols or other protic contaminants in the electrolyte to form HF and phosphoryl fluoride (POF$_3$) [41, 71]. Here the reaction with water is shown.

$$ PF_5 + H_2O \rightarrow POF_3 + 2 HF $$  \hspace{1cm} (D)

It is thought that the formation of PF$_5$ is accelerated via interactions with the common electrolyte constituents EC and DEC. These reactions are more prominent at elevated temperatures. Formed HF may react with Li-ions to form LiF, releasing a proton in the process and thus lowering the pH of the electrolyte. This will in turn lead to further HF formation by the following LiPF$_6$ interactions [71, 72].

$$ H^+ + PF_6^- \rightleftharpoons HFPF_5 $$  \hspace{1cm} (E)

$$ HFPF_5 \rightarrow HF + PF_5 $$  \hspace{1cm} (F)

The formation of HF may therefore be seen as autocatalytic. Formed PF$_5$ will also react with the electrolyte, particularly EC. This will cause polymerization of EC to occur under release of CO$_2$. The interactions of the Li-salt, most commonly LiPF$_6$, and the electrolyte are generally quite complex and not yet fully understood [71]. The most common strategies for dealing with HF formation are addition of additives, as discussed in section 1.1.4.3, and drying of solvents to very low moisture content to avoid reaction (D) from occurring in the first place. Avoiding water is overall considered a key aspect in limiting Li-ion battery aging. Li-ion cells are therefore normally assembled at low air humidity and electrodes are dried at elevated temperatures [73].

1.2.2 Aging mechanisms at anode
According to many researchers the most important aging processes in Li-ion batteries occur at the anode, at least for cells with graphite based anode materials. As these types of anodes are by far the most common, the present discussion of anode aging mechanisms will be focused on mainly these materials. Again, formation of the SEI is commonly considered the most important process. The formation of the SEI occurs right from the start of battery operation. The carbon anode will immediately upon the first charging of the battery react with EC, DMC, DEC, and/or EMC in the electrolyte to form various solid phase compounds such as ROCO$_2$Li and CO$_2$OLi, consuming active Li-ions in the process and thus lowering the available cell capacity. These formed compounds can then react further with traces of carbon dioxide or water in the electrolyte to form compounds like dimethyl-2,5-dioxahexane dicarboxylate (DMDOHC), ethylmethyl-2,5-dioxahexanedicarboxylate (EMDOHC) or diethyl-2,5-dioxahexane dicarboxylate (DECDOHC) [68, 74]. An example reaction is shown in figure 8, a suggested route to DMDOHC. Note that CO$_2$ is formed during the reactions pictured in figure 8, which might cause further side reactions or cell deformation due to the increased internal pressure.
In addition to these types of reactions, the LiPF$_6$ salt dissolved in the electrolyte may react with traces of water to form PF$_5$ and HF as discussed above. These react with Li-ions to form other solid, insulating compounds, LiF for example, that accumulate on the anode as part of the SEI as well. The traces of HF may also cause LiFMn$_{3+}$ in Mn-containing cathodes, mainly LMO, to disproportionate into Mn$^{4+}$ and Mn$^{2+}$. This will cause dissolution of cations from the cathode in the organic electrolyte. These cations will then migrate to the anode under charging conditions and form insoluble depositions on its surface, adding to the SEI [58, 68].

As mentioned above, the existence and slow growth of the SEI will not be a major issue at normal battery operating conditions, even though it will reduce the number of active cycling Li-ions, and thus available cell capacity. Furthermore, SEI growth will also slowly increase cell internal resistance over time due to the lower Li-ion diffusion rate in the growing surface layer. SEI formation is expected and can thus be accounted for. The concerning aging processes in the SEI occur at non-optimal temperatures or during overcharging of the cell. At high temperatures, the SEI will break down and crack. The SEI might also crack due to overcharging or storage at high SoC for prolonged periods. The cracking of the SEI will expose bare graphite surface which leads to accelerated formation of new SEI, thus reducing the cell capacity as active Li-ions are consumed in the process. The reason for the higher SEI formation rate is believed to be the catalytic effect of the exposed fresh graphite surface [58]. SEI cracking might also cause loss of electrical contact and loss of available active sites in the anode for Li-ion intercalation due to blockage. This might also cause graphite exfoliation. The risk of SEI cracking increases with layer thickness, meaning that these processes become increasingly important at the later stages of the battery life [68, 75].

The structural changes of the graphite itself during cycling will also cause aging. The intercalation and de-intercalation of Li-ions, particularly at higher C-rates and ΔSoC, along with solvent co-intercalation, causes the graphite layers to become increasingly disordered due to fissures, splits and cracking of the carbon material [76]. Chemical reactions producing gas might also cause cracking, especially during storage at high SoC for prolonged periods [70]. The loss of order in the graphite generally decreases electrode Li-ion capacity due to a loss of active sites. In addition, loss of order or cracking of the graphite anode will generally lead to increased SEI formation due to the increased exposed area of fresh graphite, thus leading to increased loss of active Li-ions as described above [77].
Beside SEI formation and growth and the loss of structural order within the electrode, lithium plating is yet another important graphite anode aging process to consider [67, 69]. Li-ions will under certain circumstances not be intercalated into the anode but will instead be reduced to form a metallic lithium deposition on its surface, thus adding on to the SEI [78]. This effect is especially evident when:

- The anode capacity is lower than the cathode capacity, for instance due to anode aging;
- The temperature is low;
- The C-rate under charging is high.

This may be rationalized by imagining an accumulation of Li-ions on the surface of the anode due to slow intercalation kinetics (low temperature) or lack of intercalation sites (low electrode capacity). The competing lithium plating reaction will then dominate due to the large Li-ion concentration gradient at the surface of the electrode [78]. The composition of the electrolyte and electrode materials will also affect the extent and rate of lithium plating [79]. Lithium plating causes a loss of active cycling Li-ions and an increase of internal resistance due to growth of the passivating SEI on the anode in the process. It is interesting to note that lithium plating seems unique among the Li-ion battery aging processes in that its rate is accelerated at lower temperatures [80]. Dendritic growth of metallic lithium depositions on the anode is a serious safety hazard. These dendrites might grow to breach the separator and thus cause catastrophic cell failure in the form of short-circuiting, possibly leading to ignition or explosion [37, 80, 81]. Due to the anode capacity dependence of the lithium plating reactions, the effect will usually be more evident at the later stages of battery life when anode capacity has been reduced and intercalation kinetics have slowed down due to various aging processes. Lithium plating will however tend to slow down after its onset as the resulting change in capacity balance between anode and cathode will counteract the effect [33, 80]. Figure 9 below summarizes the main degradation mechanisms at graphitic anodes.

*Figure 9: Main degradation mechanisms at graphite-based anodes [41].*
Lastly, it is worth mentioning that the main aging mechanisms are quite different for LTO-based anodes. First and foremost, no SEI formation will occur on LTO anodes due to their higher potential vs. Li/Li\(^+\) compared to graphitic anodes. For the same reason, no lithium or manganese plating will occur either. As mentioned in section 1.1.4.2, the volume change of LTO upon Li-ion intercalation is minimal which means that mechanical stress on cycling is also lessened. The main concern of LTO anode aging is instead the quite substantial production of gas at high temperatures. Still, it is commonly so that aging processes occurring at the cathode will dominate the overall aging behavior of cells with an LTO anode [53, 82, 83].

### 1.2.3 Aging mechanisms at cathode

In this section the characteristic aging mechanisms of different cathode materials will be discussed. While it generally is the case that processes as the anode will dominate the overall aging behavior of the cell, again, especially for the common graphite anodes, the cathode might also play an important role, depending on operation conditions and cell component materials. In particular, cathode aging may become the dominating factor for operation under especially harsh conditions, such as elevated temperatures [33]. First general aging mechanisms involving the cathode will be summarized, followed by a more detailed look at the specific aging behavior of common cathode materials.

The three main types of degradation processes are the same for all cathode materials: structural changes, metal dissolution and the formation of a passivating film. Sometimes this cathode film is also referred to as a SEI. One then must distinguish the anode SEI from the cathode SEI. When one talks about simply “the SEI” one will usually mean the anode SEI.

Like for the graphitic anode, volume changes caused by intercalation/de-intercalation of Li-ions during charging/discharging will induce structural changes in the cathode. This causes mechanical stress and may lead to phase transitions occurring in the material. These phase transitions may then cause further mechanical stress due to differences in specific volume between the phases. Phase transitions will also lead to a reduction of electrode capacity due to a loss of available sites for Li-ion intercalation. For these reasons, cathode materials will usually be doped in such a way as to reduce volume changes upon lithiation, for instance by incorporating aluminum or magnesium into layer structured cathode materials and cobalt, chromium, aluminum or magnesium into spinel structured ones [41].

Again, just like for the common graphite anode, there will also be film formation on the common cathodes due to side reactions with electrolyte. These reactions involving the electrolyte will however generally be slower and affect aging less than the ones occurring at the graphitic anodes. The cathode film will increase cell internal resistance and thus the available power of the cell. The cause of these reactions is the oxidation of the electrolyte by the electrode materials and/or LiPF\(_6\) decomposition [70].

Under normal operating conditions aging at the anode will dominate. As temperature increases however, the electrolyte oxidation is believed to become the crucial aging process. The main Li-ion battery cathode aging mechanisms, common to all cathode materials, are summarized in figure 10 below.
Due to the large differences in structure and chemistry of the common cathode materials, these will also generally display different aging behavior, at least in terms of what processes are the most central to the overall electrode aging. The most important differences between the aging processes of the common cathode materials will now be outlined. In terms of aging behavior, the cathode materials are divided into three categories, based on crystal structure: LMO, representing spinel structure materials; LFP, representing olivine structured materials; and the layered transition metal oxides, represented by LCO, LNO, NMC, and NCA.

1.2.3.1 Chemistry-specific cathode aging processes

1.2.3.1.1 Lithium manganese oxide

Due to the unique spinel structure of LMO and redox behavior of manganese, the dominating aging processes for these cathode materials are quite different to those occurring for the layered- and olivine structured cathode compounds. Manganese can exist in three different oxidation states: +2, +3 or +4. In the Li-ion battery cathode LiMn$_2$O$_4$, manganese is normally present in the oxidation states +3 or +4. When discharging an LMO-based battery at high C-rates, especially to low cell voltages, extra Li-ions may be intercalated into the cathode. This causes a reduction of Mn-ions from Mn$^{4+}$ to Mn$^{3+}$ in the cathode. The cause for this over-intercalation of Li-ions is the much slower diffusion of these in the cathode material compared to in the electrolyte. This causes an accumulation of Li-ions around the LMO cathode particles during discharge. This also allows for rationalization of the acceleration of this process during conditions of high C-rates and low voltages. A high C-rate indicates that more Li-ions are transferring in the cell and low voltage means fewer available sites for Li-ion intercalation in the cathode. This leads to an increased concentration gradient of Li-ions at the electrolyte-electrode interface, leading to increased rate of over-intercalation. The resulting reduction of Mn$^{4+}$ to Mn$^{3+}$ induces Jahn-Teller distortion, leading to a phase transition and corresponding volume increase of the cathode material. This phase transition also leads to a loss of cathode capacity due to a loss of active sites for Li-ion intercalation. The formed Mn$^{3+}$ may then disproportionate into Mn$^{4+}$ and Mn$^{2+}$. The
formed Mn$^{2+}$ dissolves in the organic electrolyte, as discussed in section 1.2.2. This process is catalyzed by traces of HF or other acids in the electrolyte [39, 56]. The effect is also more prominent at higher temperatures, partly due to the faster kinetics of the reaction itself and partly due to the increased rate of HF formation [32, 41]. The dissolution of manganese might contribute to as much as 20 to over 30 percent of the total capacity loss of LMO cathodes [39].

Due to these reduction and disproportionation processes, unique to these types of cathodes, LMO-based materials are especially sensitive to high C-rates and low voltages. To alleviate this problem somewhat, researchers have tried to develop special inorganic coatings of Al$_2$O$_3$, ZrO$_2$, MgO or AlF$_3$ to combat over-intercalation, or special manganese trapping separators to stop the migration of already dissolved Mn$^{2+}$ to the anode [84]. The manganese dissolution process is illustrated in figure 11.

Looking at figure 11 it is again seen how important it is to limit the exposure of the cell to water.

1.2.3.1.2 Lithium iron phosphate

Due to the stronger chemical bonds to the oxygen atoms, LFP materials are generally more stable at higher temperatures compared to both the layered transition metal oxides and LMO, in the sense that fewer side reactions with the electrolyte will occur. In fact, cell voltage would have to be as high as 5.4 V for oxidation of common electrolytes to be at all possible, far higher than the normal operation window of LFP cells [39]. There does exist an HF catalyzed iron dissolution mechanism somewhat like the dissolution of manganese from LMO cathodes, but it is considered to be of less concern for the overall aging. Dissolution of iron results in a loss of active material and thus battery capacity along with an increased internal resistance due to the growth of the anode SEI [44, 56]. In addition, the lower cell voltage when using these cathode materials leads to a lowered reactivity towards the electrolyte during storage [32]. Due to the low ionic and electronic conductivity of LFP cathodes they are especially sensitive to high C-rates, which will lead to accelerated aging [39].
1.2.3.1.3 Layered transition metal oxides

The dissolution of metal ions is in general not a concern for these compounds except for during operation at overly high voltages (beyond 4.2 V vs. Li/Li⁺). Note that this goes even for the manganese-containing NMC as manganese is only present in the inactive +4 form in this material. Capacity fade due to phase transitions is also less of a concern [73]. The formation of a cathode SEI is accelerated by high voltages and, in addition, high temperatures. Even though these processes will occur for all cathode materials, it is of special importance for this class of electrodes. These transition metal oxides can act as oxygen donors in themselves by forming a phase of low lithium/oxygen with a rock-salt structure. This rock-salt phase is also a detriment to Li-ion transport through the electrode, being of lower conductivity and thus increasing cell internal resistance. The electrolyte is oxidized during this process, and transesterification or polymerization reactions may occur, depending on the solvent composition and conditions [41]. During these reactions gas may be released, potentially causing pressure build-up and subsequent deformation of the cell [85]. While all layered transition metal oxides share this behavior, it is especially prevalent for LCO, and the less common LNO, cathodes. As the Li-ion content in these cathodes drops below around 50 percent of the maximum intercalated amount, electrolyte oxidation becomes severely limiting and for this reason one should be careful not to operate batteries with a LCO or LNO cathode at low SoCs frequently [39]. The most important aging processes of layered transition metal oxides are summarized in figure 12 below.

![Figure 12: Aging mechanisms of layered transition metal oxides [41]](image)

1.2.4 Main aging impact factors

This section aims to summarize the various factors that affect battery aging divided into calendar- and cycle aging as is often done in the literature. Calendar aging is defined as the aging of the battery that is occurring while it is not being operated, that is when no current is entering or leaving the battery. Cycle aging is the aging happening when the battery is being either charged or discharged. The goal is to recapitulate the most commonly stated functional relationships between operating conditions and aging rate. This is essential for an overall understanding of the degradation processes and their interconnections. It is also the first step to developing a holistic aging model and to optimize battery operation. As shall be seen, it will not be necessary to differentiate between various aging mechanisms in battery monitoring per se, as the actual effects on battery performance largely overlap. This allows for the definition of a relatively simple metric of overall battery aging status, the state of health.
1.2.4.1 Calendar aging

The main factors affecting the extent and rate of calendar aging are cell chemistry, temperature, calendar time and the SoC during storage. High temperatures will accelerate the rate of all side reactions, resulting in losses of active electrode material as well as of active cycling Li-ions. The effect increases exponentially as seen in figure 13, thus suggesting a Arrhenius-like description [86].

![Figure 13: Effect of storage temperature on aging in terms of percentage of capacity loss. All batteries were stored at 50 % SoC during testing [86].](image)

Very low temperatures will on the other hand lead to lithium plating on the anode becoming the dominating aging process [33]. This alludes to two different aging regions based on temperature: at low temperatures lithium plating on anode will dominate, and at high temperatures SEI formation on both electrodes (more so for the layered metal oxides, less so for LMO and especially LFP), as well as dissolution of manganese for LMO based cells, will govern the overall aging [46]. Because of this acceleration of aging at both high and low temperatures it is quite difficult to encompass the total behavior in one Arrhenius-like expression for all temperatures [87]. For operation within a limited range of temperatures this might be sufficient however [88, 89]. The temperature dependence of calendar aging has been reported to be different for different cell chemistries and even cell geometries. Waldmann et al. found that lithium plating will dominate under 25 °C, while SEI formation is more important above 25 °C for the investigated NMC cells and therefore suggested using two different Arrhenius-like expressions for these ranges, see figure 14 below [46].

![Figure 14: Arrhenius plot for high and low temperature behavior. r is the rate of reaction [46].](image)
Bandhauer et al. reached a similar conclusion, stating that battery aging will be very rapid over 50 °C, especially so for storage at high SoC due to the growth of the SEI reducing available amounts of Li-ions, and under -10 °C due to lithium plating on charging. Because of the risk of dendritic growths of lithium on the anode, charging at high C-rates when temperature is low should be avoided as far as possible as this might cause short-circuiting which might be a serious safety risk. Thermal monitoring and control is therefore essential for Li-ion batteries, particularly so for larger applications [90].

As for the effect of SoC on calendar aging, a higher SoC during storage implies a state further from the thermodynamic equilibrium of the cell. This in turn makes reactions between electrodes and electrolyte more favored. The exact mathematical relation is however not clear. Exponential [91], but also linear [88], SoC-dependencies of capacity loss during storage have been reported. Su et al. noted two distinctly different aging behaviors below and above 95 percent SoC during storage and attributed this to a change in the dominating aging mechanism perhaps due to different compounds being formed and incorporated into the SEI at SoCs above 95 percent [92]. One way of introducing the SoC dependence of calendar aging is to proceed via the Tafel equation which describes the rate of an electrochemical reaction as a function of the overpotential, the overpotential being closely related to the storage SoC [70].

Total calendar aging, measured in either capacity loss or increase of internal resistance, will not be linear over time but will generally be more rapid during the earlier stages of the battery’s life [70, 93]. Several authors have concluded a time dependence of t$^{1/2}$ for calendar aging [70, 86, 89, 94-96]. This has been widely attributed to the continuous loss of Li-ions due to the SEI formation on the anode [97]. Indeed, first-principles kinetic investigations suggest that this should be the case for graphite anodes. However, larger exponents such as 0.75 have also been reported. It is believed that this might be an indication of a more complex relationship between calendar aging and storage time, meaning that other mechanisms than just SEI formation are contributing [81, 88]. As SEI formation on the anode appears to at least be the dominating aging mechanism during storage, cells with titanate-based anodes should see relatively little calendar aging as no SEI is formed on these electrodes.

The calendar aging behavior of Li-ion batteries, measured as capacity fade, appears to be a memoryless or path independent process, meaning that future aging only depends on the current state of the battery and future operating conditions. It may therefore be considered a Markovian process. This also indicates that capacity loss represents a good measure of overall battery aging as the history of capacity fade does not need to be considered to predict its future development [89, 92]. The importance of this fact will become evident in the upcoming sections on battery modeling and capacity fade estimation.

Eddahech et al. investigated the role of battery chemistry on calendar aging. NMC, NCA, LFP and a LMO-NMC hybrid were compared. These were stored at a set of conditions of SoC and temperature. It was determined that the LMO- and NMC-based cathodes were particularly sensitive to high storage temperatures. Dissolution of manganese was confirmed to be a leading factor in the aging for these cells by post-mortem analysis. LFP was most resistant to degradation at all temperatures while NCA was found to be the middle-of-the-road choice [98].

The work done on calendar aging of cells featuring LTO-based anodes suggests that these materials will vastly outperform those with graphite based anodes in terms of capacity and internal resistance retention over time. Belharouak et al. compared the calendar aging of two LMO-cathode cells with all things the same except for the anode material, one being traditional graphite and one being LTO. Both cells were held at 70 percent SoC and 55 °C during the 28-day test, except for when measurements of internal resistance using a current pulse method were performed once per day. The striking results are
shown in figure 15 [83].

As can be seen in figure 15, the calendar aging of the LTO-based cell, measured in increase of area specific impedance, is almost non-existent over the course of testing, especially compared to the graphite-based cell. The authors attribute this extreme difference to the lack of SEI-formation in the LTO cell.

1.2.4.2 Cycle aging

Cycle aging is occurring when the battery is either being charged or discharged. Due to the dynamic nature of the processes occurring in the battery during cycling, the associated aging phenomena are more complex and multi-faceted. The number of impacting parameters is also larger. Of course, it is also impossible to separate out the always ongoing effects of calendar aging during tests of cycle aging, further complicating the study of these processes [70].

One important factor to consider is the variation of SoC, or ΔSoC, during cycling, which is usually given as an average value. It has been demonstrated that aging will be accelerated for high ΔSoC cycling. This might be related to the mechanical stress induced by the large change in intercalated lithium in the electrodes and the accompanying volume change, particularly for the cathode [99].

The C-rate appears to also play a key role. A high C-rate will, due to exothermic side reactions and the internal resistance of the cell leading to Joule heating, increase the working temperature, thus further increasing the rate of side reactions. Some authors have therefore included thermal effects into their aging models to account for this. The most straightforward method of accomplishing this is to include only Joule heating due to internal resistance, which is easily calculated from an equivalent circuit model for instance [91, 100]. The higher rate of intercalation/de-intercalation of Li-ions resulting from the higher C-rate also implies accelerated transport of Li-ions through the cell. At very high C-rates there is a chance that Li-ions are not completely intercalated/de-intercalated from the electrodes, leading to increased rates of lithium plating and dendrite formation [64, 67]. In addition, synergistic effects may exist. For instance, a low temperature along with a high C-rate and low voltage will accelerate lithium plating further [41].
Interestingly, Wang et al. introduced a C-rate dependence of cycle aging by noting that the activation energy term in their developed Arrhenius-like aging expression became smaller at higher C-rates, thereby including a factor like the one below in the final expression describing capacity loss to accommodate for this effect [97].

\[
C_{\text{loss}} \propto \left( e^{\frac{A + B\cdot C_{\text{rate}}}{RT}} \right)
\]

Where \(C_{\text{loss}}\) is the absolute loss of capacity compared to the new battery, \(R\) the gas constant, and \(T\) the temperature in Kelvin. \(A\) and \(B\) are constants, where \(A\) is large and negative and \(B\) positive. The model equation was found to fit experimental data well at varying C-rates.

Schmalstieg et al. investigated the influence of not only \(\Delta\text{SoC}\), but also of the average SoC during operation on cycle aging. Interestingly, it was noted that battery capacity degradation was accelerated at both high and low average SoC cycling, with a minimum at around 50 percent average SoC. This then means that if one wants to minimize the effects of the always occurring calendar aging by keeping the average SoC low, the rate of aging due to cycling will increase instead. Therefore there should exist an optimal average SoC where overall aging is minimized for a given battery usage pattern or application [88].

As aging due to cycling accumulates over the lifetime of the battery, there needs to be some quantitative measure of how much cycling has occurred to properly account for this effect. The two most popular options for lifetime models are cycle number counting and calculation of cumulative Ah-throughput. The Ah-throughput is the total transferred charge during the lifetime of the battery [41, 101]. Often some exponential relationship between Ah-throughput and capacity loss is found to appropriately fit data [88, 91]. When using cycle number to track the cumulative transferred charge it is necessary to define what is exactly meant by a cycle. Li-ion batteries will typically not be just repeatedly fully charged and discharged after all. One method of transferring current-time data to equivalent cycles is via the rainflow-counting algorithm as done in [96, 102].

### 1.2.5 Summary of Li-ion battery aging impact factors

As can be concluded from the discussion above, Li-ion battery aging is a highly complex issue. This is partly due to the number of influential factors but also the interactions of these. Furthermore, it is clear that the area of application and the usage pattern are essentially important to the aging process through the dependencies on C-rate, average SoC and \(\Delta\text{SoC}\). The main factors affecting calendar and cycle aging discussed in section 1.2 are summarized in table 5 below.

**Table 5: Factors affecting calendar and cycle aging.**

<table>
<thead>
<tr>
<th>Impact factors for calendar aging</th>
<th>Impact factors for cycle aging</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cell geometry</td>
<td>(\Delta\text{SoC}) during cycling</td>
</tr>
<tr>
<td>Cell chemistry</td>
<td>C-rate</td>
</tr>
<tr>
<td>Temperature</td>
<td>Cycle number or Ah-throughput</td>
</tr>
<tr>
<td>SoC during storage</td>
<td>Average SoC during cycling</td>
</tr>
<tr>
<td>Calendar time</td>
<td></td>
</tr>
</tbody>
</table>

Factors placed in the middle of table 5 affect both calendar- and cycle aging. Effects of calendar and cycle aging are normally considered additive. This means that the full aging behavior can be described using one modeling function with some factors being only dependent on average SoC, temperature and time, these thus effectively representing calendar aging. Whether this is representative of the
fundamental physical- and chemical processes causing Li-ion battery aging is very difficult to determine due to the interconnectedness of the mechanisms and their effects [86, 91, 103].

The aging mechanisms discussed above will generally lead to a loss of battery capacity and/or an increased internal resistance. The effects of Li-ion battery aging, common mechanisms, and associated important impact factors are summarized in table 6 below, based on the discussions in previous sections [69].

Table 6: Impact factors and effects of aging mechanisms [69].

<table>
<thead>
<tr>
<th>Mechanisms</th>
<th>Effects</th>
<th>Impact factors</th>
</tr>
</thead>
<tbody>
<tr>
<td>SEI growth</td>
<td>Capacity loss, resistance increase</td>
<td>Time, high temperature, high voltage, high C-rate</td>
</tr>
<tr>
<td>SEI decomposition, cracking</td>
<td>Capacity loss</td>
<td>High temperature, high voltage, high C-rate</td>
</tr>
<tr>
<td>Electrolyte oxidation</td>
<td>Capacity loss</td>
<td>High temperature, high voltage</td>
</tr>
<tr>
<td>Graphite exfoliation</td>
<td>Capacity loss, resistance increase</td>
<td>High voltage, high C-rate</td>
</tr>
<tr>
<td>Lithium plating</td>
<td>Capacity loss, resistance increase</td>
<td>Low temperature, low voltage</td>
</tr>
<tr>
<td>Electrode cracking</td>
<td>Capacity loss, resistance increase</td>
<td>Mechanical stress, high C-rate</td>
</tr>
<tr>
<td>Transition metal dissolution (Mn, Fe)</td>
<td>Capacity loss, resistance increase</td>
<td>High C-rate, low voltage, high temperature</td>
</tr>
</tbody>
</table>

Here it is also evident why the degradation of cell capacity is a sensible choice for an indicator of overall battery health when discussing battery aging. As is evident in table 6, all major aging mechanisms lead to a loss of battery capacity while only some lead to an increase of internal resistance. Therefore, capacity loss may be considered a better representation of the overall battery health in the general sense. Of course, also being able to also estimate the increase of internal resistance allows for a more nuanced picture of the occurring aging-related chemical processes.

It has also been seen that aging characteristics are dependent on cell chemistry, especially so for different anode materials. LTO-based cells display superior calendar- and cycle aging compared to those featuring a graphite anode. In addition, the main aging mechanisms are very different for LTO-based cells as no SEI formation and hardly any electrolyte oxidation will occur in cells with these anodes. Most physics-/chemistry based aging models of Li-ion battery aging are based on cells with a graphite anode and these will therefore generally not describe the aging in LTO-based cells accurately.

It is also interesting to note that overall aging behavior may depend on the interconnection of cells in parallel and/or series. Minimizing variations in thermal conditions, SoC and capacity loss between cells in the same battery pack might be crucial in preventing accelerated aging due to overcharging. Fortunately perhaps, thermal and electrical balancing are somewhat interconnected in that hotter cells are also capable of faster C-rates due to the lowered internal resistance as a result of the increased temperature [90]. Obviously, for cells connected in series the cell with the lowest capacity determines the capacity of that whole system [91]. Even though these effects seem important, relatively little material has been published on the aging behavior of entire packs or modules of cells, perhaps partly due to the increased need for measurement equipment and the increased complexity [63]. The optimal BMS would estimate the capacity of all cells in the system simultaneously and then distribute load so that the aging of cells was as even as possible, naturally an enormously challenging task [5].
Due to the vast number of interconnected processes that make up Li-ion battery aging, and because all mechanisms have not yet been fully understood, it is currently not possible to apply a first-principles aging model for battery monitoring. As an accurate model is essential to achieve acceptable results, several simplified approaches have emerged to tackle the issue. Another reason as to why simplified battery models need to be used is to ensure satisfactory computational performance. Due to the transient nature of Li-ion battery behavior, it is essential that the sampling time of measurements is sufficiently brief to allow for this behavior to be accurately captured. In this sense, a complex model might, slightly ironically, actually reduce overall accuracy of battery state estimation as the sampling time might need to be too long for the algorithm to “keep up” with new measurements. Reducing model complexity also increases hopes of flexibility as to be usable for different battery chemistries, geometries and operation conditions. This is because fewer cell characterization measurements need to be made to parameterize the model. In fact, as shall become evident in later sections on for instance dual extended Kalman filtering, online characterization might be an attractive option for certain battery models.

1.3 LITHIUM-ION BATTERY MODELING

In the clear majority of applications, the measurement data available for determination of battery status during actual operation is essentially limited to battery voltage, current and (surface) temperature. To indirectly estimate battery state and to monitor its status using just these relatively few measurements, a battery model is necessary. For instance, determining the battery SoC and/or SoH is often a goal of monitoring. A satisfactory battery model should be able to capture and replicate the behavior of the real battery to an acceptable extent whilst still achieving adequate computational performance [47]. The limiting factor for computational performance is frequently the processing power of the BMS for online solutions, offline methods usually having more leeway in this regard. The most commonly applied battery models can be divided into three categories:

- Purely empirical models,
- Electrochemical models,
- Equivalent circuit models.

Most current BMSs utilize either empirical- or equivalent circuit models [5]. There also exist crossbreeds of the three categories, such as analytical-electrochemical models [104]. Due to their relative obscurity, these types of combined approaches will not be discussed further. Out of the three main categories above, the electrochemical models may be considered the most advanced while also being the most intensive computationally. The empirical models are found at the other end of the spectrum, often utilizing relatively simple equations to simulate battery behavior and thus achieving good computational performance, this at the cost of some accuracy. The equivalent circuit models (ECMs) seem to currently constitute the happy medium between these two extremes, achieving both respectable performance and accuracy. In addition, the values of the ECM parameters, at least for simpler model variants, have physical interpretations, thus facilitating a more intuitive handling of issues such as battery aging. The three different modeling approaches will now be discussed in more detail, starting with empirical models.
1.3.1 Empirical models

Empirical models can generally be divided into analytical and stochastic models. Examples of analytical models include the Nernst model, the combined model and the kinetic battery model (KiBaM). These models use relatively simple equations to describe battery dynamics in a purely pragmatic sense without any physical or chemical interpretation. The KiBaM, which was originally developed for lead-acid batteries, will for instance model the battery charge (currently available capacity) as being in one of two hypothetical wells, either the available charge well or the bound charge well. The flowrate between these two wells depends on the difference in height of the two “liquid levels” in the wells [105]. A newer extended version of the KiBaM is the spatial kinetic battery model [106].

Stochastic models will generally model battery behavior as a discrete Markovian process, meaning that a probabilistic approach is used. Like the analytical models, these models often utilize a high level of abstraction [106].

While the empirical models might encapsulate battery behavior well, they have the quite major downside of their parameters lacking physical interpretation meaning that they are not particularly well suited for estimation of SoH or RuL [104].

1.3.2 Electrochemical models

These models are based on electrochemical kinetics and mass transfer equations. The most common approaches are the single-particle model (SPM) and the pseudo-two-dimensional model (P2D), the SPM being more simplified and the P2D more accurate, broadly speaking. The SPM models the electrodes as being spherical particles with the same total area as the real porous materials, with transport phenomena inside said hypothetical particles being considered. The SPM validity is limited for cases of high C-rates, larger cells, and when electrolyte effects are significant. The growth of the SEI layer can easily be included into the SPM to model aging effects such as loss of capacity [5, 107]. The P2D takes more physical- and chemical phenomena into account than the SPM, such as Butler-Volmer kinetics and Li-ion concentrations in both the liquid- and solid phases of the cell. Here the electrodes are modelled as consisting of several identical particles [108].

These types of models are often very computationally demanding, requiring the numerical solution of a set of interconnected PDEs in each iteration of the algorithm. This means that online monitoring is not always feasible. The large number of fitted parameters might also be questionable and overfitting may prove to be a real issue. In addition, more information about the battery needs to be gathered through characterization tests to parameterize these models in comparison to ECMs. This means that more experiments need to be performed before operation can begin, which might be expensive and time consuming [109]. The P2D and especially the SPM are very simplified models which have been developed to try to alleviate the computational strain of other electrochemical models. This seems to come at the sacrifice of some accuracy however and these simplified models are still slower than most equivalent circuit models [104].

On the other hand, electrochemical have higher potential of future development than ECMs in at least two regards. Firstly, electrochemical models could potentially be used for predictions of battery performance for varying geometric parameters and conditions and, secondly, side-reactions leading to aging could be incorporated directly into electrochemical models [109].
1.3.3 Equivalent circuit models

It appears that ECMs constitute the most popular battery models for SoC and SoH estimation in the literature, presumably as they seem to provide a reasonable compromise between accuracy and computational performance. The basic principle of an ECM is to model the dynamic behavior of the battery using simple hypothetical electric circuits consisting of elements such as resistors, voltage sources, Warburg elements and capacitors connected in series and/or parallel configurations. These elements may be seen to represent some real physical phenomena occurring in the battery, at least for simpler equivalent circuits [110]. The simplest possible ECM consists of a voltage source coupled to a resistor in series, as seen in the figure below.

![Figure 16: A simple equivalent circuit model consisting of a voltage source, dependent on the SoC, and a resistor in series.](image)

Here $V_t$ is the measurable terminal cell voltage; OCV is the open circuit voltage of the cell, a thermodynamic quantity specific to the cell chemistry; $R$ represents the internal resistance of the cell, as per Ohm’s law. The OCV of the cell is readily calculated using the equilibrium potentials of the electrodes of the cell:

$$OCV = E_{\text{cathode}} - E_{\text{anode}}$$  \hspace{1cm} (1)

The equilibrium potentials of the Li-ion battery electrodes are dependent on the stoichiometry, that is the amount of intercalated lithium ions, or the SoC, the standard potential of the electrode materials, and weakly on temperature. In addition, it is also affected by battery aging through changes in electrode geometries, such as changes to crystal structure and/or morphology, and loss of Li-ions available for cycling [109, 111]. This effect is however small and can be neglected for most purposes as will be the case in this work [65].

While the model above is extremely simplified, it still captures some fundamental battery behavior. For instance, the OCV will change as a function of SoC, that is depending on how much charge is left in the battery, and the cell voltage will also drop during load due to the internal resistance of the battery per Ohm’s law:

$$V_t = OCV(SOC) - I \cdot R$$  \hspace{1cm} (2)

The inclusion of resistor elements into the ECM also allows for thermal modeling by utilizing Joule’s first law:

$$P \propto I^2 \cdot R$$  \hspace{1cm} (3)

Here $P$ is the dissipated power. In a real battery, there are however additional effects which one might want to consider for a more realistic model. Most importantly, a real cell will not respond immediately
to an applied current. It will also not instantly decay back to the OCV after being rested. This is essentially due to diffusion effects inside the cell. This effect is most often modelled by a series of parallel resistor-capacitor (RC) couples like in the ECM below:

![Resistor-Capacitor Equivalent Circuit Model](image)

Figure 17: n resistor-capacitor equivalent circuit model.

Where $C_i$ is the capacitance of the $i$:th capacitor and $V_i$ the transient voltage over the $i$:th resistor-capacitor pair. Using Kirchhoff’s current law, which states that the sum of all currents flowing into and out of a node in an electrical circuit must be zero, and Ohm’s law, the transient voltage over a given resistor-capacitor pair for the case of zero current may be written as:

$$C_i \frac{dV_i}{dt} + \frac{V_i}{R_i} = 0$$

This differential equation gives the following solution:

$$V_i = V_{i,0} \cdot e^{-\frac{t}{R_i C_i}}$$

(5)

$V_{i,0}$ is the transient voltage over the $i$:th resistor-capacitor pair at zero time. Equation (5) clearly shows the exponential decay of the transient voltage when current is cut off. When a current $I$ is applied to the sub-circuit the following differential equation is the result:

$$\frac{dV_i}{dt} = -\frac{1}{R_i \cdot C_i} \cdot V_i + \frac{1}{C_i} \cdot I$$

(6)

The differential equation (6) may be converted to discrete time which gives the following result, without going over the details of the derivation. As shall be seen, the conversion to discrete time is a requirement for the application of the model equations in Kalman filtering methods anyway. The subscript $k$ represents the time-step and $\Delta t$ is the time between the time-steps $k$ and $k+1$.

$$V_{i,k+1} = e^{-\frac{\Delta t}{R_i C_i}} \cdot V_{i,k} + R_i \cdot (1 - e^{-\frac{\Delta t}{R_i C_i}}) \cdot I_k$$

(7)

This expression has two terms. The first represents the exponential decay of the transient voltage over time when no current is applied and the second represents the transient voltage response to a charging- or discharging current. With this expression in hand, it is simple to add RC couples to an ECM by just adding on more terms on the form of (7) for every RC couple. The terminal voltage for an $n$ RC ECM can simply be written as:
\[ V_t = OCV(SOC) - I \cdot R_0 - \sum_{i=1}^{n} V_i \] (8)

An expression on this form will be used in this work. Hu et al. compared a set of twelve different battery ECMs commonly applied for Li-ion battery modelling. It was found that the 1 RC model performed the best overall in terms of compromise between accuracy of estimation and computational performance and this will therefore be the battery model of choice in this work as well. Interestingly both Nejad et al. and Hu et al. also noted that the most suitable ECM was found to be somewhat dependent on battery chemistry, with the 1 RC hysteresis model achieving better results for LFP-based cells [104, 110]. Of course, real battery behavior is much more complex with concentration- and temperature gradients, current and material distributions just to name a few aspects which are difficult to account for using ECMs [100]. There is however no denying the success that these models have had historically [104]. A simple sketch of a 1 RC ECM, as will be used in this work, is shown below with accommodating notation for all equivalent circuit elements.

![Figure 18: A 1 resistor-capacitor equivalent circuit model.](image)

The accuracy of an n RC ECM may be increased by increasing the number of RC couples or by including some SoC- or temperature dependence into the ECM parameter values [112]. Due to the lack of appropriate battery diagnostic data, this was not possible in this work however.

### 1.4 STATE OF CHARGE ESTIMATION

For the discussion of battery technologies there exist a set of field-specific concepts. As the exact characterization of these will vary slightly from author to author the definitions used in this work will now be specified for future reference. First the state of charge (SoC) of a battery will be defined as follows:

\[ SoC = \frac{Q}{C} = \frac{\int (\eta \cdot I) \, dt}{C} \] (9)

Here Q is the current remaining charge in Ah and C the current maximum capacity in Ah of the battery. Maximum capacity is defined as the amount of charge in Ah that can be discharged from a fully charged battery, meaning a battery at the upper cut-off voltage, until the lower cut-off voltage is reached. Note that C will change over time as the battery performance degrades due to various aging processes.
Lastly, $\eta$ is the Coulombic efficiency of the cell, a measurement of how close the cell gets to total charge conservation defined as discharged/charged ampere hours. For most Li-ion cells, $\eta$ is very close to one as side reactions generally are of little concern. If the Coulombic efficiency is assumed to be exactly equal to one, as is the case in this and many other works on Li-ion battery monitoring, it does not need to be included in equation (8) and one can simply calculate the transferred charge by summarizing the product of current and $\Delta t$ over all time steps [82].

The SoC is a measurement of how much charge can be discharged from the battery in its present state as a percentage of the maximum available charge. The maximum available charge is also known as the capacity. This is then the EV equivalent of the remaining fuel gauge in internal combustion vehicles, again with the key difference that the “fuel tank” in EVs shrinks over time [64]. The concept and definition of SoC has long been debated. Here a simple engineering definition is employed for the sake of convenience. One should however keep in mind that a “true” SoC must be measured or estimated at thermodynamic equilibrium, unaffected by chemical-, thermal-, mechanical or other gradients, which is not so easily achievable and also extremely time consuming [109].

When comparing capacities, it is important to acknowledge that the capacity is sensitive to especially temperature (measured as being 7.7 percent higher at 45 °C than at 15 °C in [113]) but also the C-rate (higher discharge capacity at lower C-rates) to some extent [113]. Both effects can be rationalized by considering the accelerated chemical kinetics and mass transfer at elevated temperatures and C-rates. In this work, these effects were not considered as data on temperature and C-rate dependency was not available for the examined batteries, but it is worth noting that these effects exist and will affect the results.

The remaining charge can be measured via simply integrating the battery current over time if the initial remaining charge is known, for instance if the battery is initially fully charged (then $Q_0 = C$):

$$Q = Q_0 - \int I dt$$

As the maximum capacity of a new battery is frequently specified by the battery supplier, the SoC can then simply be calculated by dividing (10) by this known quantity. This method of calculating SoC is commonly referred to as Coulomb counting. With Coulomb counting it seems as if though the issue of calculating SoC has been solved as it simply requires measuring the applied and discharged current to/from the battery. The method has some quite severe shortcomings however. Firstly, the initial SoC of the battery must be known. This is as a rule only the case when the battery has been fully charged (SoC=1) which might be too rare an occurrence for certain applications to achieve reliable, accurate results. Secondly, due to the summation over all time steps, all slight biases and other errors of measurement sensors will accumulate over time, thus gradually degrading estimation accuracy. Thirdly, the maximum capacity must be known beforehand. For anything but a new battery this will generally not be the case and the changes to the capacity must be determined experimentally or estimated by some other method as the battery ages.

In other words, Coulomb counting is far from perfect [109]. For this reason, several other, more advanced approaches have been developed. The starting point of most of these is to not only utilize current data but also cell voltage for SoC estimation. This is often referred to as closed-loop SoC estimation. Despite these innovations Coulomb counting remains the most popular SoC estimation method in the industry [114].
SoC estimation is a relatively explored but still challenging field. The basic requirements of a SoC estimator are computational performance, accuracy and robustness. The essential detail of these algorithms is the utilization of cell voltage data along with the current data. The cell voltage behavior can be modelled as a function of the SoC, either via the OCV or directly and can thereby act as a calibration factor for the SoC calculated via Coulomb counting. The accuracy of SoC determination is essentially limited by the accuracy of the cell voltage modelling. Of course, offline SoC estimation during resting of the battery is readily performed given that the OCV-SoC relation is known. If enough time has passed since the interruption of operation, overvoltages will have decayed fully and the “pure” OCV is then directly measurable. This method is less suitable for cell chemistries with particularly flat OCV-SoC curves, especially LFP, as voltage measurements must be extremely accurate to ensure a precise SoC determination. For most battery applications, this is however of little interest as operation must be halted for extended periods to perform the measurement. Therefore, only online SoC estimation will be considered further in this work.

Online SoC estimation algorithms are plenty and remains a very active field of research. See [65] and [109] for in-depth reviews on the subject. The most popular methods are based on regression, utilizing one of the battery models outlined in section 1.4 of this work and an algorithm for interpretation of measured voltage, current and, possibly, temperature data. There exist many algorithms capable of serving this purpose. The most common are based on Kalman filters such as the linear Kalman filter (LKF) [115], the extended Kalman filter (EKF) [116], or sigma point Kalman filters (SPKFs) [117], arranged in order of increasing capability of the Kalman filter of handling non-linear systems. A smaller number of algorithms based on state observers such as the sliding mode observer [118] or the simpler Luenberg observer [119] have also been suggested.

1.5 STATE OF HEALTH AND REMAINING USEFUL LIFE ESTIMATION

The battery state of health (SoH) is an indicator of the extent of battery performance degradation due to aging processes. The need for a catch-all concept is evident when one considers the immense complexity of battery aging along with the highly limited possibilities of diagnosing the nature of the aging processes during actual battery operation. Unfortunately, there exists no set definition of SoH and several different metrics are used in the industry [64]. Most often SoH is defined as the maximum capacity of the battery in its current state as a percentage of the initial nominal capacity of the fresh battery. Nominal conditions are frequently 25°C and 1 C (one hour discharge rate) [82]. In the data used in this work capacity measurements were performed at close to these conditions.

\[
SoH = \frac{C}{C_{nom}}
\]  

(11)

As mentioned earlier, it is important to keep in mind that the cell capacity is not only a function of the extent of aging but also temperature and current. In other words, if one wants to accurately compare the capacities of an aged and a fresh battery, for instance to estimate battery SoH, testing conditions should strictly speaking be identical. In practice, such as in battery online monitoring, this is rarely achievable and these effects are therefore often ignored for the sake of practicality. As the nominal capacity of the fresh battery is normally a known quantity, SoH estimation is essentially a matter of estimating the capacity of the aged battery. The simplest way of determining the current maximum capacity of the battery is of course to measure the total transferred charge during a full discharge of the battery from a fully charged state. This is normally not possible during normal battery operation and can be time consuming and costly. It also reduces availability of the battery to perform its intended function [5]. All the issues of Coulomb counting also apply. Naturally, the SoH, as defined in (11), is closely connected to the SoC through the following relation:
\[ C = \text{SoH} \times C_{\text{nom}} = \frac{\Delta Q}{\Delta \text{SoC}} = \frac{\int_{t_1}^{t_2} (\eta_i)dt}{\text{SoC}_2 - \text{SoC}_1} \] (12)

Another commonly found SoH definition is the relative increase of internal resistance compared to that of a new cell. The easiest way of measuring battery internal resistance is by doing a pulsed charge- or pulsed discharge test, meaning that a short current signal is charged or discharged to/from the battery, respectively. The instantaneous voltage response is then equal to the internal resistance multiplied by the applied current per Ohm’s law \([120]\). This will be the method of choice for determining internal resistances in this work and will be described in more detail in a later section.

As a result of the different definitions used, discussions of SoH algorithms may become confusing and difficult. To alleviate this, efforts have been made to try to concretize the concept of SoH by establishing a correlation between capacity loss and resistance increase, thus potentially establishing a universal definition of the concept. Unfortunately, this work has not been particularly successful, even though battery-specific correlations certainly seem possible \([65, 121]\). Looking back at section 1.3, it appears that the concept of capacity loss encompasses more of the common aging processes and therefore appears to be a more meaningful metric for diagnosing battery health in terms of its safety status. The focus will thus largely be on estimating capacity degradation in this work. This is also keeping in mind the intended application, stationary renewable energy storage, where maximum power is generally of less importance than maximum energy. For say hybrid electric vehicle applications, a definition based on internal resistance might be more sensible as issues of power and heat are likely to be more essential \([65, 82]\).

Determining the SoH is critical for a number of reasons. Firstly, it allows for the prevention of critical battery failure, which is especially important due to the safety concerns of Li-ion batteries. Secondly, it allows for the optimization of battery usage and maintenance such as to maximize efficiency and return on investment \([51]\). Thirdly, it is necessary for accurate SoC estimation. When effects of operation conditions on aging for a specific battery chemistry can be predicted, battery systems can be managed in such a way as to maximize lifetime and thus efficiency and return on investment \([10]\).

SoH is closely linked with the concept of remaining useful life (RuL), the time, or alternatively number of full equivalent battery cycles, until the battery is no longer considered usable for the intended application. When RuL is zero the end of life (EoL) has been reached. EoL is in turn usually defined as when a set SoH has been reached. Note that the battery is always still technically functional at EoL. The threshold SoH at EoL depends on the intended application. For vehicle applications (EVs, HEVs) it is often set at 80 percent SoH, while it can be as low as 65 percent for stationary applications \([64]\). This means that one can determine the RuL if can predict how SoH will develop over time depending on certain key contributing aging factors. Alternatively, one can get an idea of the RuL by simple curve fitting and extrapolation of SoH data over time, as seen in the figure below. This method assumes identical future operation conditions, of course.
More advanced statistical RuL methods have also been suggested, for instance utilizing a particle filter [122, 123].

Of course, like SoC, it is not possible to directly measure SoH during battery operation, no matter the applied definition. For this reason, many online SoH estimation methodologies have been, and continue to be, suggested, varying immensely in approach and complexity. Roughly speaking, the most popular online SoH estimation methods are based either on a battery model combined with some filtering- or observer algorithm, or are based on machine learning algorithms, even though combinations have also been suggested [124].

The most common machine learning algorithms are currently based on support vector machines, fuzzy logic, or artificial neural networks. The idea is to develop a “black box” model of the battery using large amount of operational data. The “black box” model can then be used as a virtual copy of the real battery for testing, which can involve internal resistance, capacity, or both. While these types of approaches have the potential of universality and robustness there still exist some issues. For instance, they tend to be too computationally straining in order for them to be implemented in an online controller and require large amount of data to be collected. Instead, operational data is often collected over a time span and then processed in a batch-wise fashion when using these methods [7]. One often speaks of the algorithm “learning” the battery behavior. In one recent work, Klass et al. used approximately 880 000 data points in their SVM-based algorithm, which amounted to an offline training time of $1.6 \times 10^6$ core seconds ≈ 44 hours on the used computer. The data collection has to be repeated whenever a new SoH reading is required [20]. Online, adaptive training therefore seems unfeasible at this stage, even though huge efforts are being put forth in regards to streamlining the computational processes of machine learning methods [7].

In the other category of filtering- and observer based algorithms plenty of suggestions for online solutions have been suggested. These are in generally based on Kalman filtering or sliding mode observers, where Kalman based approaches appear to dominate currently [62]. In general, the struggle is to develop a combination of a battery model, which is always needed for these approaches unlike for machine learning methods, and a regression algorithm that is both sufficiently accurate and computationally efficient. This is a difficult problem in many aspects. Models and algorithms cannot be too detailed as this leads to data points having to be far apart in time which might actually reduce overall estimation accuracy. This means that the battery model and algorithm must be both simple and accurate concurrently.

Plett described methods of SoH determination using both extended Kalman filters and a dual extended Kalman filter but provided little in terms of dynamic validation [125]. The effect of filter tuning on overall performance or estimation accuracy was also not discussed. A simple empirical battery model
was used. Kim suggested a combination of a 1 RC ECM and a sliding mode observer algorithm for SoH estimation, thus potentially providing computational advantages over approaches based on Kalman filtering as heavy matrix operation are avoided in this approach [126]. Capacity fade and increase of internal resistance over an accelerated aging test are estimated with good accuracy. Remmlinger et al. used a central difference Kalman filter, incorporating the SoH directly into the state vector. Data from real operation of a hybrid vehicle was used for validation purposes. A definition of SoH based on internal resistance was used and, thus, changes to cell capacity were not considered [127]. Zou et al. developed a method based on a 1 RC ECM and two EKFs, where the two filters operate on different time scales to reduce computational strain compared to the traditional DEKF. The downside to this approach appears to be the need for batch-wise handling of data in the SoH filter. Both capacity fade and increase of internal resistance was estimated [128]. Hu et al. used a very similar approach utilizing time scale separation but with the two filters running online, however only estimating capacity and not internal resistance. Computational strain was compared favorably to the full DEKF. Filter tuning is discussed only briefly [129]. In an extensive recent review, Campestrini et al. compared a large number of different Kalman filtering methods. The focus is on SoC estimation, but the DEKF for online estimation of ECM parameters is also discussed. The issue of filter tuning is brought up as a central issue for all these methods and it is noted that filter tuning is essential in order to achieve as accurate estimation results as possible. Furthermore, it is seen that the optimal filter tuning configuration depends on not only the data used but also the algorithm itself [130].

Due to the seemingly great importance of filter tuning for the performance Kalman filtering based approaches, the need for a thorough analysis of the implications for SoH estimation of Li-ion batteries is evident.
2 Scope and aim of thesis

As renewable energy production, such as solar- and wind power, increases in importance and spread, solutions for reliable, fast and accurate battery monitoring becomes increasingly sought. The currently dominating battery technology within this field is Li-ion.

This work will examine, compare and contrast Kalman filter based approaches to Li-ion battery monitoring on the cell level with the focus on SoH estimation. The comparison will be based on the accuracy, computational performance, robustness and usability of the algorithms. SoC estimation will be discussed as well due to its near connection to SoH, especially when using Kalman filter based approaches. Two different SoC estimation algorithms will be analyzed, the extended Kalman filter (EKF) and the more involved adaptive extended Kalman filter (AEKF) which adaptively updates process- and measurement noise covariance matrices. SoH estimation, defined as the relative cell capacity compared to at the start of testing, is performed using three different Kalman filters, comparing joint- and two different kinds of dual estimation methodologies. The first will be the dual extended Kalman filter (DEKF), which utilizes two full, integrated EKFs. The two other suggested methods, the single weight dual extended Kalman filter (SWDEKF) and the enhanced state vector extended Kalman filter (ESVEKF) are attempts at reducing the complexity of the full DEKF to improve computational performance. The SWDEKF is a type of dual estimation algorithm where the weight filter state vector has been reduced in size as to only contain the cell capacity. The ESVEKF is instead a joint estimation algorithm which utilizes just a single EKF, but with the state vector extended as to contain the cell capacity. In addition, the effect of ignoring changes to cell internal resistance due to aging while estimating capacity degradation will be analyzed. Filter tuning, a commonly ignored aspect of Kalman filtering work, will be given special attention with sensitivity analyses of internal Kalman filter parameters.

To compare the Kalman filters a simple 1 RC ECM is developed and parameterized. Real battery data, obtained from the NASA Prognostics Center of Excellence, is used for all model development and validation of results [9]. The Kalman filters are compared using the same randomized current data set to simulate realistic operation conditions more accurately than the commonly found in literature accelerated aging, constant current-constant voltage (CC-CV) full charge-discharge data. All modelling and simulations will be performed using the MATLAB software.
3 METHODOLOGY

Developing an SoH monitoring algorithm based on a Kalman filter method involves several steps. First a battery model must be developed and validated. In this work a 1 RC ECM was used which means that the initial parameters of this model: OCV-SoC relationship, resistances, capacitance and cell capacity must be determined from battery diagnostic data. This involves extraction of these parameters from current and voltage measurements.

After the battery model has been parametrized, a sufficiently accurate SoC estimator must be developed. Here two different EKF-based solutions are investigated. Finally, the battery model and SoC estimator are integrated into the SoH estimator algorithm. Here three different approaches to SoH estimation are investigated, all based on Kalman filtering. First a dual extended Kalman filter (DEKF) solution, representing a somewhat standard solution to the problem of SoH estimation, is introduced [125, 131-133]. Then two potentially less computationally straining SoH estimation algorithms are suggested. One is based on a single extended Kalman filter with an extended state vector, referred to as the enhanced state vector extended Kalman filter (ESVEKF) and the other on a dual extended Kalman filter but with a shortened weight vector consisting of just a single element (cell capacity), referred to as the single weight dual extended Kalman filter (SWDEKF). All programming and plotting work was performed in the MATLAB (R2014a and R2017a) software by Mathworks. Example code can be found in the appendices of the thesis.

The following sections will outline and describe the used methodology in an approximately chronological order, starting with battery model development.

3.1 EQUIVALENT CIRCUIT BATTERY MODEL DEVELOPMENT

To establish an accurate battery model, the behavior of the real battery needs to be characterized through laboratory testing. The type and number of tests that are needed depends on the complexity of the intended model. In the case of the battery model used in this work, a 1 RC ECM, the parameters that need to be specified are the cell capacity, internal resistances $R_0$ and $R_1$, capacitance $C_1$ and the OCV-SoC relationship. Because the intention in this work is to also estimate the extent of battery aging, the development of values of parameters relevant to aging over time is also needed as reference values. To fully parameterize the 1 RC ECM, data from the following types of experiments were used:

- Low current (in this case 0.04 A) discharge tests for determination of the cell OCV-SoC relationship.
- Reference discharge tests for periodic determination of cell capacity. This type of test involves first fully charging the battery using a CC-CV method and then discharging to the cut-off voltage using a constant current.
- Pulsed current discharge tests for periodic estimation of battery transient dynamics, meaning capacitance and internal resistance parameters.

For all model development and monitoring simulations real battery data is needed. In this work the “Randomized Battery Usage Data Set” from the NASA Ames Prognostics Data Repository, which is available for download directly from the NASA Prognostics Center of Excellence website (https://ti.arc.nasa.gov/tech/dash/pcoe/prognostic-data-repository/) [9], was used for both these purposes. The purpose of this data set is to approach realistic battery operational data by randomizing the discharging and charging current in each cycle. In addition, battery diagnostic tests, reference- and pulsed discharge tests as described above in particular, are performed regularly to allow for tracking
of changes in cell behavior over time due to aging processes. Cell voltage, current and temperature was measured for all tests. Identical tests were performed for several 18650-size Li-ion batteries out of which the data from the cell RW9 was used unless otherwise stated.

### 3.1.1 Capacity determination

The cell capacity was measured by discharging a fully charged battery at 2 A constant current until the lower cut-off voltage of 3.2 V was reached. Before testing for capacity, the battery was charged using a CC-CV mode, which in this case meant that the battery was charged using a constant current of 2 A until the upper cut-off voltage of 4.2 V was reached, at which point the voltage was kept constant. Charging then continued until the current had dropped to 0.01 A. This mode of charging ensures that the battery becomes very close to fully charged. Determining cell capacity from the constant current discharge data is then merely a case of a summation of the applied current over all time-steps, $I^*\Delta t$, of the complete discharge, in other words by applying Coulomb counting. Coulomb counting could be used as the initial SoC was known to be at, or very close to, 100 % at the start of every reference discharge and as measurements were very carefully made using high-precision laboratory equipment, meaning minimal drift due to sensor bias or other measurement error.

Capacity tests were performed regularly between periods of cycling. The cycling was performed in such a way as to simulate a random usage pattern, meaning that the applied current was chosen randomly from a pre-specified set and then applied for five minutes. This procedure was repeated 1500 times between every capacity measurement. These periods of 1500 cycling steps shall be referred to as random walk (RW) phases. The exact structure of the RW phases will be explained in more detail in section 3.2 below. Reference discharge test cell voltage data recorded for battery RW9 over time is plotted below. To improve the accuracy of the Coulomb counting, a trapezoidal summation method was used to calculate the cell capacity:

$$C = \sum_k \frac{I_k + I_{k+1}}{2} \cdot \Delta t$$

The sampling time, $\Delta t$ in (13), was ten seconds for all reference discharge tests.

![Figure 20: Reference discharge voltage profiles for determination of cell capacity changes over the course of battery testing for RW9.](image-url)
Note in the figure above how upper and lower cut-off voltages did not change during testing. The first reference discharge test, performed before the start of the first RW phase, is furthest to the right in the figure above. As the cell capacity fades over time due to various aging phenomena, discharging the cell from fully a charged state at the same current and cut-off voltages will become faster. As is evident, cell capacity dropped quite severely during testing. For reference, as mentioned above, for automotive applications a Li-ion battery is usually considered to be at its EoL when its capacity has faded to 80 percent of its initial nominal value, while 65% retention of initial capacity is considered acceptable for stationary applications [134]. The measured cell capacities over the 14 first RW phases of battery RW9 are summarized in table 7 below and are also plotted in figure 21. The capacities given in table 7 are the average values of the results of the two reference discharge tests between each RW phase. For the calculation of SoH in table 7, the definition in (11) was used, with the initial capacity before the first RW phase taken as the nominal capacity. The apparent recovery of cell capacity seen in the measurement before RW phase 13 is notable. The total loss of capacity over these first 14 RW phases is more than 35 percent and is thus considered within reason for stationary applications.

Table 7: Measured cell capacities for RW9.

<table>
<thead>
<tr>
<th>Before RW phase</th>
<th>Capacity (Ah)</th>
<th>SoH (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.10</td>
<td>100</td>
</tr>
<tr>
<td>2</td>
<td>2.04</td>
<td>97.1</td>
</tr>
<tr>
<td>3</td>
<td>1.98</td>
<td>94.3</td>
</tr>
<tr>
<td>4</td>
<td>1.91</td>
<td>90.9</td>
</tr>
<tr>
<td>5</td>
<td>1.85</td>
<td>88.1</td>
</tr>
<tr>
<td>6</td>
<td>1.76</td>
<td>83.8</td>
</tr>
<tr>
<td>7</td>
<td>1.71</td>
<td>81.4</td>
</tr>
<tr>
<td>8</td>
<td>1.65</td>
<td>78.6</td>
</tr>
<tr>
<td>9</td>
<td>1.60</td>
<td>76.2</td>
</tr>
<tr>
<td>10</td>
<td>1.52</td>
<td>72.4</td>
</tr>
<tr>
<td>11</td>
<td>1.46</td>
<td>69.5</td>
</tr>
<tr>
<td>12</td>
<td>1.42</td>
<td>67.6</td>
</tr>
<tr>
<td>13</td>
<td>1.46</td>
<td>69.5</td>
</tr>
<tr>
<td>14</td>
<td>1.36</td>
<td>64.8</td>
</tr>
</tbody>
</table>
3.1.2  Open circuit voltage – state of charge expression
The data used for determination of the OCV-SoC relationship was taken from the low current (0.04 A) discharge tests. Data from one of these tests is plotted in figure 22 below. Before these tests began the battery was fully charged in a CC-CV mode, in the same way as described above.
The reason that these tests are performed at such a low current is evident when one considers the suggested 1 RC ECM, representing a reasonable, if simplified, approximation of real battery dynamics:

\[ V_{t,k} = OCV_k(SOC) - I_k \cdot R_0 - V_{1,k} \]  
\[ V_{1,k} = V_{1,k-1} \cdot e^{-\Delta t/\tau_C} + R_1 \cdot (1 - e^{-\Delta t/\tau_C}) \cdot I_{k-1} \]  

From the equations above, one can draw the conclusion that the expression for cell voltage (14) will reduce to simply the OCV in the limit of zero current after a certain time. This is because the \( e^{-\Delta t/\tau_C} \) term will always be positive and smaller than one, meaning that the transient voltage \( V_1 \) will decrease over time and eventually reach zero when no current is applied. As the full OCV-SoC relationship, from the fully charged state to the lower cut-off voltage, is desired, a low current is applied so that the OCV is changing, if slowly, while the other terms in (14) remain negligibly small. This allows the OCV to be measured.

One can then transform the OCV-time plot above to a OCV-SoC form by noticing that the applied current is of constant magnitude. As transferred charge is simply current multiplied by time one can multiply the time vector by 0.04 A, the applied current during these tests, to see how much charge has been emptied from the battery. Finally, 100 \% SoC is defined to be the charge in the battery at the start of this test, representing a fully charged battery, and 0 \% SoC is defined to be the point at which the cut-off voltage, 3.2 V, is reached. The resulting OCV-SoC plot is shown below.

\[ Figure 23: OCV-SoC for RW9 battery. \]

To utilize this data efficiently in the SoC- and SoH estimation algorithms a least-squares fitting procedure is suggested, meaning that some function on the form OCV=f(SoC) is fitted to the raw data such that the sum of the squares of residuals (error) between the fitted curve and the real data points is minimized. This is easily achieved in the MATLAB environment by using the backslash command. The only remaining task is then to determine an appropriate form of the function to be fitted to the data.
One classic approach is the so-called Nernst equation which on the following form:

$$OCV = K_0 + K_1 \cdot \ln(SOC) + K_2 \cdot \ln(1 - SOC)$$  \hspace{1cm} (16)

Plett suggested a slightly extended equation, coined the combined model [116]:

$$OCV = K_0 + K_1 \cdot \ln(SOC) + K_2 \cdot \ln(1 - SOC) - \frac{K_3}{SOC} - K_4 \cdot SOC$$  \hspace{1cm} (17)

The issue for both of these very similar equations lies in the nature of battery operation. Batteries will likely be fully charged and fully discharged several times during their lifetime. In other words, they will reach the points which should logically be equivalent to 100 % and 0 % SoC, respectively. As one approaches these points, the logarithmic terms in the equations above will go towards infinity very rapidly, thus causing the battery model equations to become very unstable and diverge. This problem is of course made worse by the inevitable existence of measurement- and modeling errors. In the worst-case scenario, the SoC might briefly jump to a negative value, causing the resulting OCV to be a complex number when applying (16) or (17). The fact that the cell capacity is continually decreasing due to aging processes makes these types of phenomena likelier over time as well. The derivatives of these functions with regards to SoC, necessary for use in extended Kalman filtering, will also display this behavior. The fully discharged state represented by a SoC equal to zero also presents an issue for the 1/SoC-term in (17). These equations can therefore be said to display a poor numerical stability and robustness to small measurement errors.

The suggested solution to these issues is to instead use a high-order polynomial for fitting the data. This is also commonly done in the literature [111, 135]. In this way one can avoid the issue of instability due briefly negative SoCs and/or complex values. The cost of this approach is slightly slower computational performance due to the larger number of coefficients compared to (16) and (17) but this is considered a small price to pay for the added stability. A 13th degree polynomial was found to fit the data quite well and was therefore chosen. The need for such a high degree polynomial is evident when considering the behavior of the fitted polynomials at very low and very high SoC/OCV, where a good fit becomes critical. The 13th degree polynomial has a maximum error of 0.0292 V which occurs at zero SoC. The fitted 13th degree polynomial as well as a fitted equation on the form of the combined model are plotted in figure 24 below. The behavior of the polynomial OCV-SoC outside of 0<SoC<1 is plotted in figure 24. Note how no spiking issues arise for the polynomial for SoCs over one or under zero while the combined model behaves strangely at SoCs equal to zero and one.
As can be seen, the fitted equation on the form of the combined model provides a much worse fit. The comparison is slightly unfair considering the much lower number of fitted coefficients in the combined model.

It is worth noting that the raw data can also be used directly for SoC estimation by utilizing a look-up table with linear interpolation between measured data points. This approach was however found to be much slower than using a fitted expression with little noticeable improvement in terms of accuracy compared to the polynomial approach and was therefore discarded. Using a fitted function also means...
that numerical derivation can be avoided when using the model in Kalman filtering, thus improving performance [136].

3.1.3 Identification of equivalent circuit model parameters

The next step of model development involved identification of the values the ECM parameters $R_0$, $R_1$ and $C_1$. Data from pulsed discharge tests were used for this purpose. For these tests discharged pulses of 1 A were applied for ten minutes, between each of which the battery was rested for 20 minutes, during which time no current was applied. At the start of the tests the battery was fully charged in a CC-CV mode and the procedure was continued until the lower cut-off voltage was reached. The resulting voltage profiles from these tests are plotted below.

![Figure 26: All pulsed discharge tests for determination of ECM parameter values for RW9.](image)

The increase in internal resistance over time can be seen by noticing how the voltage curves in figure 26 are dropping further and further down during the periods of resting. This instantaneous vertical drop upon the applied current being cut is directly proportional to the internal resistance of the cell, as seen in figure 27 below.

As all tests were performed at room temperature it was not possible to identify the temperature dependence of the ECM parameters and therefore the parameter values were assumed to be independent of the temperature. It is however worth noting, as mentioned previously, that resistance generally decreases at higher temperatures while the capacity generally increases [113]. This is especially so when one considers that the cell temperature generally was lower during the diagnostic steps for capacity and internal resistance estimation than during actual cycling. This was likely due to the extensive resting of the battery during the diagnostic tests, meaning that Joule heating (proportional to $I^2R$, see (3)) could largely be dissipated. Notice for instance in figure 31 below how the temperature rises quite quickly in the beginning of the RW phase and then remains above room temperature for the duration of RW cycling. This likely also affects the results for later RW phases as the average temperature during cycling increased over time due to the increased internal resistance of the cell caused by aging effects.
One of the great advantages of the 1 RC ECM is that model resistances and capacitance can be roughly estimated directly from pulsed discharge data using simple rules of thumb. This is shown in figure 27 below.

**Figure 27: Ballpark estimation of ECM parameters from pulsed discharge.**

Furthermore, by noting these rules of thumb it is possible to get a basic understanding of how changing the values of the ECM parameter values affects the behavior of model. While the method of visual inspection shown in figure 27 certainly is convenient, it naturally lacks accuracy. This is especially so for identification of the capacitance value as it may prove difficult to determine the exact time at which $\Delta v_\infty$ has been reached.

It is clear that this is again a question of curve-fitting, just as for determination of the OCV-SoC relationship. The difference is that here it is not possible to simply use a least squares methodology directly as the 1 RC ECM is not linear in its parameters. Consider the recursive nature of the transient voltage $V_1$ for instance. The solution here was to write a MATLAB function incorporating the full 1 RC ECM and the previously determined OCV-SoC relationship with the resistances and capacitance of the model as input parameters. The output was then the squared difference between the measured cell voltage and the cell voltage estimated by the model. The minimum of this function was then found using the built-in MATLAB function “fminsearch” which uses the Nelder-Mead simplex optimization algorithm [137]. The resulting fit for the initial pulsed discharge test of RW9 is plotted below.
As pulsed discharge tests were performed between every other RW phase (every 3000 RW steps), ECM parameter values could be updated and used as reference for SoH estimation at a later stage. As all elements of the battery models have now been identified, the result seen in figure 28 above is representative of the overall model accuracy. The model is evidently not an exact description of the battery behavior. The modelling error is comparable with reported values for common Li-ion battery models in the literature [138]. Several authors have extended their ECM by allowing for its parameters to be SoC-dependent as well [135]. This was not attempted in this work but would likely have led to a closer model fit. One issue with that approach for this work is that changes to the ECM parameter values will later be used for SoH estimation. If the ECM parameters themselves are also SoC-dependent it becomes very difficult to know how to rationally update these appropriately. Also, as will be seen later, the 1 RC ECM is certainly able to achieve accurate results when used with a Kalman filter approach despite small modelling errors. In fact, modelling errors are taken into consideration by the Kalman filter and are thus expected to exist to some extent. The errors in modelling, here defined as estimated-subtracted by measured cell voltage, of the initial pulsed discharge test of RW9 can be seen in the figure below.
The modelling error is seen to mostly lie between -30 mV and 20 mV for this case with a root-mean-square error of 11 mV. It is also notable that the largest absolute modelling errors occur at lower SoCs towards the left side of figures 29 and 28. Repeating said method of ECM parameter value identification for data sets taken from later pulsed discharge tests yielded the following development of the ECM parameters over time for battery RW9:

Table 8: Development of ECM parameters with aging for battery RW9.

<table>
<thead>
<tr>
<th>Before RW phase</th>
<th>$R_0$ [Ω]</th>
<th>$R_1$ [Ω]</th>
<th>$C_1$ [F]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.074</td>
<td>0.045</td>
<td>815.1</td>
</tr>
<tr>
<td>3</td>
<td>0.080</td>
<td>0.066</td>
<td>845.4</td>
</tr>
<tr>
<td>5</td>
<td>0.096</td>
<td>0.076</td>
<td>978.8</td>
</tr>
<tr>
<td>7</td>
<td>0.113</td>
<td>0.085</td>
<td>1205.4</td>
</tr>
<tr>
<td>9</td>
<td>0.133</td>
<td>0.089</td>
<td>1828.4</td>
</tr>
<tr>
<td>11</td>
<td>0.149</td>
<td>0.098</td>
<td>1948.6</td>
</tr>
<tr>
<td>13</td>
<td>0.169</td>
<td>0.110</td>
<td>2348.7</td>
</tr>
</tbody>
</table>

Both internal resistances parameters $R_0$ and $R_1$ evidently increase by a significant amount during the course of testing. The capacitance $C_1$ as well. Despite this changes to the capacitance was not monitored by any Kalman filtering algorithm investigated in this work. This appears to be a common choice in the literature as well [122, 128]. The reasons for this are two-fold. Firstly, as will be seen below, changing the capacitance does not reduce modelling error enough to warrant the extra computational strain required for running the extended monitoring algorithms, and, secondly, no commonly used definition of battery SoH incorporates the capacitance. Not adapting the capacitance does introduce some extra modelling error for later RW phases. This is shown in figure 30 below where the added modeling error for the pulsed discharge test before RW phase 13 is shown. Not adjusting for capacitance meant using the first measured capacitance value, 815.1 F, for all tests instead of
adjusting to the closest capacitance value as estimated via fminsearch, which for the case of RW phase 13 was 2348.7 F for battery RW9 per table 8.

![Figure 30: Difference in modelling error when not adjusting for changing capacitance.](image)

The maximum added error by not adjusting the capacitance is 40 mV in figure 30. As can be seen, the maximum overall error is however the same, approximately 80 mV at around 3.5 hours in to the test. The overall increase of modelling error for later RW phases might be due to several reasons. The initial OCV-SoC relation might have been altered, there might be additional temperature and/or C-rate effects, or phenomena not encapsulated by the simple 1 RC ECM might play a role.

The developed model may be used for estimation of battery state, meaning SoC and transient voltage \( V_1 \), and ECM parameter values \( R_0, R_1, C_1, C \), thus allowing for estimation of SoH.

### 3.2 STATE OF HEALTH ESTIMATION VALIDATION AND CONSIDERATIONS

As mentioned previously, no online reference values of capacity and resistance can be measured during operation of a Li-ion battery. For the data used in this study, tests for determination of internal resistance and cell capacity were performed periodically, between periods of operation, the so-called RW phases. In other words, the exact development or path of the cell capacity and internal resistance over time is not available. Fortunately, these parameter values change only quite slowly, especially in comparison to state variables like SoC and transient voltage, meaning that these periodic measurements are reasonable to use as practical references. For methods that attempt to estimate the cell capacity and/or resistance online, there are two different ways of validating their performance using this type of data:

1. Let the Kalman filter algorithm run through an entire RW phase with correct, as measured, starting values of capacity and internal resistances. Compare estimated values of parameters at end of RW phase simulation with measured values at that point.
2. Set starting guesses of capacity and/or internal resistance substantially off from measured values intentionally and let the filter estimation converge to the measured values at the start
of RW phase. Compare to estimated values of parameters to measured values before RW phase.

The first of these approaches has the advantage of showing the full estimated development of capacity and/or internal resistance over the course of battery operation. This would be representative of a long-term battery monitoring situation. The computation time needed for such a simulation might however be quite extensive, depending on what hardware and software is used, the length of operation between reference measurements, and the data sampling time. Many measurements taken over a long period of time need to be considered for validation of SoH-estimation methods in order for changes to the ECM parameters to be clearly distinguishable. The second method solves this issue of time consumption as data from only the first part of a given RW phase is needed for the simulation. This is also representative of the common case of ECM parameters not being known exactly and needing to be estimated online. In addition, the existing SoC reference, Coulomb counting, uses the capacity measured before a given RW phase in all calculations, meaning that the quality of this reference is worsened towards the end of a given RW phase when cell capacity will have decreased due to aging. Both validation methods will be used in this study, but the second more so due to the quicker results. The exact nature of the RW operation during cycling of the batteries will now be outlined.

Between measurements of cell capacity and transient behavior, referred to as reference- and pulsed discharge tests respectively, the battery is operated in a so-called random walk (RW) cycling mode. The structure of the data set is as follows: first reference-, pulsed-, and low current discharge tests are performed to determine initial battery capacity, OCV-SoC relation and transient dynamics. Then 1500 RW steps are performed per the following template:

1. Fully charge the battery in a CC-CV mode.
2. Pick a discharging/charging current randomly from the set [-4.5 A, -3.75 A, -3 A, -2.25 A, -1.5 A, -0.75 A, 0.75 A, 1.5 A, 2.25 A, 3 A, 3.75 A, 4.5 A]. Here, as throughout this work, negative currents indicate charging and positive currents discharging.
3. The randomly picked current is applied for 5 minutes or until the battery voltage is on the edge of the battery SOA, meaning 3.2 V or 4.2 V. This will be referred to as one RW step from here on out, regardless of duration.
4. After each RW step there is a short rest for under 1 second during which no current is applied to the battery.
   a. If 1500 RW steps, referred to as one full RW phase for here on out, have been performed, end RW cycling and proceed to diagnostic testing.
   b. Otherwise go to step 2 again.

This type of cycling data will be used for all simulations of SoC- and SoH monitoring. Typical measurements of cell voltage, current and temperature for the first 100 steps of a RW phase are displayed in figure 31. This particular example comes from the first RW phase of battery RW9.
The average duration of a RW step was quite a bit less than five minutes, as one full RW phase took around 95 hours on average. The experiments were performed at room temperature. The elevated temperature during RW cycling, likely due to Joule heating, is evident. This mode of operation is very intense as batteries are virtually constantly being either charged or discharged. This means that aging effects will be accelerated in these tests compared to most normal Li-ion battery applications.

It was discovered that negative SoC would quickly result if Coulomb counting was applied to data from later RW phases of RW9, even if the cell capacity used in the calculations was adjusted to the latest value as measured in the reference discharge tests. See for instance the SoC calculated by Coulomb counting for RW phase 10 of battery RW9 in the figure below. The calculated SoC is as low as -10 % at just after eight hours into the RW phase.
The reason for this has not been confirmed but might have to do with the fact that all reference discharge tests for capacity determination were performed at 2 A constant current and at generally lower temperatures than during RW cycling. It is widely known that higher temperatures and lower currents correlate with a higher cell capacity, meaning that the real capacity might be higher during RW phases than during reference discharge tests [82].

Having outlined the procedure of model development and special considerations regarding SoH estimation, the utilized Kalman filtering algorithms used for SoC and SoH estimation will now be explained in detail.

### 3.3 Kalman Filtering

Kalman filtering, as briefly outlined in the introduction of this work, is a family of recursive algorithms for online or batch estimation of internal, non-measurable, variables of a system with system inputs- and outputs having a certain noise or uncertainty associated with them. The goal is to minimize the mean-square estimation error of these internal variables for a stochastic system. For this reason, the Kalman filter may be seen as the stochastic, recursive equivalent of the deterministic, batch least squares method [139]. The Kalman filter has many applications such as in process control, navigation, and tracking [140]. One major advantage of the Kalman filter methodology is that the error bounds of the internal variable estimates are always calculated concurrently. One frequently mentioned downside is the large number of matrix operations needed, leading to quite low computational performance, especially as the number of estimated state variables increases [116, 141]. In fact, the application of Kalman filtering methods has largely been driven by advances in available computing power [142]. Overcoming, or at least handling of, this issue within the framework of Li-ion battery monitoring will be central in this work.

The basic principle of Kalman filtering is to construct a model of the real system which uses the internal variables which one wishes to estimate as inputs, while the outputs are some measurable variables. The output (may be a vector of values) of the model for a given system state and input is compared to the measurement. The internal variables are then updated according to the difference between the estimate and the measured value: if the difference is large than the update is large and vice versa. At the same time, measurement- and process noise is incorporated directly into the algorithm. If the measurement noise is large, updates will be smaller relative to the error and vice versa. If the measurement noise is large, the measurements are considered less important, or “less trusted”, than the model prediction. The process noise represents unmodeled inputs to the system, such as random perturbations, and modeling errors. The calculated Kalman gain matrix controls the balance of weighing between the measured values and those predicted by the model. In the following introduction to the underlying principles of Kalman filtering all equations will refer to the original algorithm developed by Rudolf Kalman during the 1950s, frequently referred to as the linear Kalman filter (LKF), for ease of explanation and because of the similar fundamental algorithm structures of all Kalman filtering approaches [143]. More advanced Kalman varieties will be introduced at a later stage. The basic Kalman filter principle may be represented graphically as so:
In figure 33 \( u_k \) represents some deterministic system input. The real, non-measurable system state is represented by the internal variables \( x_k \), which is to be estimated by the algorithm. The deterministic system input is also sent to the system model, which results in the estimated internal variables \( \hat{x}_k \) and the estimated system output \( \hat{y}_k \). The estimated output is then compared to the measured output and the difference between the measured and estimated output, often called the error, is used to update the estimate of the system state, weighted by the calculated Kalman gain matrix \( L_k \), which also considers current state uncertainty and measurement noise. Details of the algorithm will now be explained step-by-step.

The core of the Kalman filter methodology consists of two equations, referred to here as the state equation (18) and the output equation (19). For the LKF these can be written as:

\[
\begin{align*}
    x_{k+1} &= A_k \cdot x_k + B_k \cdot u_k + w_k \\
    y_k &= C_k \cdot x_k + D_k \cdot u_k + v_k
\end{align*}
\]

(18)  
(19)

Here \( k \) is the time index, \( x \) the state vector, \( u \) the deterministic system input, and \( y \) the measurable output. As an example, \( x \) will be a vector of two elements in this work, consisting of the state variables SoC and transient voltage. The measurement, \( y \), will be the cell voltage. The current will represent the deterministic system input \( u \). The matrices \( A, B, C \) and \( D \) describe the dynamics of the system and contain the model coefficients and structure. These matrices may also be time dependent. Vectors \( w \) and \( v \) represent process- and measurement noise of the system respectively. The equations (18) and (19) may together be referred to as a state-space representation of the system. It is notable that \( w \) and \( v \) must be specified a priori as part of filter initialization, often a difficult task. Furthermore, \( w \) and \( v \) are assumed to follow a certain set of assumptions: they are zero-mean and Gaussian (normally distributed) and also mutually uncorrelated [135, 142]. While these assumptions are likely rarely completely fulfilled, the results of the Kalman filter have spoken for themselves [116]. In any case, it is very useful to be able to model both measurement- and process noise during filter tuning, an activity which will be discussed in depth at a later stage in this study. The process- and measurement noises have covariance matrices \( \Sigma_w \) and \( \Sigma_v \) respectively, which are used directly in the filter algorithm.

The Kalman filter algorithm can essentially be divided into three stages: initialization, the time update and the measurement update. First the filter must be initialized before monitoring can commence. The initialization stage requires the user to specify initial guesses of the system state, \( \hat{x}_{0} \), along with the associated initial state covariance matrix, \( \Sigma_{x,0} \), and the previously mentioned process- and measurement noise covariance matrices \( \Sigma_w \) and \( \Sigma_v \). Note that from here on out estimated values will be specified with the hat symbol while measured values will not have the hat to differentiate these
from each other, just like in figure 33 above. The initial state estimate and state covariance matrix are quite straightforward to specify, \( \Sigma_{x,0} \) usually being set as the following diagonal matrix\(^2\) [144]:

\[
\Sigma_{x,0} = \text{diag}(\mathbb{E}[(x_0 - \hat{x}_0^*)(x_0 - \hat{x}_0^*)^T])
\]  

(20)

Here \( \mathbb{E} \) is the statistical expectation value operator and \( T \) represents the vector transpose. The initial guess of the system state is based on the best available information and experience. Of course, the true value of \( x_0 \) is still generally not known. This is however not a major issue for most cases for two reasons. Firstly, one will usually have some rough idea of where \( x_0 \) might lie. This is certainly the case for battery modelling as shall be seen later. It is generally best to assume that the initial guess is as poor as possible, meaning that the guess is at one end of the range of thinkable values while the real value happens to be at the other end. Secondly, the initial state covariance matrix will only be used for the first iteration. As new information in the form of measurements come in, the state covariance is updated recursively to reflect the model uncertainty. The state covariance matrix will not necessarily remain diagonal during filtering but will always be positive definite and symmetric [143, 144]. As the model converges towards true estimates of the system state, the diagonal elements of the state covariance matrix can be seen as a measurement of model or state estimate accuracy, meaning that error bounds are automatically updated during filtering. As a rule of thumb, the true value of \( x \) will be bounded by the following expression around 95% of the time if the underlying assumptions of the Kalman filter are fulfilled [116]:

\[
x = \hat{x}_k \pm 2 \cdot \sqrt{\text{diag}(\Sigma_{x,k})}
\]  

(21)

Unlike the state covariance matrix, the process- and measurement noise covariance matrices will remain constant during filtering. These two parameters represent the inherent uncertainty, or inaccuracy, of the model (values of \( \hat{x} \)) and measurements (values of \( y \)) respectively. As these are not updated recursively in the standard Kalman filtering methods, finding suitable values for these becomes crucial to achieve acceptable filter performance. The measurement noise matrix values can be estimated readily if earlier measurement data is available. For the case of a number of measurement signals \( i \) with time-independent standard deviations \( \sigma_i \), the measurement noise covariance matrix can be estimated as a diagonal matrix:

\[
\Sigma_v = k \cdot \text{diag}(\sigma_i^2)
\]  

(22)

Where \( k \) is most often one or slightly larger than one and can be seen as a tuning parameter, say if one expects larger measurement error in the future \( k \) can be made larger. If the measurement error matrix is too large, the information gathered from measurements will be ignored and the filter will just use the model predictions which can cause the filter to diverge. Setting the measurement error covariance matrix too small will cause the filter to fail to separate out sensor noise from true signal, causing reduced state estimation accuracy [144].

Most often it is the specification of the process noise covariance matrix values which represents the biggest challenge in filter tuning. Several methods have been suggested to try to rationalize the process. Because of the relative difficulty, a trial-and-error procedure is often necessary [144]. The process noise covariance matrix is in any case diagonal and time-independent in standard Kalman filter approaches. In general, finding the appropriate values of \( \Sigma_v \) and \( \Sigma_w \) will be done in one of three possible ways [145]:

\(^2\) A matrix is called diagonal if all values not on the diagonal are zero.
1. Fixing $\Sigma_v$, given that a reasonable estimate of it exists, and vary $\Sigma_w$ by trial and error until realistic, stable results are yielded.

2. The reverse situation, meaning that $\Sigma_w$ is known and $\Sigma_v$ is varied in a trial-and-error fashion.

3. Neither $\Sigma_v$ nor $\Sigma_w$ are known and both must be found by trial-and-error by varying both.

In this work the relative size of the values in the process noise covariance matrix was found to be an important factor in achieving accurate state estimation. This will be discussed further in the results section.

Finally, the LKF algorithm proper can be introduced. Note that the algorithm equations will not derived here. If the reader is interested she may see the original article by Robert Kalman for such a derivation [143]. After the initialization step, the time update can be commenced. Parameters will be notated with a minus (“−”) sign index before the measurement update and with a plus (“+”) index afterwards to differentiate these. The time update step essentially propagates the state and state covariance to the next time step through the model equations. It may thus be considered the model prediction step.

### State estimate time update:

$$\hat{x}_k = A_{k-1} \cdot \hat{x}_{k-1} + B_{k-1} \cdot u_{k-1}$$

### State covariance matrix time update:

$$\Sigma_{x,k} = A_{k-1} \cdot \Sigma_{x,k-1}^+ \cdot A_{k-1}^T + \Sigma_w$$

### Calculation of Kalman gain matrix:

$$L_k = \frac{\Sigma_{x,k}^- \cdot C_k^T}{C_k \cdot \Sigma_{x,k}^- \cdot C_k^T + \Sigma_v}$$

Note that the state estimate time update equation assumes zero process noise. In other words, it is deterministic. If the system is stable, the state covariance matrix should decrease over time. The process noise covariance matrix will always lead to larger values in the state covariance matrix, thus increasing state uncertainty. In (25) the Kalman gain matrix is strongly dependent on both the state covariance- and measurement noise covariance matrices. If the state covariance matrix is large, then the Kalman gain matrix will be large and the state estimate measurement update will be more drastic as seen below. At the same time, a large measurement error covariance matrix will lead to a small Kalman gain matrix and thus the opposite effect. In fact, the following two limits are noted [142]:

$$\lim_{\Sigma_v \to 0} L_k = \frac{1}{C_k}$$

$$\lim_{\Sigma_{x,k} \to 0} L_k = 0$$

In other words, as the measurement error approaches zero, the measurements are more “trusted”. Likewise, when the state covariance matrix, representing the state estimate uncertainty, approaches zero, model predictions are more “trusted”. The Kalman gain matrix may therefore be seen as something of a signal-to-noise metric or, as mentioned earlier, the relative trust put in measurements versus model predictions [116].
In the measurement update stage, the new measurement, \( y_k \), is utilized to update the values of the state estimate and state covariance matrix. The measurement update step may be considered a correction step where the predicted system state is adjusted by utilizing the new measurement data.

Output estimate error:
\[
e_k = y_k - \hat{y}_k = y_k - (\hat{C}_k \cdot \hat{x}_k^- + D_k \cdot u_k)
\]
(28)

State estimate measurement update:
\[
\hat{x}_k^+ = \hat{x}_k^- + L_k \cdot e_k
\]
(29)

State covariance matrix measurement update:
\[
\Sigma_{x,k}^+ = (I - L_k \cdot C_k) \cdot \Sigma_{x,k}^-
\]
(30)

If the output estimate error is large it is more likely that estimated internal variables have been inaccurately estimated and adjustments to these should therefore be large. This is reflected in (29). Note in (30) how the state covariance matrix always gets smaller due to new measurement information. Here \( I \) in (30) represents the identity matrix.

The time- and measurement update steps are then repeated in every time step for the duration of the monitoring process. The six steps in the time update and measurement update steps are essentially the same for all Kalman filter approaches with only minor modifications for some more advanced varieties. Having introduced the Kalman filter basics through the example of the LKF, a series of modified, more advanced methods are now presented. The purpose of these more advanced Kalman filters is either to improve the estimation accuracy for non-linear systems or, in the case of the adaptive extended Kalman filter, a streamlining of the filter tuning process. The last three Kalman filtering methods to be introduced are SoH estimators, meaning that also the cell capacity and, in the case of the DEKF, the internal resistance parameters \( R_0 \) and \( R_1 \) are estimated concurrently with the SoC.

### 3.3.1 Extended Kalman filter

If one studies the state- and output equations ((18) and (19)) one will notice that these describe a linear system, thus the name LKF. Because of this structure, linear systems are a requirement for using the LKF. Batteries are however strongly non-linear systems and their modelling should therefore preferably reflect this. Indeed, if the model suggested in this work, the 1 RC ECM, is examined it is found that a direct separation of the model equations into matrices A, B, C and D such as in (18) and (19) is not possible. The simple and widely accepted solution to this problem is to linearize the model equations via a truncated Taylor series expansion. After this series expansion has been performed, the same algorithm as in the LKF may be used. This is called extended Kalman filtering (EKF). The state- and output equations are now assumed to be non-linear:

\[
x_{k+1} = f(x_k, u_k) + w_k
\]
(31)

\[
y_k = g(x_k, u_k) + v_k
\]
(32)

Taylor series expansion allows for the linearization of these functions. All terms of the series but the first two are truncated:

\[
f(x_k, u_k) \approx f(\hat{x}_k, u_k) + \frac{\delta f(x_k, u_k)}{\delta x_k} (\hat{x}_k) \cdot (x_k - \hat{x}_k)
\]
(33)

\[
g(x_k, u_k) \approx g(\hat{x}_k, u_k) + \frac{\delta g(x_k, u_k)}{\delta x_k} (\hat{x}_k) \cdot (x_k - \hat{x}_k)
\]
(34)

Depending on how non-linear the system is this might be a more or less drastic approximation [140]. As will be shown, using the first two terms of the Taylor expansion will prove to be sufficient for the
purposes of battery modelling here, though some authors have suggested more sophisticated solutions to the problem of model non-linearity such as the unscented Kalman filter or the particle filter [146, 147]. Of course, the EKF requires the model equations f and g to be differentiable at all points. By simple comparison of the LKF and EKF state- and output equations it is evident that what must be determined for the LKF algorithm to be applicable for this non-linear system are equivalents of the matrices A, B, C and D. From an analytical standpoint, the A and C matrices describe how the current system state affects future system states and outputs, respectively. By comparison of (34) and (33) with (19) and (18), it is evident that the equivalents of these equations must be the Jacobian matrices. Therefore, these will be renamed accordingly:

\[
\hat{A}_{k+1} = \frac{\delta f(x_k, u_k)}{\delta x} \tag{35}
\]

\[
\hat{C}_k = \frac{\delta g(x_k, u_k)}{\delta x} \tag{36}
\]

If (35) and (36), along with (33) and (34), are inserted back into (31) and (32) the following equations are the result:

\[
x_{k+1} = \hat{A}_k \cdot x_k + f(\hat{x}_k, u_k) - \hat{A}_k \cdot \tilde{x}_k + w_k \tag{37}
\]

\[
y_k = \hat{C}_k \cdot x_k + g(\hat{x}_k, u_k) - \hat{C}_k \cdot \tilde{x}_k + v_k \tag{38}
\]

The choices of A- and C matrix equivalents are therefore validated as they serve the same functions in (37) and (38) as A and C did in (18) and (19), respectively. The parts of (37) and (38) which are not dependent on x then functionally replace the terms which are multiplied by the matrices B and D in (18) and (19). Now the exact same procedure as in the LKF, meaning the six time- and measurement update equations, may be used by incorporating these modified A and C matrices. This allows for Kalman filtering to be utilized even though the model equations are non-linear.

It is now easy to write the 1 RC ECM model equations in the matrix form appropriate for the EKF. The state equation shall be the following, utilizing the notation in figure 18:

\[
x_{k+1} = \begin{bmatrix} V_{1,k+1} \\ SOC_{k+1} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\Delta t \cdot \frac{e^{R_1 \cdot I_k}}{C} & 0 \end{bmatrix} \begin{bmatrix} V_{1,k} \\ SOC_k \end{bmatrix} + \begin{bmatrix} -\Delta t \cdot \frac{e^{R_1 \cdot I_k}}{C} \\ \Delta t \cdot \frac{e^{R_1 \cdot I_k}}{C} \end{bmatrix} \cdot I_k \tag{39}
\]

\[
y_k = V_t = OCV(SOC_k) - I_k \cdot R_0 - V_{1,k} \tag{40}
\]

This means that the state vector will consist of two elements in this case: the SoC and \( V_{1,t} \), the transient voltage over the lone RC pair of the ECM, while the output vector will consist of a single element: the cell terminal voltage. As the output vector is only of length one, the measurement noise covariance matrix will also consist of a single element. It will therefore be referred to as the measurement noise covariance from here on out. Similarly, the Kalman gain matrix will simply be a vector of length two. The deterministic input is the charging or discharging current, also a single element vector. Finally, by using (36) with the 1 RC ECM output equation in (40), the C matrix of the EKF may be written as:

\[
\hat{C}_k = \begin{bmatrix} -1 & \frac{\delta(OCV)}{\delta(SOC)} \end{bmatrix} \tag{41}
\]
Of course, as the chosen expression for the OCV-SoC relation is a simple polynomial, as described in section 3.1.2, the calculation of the partial derivative of the OCV with regards to SoC in (41) is straightforward and is easily done analytically.

### 3.3.2 Adaptive extended Kalman filter

One of the major hurdles when developing a Kalman filtering method is the process of filter tuning, particularly the identification of suitable process- and measurement noise covariance matrices. The AEKF attempts to overcome this problem by recursively updating these matrices as new measurements are made available during filtering. Being a variant of the EKF, the structure of the AEKF is very similar overall. The differing equations come in the measurement update stage of the algorithm. First the squares of the output estimation errors are summarized over a certain number of the latest time-steps, called the moving estimation window size, M. Then updated process- and measurement noise covariance matrices are updated accordingly [130, 145, 148]:

\[
H_k = \frac{1}{M} \sum_{i=k-M+1}^{k} e_i \cdot e_i^T
\]

\[
\Sigma_w = H_k + \hat{C}_k \cdot \Sigma_{x,k} \cdot \hat{C}_k^T
\]

\[
\Sigma_w = L_k \cdot H_k \cdot L_k^T
\]

Note that while one now does not need to specify process- and measurement noise covariance matrices explicitly, starting guesses for these still need to be specified. This actually means that more variables need to be pre-specified during initialization of the AEKF than of the EKF as starting guesses for process- and measurement noise covariance matrices are still needed along with the additional moving estimation window size. The adaptive updating of the process- and measurement noise covariance matrices should however make the filter more robust with regards to initialization and thus reduce the risk of the filter diverging. Before M time-steps have been performed the starting guesses for process- and measurement noise covariance matrices are used instead of (43) and (44).

### 3.3.3 Dual extended Kalman filter

The EKF and AEKF algorithms as presented thus far are SoC estimators. During filtering, all model parameters will be assumed to remain constant when using these filters. This is an approximation as the capacity, capacitance and internal resistance of the battery will change depending on operating conditions and most importantly aging. While the EKF and AEKF can handle certain model inaccuracy, the quality of the state estimation will degrade relatively rapidly if changes to parameter values are ignored. In addition, tracking of the changes to capacity and internal resistance is of interest in and of itself as these parameters are essential to monitoring of battery performance and safety. To achieve this the DEKF is introduced. The idea here is to run two EKFs in parallel, with one normal filter, the state filter, estimating the system state variables and one so-called weight filter that is estimating the model parameters concurrently. The filters are run in tandem in an intelligent way such that certain useful interconnections are utilized for increased computational performance [149].

The weight filter will have its own process- and measurement noise covariance matrices, starting guesses of the parameter values (weights) and weight covariance matrix, separate to those of the state filter. The output equation will however be the same for both filters. The weight time update equation will be very simple. The weight filter state equation is modelled to be slowly changing by applying a certain process noise, r:
\[ \theta_{k+1} = \theta_k + r_k \] (45)

The reason for this simple model is the difference in rate of change between the state variables, such as SoC and transient voltage, and the elements of the weight filter, such as cell capacity and internal resistance parameters. With regards to the state variables the weights are practically constant. Moreover, no accepted, universal equations exist for the changes in capacity or internal resistance over time.

As in the name, both filters in the DEKF are of the extended kind, meaning that Jacobian matrices are required for the linearization procedure. The Jacobians \( A \) and \( C \) for the state filter will remain the same as those in the EKF as the state- and output equations have not changed. The next step is to determine what these matrices will be for the weight filter. To differentiate the filters and their internal parameters, the subscripts \( x \) and \( w \) will be used for the state- and weight filter respectively.

First it is evident that the weight filter state equation, unlike the case for the state filter, is in fact linear in the parameters, with \( A_w \) being simply the identity matrix. \( B_w \) is naturally zero as the parameters are independent of the applied current. Since the output equation of the weight filter is the same as for the state filter, \( C_w \) will be a Jacobian matrix which needs to be calculated from the Taylor series expansion, just as for \( C \), above with the difference that the derivative is with regards to \( \theta \) instead of \( x \). What one will find when one starts to try to calculate \( C_w \) is that a recursive procedure is necessary. Like the \( C \), matrix is the partial derivative of the output function with regards to \( x \), \( C_w \) will be the total derivative of the output equation with regards to \( \theta \):

\[
\hat{C}_w = \frac{dg(x^+_k, u_k, \theta)}{d\theta}(\theta_k) \tag{46}
\]

Now note that the output function \( g \) is a function of \( \hat{x}_k^+ \) which is indeed a function of \( \theta \), but also that \( \hat{x}_k^- \) is a function of \( \hat{x}_k^{+1} \) via the time update equation as well, which in turn is also a dependent on \( \theta \). This is why the total derivative and not the partial derivative with regards to \( \theta \) is needed for the calculation of \( C_w \) \[149\]. The recursive nature of these calculations is evident, meaning that the last calculated value of \( C_w \) is needed for the calculation of the next one. \( C_w \) will simply be initialized as zero, using the following set of recursive equations, derived from the total derivation of the output equation, for calculating following values:

\[
\frac{dg(x_k^-, u_k, \theta)}{d\theta} = \frac{\delta g(x_k^-, u_k, \theta)}{\delta \theta} + \frac{\delta g(x_k^-, u_k, \theta)}{\delta x_k^-} \cdot \frac{dx_k^-}{d\theta} \tag{47}
\]

\[
\frac{dx_k^-}{d\theta} = \frac{\delta f(x_{k-1}^+, u_{k-1}, \theta)}{\delta \theta} + \frac{\delta f(x_{k-1}^+, u_{k-1}, \theta)}{\delta x_{k-1}^+} \cdot \frac{dx_{k-1}^+}{d\theta} \tag{48}
\]

\[
\frac{dx_{k-1}^+}{d\theta} = \frac{dx_{k-1}^-}{d\theta} - L_{k-1} \cdot \frac{dg(x_{k-1}^-, u_{k-1}, \theta)}{d\theta} \tag{49}
\]

Note how (48) originates from the state filter time update equation, and (49) from the state filter measurement update equation (29).

This system of recursive equations hold under the assumption that the Kalman gain, \( L \), us not a function of the weight vector \( \theta \). While this is not strictly true, it is usually not worth the extra computational

---

3 Note that the reverse is of course not true: \( \theta \) is not a function of \( x \) when using the set of equation in this work. The partial derivative is therefore sufficient for the calculation of \( C_w \).
effort to calculate the extra derivatives needed in every iteration considering how small the difference in the results is [149]. While the set of equations above may appear intimidating, a lot of results in one iteration may be reused in the next thus greatly simplifying the overall procedure.

In fact, one will find that only a relatively small number of new derivatives must be calculated in each time-step. Firstly, note that the already calculated \( C_w \) and \( A_\delta \) matrices from the state filter may be used directly in (47) and (48) like so by noting the expressions for these in (35) and (36) above:

\[
\frac{dg(x_k^-, u_k, \theta)}{d\theta} = \frac{\delta g(x_k^-, u_k, \theta)}{\delta \theta} + \hat{C}_k \cdot \frac{dx_k^-}{d\theta} \tag{50}
\]

\[
\frac{dx_k^-}{d\theta} = \frac{\delta f(x_{k-1}^+, u_{k-1}, \theta)}{\delta \theta} + \hat{A}_k \cdot \frac{dx_{k-1}^+}{d\theta} \tag{51}
\]

The Kalman gain matrix in the previous step, \( L_{k-1}^+ \), is of course also already calculated and just needs to be stored for the next iteration of the algorithm. This re-use of already calculated results also applies for the full derivatives \( C_w \) and \( \frac{dx}{d\theta} \) as well. The result is that only the Jacobians \( \frac{df}{d\theta} \) and \( \frac{dg}{d\theta} \) need to be calculated in each iteration. In the case of Li-ion battery SoH monitoring with a 1 RC ECM, the following weight vector will be used in the DEKF:

\[
\theta_k = [R_0 \quad R_1 \quad C] \tag{52}
\]

Note again how the capacitance is assumed to remain constant during cell operation and is therefore not included into the weight vector. The needed Jacobians may then be written thusly, noting that only the \( V_1 \) term in the output function depends on \( R_1 \) and likewise that only the OCV-term depends on the cell capacity:

\[
\frac{\delta g(x_k^-, u_k, \theta)}{\delta \theta} = \begin{bmatrix}
\frac{\delta g(x_k^-, u_k, \theta)}{\delta R_0} & \frac{\delta g(x_k^-, u_k, \theta)}{\delta R_1} & \frac{\delta g(x_k^-, u_k, \theta)}{\delta C}
\end{bmatrix}
= [-I_k \quad 0 \quad \delta OCV(x_k^-, u_k, \theta)] \tag{53}
\]

\[
\frac{\delta f(x_{k-1}^+, u_{k-1}, \theta)}{\delta \theta} = \begin{bmatrix}
0 & \frac{\delta V_1(x_{k-1}^+, u_{k-1}, \theta)}{\delta R_1} & 0 \\
0 & 0 & \frac{\delta SOC(x_{k-1}^+, u_{k-1}, \theta)}{\delta C}
\end{bmatrix} \tag{54}
\]

Note in the Jacobians above that the state equation \( f \) is completely independent of \( R_0 \). By noting the 1 RC ECM model equations, the partial derivative for the transient voltage with regards to \( R_1 \) can be written as the following by utilizing the product and chain rules:

\[
\frac{\delta V_1(x_{k-1}^+, u_{k-1}, \theta)}{\delta R_1} = V_{1,k-1} \cdot \frac{-\Delta t}{e^{R_1/c_1}} \cdot \Delta t/(C_1 \cdot R_1^2) + I_{k-1} \cdot \left(1 - \frac{-\Delta t}{e^{R_1/c_1}}\right) + \frac{-\Delta t}{C_1} \cdot \frac{-\Delta t}{R_1} \tag{55}
\]

The derivative of the SoC with regards to the cell capacity is also needed. It may be written as the following by using the expression for SoC in (39):

\[
\frac{\delta SOC(x_{k-1}^+, u_{k-1}, \theta)}{\delta C} = \frac{I_{k-1} \cdot \Delta t}{C^2} \tag{56}
\]
If one should want to include the final ECM parameter, the capacitance \( C_1 \) in the weight vector, the partial derivative of \( V_1 \) with regards to \( C_1 \) would also be required. This is easily derived to be:

\[
\delta V_1 (x_{k-1}^+, u_{k-1}, \theta) = \frac{(V_{1,k-1}^+ R_1 - I_{k-1} \Delta t e^{-\frac{\Delta t}{R_1 C_1}})}{C_1^2 R_1}
\]

The final partial derivative in (53), that of the OCV with regards to the capacity, may be taken as zero under the assumption that the OCV-SoC relation is independent of cell aging. Now that all Jacobians and their containing partial derivative have been determined the overall structure of the DEKF may be summarized. Again, it is essentially a question of two EKFs running in parallel.

The first step is the initialization of both filters. Process- and measurement noise covariance matrices for both filters, initial state- and weight filter covariance matrices, and starting guesses for state- and weight vectors all must be specified. The general interpretations of the effects of these internal parameters on the behavior of the filters are the same as in the EKF, perhaps expect for the weight filter process- and measurement noise covariance matrices due to the lack of a more rigorous weight filter state equation. The choice of internal parameter values for the weight filter is especially critical for this reason [130].

After initialization, the following steps will be performed in order: time update for weight filter, time update for state filter, measurement update for state filter, and lastly measurement update for weight filter. After that the cycle is repeated as such until the end of filter operation. The state filter time- and measurement update steps are identical to those in the EKF and shall therefore not be repeated here. The weight filter time update step comprises the following equations:

\[
\hat{\theta}_k = \hat{\theta}_{k-1}^w
\]

(58)

\[
\Sigma^{-}_{\theta,k} = \Sigma^{+}_{\theta,k-1} + \Sigma_r
\]

(59)

\[
L_k^\theta = \frac{\Sigma^{-}_{\theta,k} \cdot (C_k^\theta)^T}{C_k^\theta \cdot \Sigma^{-}_{\theta,k} \cdot (C_k^\theta)^T + \Sigma_e}
\]

(60)

Here \( \Sigma_r \) is the weight filter process noise covariance matrix and \( \Sigma_e \) the weight filter measurement noise covariance matrix. Note in (59) that one might certainly write this on the same form as for the equivalent equation in the state filter time update (24), that is with the \( A_w \) matrix. This is however omitted here for convenience as \( A_w \) is simply the identity matrix. The measurement update step equations for the weight filter are the following:

\[
e_k = y_k - \hat{y}_k = y_k - (C_k \cdot \hat{x}_k^- + D_k \cdot u_k)
\]

(61)

\[
\hat{\theta}_k^+ = \hat{\theta}_k^- + L_k^\theta \cdot e_k
\]

(62)

\[
\Sigma^{+}_{\theta,k} = (I - L_k^\theta \cdot C_k^\theta) \cdot \Sigma^{-}_{\theta,k}
\]

(63)

Since the weight filter and the state filter have identical output equations, the filters will also share the same equation for the error. The structure of the DEKF algorithm may be seen as a state EKF inside of a weight EKF as the weight filter time update is the first step in a given iteration and the weight filter measurement update the last. The DEKF algorithm is summarized graphically below.
Figure 34: The DEKF algorithm. Solid lines represent the paths of state- and weight vectors while dashed lines are the flows of the covariance matrices [125].

Since two filters are run in parallel, the computational strain of the DEKF might be quite large, depending on the size of state- and weight vectors and the complexity of state- and output equations. In an effort to speed up computational performance, two different methods of estimating just capacity degradation are suggested. The first of these, the enhanced state vector extended Kalman filter is, as the name suggests, based on the EKF algorithm.

3.3.4 Enhanced state vector extended Kalman filter

The ESVEKF is a simplified algorithm for the estimation of cell capacity degradation. The reason for this simplification is to achieve faster computational performance while still maintaining acceptable SoC and capacity estimation. The ESVEKF is a variant of the EKF, meaning only one Kalman filter is utilized. The state vector used in the EKF above has been extended to also feature the cell capacity:

\[ x_k = [V_1 \quad SOC \quad 1/C] \quad \text{(64)} \]

This is often called joint estimation, as opposed to the dual estimation of the DEKF and the SWDEKF which will be outlined below [149]. As the cell SoC is of course dependent on the cell capacity, the A-, B-, and C matrices of the EKF need to be adjusted accordingly. For the ESVEKF the A-, B- and C matrices will be the following:

\[
\hat{A}_k = \begin{bmatrix}
-\Delta t \\ e^{R_1C_1} \\
0 & 1 \\ 0 & 0 & -\Delta t \cdot I_k \\
0 & 1
\end{bmatrix} \quad \text{(65)}
\]

\[
\hat{B}_k = \begin{bmatrix}
R_1 \cdot (1 - e^{-\Delta t}) \\
0 \\
0 \\
0
\end{bmatrix} \quad \text{(66)}
\]

\[
\hat{C}_k = \begin{bmatrix}
-1 \\
\frac{\delta(OCV)}{\delta(SOC)} \\
\frac{\delta(OCV)}{\delta(\frac{1}{C})}
\end{bmatrix} \quad \text{(67)}
\]
The overall algorithm will then be identical to that of the normal EKF as described above. The partial derivative of the OCV with regards to $1/C$ is assumed to be equal to zero, thus echoing the earlier assumption of the OCV-SoC relation being independent of cell aging.

Note that the ESVEKF will not involve the use of recursive derivatives as the DEKF and SWDEKF. This is because the derivative with regards to $1/C$ is taken with all other elements of the state vector, that is SoC and $V_1$, as fixed. In the dual estimation approaches, the total derivative, with the accompanying recursive procedure, is needed as the elements of the state vector are functions of the elements of the weight vector. This lack of recursive derivatization means the fewer computations must be performed in each iteration of the joint estimation algorithm. Some authors have claimed that this lack of recursive derivation might lead to unstable filter behavior however [149]. Another issue might be a risk of poor numeric conditioning due to the very different rates of change of the elements in the state vector [150]. Advantages of joint estimation approaches over dual estimation should be the inclusion of any correlations between changes to especially SoC and capacity due to the common covariance matrix and the reduced number of internal filter parameters which should be conducive to easier filter tuning.

3.3.5 Single weight dual extended Kalman filter
The idea of the SWDEKF is to introduce a simple method for estimation of SoH with less computational strain than the full DEKF. By reducing the weight vector of the DEKF to just a single element, the cell capacity, the need for heavy matrix operations is dramatically reduced. The SWDEKF is otherwise identical to the DEKF in terms of overall algorithm as well as state- and output equations, bar the concatenated weight filter and accompanying matrices. This includes the need for recursive derivation in every iteration of the algorithm. Another advantage of reducing the size of the weight vector is that fewer internal filter parameter values need to be specified during the initialization step, meaning that filter tuning is simplified compared to for the full DEKF. The main concern of this method in comparison to the full DEKF is the lack of adjustments to the internal resistance parameters which should result in the filter losing accuracy over time.

In comparison to the other simplified capacity estimator, the ESVEKF, the number of tuning parameters is larger for the SWDEKF. The main advantage of the SWDEKF over the ESVEKF should in theory be the faster computational performance as the strain of the many matrix operations of the Kalman filter algorithm is reduced by scaling down matrix sizes [150]. This is so despite doubling the number of filters since the execution time of matrix multiplication- and inversion scales to the power of roughly three with the matrix size. That is for two $nxn$ matrices, the execution time of their multiplication will scale as $n^3$ [130, 151]. This means that running two filters in parallel, of course these are interconnected to an extent as well, of sizes two and one, under the assumption that the matrix operations are the limiting factor in terms of performance, is more effective than running one filter of size three as $3^3=27>(2^1+1^3)=8$. In practical use, there are of course additional factors to be considered than just matrix operations and measured actual computational times will therefore be compared.


4 Results and Discussion

4.1 Extended Kalman Filter

As the most basic variant of the Kalman filter, the LKF, is in essence not applicable for the Li-ion battery monitoring due to the strong non-linearity of the system, it was for the linearized EKF discarded in this work. The EKF, still the most basic algorithm investigated in this study, requires the allocation of four values of internal filter parameter in the initialization stage: the starting guesses of the state vector and state covariance matrix, the process noise covariance matrix, and the measurement noise covariance matrix. The effects of all these parameters on filter performance will now be presented and discussed. What will become evident is that while the EKF certainly is a very powerful SoC estimator, it does have its limitations and pitfalls. Fortunately, after a careful analysis of the filter behavior these issues can be avoided relatively easily or at least managed effectively. The EKF proves successful in overcoming most the issues of the Coulomb counting method, as discussed in section 1.4, including sensitivity to starting SoC value, and sensitivity to sensor noise. The EKF is also found to outperform Coulomb counting in terms of SoC estimation accuracy for cases of inaccurately specified cell capacities.

As the EKF, here being used purely as an SoC estimator, does not update any of the values of ECM parameters, the initial values of these were used unless otherwise stated. These values were determined using battery diagnostic test before the first RW phase as described earlier in section 3.1. The reference method for validation of the SoC estimation results was Coulomb counting for all SoC work. Coulomb counting was applicable as a reference method as the batteries were always fully charged before a RW phase began, meaning that the starting SoC was exactly known, and as current and time were measured with high precision laboratory equipment, thus minimizing sensor bias and other measurement error. It should however be noted that the quality of the Coulomb counting reference will degrade over time as the cell capacity used for its calculation will be assumed to be constant while the actual cell capacity will change due to variations in temperature, current and most importantly due to aging. Assuming that capacity degradation due to aging is the dominating effect, the Coulomb counting should be at its most accurate at the beginning of each RW phase. Therefore, mainly these operation periods were used for validation and comparison of SoC estimation results.

As Coulomb counting is already indirectly included in the model equations of the applied ECM, more specifically in the state equation, see (39), the SoC estimation of the filter should follow the Coulomb counting reference very closely when the starting value of SoC specified during initialization of the algorithm is correct, in this case SoC0=1 as the battery is known to be fully charged at the start of any given RW phase. This scenario, with the starting guess for the transient voltage being zero, is shown in figure 35 below over the first 100 RW steps of the first RW phase of battery RW9. The internal filter parameters used for obtaining the results in the two figures below are summarized in table 9.
Looking at figure 35, it is evident that the Kalman filter SoC estimation indeed follows the Coulomb counting reference very closely in this case. As none of the common issues of Coulomb counting appear in the test above no advantages of the EKF are manifested either. If one examines a later stage of the same RW phase, such as in figure 36 below, one will notice a larger difference between the EKF SoC estimate and the Coulomb counting reference. The reason for this is partly that modelling errors grow larger over time due to the ECM parameter values not being adjusted to compensate for aging effects. This means that the state covariance matrix will get bigger (individual elements will assume larger values) leading to an increased state uncertainty and thus variance in SoC estimation. The Coulomb counting reference will also be less accurate towards the end of any given RW phase as the real cell capacity has been lowered from the initial measured value due to aging effects. Since one can be sure that the Coulomb counting reference is inaccurate at this stage, but not to what extent, it is difficult to quantify the error of the EKF SoC estimation too. While it is impossible to verify SoC estimation results in this case of no valid reference it should be noted that the EKF and other Kalman based approaches have the potential of being more accurate than Coulomb counting in cases of uncertain cell capacity. This is due to the state estimate measurement update (29) adjusting the SoC estimate to match the measured cell voltage. The state estimate time update equation, keeping in mind the state equation (39) of the utilized ECM, is identical to Coulomb counting, but the additional inclusion of the cell voltage measurement in the correction stage of the EKF algorithm should allow for a more accurate SoC estimation than when solely relying on Coulomb counting.
If one keeps the same ECM parameter values, that is values of $R_0$, $R_1$, $C_1$ and cell capacity, for even longer, the resulting EKF SoC estimation error is more noticeable. The figure below shows an example simulation from the 5th RW phase of RW9, recorded a little more than three weeks into testing of RW9. Since figure 37 shows the start of a RW phase, the Coulomb counting reference may be regarded as being accurate as the capacity used for its calculation has been adjusted to the latest measured value. SoC estimation accuracy has clearly degraded compared to the results above. The need for online ECM parameter adjustment to achieve consistent acceptable SoC estimation accuracy over time is evident, as are the limitations of the EKF algorithm. Still, the SoC estimation results are not too far of the Coulomb counting reference values despite the now quite extensive modeling error. The root-mean-square error of the EKF SoC estimate compared to the Coulomb counting reference is 4.3% in figure 37. As a comparison, if normal Coulomb counting would have been applied under the same circumstances, that is with an unadjusted capacity, the resulting root-mean-square error to the Coulomb counting with adjusted capacity would be a whole 10%. This then confirmed the discussion above regarding the EKF being more accurate than Coulomb counting for cases of an inaccurate cell capacity. This is of course a great advantage of the EKF over Coulomb counting and effectively results in capacity recalibration, usually meaning a total discharge of the battery from the fully charged state, being necessary less frequently. The issue of negative SoCs discussed briefly in section 3.2 is also evident in figure 37.
After having seen two cases of the EKF not performing noticeably better than (capacity-adjusted) Coulomb counting, some scenarios which highlight its upsides will now be considered. One of the most significant strengths of the EKF, especially compared to Coulomb counting, is that the initial SoC does not need to be known exactly for SoC estimation to be successful. The filter should be able to converge to the true SoC within a reasonable time given a fairly accurate battery model, at least if values of internal filter parameters have been sensibly chosen to reflect the uncertainty of the initial state. In fact, as seen in the next figure below, the initial guess of SoC could indeed be very poor and the EKF would still find the correct value after an initial period of adjustment. The interaction between the initial guesses of the state vector and the state covariance matrix was found to be critical to this behavior and will now be analyzed.

Cell SoC must in theory always lie in the range between zero and one (0 % - 100 %). If the initial SoC should be totally unknown, the average starting SoC guess error would thus be 0.5 on average. Such a situation seems unlikely as one will usually at least have some vague idea of how charged the battery is, meaning that the real average initial error is likely smaller. The initial guess of transient voltage is easier to determine as this will almost always be zero, assuming that estimation starts when operation of the battery starts. This is because the transient voltage will naturally decay to zero over time when no current is being applied. For the analysis of the impact of the initial state vector guess on SoC estimation accuracy, only the initial SoC will therefore be varied in the analysis below. First only the initial guess of SoC was varied, meaning that all other internal filter parameters of the EKF were set as in the previous two figures. In the figure below the starting SoC guess has been set to 0.75, 0.5, and 0.05 respectively.
As can be seen in figure 38, the EKF will converge to the “true” value represented by the Coulomb counting reference despite even very poor initial guesses of SoC, thus seeming to be quite robust to poor initialization. The difference when the starting guess is poor is that it takes the filter a longer time to converge to the Coulomb counting reference, stabilizing after approximately 0.03 hours (less than 2 minutes) of operation for a starting SoC guess of 0.75 and after approximately 2 hours for a starting SoC guess of 0.05 with this set of internal filter parameters. After the filter SoC estimate has converged, the estimation is of equal accuracy for all starting guesses of SoC and virtually identical to the Coulomb counting reference.

It was found that the rate of estimation convergence to the Coulomb counting reference was also dependent on the initial SoC covariance. The case of a constant starting SoC guess of 0.5 with a varying initial SoC covariance was considered below. Since the real SoC starting value was known, an appropriate starting value for the SoC covariance could easily be determined using the rule of thumb in (20): \( \Sigma_{s,0} = (1-0.5)(1-0.5) = 0.25 \). The values for the starting guesses for SoC covariance have been set to 0.25, 0.01, and 0.001 in the figure below, all other internal filter- and ECM parameters being set as in table 9.
Here it is interesting to note that convergence is fastest for the estimate with the middle value $\Sigma_{x,0}=0.01$, which approached the Coulomb counting reference almost immediately, while the lower starting covariance value results in the estimated SoC reaching the Coulomb counting reference after around two hours. It is also evident that when the starting SoC covariance is set to 0.25 there is not enough time for the filter estimate to converge within the first 100 RW step at all. If the simulation was run for a longer period, convergence would however occur eventually, as seen below.

Note the values on the time axis in figure 40. The time needed until convergence had occurred could be very long for large values of initial SoC covariance. If the convergence behavior is this slow, issues might arise due to substantial changes to the values of ECM parameters due to aging having occurred.
before converging. If this is the case, the filter may diverge and never find the true SoC value at all. It bears repeating that the mode of cycling used for battery RW9 is very intense and aging effects are thus accelerated compared to during most normal Li-ion battery applications.

The effect of changing the starting state covariance matrix was evidently found to be quite large. The optimal value of the initial SoC covariance also appeared to be lower than 0.25, which puts the validity of the rule-of-thumb method in (20) into question. It was also notable that the filter would still converge to the Coulomb counting reference eventually, even for seemingly too high initial values of the SoC covariance, just that it would take a very long time in some cases. On the other hand, if the initial covariance was set much too large there was a risk that the filter would diverge completely. The idea when specifying the initial state covariance was then found to be something like this: specify a large initial covariance if the initial state uncertainty is large to allow for fast convergence, however risking low stability (risk of estimation diverging) in the process. On the other hand, setting a small initial covariance lessens the issues of estimation instability but might instead result in extremely slow convergence [63]. Again, a worst-case scenario in the case of Li-ion battery monitoring is that the convergence is slower that the rate of change of ECM parameter values which could lead to total divergence. Finding appropriate values of the initial state covariance matrix therefore becomes something of a balancing act, especially for more uncertain starting states.

Finally, the unvarying internal EKF parameters were examined: the process- and measurement noise covariance matrices. Simply put, these parameters decide the “trust” (or “distrust”) of the measurements relative to the model predictions. First the effects of changing the value of the measurement noise parameter were investigated. To show this in the clearest way possible some artificial noise is added to the cell voltage signal. This is done to simulate a very noisy, worst-case-scenario measurement signal as a kind of stress test of the algorithm. The added noise is Gaussian (normally distributed) and centered around the actual measured value. In other words, the accuracy of the cell voltage measurement was still the same as before but its precision has been artificially lowered. The easiest way of accomplishing this in MATLAB was to multiply each element in the vector of measured cell voltages by a normally distributed number centered on 1 with a standard variation of b in this way:

$$V_{t, \text{noisy}} = V_t \ast (1 + b \cdot \text{randn}(1, \text{length}(V_t)))$$

(68)

The MATLAB function “randn” being a generator of normally distributed random numbers centered on zero. For these tests the standard deviation of the noise was set to $5 \times 10^{-3}$ V, producing a suitably noisy voltage signal. For the three simulations in the figure below, the measurement noise covariances were set to 0.5, $5 \times 10^{-3}$ and $5 \times 10^{-5}$ V, respectively. The variation of the measurement noise covariance between these simulations was quite massive in other words (four powers of ten). One of the main advantages of Kalman filter based approaches is their ability to deal with noisy input- and output signals. By this it is meant that the algorithm can separate out the true signal from noise and thus achieve a better state estimate than what would be possible using a deterministic approach such as Coulomb counting.
The starting SoC was set to 0.5 for all tests shown in figure 41. The effect of changing the measurement noise covariance on the convergence rate of SoC estimation can be seen to be somewhat like that of changing the initial state covariance matrix. This is sensible when considering (24). The calculation of the Kalman gain matrix controls how heavily state estimations should be adjusted. If the measurement signal is known to be very noisy, meaning that a high measurement noise covariance has been specified, then the model prediction should hold more weight and vice versa. While this means more accurate SoC estimation overall, it might also mean slower convergence. This effect is observed in figure 41. If one eliminates the effect of slower convergence by specifying a correct starting SoC, then the general a trend can be seen between the logarithm of the measurement noise covariance and the root-mean-square SoC estimation error:

Figure 41: EKF SoC estimation with different measurement noise covariances.

Figure 42: Overall trend of rms SoC error versus logarithm of measurement noise covariance for cases of noisy signals and correct starting SoC for the EKF.
The results in figure 42 show the average rms SoC error versus the Coulomb counting reference over five simulations of the 300 first RW steps of the first RW phase of battery RW9. All internal filter parameters were as in table 9 with standard deviations of current- and voltage signal noise equal to values in figure 41. It is clear that the SoC estimate accuracy increases with increasing measurement noise covariance for very noisy measurement signals if convergence of the estimate does not need to be considered. Of course, if measurement noise covariance is increased even further, the EKF will essentially turn into Coulomb counting as the cell voltage signal is eventually completely ignored, resulting in an rms error of zero in the case shown in figure 42 as Coulomb counting is the reference method.

Of course, in this work where carefully recorded laboratory measurements were used, the measurement noise was generally very small, meaning that the interpretation of the cell voltage signal is straightforward. In a more practical situation this might not be the case, as can be seen in the figure below where the EKF cell voltage predictions corresponding to the same set of measurement noise covariances as in figure 41 above are shown.

As can be deduced from the plot of cell voltages above, lowering the measurement noise covariance might cause the algorithm to interpret measurement noise as true signal. This will lead to continuous overcompensation of the state estimates, per the measurement update equation (27), reducing the precision of not only the cell voltage prediction but also the SoC estimation. This overfitting-like behavior can be seen especially for the case of $\Sigma_v=5\cdot10^{-5}$ above. On the other hand, a higher value of the measurement noise covariance allows for the Kalman filter to better recognize the true cell voltage signal and thus stability and precision is clearly improved. In figure 43 this can be seen for the simulation using $\Sigma_v=0.5$, which follows the true, as measured, voltage signal quite closely despite the quite heavy added artificial noise. Considering the wide range of values of the measurement noise covariance in the figure above, one can also draw the conclusion that the SoC estimation is not especially sensitive to this parameter, even though it does have an impact. Naturally, the importance of this parameter is heightened for scenarios with higher sensor noise and bias.

![Figure 43: Cell voltage predictions of the EKF for the case of a noisy voltage signal using different measurement noise covariance matrices.](image-url)
The process noise covariance parameters can be said have a somewhat similar effect on results as the measurement noise covariance, controlling how much “trust” one puts in the model versus measurements. One difference of the process noise covariance matrix to the measurement noise covariance is that here there are two separate parameters interacting with each other. The relative magnitude of the process noise covariance of the transient voltage compared to that of the SoC estimate turned out to be an important factor in achieving overall algorithm precision and stability. The most accurate SoC estimation results were achieved when the process noise covariance of the transient voltage was set to a relatively higher value than for the SoC, such as has been the case for all simulations so far in this section. This is quite sensible when considering how the development of the ECM was conducted in this work. The state equation in the EKF calculates the SoC using only one prespecified ECM parameter, the cell capacity. This is a carefully measured value, using what is essentially Coulomb counting and an accurate 13\textsuperscript{th} degree polynomial OCV-SoC expression. On the other hand, the estimation of the transient voltage \( V_1 \) uses the estimated values of both \( R_1 \) and \( C_1 \). Due to the high complexity of battery transient behavior most of the modelling error of a given ECM will lie here. This is the reason why more advanced ECMs are generally expanded by adding on more resistors and capacitors, while the treatment of the OCV-SoC relation and the capacity determination largely remains the same [104]. When one compares the error in estimation of the OCV-SoC relation in figure 24 with the overall modeling error in 28 for instance, it is clear that much of modeling error stems from the transient behavior of the cell. Thus, the transient voltage estimate should logically be less accurate than that of SoC, which is reflected in the appropriate values in the process noise covariance matrix.

This principle of “balancing” modeling error will be seen to be an even more important aspect for SoH estimation. Finally, some practical usability aspects of the EKF will now be discussed.

In the case of the data used in this work, which is recorded from laboratory experiments, Coulomb counting is a valid SoC reference method for the reasons discussed at the start of this section. In a more realistic situation, where no reasonable reference for the SoC estimation exists, one will know that the filter has converged by gleaning at the values in the state covariance matrix. diagonal values of the state covariance matrix should be continuously decreasing, if internal filter parameters were set suitably anyway, and should eventually stabilize to a low value if the filter has converged, given that the accuracy of the applied battery model is reasonable. This will be demonstrated below by application of the rule-of-thumb in (21).

As mentioned earlier, one of the inherent strengths of Kalman filter based approaches to state estimation are the “built-in” error bounds. While (21) strictly only applies to the linear Kalman filter, it may still be used for more advanced Kalman filters to get a rough idea of the estimation certainty. In the figure below upper- and lower error bounds, calculated using (21), are plotted for SoC estimation using the EKF with noisy signals of voltage and current. Added artificial noise was on the form of (68) and had a standard deviation of \( 5 \cdot 10^{-3} \)V and \( 5 \cdot 10^{-3} \)A. The starting guess of SoC was 0.5.
It is clearly visible in figure 44 how the error bounds get closer to the estimate and the Coulomb counting reference value as filtering goes on, meaning that the SoC estimate certainty increases. It intuitively seems sensible that as the estimation certainty should increase with increasing information, in this case measurements of current and cell voltage, becoming available. This principle is reflected in (30) of the measurement update equations. After this initial period of the error bounds getting closer to the SoC estimate, the SoC estimate covariance should stabilize and remain roughly constant for the remainder of filtering unless modelling- or measurement errors change significantly in magnitude. This is shown in the figure below. Monitoring of the diagonal values of the state covariance matrix therefore becomes a useful tool in ensuring Kalman filter performance.

![Figure 44: Error bounds per (20) of SoC estimate using EKF.](image)

![Figure 45: The stabilization of SoC covariance in the EKF, indicating estimation convergence.](image)
4.2 **Adaptive Extended Kalman Filter**

The AEKF has the distinct advantage of updating process- and measurement noise covariance matrices during operation of the filter. This would in theory mean that the filter tuning process could be made simpler and also that the internal filter parameters would automatically be updated if operation conditions were to change in such a way as to affect the measurement- and/or process noise [130]. The algorithm is as mentioned earlier very similar to the EFK in most aspects and indeed produced very similar, if not even better, results in terms of accuracy and convergence rate of SoC estimation. The ECM parameter values obtained from pulsed- and reference discharge tests, as presented in tables 7 and 8 respectively, were used to obtain all AEKF results. Below the AEKF SoC estimation over the 100 first RW steps of the first RW phase of RW9 is shown against the Coulomb counting reference.

![Figure 46: SoC estimation for 100 first RW steps using AEKF with Coulomb counting reference.](image)

Note the very poor starting guess of SoC$_0$=0.05 in figure 46. The AEKF SoC estimate converges to the values measured with Coulomb counting extremely quickly and thus appears quite robust to poor initialization. It is of course interesting to compare these results with the results for the same starting SoC for the EKF in figure 38. The convergence rate of the AEKF SoC estimate is noticeably faster than for the EKF. The internal filter parameters used in obtaining the results above are summarized in table 10.

**Table 10: Internal filter parameters for AEKF for figure 46.**

<table>
<thead>
<tr>
<th>Internal filter parameters</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Starting SoC</td>
<td>0.05</td>
</tr>
<tr>
<td>Starting $V_1$</td>
<td>0.00 V</td>
</tr>
<tr>
<td>Covariance, SoC</td>
<td>$1\cdot10^{-2}$</td>
</tr>
<tr>
<td>Covariance, $V_1$</td>
<td>$1\cdot10^{-3}V^2$</td>
</tr>
<tr>
<td>Starting process noise, SoC</td>
<td>$1\cdot10^{-2}$</td>
</tr>
<tr>
<td>Starting process noise, $V_1$</td>
<td>$1\cdot10^{-2}V^2$</td>
</tr>
<tr>
<td>Starting measurement noise</td>
<td>$1\cdot10^{-2}$</td>
</tr>
<tr>
<td>Moving estimation window size, $M$</td>
<td>10</td>
</tr>
</tbody>
</table>
As can be seen in table 10, the starting state covariance matrix was the same as for the results obtained with the EKF above. Note the very high values used for starting guesses of the process noise covariance matrix and the measurement noise covariance. These starting values could furthermore be varied substantially without affecting estimation accuracy considerably. To demonstrate this behavior, the initial process noise covariance matrix was varied by multiplying the starting matrix values in table 10 by 0.01 and 100, respectively. All other internal filter parameters were as in table 10. As can be seen, convergence remains very rapid for all selected initialization values:

![Figure 47: Rapid convergence of AEKF SoC estimate for various initial process noise covariance matrices.](image)

Note the values on the axes in the figure above, the scale of the time axis has been made much smaller compared to in figure 46. As can be seen, larger values in the initial process noise covariance matrix seem to indicate faster convergence to the Coulomb counting reference. The variation of the initial process noise covariance matrix is quite large in figure 47, indicating that while its setting does matter, its effect on results was overall found to be very small. A starting process noise covariance larger than one is of course most often senseless in the context of SoC estimation. Assignment of very small values in the starting process noise covariance matrix could lead to the AEKF SoC estimate diverging for lower values in the initial state covariance matrix and should therefore be avoided.

If one instead varies the starting measurement noise covariance a similar conclusion is reached, as is seen below.
Figure 48: The quick converging of the AEKF for various starting measurement covariances.

Again, the scale of the time axis in figure 48 is notable. While the difference in results for different starting measurement covariances is relatively slim, it is seen in figure 48 that lower initial measurement noise covariance led to slightly faster convergence rates in this case. Note also the large range of values of the initial measurement noise covariance in figure 48, covering a full six powers of ten. The specification of this internal filter parameter was in other words found to be virtually arbitrary to overall filter performance in terms of accuracy and convergence rate.

The final new parameter of the AEKF compared to the EKF is the moving estimation window size, $M$, as in (42). The effects of changing this parameter were once again overall very modest. The biggest influence was seen when artificial noise was added to the measurement signals of cell voltage and current. For the results below the artificial noise was added on the form (68) with a standard deviation of $5 \times 10^{-3}$ V and $5 \times 10^{-3}$ A, respectively. All internal filter parameters except the moving estimation window size were as in table 10.

Table 11: The effect of moving estimation window size on AEKF SoC estimation accuracy.

<table>
<thead>
<tr>
<th>Moving estimation window size, $M$</th>
<th>Root-mean-square SoC error compared to Coulomb counting over 100 first RW steps</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>0.0149</td>
</tr>
<tr>
<td>100</td>
<td>0.0199</td>
</tr>
<tr>
<td>200</td>
<td>0.0222</td>
</tr>
</tbody>
</table>

It seems then that a smaller moving estimation window size is preferable from an estimation accuracy standpoint. Finding literature on reasoning of the choice of moving estimation window size seems difficult, Campestrini et al. used a moving estimation window size of ten in their recent review seemingly without motivating the choice [130]. Since this value seemed to produce acceptable results in that work and likewise when applied here it was used throughout. Sun et al. instead found a moving estimation window size of 100 to produce the best results for use in the AEKF, thus seemingly contradicting the results observed here [117]. Looking at table 11, the moving estimation window size appears to be very much a fine-tuning parameter for the case of Li-ion battery monitoring. This seems
to confirm the conclusion in [145] that the moving window size has little influence on overall filter performance.

The observed effects of changing the initial state covariance matrix of the AEKF was found to be identical to those observed for the EKF and will therefore not be discussed further. Finally, for the case of SoC estimation given a faulty assignment of cell capacity, as seen in figure 37 for the EKF, the AEKF was found to perform slightly better than the EKF, achieving a root-mean-square error of 4.2 % in the identical situation as discussed in section 4.1.

4.3 Dual Extended Kalman Filter
Having explored two different Kalman filters for SoC estimation, the focus turned towards methods for estimation of SoH. First the DEKF was investigated. This method utilizes two extended Kalman filters working in parallel, one for state estimation (e.g. SoC) and one for weight estimation (e.g. capacity). There is also some information exchange between the filters. Out of the three examined SoH estimators, the DEKF was the only one capable of estimating not only the cell capacity degradation, but also the increase of internal resistance over time. For this reason, it is very interesting to compare results of the DEKF with results for the other two SoH estimation algorithms which are only adjusting the capacity during filtering. For validation of all capacity estimation, the measured cell capacities in reference discharge tests, as described in section 3.1.1, were used as reference values.

The figure below show a plot of cell capacity estimation using the DEKF over the first full RW phase of RW9. Table 12 summarizes all the internal filter parameters that were used in initialization of the DEKF in figure 49. As can be seen, the capacity estimate converges relatively quickly to a value close to the measured capacity before the first RW phase, and then slowly decreases to eventually reach a value close to the measured value before the second RW phase by the end of the simulation, see table 7 for these measured capacities. Here the initial cell capacity value was set to 1.9 Ah. The initial guess of SoC was 0.999, and was thus taken as wholly accurate considering the fully charged state of the cell before each RW phase. It is believed that this corresponds to a realistic scenario since fully charging a cell using a constant current-constant voltage (CC-CV) scheme is always possible if the upper cut-off voltage of the battery is known. As such, the corresponding SoC initial covariance has been set to a low value, reflecting the relative certainty of the initial SoC guess.

After the initial converging mode during the first four hours or so of filtering, the DEKF estimates an approximately linear loss of capacity over time which corresponds well to data in the literature [88, 152, 153], keeping in mind the relatively unique mode of operation used for obtaining the data used in the current study. This linear capacity loss over time also seems within reason when considering the trend of measured capacities in reference discharge tests over the first twelve RW phases of RW9 as seen in figure 21. Achieving reasonable results with the DEKF required careful consideration of the internal filter parameters, filter tuning that is. This will be discussed further below.
The measured capacities via reference discharge tests were 2.10 Ah before and 2.04 Ah after the RW phase simulated in figure 49. It is difficult to exactly quantify the accuracy of the estimate of the initial capacity here due to the initial period of estimate convergence, but the highest capacity estimated by the DEKF in figure 49 is 2.11 Ah, which would be equivalent to a relative error of less than 0.5 %. The capacity estimated by the DEKF algorithm at the end of the RW phase above is 2.02 Ah, thus corresponding to a relative error of less than 1 %. The possibility of estimating the battery SoH in terms of capacity fade using the DEKF therefore seems promising.

Table 12: Internal filter parameter values for initialization of DEKF in figure 49.

<table>
<thead>
<tr>
<th>State filter internal parameters</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Starting SoC</td>
<td>0.999</td>
</tr>
<tr>
<td>Starting $V_1$</td>
<td>0.0 V</td>
</tr>
<tr>
<td>Covariance, SoC</td>
<td>$1\cdot10^{-10}$</td>
</tr>
<tr>
<td>Covariance, $V_1$</td>
<td>$1\cdot10^{-5}$V$^2$</td>
</tr>
<tr>
<td>Process noise, SoC</td>
<td>$1\cdot10^{-10}$</td>
</tr>
<tr>
<td>Process noise, $V_1$</td>
<td>$1\cdot10^{-8}$V$^2$</td>
</tr>
<tr>
<td>Measurement noise, state filter</td>
<td>$5\cdot10^{-3}$V$^2$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Weight filter internal parameters</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Starting capacity</td>
<td>1.9 Ah</td>
</tr>
<tr>
<td>Starting $R_0$</td>
<td>0.070 Ω</td>
</tr>
<tr>
<td>Starting $R_1$</td>
<td>0.032 Ω</td>
</tr>
<tr>
<td>Covariance, capacity</td>
<td>$1\cdot10^{-4}$ (Ah)$^2$</td>
</tr>
<tr>
<td>Covariance, $R_0$</td>
<td>$1\cdot10^{-9}$ Ω$^2$</td>
</tr>
<tr>
<td>Covariance, $R_1$</td>
<td>$1\cdot10^{-9}$ Ω$^2$</td>
</tr>
<tr>
<td>Process noise, capacity</td>
<td>$1\cdot10^{-11}$ (Ah)$^2$</td>
</tr>
<tr>
<td>Process noise, $R_0$</td>
<td>$1\cdot10^{-11}$ Ω$^2$</td>
</tr>
<tr>
<td>Process noise, $R_1$</td>
<td>$1\cdot10^{-11}$ Ω$^2$</td>
</tr>
<tr>
<td>Measurement noise, weight filter</td>
<td>$1\cdot10^{-2}$V$^2$</td>
</tr>
</tbody>
</table>
If the initial SoC would also be unknown, the initial SoC covariance must be increased to reflect this. In the figure below the starting SoC was set to 0.5 with an initial SoC covariance of $1 \cdot 10^{-4}$. Note how the filter will find the correct capacity and SoC concurrently despite poor initial values of both. It should be noted that the specification of initial covariance matrices was found to become more critical, and sensitive, when starting values of several closely connected parameters, such as the SoC and cell capacity, were uncertain. It was very easily so that one of the two connected parameters, such as SoC and capacity, would be over-adjusted if initial covariance matrices were not chosen carefully. This could lead to the filter diverging. This is believed to be due to a certain “entanglement” effect of the parameters due to the two filters sharing the same output equation. This will be discussed further below.

![Figure 50: SoC- (left) and capacity (right) estimates by DEKF when initial guesses of both SoC and capacity are poor for 100 first RW steps of first RW phase of RW9.](image)

Note in figure 50 above how the SoC estimate converges to the Coulomb counting reference approximately simultaneously as the capacity estimate reaches a value close to what was measured before the RW phase, in this case 2.10 Ah.

The DEKF was also found to be able to be set up to quickly estimate the internal resistance parameters, given unknown or inaccurate starting values of these. The initial guess of the cell capacity was then set to the value measured before the first RW phase, 2.10 Ah, and the related initial capacity covariance was correspondingly set to a low value of $1 \cdot 10^{-7}$ to separate out the effects of capacity estimation for these tests. The internal filter parameters relevant to estimation of internal resistance parameters were set as in table 13:

<table>
<thead>
<tr>
<th>Table 13: Internal filter parameters of DEKF for results in figure 51.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Starting $R_0$</td>
</tr>
<tr>
<td>Starting $R_1$</td>
</tr>
<tr>
<td>Covariance, $R_0$</td>
</tr>
<tr>
<td>Covariance, $R_1$</td>
</tr>
<tr>
<td>Process noise, $R_0$</td>
</tr>
<tr>
<td>Process noise, $R_1$</td>
</tr>
</tbody>
</table>
As can be seen in figure 51 both internal resistance parameter estimates would converge and stabilize despite relatively poor initial guesses of both, the one for $R_1$ being too high and one for $R_0$ being too low in the case shown. When comparing to measured values of internal resistance parameters in table 8 it may be seen that both resistance estimates are too low in figure 51, $R_0$ by 7% and $R_1$ by 13%, approximately. This might be due to temperature effects which have not been accounted for in the current model. It is widely known that the internal resistance of a Li-ion cell will be lower at higher temperatures. In this case, the average temperature during the pulsed discharge test preceding the first RW phase of RW9 was 23.4 °C while it was 29.8 °C during cycling. The average temperature was in fact lower for all pulsed discharge tests in comparison to the subsequent RW phases, as seen in table 14.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>23.4</td>
<td>29.8</td>
<td>6.4</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>23.8</td>
<td>29.0</td>
<td>5.2</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>23.3</td>
<td>28.7</td>
<td>5.4</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td>29.0</td>
<td>35.5</td>
<td>6.5</td>
</tr>
<tr>
<td>9</td>
<td></td>
<td>30.0</td>
<td>35.2</td>
<td>5.2</td>
</tr>
<tr>
<td>11</td>
<td></td>
<td>30.1</td>
<td>34.3</td>
<td>4.2</td>
</tr>
<tr>
<td>13</td>
<td></td>
<td>28.9</td>
<td>34.2</td>
<td>5.3</td>
</tr>
</tbody>
</table>

Since the temperature dependence of the internal resistance parameters can be quite large, and was not determinable for the used data set, there is reason to believe that the filter is converging to the correct values despite the differences to measured values [127, 154]. While unfortunately there is no way of absolutely validating these estimates of internal resistance, some weight must be put on the ability of the algorithm to accurately predict the SoC and capacity using these values. In fact, the values of internal resistance estimated by the DEKF were found to consistently produce better results than those from the pulsed discharge tests when inserted into the EKF algorithm in terms of root-mean-square error of SoC estimation.
To clearly illustrate the advantages of the DEKF over the EKF for long-term SoC estimation another set of operational data from another battery was used. While still belonging to the “Randomized Battery Usage Data Set”, the Li-ion battery RW2 was operated quite differently compared to RW9 during the RW phases. Here it was the duration of charging that was randomized from the set [0.5 hours, 1 hour, 1.5 hours, 2 hours, 2.5 hours] and the charging current was instead always 2 A. The battery was charged using this current for the randomized duration or until the cell voltage reached the upper cut-off voltage 4.2 V. At this point, the charging changed to a constant voltage mode which continued until the applied current dropped below 0.01 A or the randomly chosen duration had ended. This is then essentially CC-CV charging, meaning that the battery is periodically fully charged and thus often reaches points of 100 % SoC. Discharging was done as with RW9, meaning that a random current between 0.5 A and 4 A is applied for five minutes or until the lower cut-off voltage is reached. A plot of a series of charge- and discharge cycles in the form of measured cell voltage and current from this data is shown in the two figures below.

![Figure 52: Measured cell voltage for RW cycling using CC-CV mode charging for RW2. Note the periods of constant voltage at 4.2 V.](image)

The points in figure 53 below were the current approaches zero will roughly represent points of 100 % SoC.

![Figure 53: Current over time for cycling of battery RW2.](image)
Below the SoC estimated by the DEKF for this data is shown, along with a Coulomb counting reference which uses the capacity measured in reference discharge tests before cycling began.

Due to the existence of these points of almost guaranteed 100 % SoC it is easy to see in figure 54 that the DEKF performs rather well in terms of predicting the SoC and thus cell capacity. The Coulomb counting, which of course does not adjust for changes to cell capacity, quickly loses its accuracy and even predicts an SoC below zero at certain points in figure 54. Of course, like Coulomb counting the EKF does not adjust for changes to cell capacity either and would thus struggle to perform much better in this scenario. It is also clear that the capacity fade is very rapid at these cycling conditions of high ΔSoC, as was expected from the discussion of cycle aging impact factors in section 1.2.4.2. In addition, some other advantages of the DEKF became evident with this test. Firstly, no data from reference discharge tests needed to be considered for determination of initial cell capacity as a rough initial cell capacity could be guessed from the earlier analysis of the similar battery RW9 and this estimate was then adjusted by the DEKF online to fit the data. Secondly, no pulsed discharge tests for determination of the values of internal resistance parameter values were needed as, again, rough guesses using experience from battery RW9 were sufficient. The only battery-specific data that was required for the initialization of the DEKF for the “new” battery was low current discharge data for determination of the OCV-SoC relationship of the cell, which was fitted with a 13th degree polynomial just as with RW9.

In other words, using the DEKF instead of the EKF could potentially reduce the need for likely time-consuming and expensive cell-characterization tests, at least if rough estimates of initial cell parameters are available. Another advantage of using the cycling data from RW2 for algorithm testing and validation is that points of negative SoC, as mentioned in section 3.2, are avoided. While it is difficult to quantify the effect that these points had on results, it does appear that especially estimates of capacity are more stable when using the data from RW2. For instance, it was found that the capacity estimate was less dependent on the initial guessed value and would eventually converge to a common value independent of the starting guesses more often. This is shown in the figure below where starting guesses of cell capacity were 1.7 Ah, 1.9 Ah and 2.2 Ah respectively.

![Figure 54: SoC by Coulomb counting and DEKF for CC-CV mode charge cycling of RW2.](image-url)
Figure 55: Convergence of capacity estimates despite large disparity in starting values using DEKF and data from battery RW2.

Note the relatively turbulent initial capacity estimates, followed by stabilization and convergence to a common value. The line representing measured capacity in figure 55 shows the capacity measured in reference discharge tests before cycling began. The cell capacity measured after the cycling in figure 55 was 1.92 Ah, thus closely matching the estimated value. The relatively slow rate of convergence is also notable. For the data acquired from RW9, the capacity estimates tended to diverge if the convergence was not very quick. The fact that such slow convergence as seen in figure 55 was at all possible indicates a certain robustness of the capacity estimation. Certainly, convergence could also be made quicker by increasing the initial capacity covariance, as seen below.

Figure 56: Quick convergence of capacity estimates with different initial guesses of capacity for DEKF using battery data from RW2.
Note the difference in the scale of the time axes in figures 55 and 56. As can be seen to the left of figure 56, higher values of initial capacity covariance do seem to cause an increased initial estimate turbulence, which might be an issue. All internal filter parameters were set the same in obtaining the results shown in the two figures above, bar the initial capacity covariance which was set to $10^6$ (Ah)$^2$ and $10^3$ (Ah)$^2$ in figure 55 and 56 respectively.

As is evident per table 12, the number of internal filter parameters that must be set for initialization of the DEKF is very large, much larger than for the EKF: 17 for the DEKF compared to just 7 for the EKF. This is troublesome, especially considering the lack of strict, intuitive guidelines as how to set most of these, even though some general principles and strategies learned from the EKF tuning may be applied. Also, the curse of dimensionality means that exploring combinations of these parameters quickly becomes unmanageable. Of course, the state- and weight filters and their internal filter parameters will interact with each other as well. This means that also the relative magnitude of parameter values in the filters must also be considered. The trial-and-error approach of identifying covariance matrix values outlined in section 3.3 of course becomes exponentially more time consuming with an increasing number of parameters to be set. Despite these issues, which hindered the full characterization of DEKF filter tuning behavior, some general trends could be elucidated.

For the initial guesses and initial covariance matrices the same general principle as for the EKF applies, that is if initial guesses are uncertain or intentionally poor then the initial covariance matrix/matrices should be set larger to ensure correct convergence. Values of the process- and measurement noise covariance matrices, which are now two in number respectively, are still the most challenging to determine and a trial-and-error approach is a must for most cases, including this work. As was seen in the discussion of the EKF above, by balancing the process- and measurement noise covariance matrices it was possible to achieve accurate estimation despite very noisy input- and output signals for instance. In the case of using two filters simultaneously, this procedure becomes more complex. As can be seen in table 12 and in earlier descriptions of the DEKF algorithm, measurement noise is to be set separately for the state- and weight filters. This might seem odd considering that the two filters share the same output equation and, thus, measurement of cell voltage. Much like when considering the very simple state equation of the weight filter, this is a question of the difference in rates of change of the elements of the state vector compared to those in the weight vector.

Take the cell capacity as an example. Say that in a certain time step a difference between the predicted- and measured cell voltage is recorded, an error per (28). The goal is now to adjust first the state vector and then the weight vector by using this error multiplied by the corresponding Kalman gain matrix per (29) and (62). Looking at the output equation, this is a summation of three terms: the OCV, which is a function of the SoC; the transient voltage $V_{trans}$, and the instantaneous ohmic resistance response term $I\cdot R_0$. The error in the calculated cell voltage might be due to an error in the estimation of any of these terms individually, some combination thereof, or all of them simultaneously. For the EKF the situation is now relatively simple. Since only the SoC and the transient voltage are adjusted on an online basis, one will simply adjust one or both based on the current state covariance matrix and the C matrix, along with the specified constant measurement noise covariance. For the DEKF things get more complicated. A faulty SoC prediction might be due to either the state filter as in the EKF, but might also be caused by a faulty cell capacity estimate. This is because the SoC is a function of the cell capacity, as is evident by considering its definition in (9). The same situation applies for the transient voltage term $V_{trans}$ and the estimation of the internal resistance parameter $R_0$. To be able to separate these effects, setting different measurement noises for the state- and weight filters might be very useful. As is known, the elements of the state vector will change much faster than those in the weight vector. This is reflected by the weight vector being initialized with a larger measurement noise covariance than the state filter.
The balance between the measurement noises was found to be critical for the proper convergence of the filter. It should however be noted that a similar separation-of-time scales effect could be achieved by setting the measurement noises identically for the two filters and then lowering the values in the weight filter process noise covariance matrix accordingly. Examples of this may be seen in the results section 4.6.2 below. Campestrini et al. applied different measurement noise covariance matrices in the state- and weight filters of their DEKF approach without motivation.

The process noise covariance matrix plays quite a different role in the weight filter than in the state filter as it is used to “drive” the process, for the lack of a better weight filter state equation. This was especially evident for the estimation of cell capacity. Setting the capacity process noise covariance too high resulted in a too fast capacity degradation as shown in the figure below. The opposite was also applicable, setting the weight filter process noise covariance matrix too low results in no changes to parameter values over time at all. Another phenomenon could be observed if the state filter SoC estimation process noise covariance was set low while the weight filter capacity estimation process noise covariance was set to high values. This would lead to estimates of both SoC and capacity diverging. This is because the capacity estimate is overcorrected to adjust for errors in SoC, but the algorithm still considers the SoC estimate from the state filter to be relatively accurate due to the low SoC covariance. The result is a self-accelerating effect where the capacity estimates become increasingly inaccurate causing the SoC estimates to diverge without the state filter “intervening” enough to adjust the faulty SoC estimate.

![Figure 57: Capacity estimation using the DEKF over the first RW phase of RW9 with the capacity process noise covariance set too high. Internal filter parameters are like in table 12 except for capacity process noise covariance=1⋅10⁻⁸.](image)

Again, the capacity process noise covariance was found to be especially important. For the internal resistances \( R_0 \) and \( R_1 \), the relevant values of the process noise covariance matrix seemed to mostly control the noise of the estimation, meaning that appropriate values of these parameters could be found by simple visual inspection. This can be seen in the figure below where estimations of internal resistances have been performed using identical internal filter parameters to those used to obtain the results in figure 51 bar the values of process noise covariance for the internal resistance parameters.
Note how effect of varying the process noise covariances only become evident after some time has passed. This is due to initial covariance matrices being set identically for all simulations in figure 58.

As is evident, a higher process noise covariance leads to a noisier estimate of the internal resistance parameters. It is notable how the relation between the process noise covariance and noise level of the estimation appears very similar for \( R_0 \) and \( R_1 \). For this reason, it seemed sensible to keep the values in the process noise covariance matrices identical for these.

Finally, it is interesting to note that estimation of internal resistances \( R_0 \) and \( R_1 \) was much more straightforward then the estimation of the cell capacity from a filter tuning perspective. The DEKF was found to estimate these parameters with a level of robustness that was simply not possible for the cell capacity, especially so when data from battery RW2 was used. It is thought that this might be due to the “entanglement” effect of the SoC and the cell capacity as mentioned above. From this perspective, investigation of a correlation between cell capacity loss and internal resistance gain due to aging would possibly be of high interest. Theoretically, online estimation of internal resistance parameters should also allow for more accurate short-term SoC estimation as effects of, say, changing temperatures could be indirectly controlled for.

4.4 Enhanced State Vector Extended Kalman Filter

The first of the suggested simpler Kalman filter SoH estimation algorithms, the ESVEKF, uses a single state filter to estimate the cell capacity. Due to the similarity to the already analyzed EKF, a lot of lessons learned in studying that approach could be applied here as well. The filter was found to be able to estimate the cell capacity rapidly and accurately despite poor initializing values, as can be seen below. The ESVEKF does not estimate the internal resistance parameters of the cell. Because of this fact, the measured values from pulsed discharge tests were used for all results. The results in figure 58 are quite similar in terms of accuracy and overall appearance to what was observed for the DEKF in figure 49.
Figure 59: Capacity estimation over entire first RW phase of battery RW9 using ESVEKF.

After the initial stabilizing period, SoC estimation is accurate and remains so for the duration of the first RW phase:

Figure 60: SoC estimation using the ESVEKF for a poor initial guess of cell capacity.

For filter tuning of the ESVEKF, the found-to-be successful internal filter parameter values of the EKF were the starting point. Indeed, the most fruitful results of the ESVEKF were achieved with internal filter parameters set very similarly. The values of internal filter parameters used for results shown in figures 59 and 60 are summarized below.
Table 15: Internal filter parameters of the ESVEKF for the results in figures 59 and 60.

<table>
<thead>
<tr>
<th>State filter internal parameters</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Starting SoC</td>
<td>0.999</td>
</tr>
<tr>
<td>Starting V₁ (V)</td>
<td>0.00 V</td>
</tr>
<tr>
<td>Starting C</td>
<td>1.9 Ah</td>
</tr>
<tr>
<td>Covariance, SoC</td>
<td>1∙10⁻⁴</td>
</tr>
<tr>
<td>Covariance, V₁</td>
<td>1∙10⁻⁶ V²</td>
</tr>
<tr>
<td>Covariance, 1/C</td>
<td>1∙10⁻⁴(1/Ah)²</td>
</tr>
<tr>
<td>Process noise, SoC</td>
<td>1∙10⁻¹⁰</td>
</tr>
<tr>
<td>Process noise, V₁</td>
<td>1∙10⁻⁴ V²</td>
</tr>
<tr>
<td>Process noise, 1/C</td>
<td>1∙10⁻¹¹(1/Ah)²</td>
</tr>
<tr>
<td>Measurement noise</td>
<td>5∙10⁻³ V²</td>
</tr>
</tbody>
</table>

The number of internal filter parameters has been reduced from the 17 of the DEKF to just 10 in the ESVEKF, which is just three more than the EKF. Due to the single filter approach, only one measurement noise covariance value must be assigned, making finding an appropriate value for this parameter as intuitive as in the EKF. The algorithm was found to be quite insensitive to the starting guess of cell capacity, converging to a capacity value close to the measured 2.10 Ah for a wide array of different initializing values:

![Figure 61: Capacity estimation using the ESVEKF converging to measured value independently of starting guess.](image)

For comparisons sake, all internal filter parameters except the starting guess of capacity were the same in figure 61 as in figures 59 and 60. The ESVEKF would also quickly converge even if the starting guess of SoC also was uncertain. In the figure below the initial capacity was set to 1.9 Ah while the initial SoC was set to 0.5. All other internal filter parameters were the same as for previously shown ESVEKF results. Just like for the DEKF, the capacity and SoC estimates converge approximately simultaneously and SoC estimation remains close to the Coulomb counting reference thereafter.
Like for the DEKF, data from the battery RW2 was also examined in the ESVEKF. Also here capacity estimates quickly converge and the algorithm appears relatively robust to initialization. Of course, unlike for the DEKF, initial values of internal resistance parameters and capacitance were required. Again, the method based the MATLAB function “fminsearch”, as outlined in section 3.1.3, was applied to identify these values. The measured ECM parameters of battery RW2 before cycling began were:

<table>
<thead>
<tr>
<th>R_0</th>
<th>R_1</th>
<th>C_1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.099 Ω</td>
<td>0.055 Ω</td>
<td>1447 F</td>
</tr>
</tbody>
</table>

Figure 62: Convergence of SoC using the ESVEKF despite poor guesses of both SoC and capacity.

Figure 63: Rapid convergence of capacity estimates with three different initial guesses using ESVEKF and data from battery RW2
It is seen in figure 63 that convergence is very rapid and virtually independent of the initial guess of capacity, which were 3.0 Ah, 2.0 Ah, and 1.0 Ah here, respectively. Like for the DEKF, capacity estimates are initially relatively noisy, but then stabilize quite well. Again, the ESVEKF was found to be more robust with regards to initialization than the DEKF, perhaps simply since there are fewer internal filter parameters which to manipulate.

It is likely that periodic calibration of the ECM parameter values is necessary for the long-term use of the ESVEKF. This is evident when the output equation of the Kalman filter is considered. Only the OCV-term is adjusted for aging effects in the ESVEKF. The effects of this will be analyzed further in section 4.6.2. Short-term, noting again that the rate of battery aging is highly accelerated in the investigated data compared to for most normal Li-ion battery applications, convergence to measured values was found to be rapid and accuracy of the capacity estimation was good compared to values measured in reference discharge tests. In addition, the filter tuning process was heavily streamlined compared to for the DEKF due to the much smaller number of internal parameters to be considered. The more intuitive filter tuning process might well be a significant advantage in real-life applications.

4.5 SINGLE WEIGHT DUAL EXTENDED KALMAN FILTER

The SWDEKF uses the DEKF algorithm but with a weight vector consisting of only the cell capacity, meaning that neither internal resistance- nor capacitance parameters are adjusted during filtering. The purpose of this truncation of the weight vector was to reduce the computational strain compared to the full DEKF whilst still achieving results of acceptable quality in terms of estimation of cell capacity and, thus, state of health. While the number of variables being adjusted is the same as in the ESVEKF, they are here split into two filters, which should theoretically allow for more accurate tuning of overall estimation accuracy as well as improved computational performance. On the other hand, the increased number of filter tuning parameters makes the tuning process more demanding and, possibly, time consuming. As is evident from the figure below, the cell capacity estimates using this filter could also quickly converge to the measured values despite a poor initialization. Just like for the ESVEKF, the internal resistances and capacitances as measured in the pulsed discharge tests were used for all SWDEKF work.

![Figure 64: Convergence of capacity estimate for SWDEKF despite poor initial guess.](image-url)
It is interesting to note the two apparent plateaus in the capacity estimation, very similar to the DEKF results pictured in figure 49. The following internal filter parameters were used to obtain the results above:

Table 17: Internal filter parameters for SWDEKF results in figure 64.

<table>
<thead>
<tr>
<th>State filter internal parameters</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Starting SoC</td>
<td>0.999</td>
</tr>
<tr>
<td>Starting V₁</td>
<td>0.0 V</td>
</tr>
<tr>
<td>Covariance, SoC</td>
<td>$1 \times 10^{-8}$</td>
</tr>
<tr>
<td>Covariance, V₁</td>
<td>$1 \times 10^{-4}$ V²</td>
</tr>
<tr>
<td>Process noise, SoC</td>
<td>$1 \times 10^{-15}$</td>
</tr>
<tr>
<td>Process noise, V₁</td>
<td>$1 \times 10^{-9}$ V²</td>
</tr>
<tr>
<td>Measurement noise, state filter</td>
<td>$5 \times 10^{-6}$ V²</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Weight filter internal parameters</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Starting capacity</td>
<td>1.9 Ah</td>
</tr>
<tr>
<td>Covariance, capacity</td>
<td>$5 \times 10^{-6}$ (Ah)²</td>
</tr>
<tr>
<td>Process noise, capacity</td>
<td>$1 \times 10^{-10}$ (Ah)²</td>
</tr>
<tr>
<td>Measurement noise, weight filter</td>
<td>$1 \times 10^{-3}$ V²</td>
</tr>
</tbody>
</table>

As was the case for the DEKF, balancing the weight- and state filter internal filter parameters was found to be critical for the overall performance of the SWDEKF. In fact, the overall behavior of the filter was very similar to that of the DEKF. Rate of convergence to measured capacity values was quite rapid for a wide array of starting guesses. The figure below displays the convergence behavior of SWDEKF capacity estimates for a wide array of starting guesses of cell capacity for the start of the first RW phase of battery RW9.

![Figure 65: Convergence of capacity estimates to measured value for RW9 for different starting guesses in SWDEKF.](image)

It can be seen when comparing figures 65 and 61 that the ESVEKF and the SWDEKF appear to converge at approximately the same rate, around three hours into the RW phase. The spread of estimated capacities for the different starting guesses is slightly wider for the SWDEKF. This might however be an issue of filter tuning and could possibly be improved by enough trial-and-error testing of internal filter parameters.
Again, the cycling data from battery RW2 was also used with the SWDEKF. While results were found to be like those for the DEKF and the ESVEKF, convergence was found to be generally slower, as is evident in the figure below.

Unlike for the other two investigated capacity estimation algorithms, this slow convergence behavior was not helped by increasing the initial capacity covariance. This would only have the effect of introducing more turbulent behavior and spiking at the start of filtering, actually slowing down convergence compared to when using a relatively low initial capacity covariance such as in figure 66.
In terms of long-term capacity estimation, the same concerns as for the ESVEKF apply. The results of short-term capacity estimation simulations where overall slightly worse for this algorithm than for ESVEKF in terms of accuracy and convergence rate. Again, this might be due to filter tuning issues however. The higher number of internal filter parameters means that finding appropriate values of these was more difficult than for the ESVEKF.

4.6 COMPARISON OF ALGORITHMS AND SUMMARY

4.6.1 State of charge algorithms
The two investigated SoC estimation algorithms both performed very well in terms of accuracy to the Coulomb counting reference. To compare the accuracy of SoC estimates quantitatively, internal filter parameters were set up identically for the two filters, with the starting values of process- and measurement noise covariance matrices for the AEKF being the same as the equivalent constant values in the EKF. The filters were then used to simulate the 300 first RW steps of the first RW phase of battery RW9, which is made up of approximately 70 000 data points. The 300 first RW steps were chosen for this comparison as this amount of data allows for differences in accuracy between the algorithms to be clearly distinguishable but it still early enough in the RW phase to be able to reasonably assume that the values of ECM parameters are roughly the same as before cycling began, meaning that Coulomb counting represents a good reference for SoC estimates. Choices of internal parameters for the two filters are shown in the table below:

<table>
<thead>
<tr>
<th>State filter internal parameters</th>
<th>EKF</th>
<th>AEKF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Starting SoC</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>Starting V₁</td>
<td>0.0 V</td>
<td>0.0 V</td>
</tr>
<tr>
<td>Covariance, SoC</td>
<td>1·10⁻²</td>
<td>1·10⁻²</td>
</tr>
<tr>
<td>Covariance, V₁</td>
<td>1·10⁻³V²</td>
<td>1·10⁻³V²</td>
</tr>
<tr>
<td>(Starting) Process noise, SoC</td>
<td>1·10⁻¹⁰</td>
<td>1·10⁻¹⁰</td>
</tr>
<tr>
<td>(Starting) Process noise, V₁</td>
<td>1·10⁻⁴V²</td>
<td>1·10⁻⁴V²</td>
</tr>
<tr>
<td>(Starting) measurement noise</td>
<td>5·10⁻³ V²</td>
<td>5·10⁻³ V²</td>
</tr>
<tr>
<td>Moving estimation window size, M</td>
<td>-</td>
<td>10</td>
</tr>
</tbody>
</table>

In addition, artificial random noise was added to measured cell voltage and current signals on the form of (68) with standard deviations of 5·10⁻³ V and 5·10⁻³ A. ECM parameters were set as measured values in pulsed- and reference discharge tests before the first RW phase of RW9 for both compared algorithms. This setup produced the following results:

<table>
<thead>
<tr>
<th></th>
<th>EKF</th>
<th>AEKF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Root-mean-square SoC error</td>
<td>0.0142</td>
<td>0.0176</td>
</tr>
<tr>
<td>Execution time (s)</td>
<td>5.9</td>
<td>7.5</td>
</tr>
</tbody>
</table>

Looking at table 19, the results point towards the EKF being the superior alternative, both producing a more accurate result and being faster to execute. What is not included in table 19 however is the time needed for filter tuning. This process was much easier for the AEKF as parameters for process- and measurement noise covariance are updated recursively during filtering, meaning that initialization was less sensitive. Of course, the window size needs to be set in the AEKF as well, but as discussed above this was not a major issue and its value had relatively little influence on the accuracy of estimation.

---

4 Compared to the Coulomb counting reference.
results. It may well turn out to be advantageous to apply an AEKF in situations where initialization is difficult to rationalize or where earlier operation data is scarce. For the purposes of this work the added accuracy and speed of the EKF was prioritized higher than robustness to initialization however, especially considering how much emphasis was put on characterizing filter behavior and filter tuning.

4.6.2 State of health estimation algorithms

It is clear that all investigated SoH estimation algorithms were capable of estimating cell capacity during online operation. This was the case even if initial capacity guesses were very inaccurate, which would result in all algorithms relatively quickly converging to measured values of capacity, given that all other ECM parameters were correctly set in accordance with measured values. The DEKF also had the advantage of being able to estimate the internal resistance parameters of the cell, which should increase accuracy of estimations both short- and long term due to the increased model accuracy. As all model parameters, see table 8 for the current case of a 1 RC ECM, change slowly over time as the battery ages, the accuracy of the ESVEKF and SWDEKF should falter over longer periods. In addition, internal resistance is known to be a function of both temperature and current which theoretically should allow for the DEKF to be more accurate over shorter time periods as well. This is especially the case if conditions during measurements of model parameters are substantially different than during operation such as the case for capacity measurements of battery RW9.

One recurring issue when discussing SoH estimation methods is the question of validation. In the case of SoC estimation, this can be measured accurately under certain operation conditions using laboratory equipment by Coulomb counting, at least over shorter periods of time when cell capacity can be assumed to remain constant. In other words, a clear reference point exists for which to aim. Such a reference method does not exist for online capacity- and/or internal resistance estimation as these quantities can only be calculated using data taken from special operation modes, such as the pulsed- and reference discharge tests used in this work. Two methods of validation, as outlined in section 3.2, were used in this work, utilizing the fact that cell capacity was measured in reference discharge tests before and after any given RW phase. Using both methods of validation, all algorithms could estimate the capacity changes over a full, intense usage period of over 90 hours for battery RW9, as well as the over longer time periods. It was found that differences in SoH estimation between the investigated algorithms became more significant for longer simulation periods.

To try to quantitatively compare the algorithms, a case study consisting of the long-term capacity estimation for battery RW2 was considered. The purpose of this test is to observe and compare the long-term behavior of the filters in a more realistic prolonged battery monitoring situation. The reason for using data from this battery and not RW9 was due to the more stable estimation results obtained using RW2. The reasons for this higher stability, or robustness with regards to initialization, are believed to be the prolonged periods of charging and lack of points of negative SoC, as calculated by Coulomb counting, in the data. The internal filter parameters of the DEKF and SWDEKF were set identically in terms of the shared internal filter parameters for the results below to try to make the comparison as fair as possible. The internal filter parameters of the algorithms for this long-term test are summarized in table 20 below. The estimated capacities of RW2 using the three algorithms for data recorded over more than 1200 hours of RW cycling are shown in the figure below.
It is of course the case that filter tuning will affect the results in figure 68. The results should be considered merely representative of a typical situation and have not been optimized with regards to filter tuning. It is seen that DEKF performs the best in terms of accuracy, followed by SWDEKF and then ESVEKF. Root-mean-square errors of the SoH estimates for the algorithms are shown in table 21. The error was defined as the difference between the measured and estimated capacities at the end of a given RW phase. The starting capacity of RW2 was 1.98 Ah, meaning that the measured SoH at the end of the data in figure 68 is 77%. The test above thus represents more than a typical full Li-ion battery lifetime in an automotive application.
Table 21: Root-mean-square capacity estimation errors for results in figure 65 for investigated SoH estimation algorithms.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>RMS error with regards to capacity as measured in reference discharge tests</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEKF</td>
<td>2.2 %</td>
</tr>
<tr>
<td>SWDEKF</td>
<td>8.3 %</td>
</tr>
<tr>
<td>ESVEKF</td>
<td>10.4 %</td>
</tr>
</tbody>
</table>

The tests of long-term capacity estimation above were performed in the same way for all compared algorithms. The filters were initialized with the ECM parameter values and cell capacity as measured for RW2 in reference- and pulsed discharge tests before cycling began. These initial ECM parameter values may be found in table 16 above. The filters were then used to simulate the entire RW phases, that is until the next batch of cell diagnostic tests were performed. The estimated values of cell capacity at the end of said RW phases were then used as initialization values for the simulation of the subsequent RW phase. In addition, the initialization values of internal resistance parameters $R_0$ and $R_1$ were also updated in the same way as the capacity for the DEKF. The filters which do not adapt internal resistance values during the filtering process, the ESVEKF and SWDEKF, used the initial values of these parameters throughout testing for fairness of comparison.

As can be seen in figure 68, all algorithms perform similarly in terms of accuracy of estimation to measured capacities at the start of the testing, especially the ESVEKF and the DEKF. The difference in error compared to the measured values of capacity was indeed much smaller over the first four data points, measured over more than 500 hours of simulated operation, than during the full test for all algorithms:

Table 22: Root-mean-square error over first four measurement points of long-term capacity estimation test.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>RMS error with regards to capacity as measured in first four reference discharge tests</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEKF</td>
<td>1.3 %</td>
</tr>
<tr>
<td>SWDEKF</td>
<td>5.9 %</td>
</tr>
<tr>
<td>ESVEKF</td>
<td>1.9 %</td>
</tr>
</tbody>
</table>

For later data points, the capacity estimation accuracy of especially the algorithms not adjusting for changes in internal resistance suffers. That said, the accuracy of the DEKF capacity estimate also appears to decrease towards the later RW phases. While the lack of adjustment of the capacitance parameter in any investigated algorithm likely plays a role, temperature effects might also be important. It should however be noted that the change in average temperature during cycling over time is much smaller for battery RW2 than it was for RW9. This is likely due to the lower average current applied during charging of RW2 using the CC-CV mode as Joule heating (see (3)) is proportional to the applied current squared.

At first glance it might appear odd that not adjusting for changes to internal resistance parameters would result in too high capacity estimates for later data points, as seen for the ESVEKF and SWDEKF in figure 68. Looking at the output equation the opposite effect seems expected. When internal resistance parameters $R_0$ and $R_1$ are not adjusted to account for aging effects they will quickly be smaller than the values one would measure for the real cell in a pulsed discharge test or estimate with the DEKF. Disregarding changes to the capacitance parameter due to aging, the absolute values of the terms in the output equation which are being multiplied by $R_0$ and $R_1$ will both be too small. For illustrations sake, the output equation is rewritten as such by utilizing (14) and (15):
\[ OCV(SOC) = V_t + R_o \cdot I + R_1 \cdot I \cdot \left(1 - e^{-\frac{\Delta t}{C}}\right) + \text{(term not dependent on } I) \] (69)

Given that all Kalman filter algorithms will adjust the state- and, in the case of the DEKF and SWDEKF at least, the weight vector to make the predicted terminal voltage \( V_t \) be as close to the measured value as possible in each time step, the terminal voltage term in (69) may be seen as fixed in this discussion. Now consider that the situation can be divided into two cases: during discharge (current is positive) and during charge (current is negative). For discharge with too low, unadjusted values of \( R_0 \) and \( R_1 \), it is seen that the estimated OCV would need to be lower for the same cell voltage \( V_t \) to be predicted. This is of course equivalent to a lower estimated SoC. During discharge this would mean that the cell capacity would need to be lower to achieve a lower SoC for a given discharge current and duration. For the other case, that of charging the cell, a similar effect should be expected. For non-adjusted values of \( R_0 \) and \( R_1 \), the OCV, and thus SoC, would need to be higher for the same given terminal voltage prediction. For charging this is equivalent to a larger positive change in SoC for the same current and time, meaning a smaller capacity again be expected to be predicted here as well. So why is the opposite effect observed, that is capacity estimates of the algorithms that do not adjust for changes to internal resistance parameters becoming increasingly too large over time?

The explanation is believed to lie in the allocation of modelling error in the state process noise covariance matrix. As mentioned in the discussion of the EKF, it is intuitive to consider the estimate of the transient voltage \( V_t \) to be less accurate than that of the SoC and to thusly reflect this in the process noise covariance matrix by specifying a larger value for the transient voltage covariance. This appears to be fine for the case of a relatively accurate model with accurately specified parameters. In the case of the ESVEKF and SWDEKF, where the internal resistance parameters are not adjusted over time, this will however have the effect of “swallowing up” errors of the SoC- and capacity estimation as modeling error increases. Essentially, the modelling error of the \( V_t \)-term will increase much faster compared to the OCV-term over time when changes to internal resistance parameters are not considered. Examine the output equation rewritten yet again:

\[ V_t + R_0 \cdot I = OCV(SOC) - V_t \] (70)

For discharge the left side of (70) will be too small and during charging it will be too large because of the error in \( R_0 \). Due to the process noise covariance being set to a larger value for \( V_t \) than for SoC, it will be adjusted more to make up the difference to \( V_t \). What is observed then is a smaller change in OCV for a given current over a certain time, leading to a larger cell capacity being predicted, and thus explaining the observed effect. One could consider it a misallocation of the error due to the balance of modeling error being shifted unevenly as the battery ages. This might also explain why the same effect is seen to a smaller extent in the DEKF as well as it still is a question of \( V_t \) becoming increasingly inaccurate over time for this algorithm, but here due to an inaccurate value of the capacitance parameter. Incorporating the capacitance into the weight vector of the DEKF might improve long-term capacity estimates for this reason.

Overcoming this type of issue within the context of the ESVEKF and SWDEKF algorithms appears difficult, even though the effects could likely be made less impactful through careful filter tuning. The issue of the shifting of modeling error seems to be an inherent problem of all Kalman filtering algorithms. Perhaps some adaptive solution for online estimation of process noise could be the solution as the problem appears to stem from the assignment of constant process noise covariance matrices. It might also be worth considering the approaches suggested by Zou et al. and Hu et al. which incorporate a difference in time scales between the state and weight filters.
While the DEKF displayed the most accurate results both long-term and short-term, it was also the most computationally taxing algorithm. A comparison of the three SoH estimation methods in terms of execution time is shown in the table below.

**Table 23: Execution times of SoH estimation algorithms for simulations of 300 RW cycles.**

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Execution time (s)</th>
<th>Percentage improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEKF</td>
<td>10.7</td>
<td>0 %</td>
</tr>
<tr>
<td>ESVEKF</td>
<td>6.5</td>
<td>39 %</td>
</tr>
<tr>
<td>SWDEKF</td>
<td>8.5</td>
<td>21 %</td>
</tr>
</tbody>
</table>

The algorithm execution time was recorded using the “timeit” MATLAB function for a simulation over the 300 first RW steps of the first RW phase of battery RW9. The displayed values were the average execution time of one hundred such simulations. There is clearly a substantial difference in algorithm computational performance. It is interesting to note that dividing the burden into two filters, such as in the SWDEKF, instead of combining into one larger filter, such as in the ESVEKF, does not appear to be faster, despite the reduced matrix dimensions as discussed in section 3.3.5. For the application investigated in this study, it appears that the calculation of the recursive derivatives in the SWDEKF outweighs the reduced strain of matrix operations. This seems to be confirmed by the relatively small difference in execution time between the SWDEKF and the DEKF as well. It is thought that the computational advantages of the dual filter approach would be more evident for larger state- and/or weight vectors. Some general observations when studying the SoH estimation algorithms will now be discussed.

One recurring phenomena discovered when trying to estimate the cell capacity in an online mode is the need for relatively quick convergence. It was observed that when intentionally providing the filter with inaccurate initial guesses of cell capacity, the filters would often diverge unless relatively quick convergence to measured values was ensured by adjusting the initial capacity covariance parameter. This effect was found to be much more prominent when using the data from battery RW9 than when using the CC-CV mode charging data from RW2. The reason for this need for quick convergence is likely the accumulative nature of Coulomb counting, as it is featured in the state equation of all filters. If the initial guess of the cell capacity is lower than the measured value, the filter SoC estimate will get further and further away from the Coulomb counting reference SoC values calculated using the measured capacity during a given period of constant current charge or discharge such as during the RW phases of battery RW9. For the Kalman filter SoC estimate to approach the reference values the capacity estimate would now need to assume a higher value than what was measured. This can be explained mathematically by noting that one divided by the capacity will be directly proportional to the slope of the SoC-time curve when current is held constant, meaning that the sum of all currents over said period is also a constant:

$$SOC_{k+1} = a + b \cdot t = SOC_k + \frac{\sum_k I_k \cdot \Delta t_k}{C} = SOC_k + \left(\frac{I_{sum}}{C}\right) \cdot t_{sum}$$  \(71\)

This overcompensation of the capacity estimate might cause serious convergence issues and should preferably be avoided by ensuring either that the initial capacity estimate is reasonably accurate or by careful tuning of the initial capacity covariance. If filter tuning is reasonable, capacity estimates for cases of a faulty initial guess of the capacity are therefore expected to oscillate to some extent before converging. This principle is only strictly applicable for cases of periods of constant current however. In the data used in this work, the applied current was very dynamic, meaning that this effect should
be reduced. This is because the SoC estimated with a faulty capacity might “accidentally” coincide with the correct SoC value when the applied current changes. This principle is shown in the figure below.

![Figure 69: Principle of “accidental” convergence of SoC estimation for faulty concurrent estimation of capacity. The red line represents an SoC estimate with a low capacity, the green line high capacity, and the blue line is the in-between case.](image)

In figure 69 the blue line represents the “true” SoC, which could for instance be represented by Coulomb counting using a correct capacity. The red line represents a concurrent estimate of SoC and cell capacity where the capacity is too low and the green line the same situation but with the capacity being estimated as a too high value. During the initial 15 minutes of constant current charging, the SoC estimates are diverging due to the variations in capacity. As the current changes to a discharging mode, as after 15 minutes in the figure above, the SoC estimates will however tend to converge again. Note that time and SoC in figure 69 are only there for the sake of illustration of the discussed principle and do not represent realistic values.

This overcompensation phenomenon is also likely part of the reason why estimation of internal resistance parameters appeared more robust than estimating cell capacity as the cell voltage response to changes of these parameters is immediate, as seen in figure 27. It also means that it is more difficult to estimate the cell capacity under conditions of rapidly varying current. This is because changes in cell capacity become less noticeable per the principle shown in figure 69. In short, there is not enough time for the inaccurate slope of the SoC-time curve to cause the SoC estimate to drift substantially from the correct value, given that the starting SoC is reasonably accurate. And of course, it is only errors in SoC, leading to errors in OCV, which are processed by the Kalman filter algorithm to update estimates of state and system weights. This would also partly explain the observed more stable capacity estimation behavior when using data from battery RW2. The relatively long periods of charging in a CC-CV mode provides the algorithm enough time to accurately adapt the capacity estimation.

Finding the suitable internal parameters was cumbersome for the SWDEKF. The algorithm lacks the simplicity of the ESVEKF and the tunability of the full DEKF. As modelling errors inevitably increase during cycling in the RW phases, all adjustments must be included in the capacity estimation. This meant that estimates tended to either be very noisy, perhaps due to the algorithm including short term variations in internal resistance into the capacity estimate, or to change too quickly over time unless choices internal parameters were considered very carefully. It appeared that this issue was of
less concern for the ESVEKF which produced more stable results. This is believed to be due to the capacity- and SoC estimates sharing the same covariance matrix due to the joint estimation approach. While initializing state covariance matrices were always chosen to be diagonal this property would quickly be lost during the process of filtering due to the state covariance measurement update equation (28). The covariance of the SoC- and capacity estimate, meaning the value of positions (2,3) and (3,2) in the state covariance matrix of the ESVEKF, might be a very important parameter for accurate and robust capacity estimation. This is of course missing in both dual filter approaches. The loss of information on relevant cross-correlations between state- and weight estimations in the dual filter approaches was also speculated to possibly decrease overall algorithm accuracy by Plett [150].

Also, likely due to the expected increase of process noise during operation as modelling errors become more severe, which is of course very difficult to predict quantitatively, it was found to be much easier to achieve reasonably accurate short-term capacity estimates for all investigated SoH estimation algorithms. More specifically, towards the beginning of a given RW phase. This was especially evident when using the algorithms that just adjusted the cell capacity. Quick convergence of the capacity estimate could be ensured, as discussed above, quite handily, but all the capacity estimates of suggested algorithms tended to drift over longer time periods unless the filter parameters were tuned in hindsight. Again, the ESVEKF seemed to perform well in this regard and would converge very quickly to measured values in these types of scenarios were all ECM parameters except the cell capacity were set to the latest measured values.

Looking at filter tuning more generally, it is easy to confirm the conclusion of Campestrini et al. that it is essentially important for estimation results of Kalman filtering approaches [130]. While mostly algorithms for SoC estimation were investigated in that work, it has been seen here that filter tuning is as, if not even more, important for SoH estimation as well. This was most striking for the dual filter approaches, the DEKF and the SWDEKF. Not only do both filters involve many internal filter parameters, they also require the “balancing” of the two filters. Because the state- and weight filters in the DEKF and SWDEKF share the same output equation, it becomes very easy to overcompensate for errors in the state filter by excessive adjusting of the weight filter. In this work, the balancing of the two filters to achieve realistic results was performed essentially by combining known parameter values from literature and trial-and-error, utilizing visual inspection. This appears inefficient and the filter tuning should be studied further to find a way of streamlining the process. As the ESVEKF only utilized a single filter it did not suffer from these issues of difficult filter tuning to the same extent. As it stands, this is a major advantage of the ESVEKF, especially considering that results were comparable to those obtained with DEKF and SWDEKF, at least for short-term estimation.

Looking at data from battery RW2, where charging was done in a CC-CV mode, is very interesting. While this mode of operation certainly can be considered less realistic than what was used for cycling of battery RW9, it does allow for control of capacity estimation results through the points of 100 % SoC. Firstly, it appears that results for all filters were more stable for RW2. It is suspected that this is at least partly due to the lack of negative SoCs, as discussed briefly in section 3.2, but perhaps also because the relatively long charging periods allow for capacity estimates to stabilize, as mentioned in the discussion of (67). The issues of negative SoCs for RW9 could possibly be resolved by integrating a temperature- and/or current dependence for the cell capacity. As, again, the data from RW9 attempts to resemble a realistic scenario, it appears that establishing these types of functional dependencies is of high importance, not only for the accuracy, but also for the stability of monitoring algorithms. The higher stability of results is evident in the apparent ability of capacity estimates to converge slowly to the right values, thus seeming to at least partly avoid the issues of having to ensure quick convergence as discussed above.
5 CONCLUSIONS AND FUTURE WORK

Two methods of SoC estimation and three methods of SoH estimation were examined in this work. All were based on Kalman filtering approaches. Both investigated SoC estimation algorithms, the extended Kalman filter and the adaptive extended Kalman filter, performed similarly in terms of accuracy to a Coulomb counting reference. Both were also proven to be able to overcome most of the commonly cited shortcomings of Coulomb counting. It was however found that the extended Kalman filter could be made to achieve both a slightly higher accuracy and faster computational performance if internal filter parameters were adjusted appropriately. As the process of finding suitable values for these internal filter parameters was rather painless in the case of the EKF this algorithm was found to be more promising than the AEKF for the application of Li-ion battery SoC estimation. It was however noted that the accuracy of SoC estimation will quickly degrade no matter what regression algorithm was used if the cell capacity was not adjusted to account for effects of battery aging.

The SoH algorithms were all able to accurately estimate the cell capacity over short time scales. The ESVEKF was found to be the easiest to use from a filter tuning perspective due to the lower number of internal filter parameters and was also found to be slightly faster than the dual filter based SWDEKF. In addition, the ESVEKF capacity estimate could be made to converge the fastest to measured capacities of all algorithms. In terms of long-term, continuous capacity estimation the DEKF algorithm performed the best of all investigated algorithms by far. It is believed that this is due to this algorithm also adjusting for changes to internal resistance parameters as the battery ages. The impact of filter tuning on estimation of especially cell capacity was found to be especially substantial, much more so than for online estimation of internal resistance, another common state of health indicator. The difficulty of achieving robust capacity estimation is believed to be due to the entangled effects of the state of charge and the cell capacity on the cell voltage but this needs to be investigated further. Time-scale separation of filters or application of a separate output equation for the weight filter appears promising routes to explore in this regard. The relative ease of estimating internal resistance parameters also promotes the idea of developing some more general idea of state of health and thereby finding some correlation between internal resistance gain and capacity fade.

As this work was only concerned with simulations, real online applications must be explored further. This includes integration of algorithms into some sort of BMS with the goal of online capacity- and/or internal resistance monitoring. As these parameters are normally not measurable during actual battery operation, incorporating such an algorithm would allow not only for the estimation of battery SoH, but also allows for the automated collection of monitoring data. Currently, the largest obstacle that still must be overcome in developing a more holistic battery aging model capable of predicting the rate of battery aging as a function of operation conditions, such as temperature and usage profile, is the need for huge amounts of laboratory data. This data needs to be obtained from carefully organized vast test matrices, covering all possible operation scenarios. Due to the slow nature of battery aging this data acquisition is very time consuming. Currently, the answer to this has been to perform so-called accelerated aging tests, which might instead fail to approximate real battery operation faithfully enough. Online, continuous, recording of capacity degradation or internal resistance increase along with the traditional parameters of current, voltage and temperature would allow for the establishing a database. This database could then be used to develop an aging model, perhaps using a methodology based on machine learning principles.

Another important factor to consider is the transition of monitoring algorithms from the cell scale to the battery pack scale. Reports on the aging behavior of entire packs or modules of Li-ion cells are rare and lots of work remains to be done within this area [57].


## References


8. **Bole, B., C.S. Kulkarni, and M. Daigle.** *Adaptation of an electrochemistry-based li-ion battery model to account for deterioration observed under randomized use.* In *Proceedings of Annual Conference of the Prognostics and Health Management Society, Fort Worth, TX, USA*. 2014.


10. **Ciez, R.E. and J.F. Whitacre, The cost of lithium is unlikely to upend the price of Li-ion storage systems.** Journal of Power Sources, 2016. **320**: p. 310-313.

Ahmadi, L., et al., *Energy efficiency of Li-ion battery packs re-used in stationary power applications*. Sustainable Energy Technologies and Assessments, 2014. 8: p. 9-17.


Swierczynski, M., et al., *Selection and Performance-Degradation Modeling of LiMO2/Li4Ti5O12 and LiFePO4/C Battery Cells as Suitable Energy Storage Systems for Grid Integration With Wind


50. Chanson, C., Safety of lithium-ion batteries. The European Association for Advanced Rechargeable Batteries, 2013.


Kim, Y., *Encapsulation of LiNi0.5Co0.2Mn0.3O2 with a thin inorganic electrolyte film to reduce gas evolution in the application of lithium ion batteries*. Physical Chemistry Chemical Physics, 2013. **15**(17): p. 6400-6405.


120. Zhang, C., et al., Identification of dynamic model parameters for lithium-ion batteries used in hybrid electric vehicles. 2010.


7 APPENDICES

The following appendices contain the MATLAB code as used in the thesis to obtain the presented results. For these algorithms to be used, some data needs to be loaded. The battery data set from the NASA Ames Prognostics Center of Excellence called “Randomized Battery Usage Data Set” was used throughout this work and is available here: [https://ti.arc.nasa.gov/tech/dash/pcoe/prognostic-data-repository/](https://ti.arc.nasa.gov/tech/dash/pcoe/prognostic-data-repository/).

APPENDIX 1: CODE FOR PLOTTING AND ORGANIZATION OF BATTERY DIAGNOSTIC DATA

This code was mostly adapted from the MATLAB file provided with the battery data. Capacity is measured for all RW phases. These values were saved as “capacity” for later use in Kalman filter codes. The structure “RWDis” containing all the indices of RW steps was also saved for later use.

```matlab
close all; clear all;
% Change this data file name below to plot different files
load RW9

%% First we show a sample plot of the low current discharge data
steps = data.step; % save steps array to new variable

% search for instances of low current discharge step
inds = []; % initialize index array
for (i=1:length(steps))
    if (strcmp(data.step(i).comment,'low current discharge at 0.04A'))
        inds = [inds, i];
    end
end

% plot data from first low current discharge step
RT = steps(inds(1)).relativeTime/3600; % relative time in hours
V = steps(inds(1)).voltage; % voltage
I = steps(inds(1)).current; % current

figure % plot voltage profile
plot(RT,V)
title('Voltage for low current discharge')
xlabel('Time (h)');
ylabel('Voltage (V)');

figure % plot current profile
plot(RT,I)
title('Current for low current discharge')
ylim([0,1])
xlabel('Time (h)');
ylabel('Current (A)');

%% Next, the constant load profiles that are run after every 50 random walk
% discharge cycles are plotted using

% initialize plot
figure, hold on
xlim([-1,2.5])
ylim([3,4.25])
xlabel('Time (h)');
ylabel('Voltage (V)');
```
title('Reference Discharge Profiles')

for i = 1:length(steps) % search through the array of step structures
    if strcmp(steps(i).comment,'reference discharge')
        RT = steps(i).relativeTime/3600; % relative time in hours
        V = steps(i).voltage; % voltage
        plot(RT,V,'k')
    end
end

%% We can benchmark the battery’s capacity by integrating current over the reference cycles. The next plot shows this capacity measurement vs date.

counter = 0; % initialize counters

for i = 1:length(steps) % search through the array of step structures
    if strcmp(steps(i).comment,'reference discharge')
        counter = counter+1;

        % save the date of this reference discharge step
        date(counter) = datenum(steps(i).date);

        % save the battery capacity measured by integrating current
        capacity(counter) = trapz(steps(i).relativeTime/3600,data.step(i).current);
    end
end

% plot the data
figure
plot(date,capacity,'o')
datetick('x','mmm/yyyy')
xlabel('Date');
ylabel('Capacity (Ah)');
title('Degradation of Measured Capacity')

%% The next code snippet shows how the resting periods after each reference discharge can be grouped with the corresponding reference discharge cycle and added to the plot above.

% initialize Reference Discharge Struct that will hold indexes of reference discharge steps and the rest steps that come immediately before and after
RefDis.repition{1}.indexes = 0;

% initialize loop parameters
counter = 0;
for i = 1:length(data.step) % search through the array of step structures
    if strcmp(steps(i).comment,'reference discharge')
        counter = counter+1;
        if strcmp(steps(i-1).comment,'rest post reference charge')
            RefDis.repition(counter).indexes = [i-1, i];
        else
            RefDis.repition(counter).indexes = i;
        end
    end
    if strcmp(steps(i).comment,'rest post reference discharge')
        RefDis.repition(counter).indexes = [RefDis.repition(counter).indexes, i];
    end
end
end
end

% now plot each of the grouped reference discharge cycles
figure, hold on
for i = 1:length(RefDis.repition)
    % stitch together all of the substeps identified in the repitition
    % field
    step = steps(RefDis.repition{i}.indexes(1));
    % if the first index is a rest period, then make it end at relativeTime == 0
    if(strcmp(step.comment,'rest post reference charge'))
        step.relativeTime = step.relativeTime - step.relativeTime(end);
    end
    for j = 2:length(RefDis.repition{i}.indexes)
        step.relativeTime = [step.relativeTime
        step.relativeTime(end)+steps(RefDis.repition{i}.indexes(j)).relativeTime];
        step.voltage = [step.voltage
        steps(RefDis.repition{i}.indexes(j)).voltage];
        step.current = [step.current
        steps(RefDis.repition{i}.indexes(j)).current];
        step.temperature = [step.temperature
        steps(RefDis.repition{i}.indexes(j)).temperature];
    end
    plot(step.relativeTime/3600,step.voltage,'k')
    title('Reference discharge profiles and post-discharge rest periods')
    xlabel('Time (h)')
    ylabel('Voltage (V)')
end

%% Example Plots of Pulsed Load Charging and Discharging Cycles

PulsedDis.repition{1}.indexes = 0;
counter1 = 0;
counter2 = 0;
% find Pulsed Load discharge characterization cycles
for i = 2:length(data.step)
    if(strcmp(data.step(i).comment,'pulsed load (discharge)')... || strcmp(data.step(i).comment,'pulsed load (rest)'))
        % if just starting the pulsed load cycle
        if(strcmp(data.step(i).comment,'pulsed load (rest)') && ~strcmp(data.step(i-1).comment,'pulsed load (discharge)'))
            counter1 = 1;
            counter2 = counter2+1;
            PulsedDis.repition{counter2}.indexes(counter1) = i;
        else
            counter1 = counter1+1;
            PulsedDis.repition{counter2}.indexes(counter1) = i;
        end
    end
    if(strcmp(data.step(i).comment,'rest post pulsed load or charge'))
        % check to see if this precedes a pulsed load (discharge)
        if(strcmp(data.step(i-1).comment,'pulsed load (discharge)'))
            counter1 = counter1+1;
            PulsedDis.repition{counter2}.indexes(counter1) = i;
        end
    end
end
% now plot each of the grouped pulsed load discharge cycles
figure, hold on
for i = 1:length(PulsedDis.repition)
    % stitch together all of the substeps identified in the repetition
    step = steps(PulsedDis.repition{i}.indexes(1));
    for j = 2:length(PulsedDis.repition{i}.indexes)
        step.relativeTime = [step.relativeTime
                            step.relativeTime(end)+steps(PulsedDis.repition{i}.indexes(j)).relativeTime
                        ];
        steps(PulsedDis.repition{i}.indexes(j)).voltage;
        steps(PulsedDis.repition{i}.indexes(j)).current;
        steps(PulsedDis.repition{i}.indexes(j)).temperature;
    end
    plot(step.relativeTime/3600,step.voltage,'k')
    title('Pulsed Discharge')
    xlabel('Time (h)')
    ylabel('Voltage (V)')
end

%% Identify all random walk steps and plot current, voltage and temperature
%% for chosen steps and phase
RWDis.repition{1}.indexes = 0;
counter1 = 0;
counter2 = 0;
lastStep = -1;
% find indexes of all RW steps
for i = 1:length(steps)
    if (strcmp(steps(i).comment,'discharge (random walk)')||...
        strcmp(steps(i).comment,'rest (random walk)')||...
        strcmp(steps(i).comment,'charge (random walk)'))
        % if there is a break in the step sequence then separate sequence
        % into new list
        if (i-lastStep > 1)
            counter1 = 1;
            counter2 = counter2+1;
            RWDis.repition(counter2).indexes(counter1) = i;
        else
            counter1 = counter1+1;
            RWDis.repition(counter2).indexes(counter1) = i;
        end
        lastStep = i;
    end
end
% identify index sequence for first 50 RW steps from first RW phase
RWsteps=50;
RWphase=1;
inds = RWDis.repition(RWphase).indexes(1):RWDis.repition(RWphase).indexes(RWsteps*2)
% stitch all of the indexed steps together
step = steps(inds(1));
for i = 2:length(inds)
    step.relativeTime = [step.relativeTime
                        step.relativeTime(end)+steps(inds(i)).relativeTime];
    step.voltage = [step.voltage steps(inds(i)).voltage];
    step.current = [step.current steps(inds(i)).current];
    step.temperature = [step.temperature steps(inds(i)).temperature];
end

% plot Voltage
figure
subplot(3,1,1)
plot(step.relativeTime/3600,step.voltage,"k")
title(sprintf('First %d RW steps',i))
xlabel('Time (h)')
ylabel('Voltage (V)')

% plot Current
subplot(3,1,2), hold on
plot(step.relativeTime/3600,step.current,"k")
xlabel('Time (h)')
ylabel('Current (A)')

% plot temperature
subplot(3,1,3), hold on
plot(step.relativeTime/3600,step.temperature,"k")
ylim([15, 40])
xlabel('Time (h)')
ylabel('Temperature (C)')

APPENDIX 2: CODE FOR EXTENDED KALMAN FILTER

close all, clear all
load RW9 % loading battery data file
load RWDIs % structure containing indeces of RW phases and steps in battery data
load capacity % vector containing all capacities for the battery in question measured during reference discharge tests
steps = data.step; % save steps array to new variable
% Data selection
RWsteps=100; % number of RW steps from start of phase
phase=1; % RW phase
inds2 = RWDIs.repetition(phase).indexes(1):RWDIs.repetition(phase).indexes(RWsteps*2);
% Stich all of the indexed steps together
step = steps(inds2(1));
for i = 2:length(inds2)
    step.relativeTime = [step.relativeTime
                         step.relativeTime(end)+steps(inds2(i)).relativeTime]; % time of data points (s)
    step.voltage = [step.voltage steps(inds2(i)).voltage]; % cell voltage (V)
    step.current = [step.current steps(inds2(i)).current]; % current (A)
    step.temperature = [step.temperature steps(inds2(i)).temperature]; % temperature (deg. C)
end
% Get cell capacity for RW phase from reference discharge tests
step.capacity=(capacity(phase*2)+capacity(phase*2 - 1))/2; % average of the two measurements before each RW phase
%% Coulomb counting for SOC reference
for i = 1:length(step.current)
    if i==1
        dt=step.relativeTime(i); % length of time step (s)
        SOC_0=1; % starting SOC value, representing a fully charged battery
    else
        dt=step.relativeTime(i)-step.relativeTime(i-1);
        current=(step.current(i)+step.current(i-1))/2; % average current between timesteps
        SOC_0=SOC(i-1);
    end
    SOC(i)=SOC_0-(current*dt/(step.capacity*3600)); % Coulomb counting.
end

%% Kalman initialization and internal parameters
SOC_0=0.9; % starting guess of SOC
V1_0=0; % starting guess of transient voltage (V1)
x_hat_0=[V1_0; SOC_0]; %Starting state vector x
covx_p=[1e-3 0;0 1e-2]; % initial state covariance matrix
covw=[1e-6 0;0 1e-10]; % process noise covariance matrix
covv=5e-3; % measurement noise covariance (matrix)

%% Adding artificial noise to signal
step.voltage=step.voltage.*(1 + randn(1,length(step.voltage)) * 5e-3 ); % adding artificial random noise to measurement signal

%% ECM parameters from pulsed discharge test
R0=0.074; % Ohm
R1=0.045; % Ohm
C1=815; % Farad

%% Extended Kalman filtering
for i=1:length(step.relativeTime) % filtering over all selected time steps
    %% Time update
    if i==1 % time step
        dt=step.relativeTime(i);
    else
        dt=step.relativeTime(i)-step.relativeTime(i-1);
    end
    A=[ exp(-dt/(R1*C1)) 0 ; 0 1 ]; % calculating A matrix
    B=[ R1 * (1-exp(-dt/(R1*C1))) ; -dt/(step.capacity*3600) ]; % calculating B matrix
    if i==1 % state estimate time update
        x_hat_minus=A*x_hat_0 + B*step.current(i);
    else
        x_hat_minus=A*x_hat_plus(:,i-1) + B*step.current(i-1);
    end
    covx_m=A*covx_p*A' + covw; % state covariance matrix time update
    [OCV, dOCVdSOC]= OCV_SOC(x_hat_minus(2)); % function for the OCV-SOC relationship, also outputs the derivative of this function.
    C(i,:)=[-1, dOCVdSOC]; % calculating C matrix
    Yest(i)=OCV - R0*step.current(i) - x_hat_minus(1); % output equation
    %% Measurement update
    err(i)=step.voltage(i)-Yest(i); % error of predicted cell voltage
    L(:,i)=covx_m*C(i,:)'*( (C(i,:)*covx_m*C(i,:)' + covv )^(-1); % calculating Kalman gain matrix
    x_hat_plus(:,i)=x_hat_minus + L(:,i)*err(i); % state estimate measurement update
    covx_p=( eye(2) - L(:,i)*C(i,:) )*covx_m; % state covariance matrix measurement update
end

%% Plotting results
figure % plot of Kalman SOC estimation in comparison to Coulomb counting SOC
APPENDIX 3: CODE FOR ADAPTIVE EXTENDED KALMAN FILTER

close all, clear all, clc
load RW9 % loading battery data file
load RWDis % structure containing indeces of RW phases and steps in battery data
load capacity % vector containing all capacities for the battery in question measured during reference discharge tests
steps = data.step; % save steps array to new variable
% Data selection
RWsteps=100; % number of RW steps from start of phase
phase=1; % RW phase
inds2 = RWDis.repetition{phase}.indexes(1):RWDis.repetition{phase}.indexes(RWsteps*2);
% Stich all of the indexed steps together
step = steps(inds2(1));
for i = 2:length(inds2)
    step.relativeTime = [step.relativeTime step.relativeTime(end)+steps(inds2(i)).relativeTime]; % time of data points (s)
    step.voltage = [step.voltage steps(inds2(i)).voltage]; % cell voltage (V)
    step.current = [step.current steps(inds2(i)).current]; % current (A)
    step.temperature = [step.temperature steps(inds2(i)).temperature]; % temperature (deg. C)
end
% Get cell capacity for RW phase from reference discharge tests
step.capacity=(capacity(phase*2)+capacity(phase*2-1))/2; % average of the two measurements before each RW phase
% Coulomb counting for SOC reference
for i = 1:length(step.current)
    if i==1
        dt=step.relativeTime(i); % length of time step (s)
        SOC_0=1; % starting SOC value, representing a fully charged battery
    else
        dt=step.relativeTime(i)-step.relativeTime(i-1);
        current=(step.current(i)+step.current(i-1))/2; % average current between timesteps
        SOC_0=SOC(i-1);
    end
    SOC(i)=SOC_0-(current*dt/(step.capacity*3600)); % Coulomb counting. Conversion factor 3600 to convert capacities to unit A*s from A*h
end
% Kalman initialization.
% Kalman initialization and internal parameters
SOC_0=0.9; % starting guess of SOC
V1_0=0; % starting guess of transient voltage (V1)
x_hat_0=[V1_0; SOC_0]; %Starting state vector x
covx_p=[1e-3 0; 0 1e-2]; % initial state covariance matrix
M=10; % moving estimation window size for AEKF
Q_0=[1e-2 0; 0 1e-2]; % starting process noise covariance matrix
R_0=1e-2; % starting measurement noise covariance
step.voltage=step.voltage.*(1 + randn(1,length(step.voltage)) * 5e-3); % adding artificial random noise to measurement signal
R0=0.074; % Ohm
R1=0.045; % Ohm
C1=815; % Farad

%% Kalman filtering
for i=1:length(step.voltage)
    if i==1 % time step
        dt=step.relativeTime(i);
    else
        dt=step.relativeTime(i)-step.relativeTime(i-1);
    end
    A=[ exp(-dt/(R1*C1)) 0 ; 0 1 ];
    B=[ R1 * (1-exp(-dt/(R1*C1))) ; -dt/(step.capacity*3600) ];

    if i==1
        x_hat_minus=A*x_hat_0 + B*I(i); % time update state estimate
        covx_m=A*covx_p*A' + Q_0; % time update state covariance matrix
    else
        x_hat_minus=A*x_hat_plus(:,i-1) + B*I(i-1);
        covx_m=A*covx_p*A' + Q(:,:,i-1);
    end
    [OCV, dOCVdSOC]= OCV_SOC(x_hat_minus(2)); % function for the OCV-SOC relationship, also outputs the derivative of this function
    C(i,:)=[-1, dOCVdSOC ]; % calculating C matrix
    Yest(i)=OCV - R0*I(i) - x_hat_minus(1); % output equation
    err(i)=Y(i)-Yest(i); % error of predicted cell voltage

    if i<=M
        R(i)=R_0;
    else
        H(i)=(1/M)*sum(err(i-M+1:i).^2);
        R(i)=H(i) + C(i,:)*covx_m*C(i,:); 
    end

    % Kalman gain and measurement updates to x and its covariance
    if i==1
        L(:,i)=( covx_m*C(i,:)')*( (C(i,:)*covx_m*C(i,:)') + R_0 )^-1; % Kalman gain matrix
        x_hat_plus(:,i)=x_hat_minus +L(:,i)*err(i); % measurement state update
        covx_p=( eye(2) - L(:,i)*C(i,:) )*covx_m;
    else
        L(:,i)=( covx_m*C(i,:)')*( (C(i,:)*covx_m*C(i,:)') + R(i ) )^-1; % Kalman gain matrix
        x_hat_plus(:,i)=x_hat_minus +L(:,i)*err(i); % measurement state update
        covx_p=( eye(2) - L(:,i)*C(i,:) )*covx_m;
    end

    % Updating process noise covariance
    if i<=M
        Q(:,:,i)=Q_0;
    else
        Q(:,:,i)=L(:,i)*H(i)*L(:,i)';
    end
end

%% plotting results
figure % plot of Kalman SOC estimation in comparison to Coulomb counting
A  P  P  E  N  D  I  X  4:  C  O  D  E  F  O  R  D  U  A  L  E  X  T  E  N  D  E  D  K  A  L  M  A  N  F  I  L  T  E  R

```matlab
close all, clear all
load RW9 % loading battery data file
load RWDis % structure containing indeces of RW phases and steps in battery data
load capacity % vector containing all capacities for the battery in question measured during reference discharge tests
steps = data.step; % save steps array to new variable
% Data selection
RWsteps=100; % number of RW steps from start of phase
phase=1; % RW phase
inds2 = RWDis.repition{phase}.indexes(1):RWDis.repition{phase}.indexes(RWsteps*2);
% stich all of the indexed steps together
step = steps(inds2(1));
for i = 2:length(inds2)
    step.relativeTime = [step.relativeTime step.relativeTime(end)+steps(inds2(i)).relativeTime]; % time of data points (s)
    step.voltage = [step.voltage steps(inds2(i)).voltage]; % cell voltage (V)
    step.current = [step.current steps(inds2(i)).current]; % current (A)
    step.temperature = [step.temperature steps(inds2(i)).temperature]; % temperature (deg. C)
end
% get cell capacity for RW phase from reference discharge tests
step.capacity=(capacity(phase*2)+capacity(phase*2-1))/2; % used cell capacity is average of the two measurements before each RW phase
% Coulomb counting for SOC reference
for i = 1:length(step.current)
    if i==1
        dt=step.relativeTime(i);
        I=step.current(i);
        SOC_0=1; % starting value
    else
        dt=step.relativeTime(i)-step.relativeTime(i-1);
        I=(step.current(i)+step.current(i-1))/2; % average current between timesteps
        SOC_0=SOC(0)-((I*dt/(step.capacity*3600))); % adjusted capacity for temperature according to Farmann, 2015: 7.7% higher capacity per 30 deg C, average temp. of capacity measurement: 24.96 deg C, average temp. of 50 first RW cycles: 31.94 deg C.
    end
end
% Dual Kalman filter initialization
% State filter internal paramters
SOC_0=0.999; % starting guess of SOC
V1_0=0; % starting guess of transient voltage (V1)
x_hat_0=[ V1_0 ; SOC_0 ]; % initial state vector
covx_plus=[1e-5 0;0 1e-10]; % initial state covariance matrix for state filter
```
\( \text{sigma} = 5\text{e}^{-3} \); measurement noise covariance (matrix) for state filter
\( \text{covw} = \begin{bmatrix} 1\text{e}^{-8} & 0 \\ 0 & 1\text{e}^{-10} \end{bmatrix} \); process noise covariance matrix state filter

\% weight filter internal parameters
\( \text{cap}_0 = 1.9 \); starting guess of cell capacity
\( R_0 = 0.074 \); starting guess of \( R_0 \)
\( R_1 = 0.045 \); starting guess of \( R_1 \)
\( \text{covohm} = \begin{bmatrix} 1\text{e}^{-9} & 0 \\ 0 & 1\text{e}^{-10} \end{bmatrix} \); initial state covariance matrix for weight filter

\( \text{sigmav} = 2 \); measurement noise covariance (matrix) for weight filter
\( \text{covr} = \begin{bmatrix} 1\text{e}^{-11} & 0 & 0 \\ 0 & 1\text{e}^{-11} & 0 \\ 0 & 0 & 1\text{e}^{-9} \end{bmatrix} \); process noise covariance matrix for weight filter

\% weight filter internal parameters
\( \text{cap}_0 = 1.9 \); starting guess of cell capacity
\( R_0 = 0.074 \); starting guess of \( R_0 \)
\( R_1 = 0.045 \); starting guess of \( R_1 \)
\( \text{covohm} = \begin{bmatrix} 1\text{e}^{-9} & 0 & 0 \\ 0 & 1\text{e}^{-9} & 0 \\ 0 & 0 & 1\text{e}^{-4} \end{bmatrix} \); initial state covariance matrix for weight filter
\( \text{sigmae} = 1\text{e}^{-2} \); measurement noise covariance (matrix) for weight filter
\( \text{covr} = \begin{bmatrix} 1\text{e}^{-11} & 0 & 0 \\ 0 & 1\text{e}^{-11} & 0 \\ 0 & 0 & 1\text{e}^{-9} \end{bmatrix} \); process noise covariance matrix for weight filter

\% Adding artificial noise to signal
\( \text{step.voltage} = \text{step.voltage} \times (1 + \text{randn}(1, \text{length(step.voltage)}) \times 5\text{e}^{-3}) \); adding artificial random noise to measurement signal

\% starting guesses for cell parameters
\( C_1 = 815.1 \); Farad
\( \text{ohm}_0 = \begin{bmatrix} R_0 \\ R_1 \\ \text{cap}_0 \end{bmatrix} ; \% weight vector

\% Dual extended Kalman filtering
for \( i = 1: \text{length(step.relativeTime)} \); filtering over all selected time steps
\% time update weight filter then state filter
if \( i = 1 \); time step
\( \text{dt}(i) = \text{step.relativeTime}(i) \);
else
\( \text{dt}(i) = \text{step.relativeTime}(i) - \text{step.relativeTime}(i-1) \);
end
if \( i = 1 \); weight filter time update
\( \text{A}(:,:,i) = \begin{bmatrix} \exp(-\text{dt}(i)/(\text{ohm}_0(2)*C_1)) & 0 \\ 0 & 1 \end{bmatrix} ; \begin{bmatrix} -\text{dt}(i)/(\text{ohm}_0(3)*3600) \end{bmatrix} ; \text{x}_0 = \text{A}(:,:,i) \times \text{x}_0 + \text{B} \times \text{I}(i) ; \% state filter time update
else
\( \text{ohm}_0 = \text{ohm}_0 + \text{dt}(i)/(\text{ohm}_0(3)*3600) \);
\( \text{x}_0 = \text{A}(:,:,i) \times \text{x}_0 + \text{B} \times \text{I}(i) ; \% state filter time update
end
\% time update weight filter covariance matrix
\( \text{covx} = \text{A}(:,:,i) \times \text{covx} + \text{covw} ; \% time update state filter covariance matrix
\% calculating C matrices for both filters
\% derivative of OCV-SOC polynomial
if \( i = 1 \); calculation of C matrix for weight filter requires recursive procedure
\( \text{C}(i,:) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} ; \% initializing values
\text{F}(i,:) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} ; \% dx PLUS_k-1/dohm
\text{E}(i,:) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} ; \% dx_MINUS_k/dohm
else
\% partial derivatives
\( \text{dSOCdC}(i) = \text{I}(i) \times \text{dt}(i)/3600 \times (\text{ohm}_0(3) \times 2) ; \% \text{dSOC}/dC
\text{dgdC} = 0 ; \% dg/dC
\text{dxR1_plus1}(i) = \text{x}_0 \times \text{dt}(i) \times \exp(-\text{dt}(i)/(\text{ohm}_0(2)*C_1)) / (\text{C}_1 \times (\text{ohm}_0(2)^2) \times 2) ; \% \text{dxR1_plus2}(i) = \text{I}(i-1) \times (1 - \exp(-\text{dt}(i)/(\text{ohm}_0(2)*C_1)) - \text{dt}(i) \times \exp(-\text{dt}(i)/(\text{ohm}_0(2)*C_1)) / (\text{ohm}_0(2)*C_1)) ; \% \text{dxR1_plus}(i) = \text{dxR1_plus1}(i) + \text{dxR1_plus2}(i) ; \% \text{dV1}/\text{dR1} at \text{xR1_plus_k-1}
dxR1_minus1(i) = x_hat_minus(1) * dt(i) * exp(-dt(i)/(ohm_hat_minus(2)*C1)) / (C1*(ohm_hat_minus(2))^2);

dxR1_minus2(i) = I(i) * (1 - exp(-dt(i)/(ohm_hat_minus(2)*C1)) - dt(i) * exp(-dt(i)/(ohm_hat_minus(2)*C1)) / (ohm_hat_minus(2)*C1));

dxR1_minus(i) = dxR1_minus1(i) + dxR1_minus2(i);

% dV1/dR1 at current

%% full Jacobians

dgdohm=[-I(i), -dxR1_minus(i), dgdC(i)]; % [dg/dR0 dg/dR1 dg/dC]
dfdohm=[0, dxR1_plus(i), 0; 0, 0, dSOCdC(i-1)]; % [dV1/dR0 dV1/dC; dSOC/dR0 dSOC/dR1 dSOC/dC]

%% recursive derivatives

F(:,:,i)=E(:,:,i-1) - L(:,i-1)*C_ohm(i-1,:);
E(:,:,i)=dfdohm + A(:,:,i)*F(:,:,i);
C_ohm(i,:)=dgdohm + C(i,:)*E(:,:,i);

end

%% measurement update for state filter

L(:,i)=covx_minus*C(:,:,i-1) * (C(:,:,i-1) * covx_minus * C(:,:,i-1)' + sigmav)^-1; % Kalman gain matrix for state filter

Yest(i)=OCV - ohm_hat_minus(1)*I(i) - x_hat_minus(1); % output equation
erri(i)=Y(i)-Yest(i); % calculation of error of cell voltage prediction

x_hat_plus(:,i)=x_hat_minus + L(:,i)*err(i); % state estimate measurement update

covx_plus=( eye(2) - L(:,i)*C(i,:)' ) * covx_minus; % state filter covariance matrix measurement update

%% measurement update for weight filter

L_ohm(:,i)=covohm_minus*C_ohm(i,:)' * (C_ohm(i,:)' * covohm_minus * C_ohm(i,:)' + sigmaw)^-1; % Kalman gain matrix for weight filter

ohm_hat_plus(:,i)=ohm_hat_minus + L_ohm(:,i)*err(i); % state estimate measurement update

covohm_plus=( eye(3) - L_ohm(:,i)*C_ohm(i,:)' ) * covohm_minus; % weight filter covariance matrix measurement update

end

%% plotting of results

figure
plot(step.relativeTime/3600,SOC,step.relativeTime/3600,x_hat_plus(2,:),'Linewidth',2)
legend('CC SOC','Kalman SOC');
title('Kalman SOC')

set(gca,'FontSize',22)
ylim([0, 1])

xlim([0, max(step.relativeTime/3600)])
xlabel('Time (h)')
ylabel('SOC')

figure
plot(step.relativeTime/3600,ohm_hat_plus(3,:),'Linewidth',2)
title('Estimated capacity')
set(gca,'FontSize',22)
ylim([1.5, 2.3])

xlabel('Time (h)')
ylabel('Capacity (Ah)')

figure
plot(step.relativeTime/3600,ohm_hat_plus(1,:),step.relativeTime/3600,ohm_hat_plus(2,:),'Linewidth',2)
set(gca,'FontSize',22)
title('Estimated internal resistances')
legend('R_0','R_1')

set(gca,'FontSize',22)

xlabel('Time (h)')
APPENDIX 5: CODE FOR ENHANCED STATE VECTOR EXTENDED KALMAN FILTER

```matlab
close all, clear all
load RW9 % loading battery data file
load RWDIs % structure containing indeces of RW phases and steps in battery data
load capacity % vector containing all capacities for the battery in question measured during reference discharge tests
steps = data.step; % save steps array to new variable
%% Data selection
RWsteps=100; % number of RW steps from start of phase
phase=1; % RW phase
inds2 = RWDIs.repeatition(phase).indexes(1):RWDIs.repeatition(phase).indexes(RWsteps*2);
%% Stich all of the indexed steps together
step = steps(inds2(1));
for i = 2:length(inds2)
    step.relativeTime = [step.relativeTime step.relativeTime(end)+steps(inds2(i)).relativeTime]; % time of data points (s)
    step.voltage = [step.voltage steps(inds2(i)).voltage]; % cell voltage (V)
    step.current = [step.current steps(inds2(i)).current]; % current (A)
    step.temperature = [step.temperature steps(inds2(i)).temperature]; % temperature (deg. C)
end
%% Get cell capacity for RW phase from reference discharge tests
step.capacity=(capacity(phase*2)+capacity(phase*2-1))/2; % average of the two measurements before each RW phase
%% Coulomb counting for SOC reference
for i = 1:length(step.current)
    if i==1
        dt=step.relativeTime(i); % length of time step (s)
        SOC_0=1; % starting SOC value, representing a fully charged battery
    else
        dt=step.relativeTime(i)-step.relativeTime(i-1);
        current=(step.current(i)+step.current(i-1))/2; % average current between timesteps
        SOC(i)=SOC(i-1);
    end
    SOC(i)=SOC_0-(current*dt/(step.capacity*3600)); % Coulomb counting. Conversion factor 3600 to convert capacities to unit A*s from A*h
end
%% Kalman initialization and internal parameters
SOC_0=0.999; % starting guess of SOC
V1_0=0; % starting guess of transient voltage (V1) (V)
cap_0=1.9; % starting guess of cell capacity (Ah)
x_hat_0=[V1_0; SOC_0; 1/cap_0]; %Starting state vector x
covx_p=[1e-4 0 0; 0 1e-10 0; 0 0 1e-4]; % initial state covariance matrix
covv=[1e-4 0 0; 0 1e-10 0; 0 0 1e-11]; % process noise covariance matrix
covv_e=5e-3; % measurement noise covariance (matrix)
%% ECM parameters from pulsed discharge test
R0=0.074; % Ohm
R1=0.045; % Ohm
C1=815.1; % Farad
%% Kalman filter with enhanced state vector
for i=1:length(step.voltage) % filtering over all selected time steps
    %% Time update
    if i==1
        dt=step.relativeTime(i); % time step (s)
```
A=[ exp(-dt/(R1*C1)) 0 0 ; 0 1 -dt*step.current(i)/3600 ; 0 0 1 ];
B=[ R1 * (1-exp(-dt/(R1*C1))) ; 0 ; 0 ];
\[ x_{hat\_minus} = A*x_{hat\_0} + B*step.current(i); \]

% update

else
\[ dt = step\text{.relativeTime}(i) - step\text{.relativeTime}(i-1); \]
A=[ exp(-dt/(R1*C1)) 0 0 ; 0 1 -dt*I(i-1)/3600 ; 0 0 1 ];
B=[ R1 * (1-exp(-dt/(R1*C1))) ; 0 ; 0 ];
\[ x_{hat\_minus} = A*x_{hat\_plus}(:,i-1) + B*step\text{.current}(i-1); \]
end

covx_m = A*covx_p*A' + covw; % state covariance matrix time update

[OCV, dOCVdSOC] = OCV_SOC(x_hat_minus(2)); % function for the OCV-SOC relationship, also outputs the derivative of this curve
%
%% Calculating the C matrix
dSOCdC(i) = -I(i)*dt/3600; %dSOC/d(1/C)) as x(3)=1/C.
dgdC(i) = dOCVdSOC*dSOCdC(i);
C = [-1, dOCVdSOC, dgdC(i)]; %C=dg/dx

Yest(i) = OCV - R0*I(i) - x_hat_minus(1); % output equation

err(i) = Y(i) - Yest(i); % error of predicted cell voltage
%
%% Measurement update
L(:,i) = covx_m*C(i,:)'*(C(i,:)*covx_m*C(i,:) + covv)^-1; % Kalman gain
\[ x_{hat\_plus}(:,i) = x_{hat\_minus} + L(:,i)*err(i); \]
%
%% plotting results
figure % plot of Kalman SOC estimation in comparison to Coulomb counting
plot(step\text{.relativeTime}/3600, SOC, step\text{.relativeTime}/3600, x_hat_plus(2,:),'LineWidth',2)
legend('CC SOC', 'Kalman SOC');
set(gca,'FontSize',22)
title('Kalman SOC')
ylim([0, 1])
xlabel('Time (h)')
ylabel('SOC')

figure % plot of the estimated capacity
plot(step\text{.relativeTime}/3600,1./x_hat_plus(3,:), 'LineWidth',2)
set(gca,'FontSize',22)
ylim([1.5 2.3])
xlabel('Time (h)');
ylabel('Capacity (Ah)');

APPENDIX 6: CODE FOR SINGLE WEIGHT DUAL EXTENDED Kalman filter

close all, clear all, clc
load RW9 % loading battery data file
load RWDIs % structure containing indeces of RW phases and steps in battery data
load capacity % vector containing all capacities for the battery in question measured during reference discharge tests
steps = data\text{.step}; % save steps array to new variable
%
%% Data selection
RWsteps=100; % number of RW steps from start of phase
phase=1; % RW phase
inds2 = RWDIs\text{.repition}(phase).indexes(l):RWDIs\text{.repition}(phase).indexes(RWsteps*2);

%% Stich all of the indexed steps together
step = steps(inds2(l));
for i = 2:length(inds2)
    step.relativeTime = [step.relativeTime
    step.relativeTime(end)+steps(inds2(i)).relativeTime]; % time of data points (s)
    step.voltage = [step.voltage steps(inds2(i)).voltage]; % cell voltage (V)
    step.current = [step.current steps(inds2(i)).current]; % current (A)
    step.temperature = [step.temperature steps(inds2(i)).temperature]; %
    % temperature (deg. C)
end
%% Get cell capacity for RW phase from reference discharge tests
step.capacity=(capacity(phase*2)+capacity(phase*2-1))/2; % average of the
% two measurements before each RW phase
%% Coulomb counting for SOC reference
for i = 1:length(step.current)
    if i==1
        dt=step.relativeTime(i); % length of time step (s)
        SOC_0=1; % starting SOC value, representing a fully charged battery
        else
        dt=step.relativeTime(i)-step.relativeTime(i-1);
        current=(step.current(i)+step.current(i-1))/2; % average current
        between timesteps
        SOC_0=SOC(i-1);
        SOC(i)=SOC_0-(current*dt/(step.capacity*3600)); % Coulomb counting.
    Conversion factor 3600 to convert capacities to unit A*s from A*h
end
%% Kalman initialization and internal parameters
%% State filter internal parameters
SOC_0=0.999; % starting guess of SOC
V1_0=0; % starting guess of transient voltage (V)
x_hat_0=[ V1_0 ; SOC_0 ]; % starting state vector
covx_plus=[1e-6 0; 0 1e-8]; % guess initial covariance matrix for state
    filter
    sigmav=1e-8; % measurement noise covariance matrix for state filter
covw=[ 1e-8 0 ; 0 1e-12 ]; % state filter process noise covariance matrix
%% Weight filter internal parameters
covohm_plus=1e-3; % guess initial covariance matrix for weight filter
    sigmaw=1e-4; % measurement noise covariance (matrix) for weight filter
covr=1e-11; % weight filter process noise covariance (matrix)
%% Starting guesses/values of cell parameters and ECM parameters
cap_0=1.9; % initial guess of cell capacity (Ah)
R0=0.074; % Ohm
R1=0.045; % Ohm
C1=815; % Farad
ohm_hat_0=cap_0; % weight filter state (vector)
%% Single weight dual extended Kalman filtering
% "minus" indicates estimates after time update and "plus" estimates after
% measurement update
for i=1:length(step.voltage)
    if i==1 % time step
        dt(i)=step.relativeTime(i);
        ohm_hat_minus=ohm_hat_0; % time update weight filter
        A=[ exp(-dt(i)/(R1*C1)) 0 ; 0 1];
        B=[ R1 * (1-exp(-dt(i)/(R1*C1))) ; -dt(i)/(ohm_hat_minus*3600)];
        x_hat_minus=A*x_hat_0 + B*I(i); % time update state filter
    else
        dt(i)=step.relativeTime(i)-step.relativeTime(i-1);
        ohm_hat_minus=ohm_hat_plus(i-1);
        A=[ exp(-dt(i-1)/(R1*C1)) 0 ; 0 1];
        B=[ R1 * (1-exp(-dt(i-1)/(R1*C1))) ; -dt(i-1)/(ohm_hat_minus*3600)];
    end
end
% time update weight filter covariance matrix

covohm_minus=covohm_plus + covr;  % time update weight filter covariance matrix

covx_minus=A*covx_plus*A' + covw;  % time update state covariance matrix

[OCV, dOCVdSOC]=OCV_SOC(x_hat_minus(2));  % function for the OCV-SOC relationship, also outputs the derivative of this curve.

C(i,:)=[-1, dOCVdSOC ];  % C_x=dx/dx

% Measurement update for state filter as well as calculation of error

L(:,i)=covx_minus*C(i,:)' * ( (C(i,:) * covx_minus * C(i,:,:)' + sigmav )^-1 );  % Kalman gain matrix for state filter

Yest(i)=OCV - R0*I(i) - x_hat_minus(1);  % output equation

er(i)=Y(i)-Yest(i);  % error of predicted cell voltage

x_hat_plus(:,i)=x_hat_minus + L(:,i)*err(i);  % state filter measurement update

covx_plus=( eye(2) - L(:,i)*C(i,:) )*covx_minus;  % state filter covariance matrix measurement update

%% Measurement update for weight filter

if i==1  % calculation of C matrix for capacity filter requires recursive derivatives
    C_ohm(i,:)=0;  % initializing values

    F(:,i)=[0 ; 0];  % dx_plus_k-1/dC
    E(:,i)=[0 ; 0];  % dx_minus_k/dC

else  % Partial derivatives

    dSOCdC(i)=I(i)*dt(i)/(3600*(ohm_hat_minus)^2);  % inner derivative

    dgdC(i)=0;
    dfdC=[0 ; dSOCdC(i-1)];  % Recursive derivatives

    F(:,i)=E(:,i-1) - L(:,i-1)*C_ohm(i-1);  % C_ohm(i,:)

    E(:,i)=dfdC + A*F(:,i);  % C_ohm(i)=dgdC(i) + C(i,:)*E(:,i);

end  % Recursive derivatives

% Kalman gain matrix for weight filter

ohm_hat_plus(i)=ohm_hat_minus + L_ohm(i,:)*err(i);  % updating weight estimate post measurement

covohm_plus=( 1 - L_ohm(i) * C_ohm(i) )*covohm_minus;  % weight filter covariance matrix measurement update
end  % recursive derivatives

%% Plotting of results

figure  % plot of Kalman SOC estimation in comparison to Coulomb counting

plot(step.relativeTime/3600,SOC,step.relativeTime/3600,x_hat_plus(2,:),'LineWidth',2)

legend('CC SOC','Kalman SOC');
title('Kalman SOC')
set(gca,'FontSize',22)
ylim([0, 1])
xlabel('Time (h)')
ylabel('SOC')

figure  % plot of estimated capacity

plot(step.relativeTime/3600,ohm_hat_plus,'LineWidth',2)
title('Estimated capacity')
set(gca,'FontSize',22)
ylim([1.8, 2.3])
legend('Capacity (Ah)')
xlabel('Time (h)')
ylabel('Capacity (Ah)')