

Applications of Mathematical Models for Lithium-Ion Battery Management Systems

by

Moin Ahmed

A thesis

presented to the University of Waterloo

in fulfillment of the

thesis requirement for the degree of

Doctor of Philosophy

in

Chemical Engineering

Waterloo, Ontario, Canada, 2024

© Moin Ahmed 2024

Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

External Examiner	Dr. Xianke Lin
Supervisor(s)	Dr. Aiping Yu
	Dr. Michael Fowler
Internal Member	Dr. Ali Elkamel
	Dr. Yverick Pascal Rangom
Internal External Member	Dr. Amir Khajepour

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

Moin Ahmed, Yun Zheng, Amine Khalil, Hamed Fathiannasab, and Zhongwei Chen, “The role of Artificial Intelligence in the Mass Adoption of Electric Vehicles”. Joule, 2021

M.A. and Y.Z. conceived the review article. M.A. conducted relevant literature reviews and wrote the original manuscript. Y.Z. did major revisions and conducted literature review accordingly. H.F. conducted relevant literature review and wrote part of the section on battery-management systems. A.A. edited and proofed the manuscript. M.A., Y.Z., A.A., and Z.C. discussed, reviewed, and revised the manuscript. Z.C. and Y.Z. provided supervision throughout the manuscript-writing process. M.A. and Y.Z. contributed equally to this work. All authors approved the final version of the manuscript.

Moin Ahmed, Zhiyu Mao, Yun Zheng, Tao Chen, and Zhongwei Chen, Electric Vehicle Range Estimations using Regression Techniques, World Electric Vehicle Journal, 2021”.

M.A. conceptualized the project, wrote the relevant project code, and wrote the original manuscript. Z.M. and Y.Z. made major revisions to the manuscript and provided valuable feedback throughout the course of the project. T.C. and Z.C. provided supervision throughout the course of the project. All authors approved the final version of the manuscript.

Moin Ahmed, Xingling Zhang, Zhiyu Mao, Tao Chen, Zhongwei Chen, “Fault Detection in Lithium-ion Battery Pack of Electric Vehicle using Auto-Encoder”. Ready for submission.

M.A. and X.Z. conceptualized the project. M.A. wrote the project code and origin manuscript. M.A., X.Z., and Z.M. discussed the project results throughout the course of the project. X.Z. and Z.M. edited the manuscript. Z.C. provided supervision throughout the course of the project. All authors approved the final version of the manuscript.

Moin Ahmed, Zhiyu Mao, Michael Fowler, Zhongwei Chen. User-Specific Range Estimations Using Application-Scale Electric Vehicle and Continuum-Scale Battery Cell Electrochemical Models. Ready for submission.

M.A. conceptualized the research idea, wrote the code, conducted the research work, and wrote the original manuscript. Z.M. provided valuable feedback throughout the research project and edited the manuscript. M.F. and Z.C. provided supervision throughout the course of the project. All authors approved the final version of the manuscript.

Moin Ahmed, Yunpeng Liu, Zhiyu Mao, Michael Fowler, Zhongwei Chen. Comparative Analysis of Computational Times of Lithium-Ion Battery Management Solvers and Battery Models Under Different Programming Languages and Computing Architectures. Batteries. [Accepted]

M.A. conceptualized the research idea, wrote the code, conducted the research work, and wrote the original manuscript. Y.L. and Z.M. provided valuable feedback throughout the research project and edited the manuscript. M.F. and Z.C. provided supervision throughout the course of the project. All authors approved the final version of the manuscript.

Abstract

Lithium-ion batteries (LIBs) are the most widely used electrochemical storage technology in contemporary electrified vehicles (such as electric vehicles (EVs)), portable consumer products, and renewable energy generation applications. While industries pertaining to LIB manufacturing are now mature, LIB also benefits from enhanced performance in terms of high energy ($\sim 300\text{-}500 \text{ WhrL}^{-1}$ depending on the battery cell geometry) and power densities, long cycling life (> 1000 cycles), low memory effects, and low discharge rates. The battery cells of LIBs are commercially available in various form factors, including cylindrical, pouch, and prismatic form factors. In the above-mentioned applications, these battery cells are electrically connected in series and parallel combinations to output the desired energy and power requirements. This battery pack in term is connected to the battery management system (BMS) which serves various purposes, including (1) sensing for voltage, current, and temperature, (2) protection against extreme conditions such as excessive currents, under and high voltage limits, etc., (3) interface with the user on useful information such as charge control and range estimation and (4) Battery state estimation for performance management and diagnostics. In this thesis, research works for range estimations, early fault detection, and battery state estimations have been included.

Accurate range estimates for EVs are deemed an attractive alternative to conventional internal combustion engines due to their low carbon footprint, low running cost, and higher energy efficiency. However, currently, they suffer from a lower range than conventional vehicles, which induces range anxiety for consumers. First, the EV parameters that strongly impact its range are determined using data-driven techniques. A detailed dataset of commercial EV models manufactured from 2008 to 2021 was collected through web mining. A strong correlation between battery capacity, top speed, curb weight, and acceleration with range was observed. Furthermore, machine learning algorithms were trained and tested on this dataset, with the lowest root-mean-squared error of 19.5 miles. Additionally, a simple linear relationship between the EV range and EV model, battery, and performance parameters was determined to be convenient for EV consumers. Second, the EV system-level vehicle dynamics model and a physics-

based battery cell model were utilized to estimate the range for specific vehicles on custom user-specific drive cycles. For this purpose, two types of commercially available battery cells that are utilized in various EV models were selected. While most of the battery cell's electrochemical parameters from prior literature were used, some of these parameters were estimated using a genetic algorithm (GA). Meanwhile, the vehicle dynamics model was used to determine the battery pack energy and current requirements for various EV models. The drive cycles used in the mentioned model were the highway fuel economy test cycle (HWFET) and urban dynamometer driving schedule (UDDS). These requirements were then inputted into the battery cell models for range estimation. The estimated range was compared to the ranges disclosed by the United States Environmental Protection Agency (EPA).

The BMS utilizes the sensor readings from the current, voltage, and temperature readings to estimate the battery diagnostics and ensure user safety. Under extreme conditions, LIBs can undergo thermal runaway leading to battery pack fires and explosions, severely jeopardizing user safety. Hence, early fault detection of an EV battery pack can be a critical asset for EV user safety and battery pack longevity. In this work, an autoencoder was trained and applied to a real-world dataset consisting of 100 EVs. Furthermore, the voltage and temperature time series were compared as input for the fault detection. Temperature-based autoencoder was successfully able to detect a faulty battery pack from normal functioning ones in EVs. Lastly, a process flow of battery pack fault detection, with autoencoder, for large-scale EV applications is discussed.

Finally, the last set of research works concerns the state estimations, specifically the state-of-charge (SOC), of LIB. The complex and non-linear electrochemical behavior of the LIBs poses a significant academic and commercial challenge for its state estimations. While cell-level simpler equivalent circuit models (ECM) are commonly used by battery management systems (BMS) hardware, continuum-scale electrochemical models (EM) are attractive due to their higher accuracy, higher fidelity, and ease of integration with thermal and degradation models. However, compared to ECMs, EMs can have higher solution times and their numerical schemes require more extensive mathematical and computational expertise. Various reduced-order EM battery models, specifically the single particle model (SPM), enhanced single particle model

(ESPM), reduced order pseudo-two-dimensional (P2D), and their computationally efficient numerical schemes have been proposed in the literature. The computational performance of some of these battery models and solvers has been compared under different programming languages (Python and C++) and computational hardware specifications (hardware specifications representative of embedded, personal computing, and cloud systems). C++ programming language displayed at least a 10-fold reduction in solver and battery model solution times with the exact figure dependent on the cycling steps. Meanwhile, the embedded systems were able to perform the simulations using reduced-order battery solvers and battery models even with the slower-performing Python programming language, making them a reasonable candidate for embedded systems.

Acknowledgments

First and foremost, the author would like to acknowledge his previous supervisor Dr. Zhongwei Chen, and current supervisors, Dr. Aiping Yu and Dr. Michael Fowler, for their invaluable guidance and support throughout my Ph.D. program. The author would also wholeheartedly thank them for believing in my abilities and providing me with this excellent research and development opportunity. I have acquired countless skills within the domain of battery mathematical models and software development during this time.

Moreover, the author is honored to have Dr. Ali Elkamel (University of Waterloo), Dr. Yverick Pascal Ransom (University of Waterloo), Dr. Amir Khajepour (University of Waterloo), and Dr. Xianke Lin (Ontario Tech University) as his Ph.D. defense committee members. These committee members have provided valuable advice to address various research challenges.

The author would also like to thank and acknowledge his research contributors, Dr. Zhiyu Mao, Dr. Yun Zheng, Xinglin Zhang, and Dr. Tao Chen for their technical advice and recommendations throughout the relevant research projects. Moreover, the author is also grateful to his colleagues and peers in his research group with whom the author has spent memorable times during his Ph.D. program.

Last, but not the least, I would like to wholeheartedly thank my family for their consistent love, support, patience, and encouragement throughout my Ph.D. program.

Table of Contents

Examining Committee Membership	ii
Author's Declaration.....	iii
Statement of Contributions	iv
Abstract.....	vi
Acknowledgments.....	ix
List of Tables	xv
List of Illustrations.....	xvi
List of Abbreviations	xxii
List of Symbols	xxiv
1 Introduction	1
1.1 Challenges and Motivations	1
1.2 Project Objectives.....	4
1.3 Thesis Outline.....	5
2 Background and Literature Review.....	7
2.1 Chapter Summary	7
2.2 Introduction to Lithium-Ion Batteries and Basic Operations.....	7
2.3 Equivalent Circuit Models.....	9
2.4 Pseudo-Two-Dimensional Model.....	10
2.5 Simplified Continuum-Scale and Reduced Order Models	18
2.6 Overview of the Application of Artificial Intelligence within Electric Vehicle Applications ..	22

2.7	Physics-Based Continuum-Scale Battery Models and Machine Learning Models for Battery Management System Applications	26
2.7.1	Continuum Model Parameter Estimations	26
2.7.2	Electric Vehicles Range Estimations	27
2.7.3	Fault Detection	29
3	Range Estimations.....	32
3.1	Chapter Objective	32
3.2	Chapter Introduction.....	32
3.3	Main Contributions.....	35
3.4	Methodology.....	35
3.4.1	Data Collection and Preprocessing	35
3.4.2	Linear Regression Parameters using Bootstrapping.....	38
3.4.3	Regression Techniques.....	39
3.4.4	Model Training.....	42
3.4.5	Model Evaluation	42
3.4.6	Vehicle Dynamics Model.....	42
3.4.7	Battery Cell Model.....	46
3.4.8	Battery Cell Parameter Estimations using Genetic Algorithm.....	49
3.4.9	Experimental	50
3.5	Results and Discussions.....	51
3.6	Conclusions	66
3.7	Supporting Information	68

4	Fault Detection Using Machine Learning Techniques	72
4.1	Chapter Objective	72
4.2	Chapter Introduction.....	72
4.3	Main Contributions.....	75
4.4	Methodology.....	76
4.4.1	Data Acquisition and Preprocessing	76
4.4.2	Variation Coefficient.....	77
4.4.3	Autoencoder	79
4.4.4	Threshold Determination.....	83
4.5	Results and Discussion	83
4.6	Conclusions	90
5	Comparative Analysis of Computational Times of Lithium-Ion Battery Management Solvers and Battery Models Under Different Programming Languages and Computing Architectures for States Estimation	92
5.1	Chapter Objective	92
5.2	Chapter Introduction.....	92
5.3	Main Contributions.....	99
5.4	Methodology.....	100
5.4.1	Electrode Concentration Solver	100
5.4.2	Electrolyte Concentration Solver	105
5.4.3	Battery Models	108
5.4.4	Energy Balance	114

5.4.5	Sigma-Point Non-Linear Kalman Filter	117
5.4.6	Programming Language	121
5.4.7	Hardware Specifications	122
5.5	Results and Discussions.....	123
5.6	Conclusions	134
5.7	Supporting Information	135
5.7.1	Section S1: Electrode Open-Circuit Potentials	135
5.7.2	Section S2: Crank Nickelson Scheme for Lithium Diffusivity in the Solid Electrode Region – Inverse Matrix Operation vs. Thomas Algorithm	136
5.7.3	Section S3: Optimization of the Eigen Expansion Solver for the Lithium Diffusivity in the Solid Electrode Region.....	138
5.7.4	Section S4: Polynomial Approximation for the Lithium Diffusivity in the Solid Electrode Region	139
	Section S5: Electrode Lithium concentration solvers under static loads.....	140
5.7.5	Section S6: Absolute error of the electrode solvers compared to the Crank-Nicolson Method 141	
5.7.6	Section S7: Finite Volume Method for Lithium-ion Diffusivity in the Electrolyte Region – Inverse Matrix Operation vs. Thomas Algorithm	141
5.7.7	Section S8: Battery models under static battery cycling.....	143
5.7.8	Section S9: Comparison between the isothermal single particle model and enhanced self- correcting equivalent circuit model.....	143
6	Thesis Conclusions and Future Works	146

6.1	Conclusions	146
6.2	Future Works	151
Appendix A: Volume Averaging of Partial Differential Equations of Multiphase System		181
Appendix B: Non-linear Kalman Filter		183

List of Tables

Table 1: Boundary Conditions of P2D Governing PDEs.....	17
Table 2: Information Contained in Electric Vehicle (EV) Dataset	36
Table 3: Parameters for the Vehicle Dynamics Simulations	45
Table 4: Correlations between independent variables and EPA.	53
Table 5: The ML model parameters.....	59
Table 6: Model evaluations from MAE, MAPE, and R^2	60
Table 7: RMSE from model evaluations.....	60
Table 8: Parameters for the battery cell model.	64
Table 9: Autoencoder Specifications	82
Table 10: Battery Cell Parameters	116
Table 11: CPU and Memory Hardware Specifications used in this work.	123
Table 12: Solution times of the different solution methods performed in Python programming language in the PC system.....	137
Table 13: Model parameters for enhanced self-correcting equivalent circuit model.....	144

List of Illustrations

Figure 1: (A) Ragone plot emphasizing the higher volumetric and specific energy density of LIB compared to those of other battery chemistries ¹⁵ . (B) The reduction of battery cell and pack prices in recent years ¹²	2
Figure 2: (A) Electric vehicle (application of LIB battery cell) with a battery pack and battery management system. (B) Various commonly found commercial form factors of LIB battery cells. (C) Various functions of the battery management system in applications containing LIB battery pack. (D) Sensors used in the LIB battery pack, including the ones for the terminal cell/module voltage and temperature. (E) Modeling approaches used in the LIB modeling are categorized based on their simulation lengths.....	3
Figure 3: Pictorial representation of (A) different regions across the length of the battery cell and (B) flow of electrons and lithium ions during battery cell charge.	9
Figure 4: Equivalent Circuit Model Schematic for (A) Thevenin and (B) self-correcting cell models.....	10
Figure 5: Coordinate System for P2D Model ²⁹	12
Figure 6: The plots from the fitted functions that represent the thermodynamic potentials of the (A) positive and (B) negative electrodes at room temperature. (the plots have been generated from the functions provided by references. ^{19,31})	13
Figure 7: (A) The computation requirements and model accuracy of phenomenological battery cell-level equivalent circuit models (ECM) and physics-based continuum-level electrochemical models (EM). ⁴⁶ (B) The coordinate systems and governing PDEs of the single particle model (SPM). ⁴⁷	20
Figure 8: (A) Process flow for reinforcement learning. (B) Commonly used architecture for convolution neural networks (CNN). (C) Neural network of recurrent neural networks (RNN). RNN, commonly used for time series, has the output of the neural network at (t-1) time-step as the input of the at t time-step. The image shows a single RNN unrolled in time. (D) Depiction of the process flow of particle swarm intelligence (PSO). (E) Depiction of the process flow of genetic algorithm (GA). GA, commonly used for	

search and optimization engineering problems, has an initial population of solutions. During each iterative process, the population undergoes mutation and crossover operations to find the solution.	25
Figure 9: Flowchart of regression process.	36
Figure 10: EV model, battery, and performance features considered from the dataset.	37
Figure 11: (A) HWFET and (B) UDDS drive-cycles	45
Figure 12: A plot, along with the line of best fit and Pearson correlation coefficient between (A) EPA and WLTP (B) EPA and NEDC. All units are in km. Furthermore, the bootstrapped linear regression parameters for each iteration are presented for (C) EPA and WLTP and (D) EPA and NEDC.....	52
Figure 13: Plots of independent variables against EPA range, where the independents variables are (a) model year (b) battery capacity (in kWh), (c) number of battery cells in an EV, (d) number of battery modules, (e) EV battery pack voltage, (f) EV top speed (in km/h), (g) EV acceleration from 0 to 100 km/h (in s), (h) EV curb weight (in kg), and (I) Gross Vehicle Weight Rating (in kg).	55
Figure 14: Scatter plot of independent variables with EPA with scatter points colored differently based on EV body style.....	56
Figure 15: Histogram and boxplots of (A) EV body style, (B) battery cooling mechanism, and (C) type of lithium-ion battery used. The histogram plots the frequency (referred to as the count) of datapoints against the qualitative independent variables. Boxplot displays the average and quartiles of the EPA range against the qualitative independent variables.	58
Figure 16: Resultant variables calculated from the vehicle dynamics model. The top (Figure (A) and (B)), middle (Figure (C) and (D)), and bottom row (Figure (E) and (F)) represent the calculated motor speed, required battery pack power, required battery pack current for HWFET and UDDS drive cycles.	62
Figure 17: (A) The experimental and simulated discharge profile at two C-rates (1/3 C and 1/2 C) and various temperatures (10°C, 25°C, and 40°C). (B) Experimental and simulated HPPC discharge profile. The experimental and simulated battery cell SOC for the (C) constant current and (D) HPPC discharge profiles.	65
Figure 18: Open-circuit potential at room temperature for (A) positive and (B) negative electrodes.	68

Figure 19: (A) The battery cell terminal potential at various temperatures at 0.05 C. (B) The change of the battery cell's open-circuit potential with respect to temperature. Open-circuit potential of the (C) positive and (D) negative electrode with respect to temperature.....	70
Figure 20: (A) The absolute error between the simulated and experimental battery cell terminal voltage and (B) its distribution for HPPC cycle. (C) The absolute error between the simulated and experimental battery cell terminal voltage and (D) its distribution for the custom discharge cycle.....	71
Figure 21: (a) An example illustration an EV with a battery pack, comprising of battery cells, and connected to a BMS, (b) an example of a battery pack with terminal voltage and temperature sensors, (c) block diagram portraying the dataset and the methodology in this work.	77
Figure 22: (a) The proposed autoencoder structure. (b) Autoencoder model training process.....	80
Figure 23: The variation coefficient of (a) voltage and (b) temperature of a normally functioning EV and a faulty EV, referred to as “normal” and “faulty” in the figure legends.....	84
Figure 24: The MSE from the temperature autoencoder for (a) faulty and (b) a representative normal EV.	85
Figure 25: MSE distributions with different individual EV MSE values based on (a) mean, (b) 99.975th percentile, and (c) 100th percentile. The red outline indicates the histogram bar with the faulty EV.....	86
Figure 26: The MSE from the temperature autoencoder for (a) a normal functioning EV (EV1), (b) another normal functioning (EV2), and (c) the faulty EV. The distribution of MSEs for (d) EV1, (e) EV2, and (f) all normally functioning EVs compared to the faulty EV.....	88
Figure 27: MSE distributions with different individual EV MSE values based on the (a) mean and (b) 99.975th percentile. The red outline indicates the histogram bar with the faulty EV. These plots help identify the threshold value.	89
Figure 28: Recommended process flow for battery pack fault detection for cloud-based BMS in EV applications.	90

Figure 29: (A) Battery pack, comprising various battery cells, connected to a battery management system (B) Various form factors of commercial lithium-ion batteries used in battery packs. (C) coordinate systems, a system of partial differential equations (PDEs), and their respective boundary conditions.....	97
Figure 30: (A) Depiction of the single particle model and (B) various thermal sources considered in the lumped thermal model.	115
Figure 31: (A) Conceptual class diagram representing the codebase object-oriented architecture of the codebase. Class diagrams of the (B) Electrode, (C) BaseCycler	122
Figure 32: The solution of various solvers for the PDE about the lithium diffusivity in the solid electrode region for (A) negative electrode and (B) positive electrode under discharge-rest, (C) negative electrode and (D) positive electrode under charge-rest, and (E) negative electrode and (F) positive electrode under HPPC cycling steps.....	125
Figure 33: The solution times of various cycling steps in (A) PC, (B) cloud, and (C) embedded systems for the negative electrode. The solution times of the cycling steps performed in the Python and C++ programming language on the PC using (D) Eigen and (E) Polynomial solvers. (F) The solution times of the Polynomial solver on the embedded system are presented.	127
Figure 34: (A) Electrolyte concentration at different times across the battery cell during discharge. (B) The solution times of the FVM solver in Python and C++ programming languages across various computing systems.....	129
Figure 35: Time series of the battery cell terminal voltage as calculated from various battery models under (A) constant current discharge, (B) constant current charge, and (C) dynamic HPPC cycling. (D) Solution times of various charging steps as conducted using different battery models in Python in PC. (E) The solution times of the SPM under various cycling steps under a PC system. (F) The solution times of isothermal and non-isothermal SPM. (G) Comparison of the computation time between Python and C++ programming languages. (H) The solutions times of the isothermal discharge step for different battery models using Python in embedded system.	131

Figure 36: (A) True (or intended) vs. the actual noisy applied current. (B) The actual and SPKF estimated outputs. The estimated SOC for the (C) negative and (D) positive electrodes. (E) The computational times of the SPM vs. SPM with SPKF across multiple computing systems.	133
Figure 37: The (A) positive and (B) negative electrode's open-circuit potential.	136
Figure 38: The (A) solution times, (B) plots, and the (C) absolute errors from the use of various number of roots for the calculation of the negative electrode's state-of-charge (SOC) during the battery cell discharge. Similarly, the (A) solution times, (B) plots, and the (C) absolute errors of the positive electrode's state-of-charge (SOC) during the battery cell. These simulation times are from the simulations performed on the personal computer in the Python programming language.....	138
Figure 39: The polynomial approximation of various orders on the volume averaged equation of the partial differential equation pertaining to the lithium diffusivity in the solid electrode region.	139
Figure 40: The solution of various solvers for the PDE pertaining to the lithium diffusivity in the solid electrode region for (A) negative electrode and (B) positive electrode under discharge, (C) negative electrode and (D) positive electrode under charge cycling steps.	140
Figure 41: Absolute error between the Crank-Nicolson method and the eigen-expansion and polynomial expansion methods for the (A) negative electrode and the (B) positive electrode under the discharge cycling step. Furthermore, the absolute error for these methods for the (C) negative and the (D) negative electrodes under the HPPC cycling step.	141
Figure 42: The results from the inverse and the Thomas algorithm for the FVM solver on the electrolyte concentration PDE after 4000s of constant discharge.....	142
Figure 43: Battery models under constant current (A) discharge and (B) charge static battery cycle.....	143
Figure 44: (A) The electrical circuit for the ESC. (B) Actual and fitted battery cell open-circuit voltage (OCV). The dashed lines depict the electrode's open-circuit potentials (OCP). (C) The obtained time-terminal voltage comparison from the SPM and ESC.	145

Figure 45: Inhouse BMS system, currently being designed, tested, and optimized, by the thesis author. This current version includes voltage regulation (step-down buck converter), cell balancing, cell terminal voltage readings and current readings.	153
Figure 46: The step function represents the presence of α -phase in a one dimensional co ordinate system (left), while its derivative is the Dirac delta function (right).	181

List of Abbreviations

ANN	Artificial Neural Network
BC	Boundary Conditions
BEV	Battery Electric Vehicles
BMS	Battery Management System
BV	Butler Volmer Equation
CKDF	Centered Difference Kalman Filter
CI	Computational Intelligence
DL	Deep Learning
ECM	Equivalent Circuit Model
ESC	Enhanced Self-Correcting Equivalent Circuit Model
ESPM	Enhanced Single Particle Model
EV	Electric Vehicle
FVM	Finite Volume Method
GA	Genetic Algorithm
GHG	Green House Gases
HEV	Hybrid Electric Vehicles
IC	Initial Condition
LCO	Lithium-Cobalt Oxide
LIB	Lithium-Ion Battery
ML	Machine Learning
MSE	Mean-Squared Error
NMC	Nickel Manganese Cobalt
NN	Neural Network
OCV	Open Circuit Potential

OOP	Object-Oriented Programming
ODE	Ordinary Differential Equation
PDE	Partial Differential Equation
PBM	Physics Based Model
PHEV	Plug In Hybrid Electric Vehicles
RK4	Runge-Kutta Fourth Order Scheme
ROM	Reduced Order Model
SOA	Safe Operating Conditions
SOC	State of Charge
SPM	Single Particle Model
SVM	Support Vector Machine
TDMA	Tridiagonal Matrix

List of Symbols

ϵ_l	Volume fraction of the active material in the electrode l
$\epsilon_{e,j}$	Volume fraction of the electrolyte in the region j
$\mathcal{L}$	Mean-Squared Error
ρ	Density [kg/m^3]
σ	Root-Mean Square Error
δ_V	Variation Coefficient
A_{cell}	Surface Area of the battery cell
$a_{desired}$	Desired acceleration [m^2/s]
a_{actual}	Actual acceleration [m^2/s]
$a_{s,j}$	Specific interfacial area of the electrode [m^2/m^3]
$c_{e,init}$	Initial lithium-ion concentration across the battery cell [mol/m^3]
$c_{s,l}$	Lithium-ion concentration at the electrode l [mol/m^3]
$D_{s,l}$	Lithium-ion diffusivity of the solid electrode region l [m^2/s]
$D_{e,j}^{eff}$	Effective Lithium-ion diffusivity in the electrolyte region in the region j [m^2s^{-1}]
F	Faraday's Constant (96487 C/mol)
$F_{acceleration}$	Acceleration Force [N]
$F_{aerodynamics}$	Aerodynamics Force [N]
F_{grade}	Force experienced from the road grade [N]

$F_{rolling}$	Rolling Force [N]
j_l	The lithium-ion flux into or out of the electrode l [$mol/(m^2s)$]
h	Heat transfer coefficient [$W/(m^2K)$]
I	Applied battery cell terminal current [A]
I_{pack}	Applied battery pack current [A]
$m_{equivalent}$	Equivalent vehicle mass [kg]
$P_{battery}$	Power required by the EV battery pack [W]
P_{motor}	Power required by the EV motor [W]
R_{int}	Battery cell internal resistance from the electrolyte solution [Ω]
R_l	Particle radius of the active material in the electrode l [m]
T_{amb}	Ambient temperature [K]
T_{cell}	Battery cell temperature [K]
S_l	Electrochemically active area of the electrode l [m^2]
t	Time [s]
U_j	Open-circuit potential of the electrode l [V]
v_{actual}	Actual speed [m/s]
$v_{desired}$	Desired Speed [m/s]
x	Coordinate across the battery cell [m]
U_j	Open-circuit potential of the electrode j [V]

V_{cell}	Battery Cell Terminal Voltage [V]
Vol_j	Volume of the electrode j [m^3]
Vol_{cell}	Volume of the battery cell [m^3]

1 Introduction

1.1 Challenges and Motivations

There has been a significant push towards electrification of transportation and energy generation from sustainable and renewable energy sources in recent decades from governmental, industrial, and academic entities.¹⁻⁴ Electrified vehicles, including electric vehicles (EV), renewable power plants, and also portable consumer products rely heavily on energy storage systems. Currently, lithium-ion batteries (LIBs) are ubiquitously applied as energy storage technology in the mentioned application areas^{5,6}. This dominance of LIB as a contemporary energy storage system can be attributed to its industrial maturity and its advantages over other battery systems.⁷ These advantages include higher specific and volumetric energy densities (Figure 1A), lower self-discharge rate, and higher cycling life compared to other battery chemistries (including nickel-cadmium, nickel metal hydrides, and lead-acid batteries).⁷⁻⁹ Consequently, there has been industrial and academic interest in various areas of LIB research and development, including the design of LIB materials and improvement in their manufacturing processes.^{10,11} Furthermore, the recent advances in lithium-ion manufacturing have resulted in lower battery cell and battery pack costs (Figure 1B)¹². Another area of research, within the LIB domain, is the functionality of the battery management systems (BMS).^{13,14}

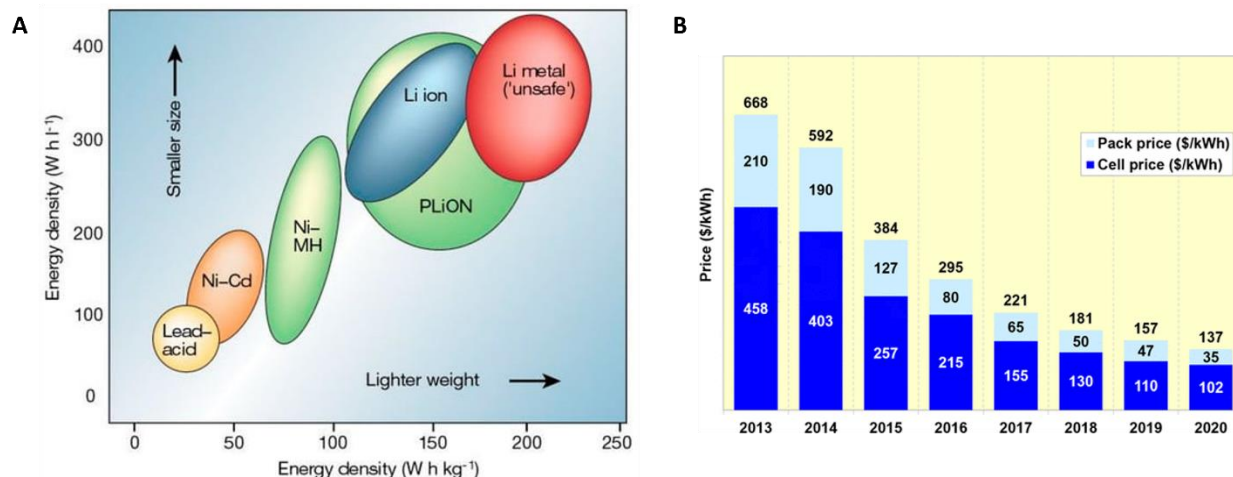


Figure 1: (A) Ragone plot emphasizing the higher volumetric and specific energy density of LIB compared to those of other battery chemistries ¹⁵. (B) The reduction of battery cell and pack prices in recent years ¹².

Within a LIB application area (e.g., in EVs), the battery pack hosts multitudes of battery cells connected in series and parallel to output the desired voltage and current (Figure 2A). These LIB battery cells can exhibit various form factors, including cylindrical, pouch, or prismatic (Figure 2B) ^{4,7}. The battery pack is also connected to the battery management system (BMS) which ensures the safe and performant operations of the battery pack as well as interfacing the relevant information to the end-user (Figure 2C). Functions of the BMS can include (1) sensing for voltage, current, and temperature, (2) protection against extreme conditions such as excessive currents, under and high voltage limits, etc., (3) interface with the user on useful information such as charge control and range estimation and (4) Battery state estimation for performance management and diagnostics.^{13,16} Moreover, within the battery pack, there are voltage and thermal sensors that provide battery terminal voltage and temperature readings for its calculations. Before proceeding, it is noteworthy to mention that various mathematical approaches have been explored in the literature for battery material design, battery cell simulations, and battery pack management.^{17,18} These mathematical can be categorized based on the lengths of the modeling outputs (Figure 2E). While continuum-scale equivalent-circuit models are common in practical BMS applications, physics-based continuum scale models have gained academic interest for BMS applications.¹⁹

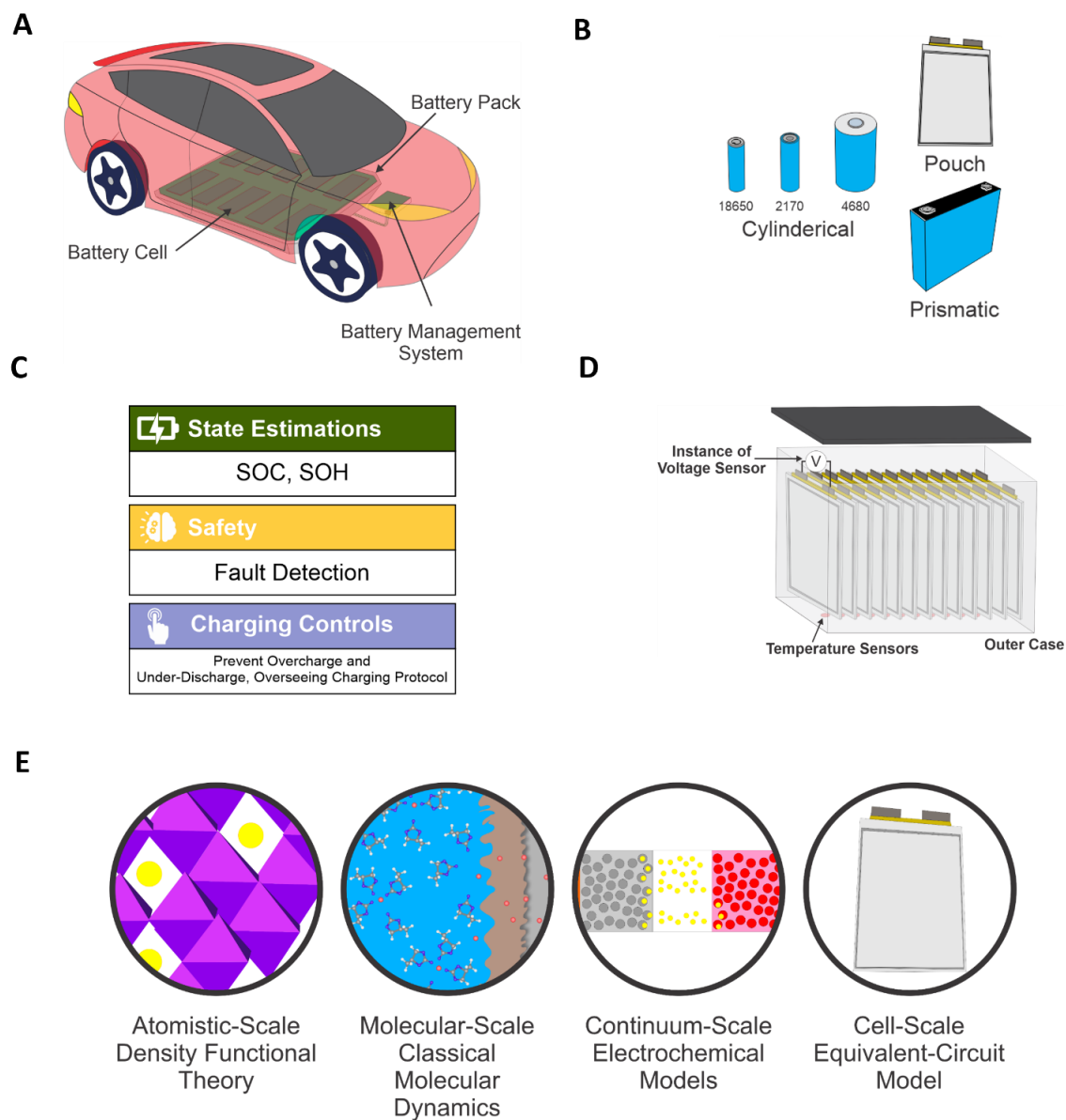


Figure 2: (A) Electric vehicle (application of LIB battery cell) with a battery pack and battery management system. (B) Various commonly found commercial form factors of LIB battery cells. (C) Various functions of the battery management system in applications containing LIB battery pack. (D) Sensors used in the LIB battery pack, including the ones for the terminal cell/module voltage and

temperature. (E) Modeling approaches used in the LIB modeling are categorized based on their simulation lengths.

The electrochemical and thermal phenomena occurring in battery cell operations are complex, coupled, and non-linear in nature. Hence, the mathematical formulation required for battery cell estimations tends to be tedious and computationally demanding. Moreover, the BMS systems are usually embedded systems with very limited computational and memory power. Furthermore, for battery pack BMS calculations, consideration for (1) the battery cell property and performance variability across the battery pack and (2) the external conditions need to be taken into account which further requires computational demand. Hence there has been active academic interest in making these BMS calculations computationally and practically manageable while maintaining their accuracy.

1.2 Project Objectives

The objective of the research works included in this thesis is to improve the functionality of the embedded and cloud-based BMS in areas of accurate range estimations, early fault detection, and state estimations using battery models. The intended BMS applications are targeted towards electric vehicles. All of the simulations were performed using an in-house codebase.

1. EV range estimation: The BMS can interface with the user by providing useful information about the battery pack including the battery pack's range estimation. Range estimation was performed using vehicle dynamics and a physics-based battery model. The first research work on range estimation focuses on determining the main parameters that influence the electric vehicle's range. For this purpose, a dataset of over 200 EVs was collected through web scraping from the internet. Various self-learning linear regression models were employed to find the main vehicle and battery parameters that affect the EV's range. Later, to form a mathematical formulation for range estimation on custom drive cycles, application-level vehicle dynamics models and physics-based continuum-scale models were implemented together on Tesla Model 3. From the Tesla Model 3,

vehicle parameters were used to find the battery pack requirements during different drive cycles. The fitted battery cell models were then used to estimate the range of the EV.

2. Battery pack early fault detection: Under extreme conditions, LIBs can undergo thermal runaway leading to battery pack fires and explosions, severely jeopardizing user safety. Early fault detection for a BMS is therefore a desired functionality. In this work, an autoencoder was trained and applied to a real-world dataset consisting of 100 EVs. Temperature-based autoencoder was successfully able to detect a faulty battery pack from normal functioning ones in EVs. Lastly, a process flow of battery pack fault detection, with autoencoder, for large-scale EV applications is discussed.
3. Solver times and computational requirements of various solvers and battery models for state estimations: Phenomenological cell-level equivalent circuit models are commercially utilized for battery pack state estimation due to their ease of computational implementation, lesser mathematical rigor, and faster solution times. In this research work, an attempt was made to estimate the solution times of various full-order and reduced-order models as well as various continuum-scale battery cell models in various programming languages (Python and C++) and different hardware specifications (hardware specifications representative of embedded, personal computing, and cloud systems). The solution times when thermal models and non-linear Kalman filters for battery cell estimations were also tested and evaluated.

1.3 Thesis Outline

As mentioned in the previous subsection, the main thesis objectives are on the application of mathematical models for the functionalities of the BMS in the LIB-based battery packs. The thesis consists of six chapters and their brief content is outlined below.

- Chapter 1 presents the thesis's general overview, motivations, and outline.
- Chapter 2 serves as the thesis background and also provides a related literature review overview. Hence, it briefly provides an overview of the phenomenological battery-cell level, physics-based continuum-scale battery cell models, and machine-learning algorithms relevant to BMS

applications. This is followed by their use in BMS applications in research, including EV range estimations, early fault detection, and battery cell and pack state estimations.

- Chapter 3 provides the introduction, methodology, results and discussions, and conclusion subsections on the conducted research work for EV range estimations. This is followed by the section on supporting information that provides additional information to support the results and discussions.
- Chapter 4 provides the same subsections as its preceding chapter except that its main focus is on the application of machine learning models for battery pack early fault detection.
- Chapter 5 provides the introduction, methodology, results and discussions, conclusions, and supporting information on the use of various solvers and battery models for state estimation. These models are implemented in various hardware whose specifications resemble those of personal computing, embedded systems, and cloud computing.
- Chapter 6 concludes all the previous chapters.

2 Background and Literature Review

This chapter is reprinted in adapted from the article below with permission from Joule.

Moin Ahmed, Yun Zheng, Amine Khalil, Hamed Fathiannasab, and Zhongwei Chen, “The role of Artificial Intelligence in the Mass Adoption of Electric Vehicles”. Joule, 2021

2.1 Chapter Summary

As mentioned in the prior section, the objective of the research works in this thesis pertains to the mathematical models and algorithms for the BMS functionalities. As such this section focuses on providing relevant background and literature work. For the background, first, the macro-scale phenomena during battery cell operations are discussed. Then the equations (along with their brief explanation) behind battery cell-level phenomenological models, specifically equivalent circuit model, and continuum-scale physics-based pseudo-two-dimensional are presented. A discussion on simplified physics-based continuum-scale models and their reduced-order formulation, as found in the literature, is presented. Then, a discussion on the background and application of machine learning algorithms for practical battery management systems is followed by the use of these models for BMS applications, specifically range estimations for electric vehicles (EV) applications, battery pack early fault detection, and battery pack states estimations.

2.2 Introduction to Lithium-Ion Batteries and Basic Operations

As is the case for electrochemical systems, the lithium-ion battery (LIB) consists of current collectors, and two electrodes (here denoted by the positive and negative electrodes) separated by the electrolyte. Specifically, when categorically defining the regions across the thickness of the LIB battery cell, there is a current collector, negative, electrode, separator, positive electrode, and, finally, another current collector (Figure 3A). The separator, usually made from polymeric materials, provides a mechanical and electrical barrier between the negative and positive electrodes. It allows the movement of ions in the electrolyte. The electrolyte solvent is organic and has dissolved lithium salts. Foils of copper and aluminum are prevalent as current collectors adjacent to the negative and the positive electrode, respectively.⁷ The electrodes are a

coating of a mixture of electrochemically active and inactive components on the current collectors. The electrochemically inactive components do not take part in the reduction-oxidations reactions but rather enhance the electrode's binding to the current collector and electric conductivity. Graphite is dominantly used as the electrochemically active material in the LIB's negative electrode. Meanwhile, various lithium transition metal oxides are used as the electrochemically active material in the positive electrode. These materials vary in their crystalline structure as well as electrochemical and mechanical properties ^{7,15}.

During battery operations (i.e., charge or discharge), there is a flow of charge-neutral lithium within the electrodes, lithium-ions within the electrolyte, and electrons through the external circuit as well as the lithium-ion reduction-oxidation reactions at the electrolyte-electrode interface. For instance, when the LIB undergoes discharge, the lithium atoms within the negative electrode transverse through its crystalline structure to the electrode-electrolyte interface. There, the lithium atoms get oxidized to the lithium-ion and dissolve into the electrolyte. Also, the reduction of lithium-ions in the electrolyte near the positive electrode-electrolyte interface happens when the reduced lithium-ion enters the positive electrode active material crystalline structure and then diffuses into its bulk. Within the electrolyte, there is a diffusion of lithium-ion, which, in general, is motivated by the lithium-ion concentration difference, the presence of an electric field, and solvent convection. Meanwhile, there is a flow of electrons, the amount of which is equal to the number of lithium ions reduced or oxidized, within the external circuit from the negative terminal end of the battery cell to the positive one. The flow of ions and electrons is illustrated in Figure 3B.

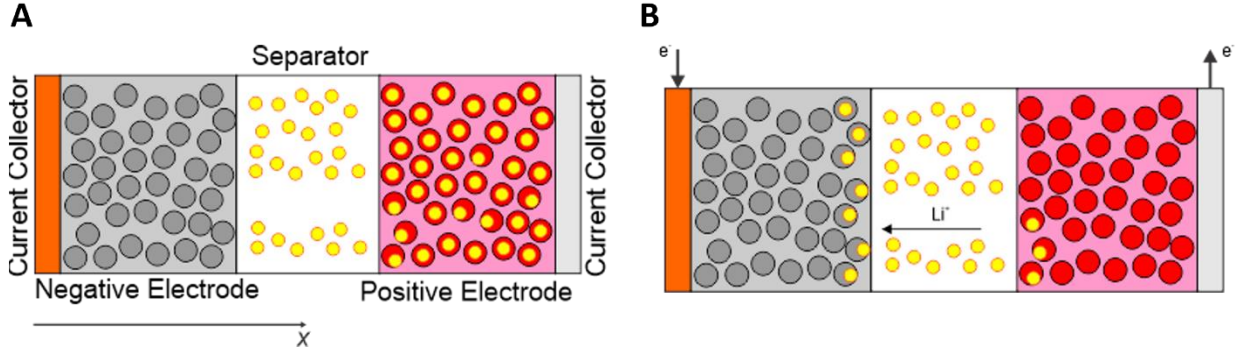


Figure 3: Pictorial representation of (A) different regions across the length of the battery cell and (B) flow of electrons and lithium ions during battery cell charge.

2.3 Equivalent Circuit Models

Equivalent Circuit Models (ECM) attempt to simulate the battery cell terminal voltage and state-of-charge (SOC) based on the charge/discharge current using the behavior of a circuit of electric components. These models are simpler to implement, but they are not based on underlying physical phenomena occurring in the batteries during operations.²⁰ In addition, various battery thermal and aging models can be incorporated with the ECM to account for non-isothermal conditions and battery aging.^{21,22} Thevenin and the self-correcting cell model are the most prominent and commonly utilized ECM models (Figure 4). Thevenin models the terminal voltage polarization during battery operation using multiple resistor-capacitor (RC) pairs (Figure 4A), while the self-correcting cell model further models the cell voltage hysteresis by including a hysteresis circuit element (Figure 4B). It has been observed that terminal cell voltages converge to a voltage other than the cell's rest potential at a given SOC, and this hysteresis cell element accounts for such discrepancies.²³ Since the Thevenin model is expected to be used in the research projects, its relevant mathematical equations are presented below.

For practical reasons, the time in the Thevenin model is discretized, and the battery cell's SOC, the current through the resistor in the n^{th} RC pair, and the terminal voltage are given, respectively as follows (Equation 1 and Equation 2).²³

$$SOC[k + 1] = SOC[k] - \frac{\Delta t}{Q} \eta[k] i[k]$$

Equation 1

$$i_{R_n}[k + 1] = \exp\left(-\frac{\Delta t}{R_n C_n}\right) i_{R_n}[k] + \left(1 - \exp\left(-\frac{\Delta t}{R_n C_n}\right)\right) i[k]$$

Equation 2

Where k , Δt , OCV, and Q represent the time-step, time period between the current and the last steps, open-circuit potential, and the nominal cell capacities, respectively. It should be noted the initial values for the SOC, the current through the resistor in the n^{th} RC pair, and the terminal cell voltage need to be inputted into the ECM models. Also, the function of the OCV with respect to the SOC (i.e., $SOC[k]$) needs to be known before using the ECM model. Experimentally this OCV function is found by fully charging and discharging the battery cell at a low current and averaging the recorded cell terminal voltages at every SOC.²³

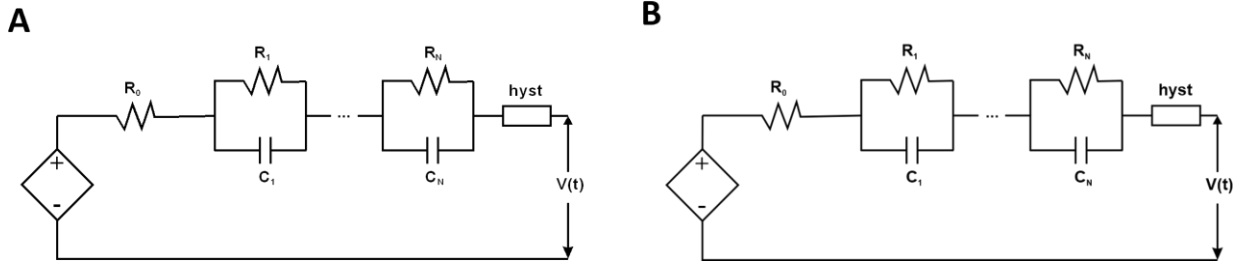


Figure 4: Equivalent Circuit Model Schematic for (A) Thevenin and (B) self-correcting cell models.

2.4 Pseudo-Two-Dimensional Model

Within the realm of LIB physics-based modeling, the model approaches can be categorized by the length scale of the modeled variable(s)/ parameter(s) /phenomenon(s)^{19,24}. On one end, the molecular scale approaches, which use density functional theory and molecular dynamics, have been applied to research the material properties of LIB electrodes and electrolytes²⁵. On the other end, the application level calculates the energy and power requirements of the battery pack in an application using the dynamic forces

experienced by that particular application during its usage. In between, the microscale models attempt to calculate the quantify the concentrations of lithium and the potential in the electrode and electrolyte regions using mass transport and charge conservation. In this approach, each spatial point during the simulation process is considered to possess a distinct chemical composition. This makes them computationally expensive and programmatically challenging to implement for battery material design and controls. On the other hand, continuum scale models assume an aggregate material property within the electrode region. The remainder of this subsection provides the continuum scale pseudo-two-dimensional (P2D) equations about mass- and charge-conservation of the lithium-ions within the solid electrode and liquid electrolyte region.

Thomas Fuller, Marc Doyle, and John Newman introduced the continuum-scale LIB battery cell model, referred to as the Doyle-Fuller-Newman or pseudo-two-dimensional (P2D) model, and this model is still widely used to date ^{26–28}. In this model, the lithium mass and charge in two distinct phases, the solid electrode and liquid electrolyte are considered. The x -coordinate axis starts at the negative electrode-current collector interface (i.e., where $x = 0$) and has the maximum value (i.e., $x = L_{max}$) at the positive electrode-current collector interface. Note that within the negative and positive electrode regions, both solid electrode and liquid electrolyte phases are present. For the electrolyte phase, the mass and charge conservation partial differential equations (PDEs) for the concentrated solution theory are utilized. Furthermore, the point form of Ohm’s law is used for the potential variation in the solid electrode region. Moreover, the spherical coordinate system is used for the diffusion of the lithium in the solid electrode phase within the electrode regions. A perfect sphere, of radius R_s , is used to represent an electrode particle at a particular point in the x -coordinate system. Hence, the PDE pertaining to the mass balance in solid from the concentration gradient in the radial coordinate system is used. A schematic of the spatial co-ordinates across the battery cell is illustrated below (Figure 5).^{19,28}

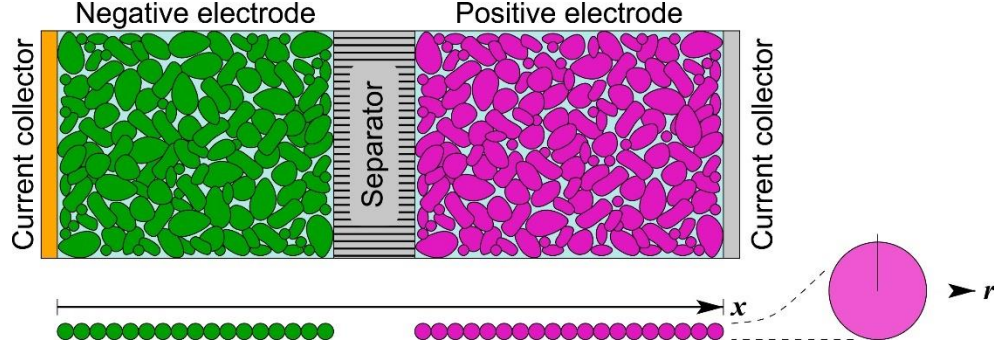


Figure 5: Coordinate System for P2D Model²⁹

Before proceeding, the brief explanations of the overpotentials, the electrode's state-of-charge (SOC_j), and the molar flux of ions/electrons are presented in these paragraphs. The continuum-scale models presented in this section attempt to model the overpotential of the LIB during its charge/discharge operations. Overpotential in this thesis refers to the difference between the electrode or cell terminal potential with its thermodynamic equilibrium potential at ambient temperature and pressure. The overpotential is the difference between thermodynamic terminal potential and the actual terminal potential during the operations. The source of overpotential can arise from the surface overpotential, the ohmic potential drop across the battery cell constituents, and the concentration overpotential³⁰. The open-circuit voltage (OCV) of an electrochemical cell, including LIB, is its terminal potential when at rest. During the charge/ discharge process, the average concentration of lithium within an electrode changes. This is referred to as the change of lithiation of the electrode. The state of charge (SOC_j) of an electrode, which refers to the concentration of lithium concentration within is defined as follows.

$$SOC_n = \frac{\theta_n - \theta_{n,0\%}}{\theta_{n,100\%} - \theta_{n,0\%}}$$

$$SOC_p = \frac{\theta_p - \theta_{p,0\%}}{\theta_{p,100\%} - \theta_{p,0\%}}$$

Equation 3

Where the subscript n and p denotes the negative and positive electrodes, respectively. Furthermore, $\theta_{p,0\%}$, $\theta_{p,100\%}$, and $\theta_{n,0\%}$, $\theta_{n,100\%}$ are the allowed minimum and maximum states of lithiation of the respective electrodes. The functions representing the experimental thermodynamic potentials of the positive and

negative electrodes as the function of their lithiation are used in the continuum-scale models (the plots from these functions, of various commonly used commercial electrode chemistries, are presented in the figure below).

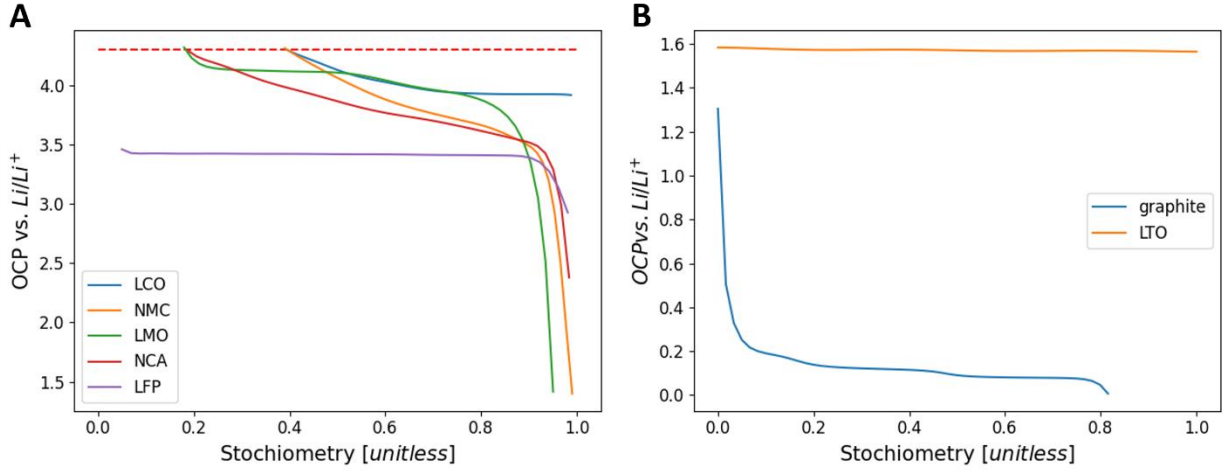


Figure 6: The plots from the fitted functions that represent the thermodynamic potentials of the (A) positive and (B) negative electrodes at room temperature. (the plots have been generated from the functions provided by references.^{19,31})

In the P2D model, the Butler Volmer (BV) equation is used to calculate the charge transfer current at the electrode-electrolyte interface within the electrode regions (Equation 4).^{19,28,30} It is important to note that in this thesis, molar flux (in mol m⁻² s⁻¹) is used instead of the current density (in A m⁻²). Therefore, Faraday's constant is used for relevant conversions of the exchange current density.

$$J_j(c_{s,e}, c_e, \phi_s, \phi_e) = \frac{i_{0,j}}{F} \left[\exp\left(\frac{\alpha_a F}{RT} \eta_{s,j}\right) - \exp\left(-\frac{\alpha_c F}{RT} \eta_{s,j}\right) \right] \quad \text{Equation 4}$$

Where the $i_{0,j}$ is referred to as the exchange current density of an electrode j . α_a and α_c denotes the rate of anodic and cathodic directions, respectively. The surface overpotential of the electrode, η_s , represents the potential difference between the solid surface and the immediate liquid potential above the solid's thermodynamic potential (mathematically represented by Equation 5).^{19,30}

$$\eta_{s,j} = \phi_s - \phi_e - U_{ocp}(c_{s,e}) \quad \text{Equation 5}$$

Where, the open-circuit potential, U_{ocp} , of the electrode depends on the surface lithium concentration in the electrode particle and is generally explicitly defined by a function during simulation runs. Moreover, the exchange current density in the above BV equation is represented as follows (Equation 6).¹⁹

$$i_{0,j} = Fk c_e^{1-\alpha} (c_{s,max} - c_{s,e})^{1-\alpha} c_{s,e}^\alpha \quad \text{Equation 6}$$

Where k (in $\text{mol}^{1-\alpha} \text{m}^{4-3\alpha} \text{s}^{-1}$) is the effective reaction rate constant of the redox reaction at the electrode-electrolyte interface. $c_{s,max}$ (in mol m^{-3}) denotes the maximum lithium concentration (or capacity) of the electrode.

The driving force of the lithium atom within the solid electrode is assumed to be the concentration gradient. Mathematically, with the application of Fick's second law, this can be represented as follows (Equation 7).³²

$$\frac{\partial c_s}{\partial t} = \nabla \cdot (D_s c_s) \quad \text{Equation 7}$$

Where c_s denotes the lithium atoms in the solid electrode region. Moreover, D_s [in m^2/s] denotes the diffusivity of the solid lithium in the electrode material in question. In P2D formulation, where a spherical coordinate system is considered for the above PDE, the above PDE is presented as follows (Equation 8).

$$\frac{\partial c_{s,j}}{\partial t} = \frac{1}{r^2} \frac{\partial}{\partial r} \left(D_{s,j} \frac{\partial c_{s,j}}{\partial r} \right) \quad \text{Equation 8}$$

Meanwhile, the macroscopic point form of Ohm's law is given below (Equation 9).

$$\nabla \cdot \mathbf{i}_s = \nabla \cdot (-\sigma \nabla \phi_s) = 0$$

Equation 9

After the application of volume average technique is applied to the above PDE (refer to Appendix A on the PDE volume averaging technique) and some rearrangements, we obtain the following (Equation 10).³³

$$\frac{\partial}{\partial x} \left(\sigma_{eff,j} \frac{\partial \phi_{s,j}}{\partial x} \right) = a_{s,j} F J_j$$

Equation 10

Along the derivation of the above equation, the specific interfacial surface area, $a_{s,j}$, is introduced and it is representative of the total surface area of the electrode particles per unit volume within a volume (defined in Equation 11). It is defined as follows.³³

$$a_{s,j} = \frac{3\epsilon_{s,j}}{R_j}$$

Equation 11

In the above equation, the term $\epsilon_{s,j}$ is the volumetric ratio of the active material. Furthermore, effective conductivity, σ_j^{eff} is introduced and it is usually assumed to be modelled using the Bruggeman correlation ($\sigma_j^{eff} = \sigma_j \epsilon_{s,j}^{brugg}$). Note that the constant *brugg* represents the Bruggeman co-efficient.

The movement of a chemical species (c_e) in a concentrated solution is assumed to be driven by the concentration gradient, migration, and solvent convection. The mass transport of the concentrated solute species and the resulting current density are represented by the following differential equations.³⁴

$$\frac{\partial c_e}{\partial t} = \nabla \cdot \left[D_e \left(1 - \frac{d \ln c_0}{d \ln c} \right) \nabla c_e - \frac{\mathbf{i} \cdot \nabla t_+^0}{z_+ v_+ F} - \nabla \cdot (c_e \mathbf{v}_0) \right]$$

Equation 12

$$\mathbf{i} = -\kappa \nabla \phi - \frac{\kappa}{F} \left(\frac{s_+}{n v_+} + \frac{t_+^0}{z v_+} - \frac{s_0 c}{n c_0} \right) \nabla \mu_e$$

Equation 13

In the above equations, it is commonly assumed that the term $\frac{d \ln c_0}{d \ln c}$, ∇t_+^0 and the solvent convection ($\mathbf{v}_0$) are approximately zero. Again applying the volume averaging technique to the LHS and RHS of the above equations, the following resultant differential equations are obtained.^{28,33}

$$\frac{\partial(\epsilon_e c_e)}{\partial t} = \nabla \cdot (D_{e,eff} \nabla c_e) + a_s(1 - t_+^0)J_j \quad \text{Equation 14}$$

$$\frac{\partial}{\partial x} \left(\kappa_{eff} \frac{\partial \phi_e}{\partial x} + \kappa_{D,eff} \frac{\partial \ln c_e}{\partial x} \right) + a_s FJ(c_{s,surf,j}, c_e, \phi_{s,j}, \phi_e) = 0 \quad \text{Equation 15}$$

The boundary and initial conditions of the above differential equations are tabled below (Equation 16- Equation 19).

Table 1: Boundary Conditions of P2D Governing PDEs

	$\frac{\partial \bar{\phi}_s}{\partial x} \Big _{x=0} = -\frac{i_{app}}{A * \sigma_{eff}}$	
Charge conservation in the electrode	$\frac{\partial \bar{\phi}_s}{\partial x} \Big _{x=L^-+L^+} = 0$	Equation 16
	$\frac{\partial \bar{\phi}_s}{\partial x} \Big _{x=L} = -\frac{i_{app}}{A * \sigma_{eff}}$	
	$\varepsilon_e \bar{\Gamma}_e = -\kappa_{eff} \nabla \bar{\Phi}_e - \kappa_{D,eff} \nabla \ln \bar{c}_e$	
Charge conservation in the electrolyte	$-\kappa_{eff} \frac{\partial \bar{\Phi}_e}{\partial x} - \kappa_{D,eff} \frac{\partial \ln \bar{c}_e}{\partial x} \Big _{x=0} = 0$	Equation 17
	$-\kappa_{eff} \frac{\partial \bar{\Phi}_e}{\partial x} - \kappa_{D,eff} \frac{\partial \ln \bar{c}_e}{\partial x} \Big _{x=L} = 0$	
Mass conservation in the electrode	$\frac{\partial \bar{c}_s}{\partial r} \Big _{r=0} = 0$	Equation 18
	$D_s \frac{\partial \bar{c}_s}{\partial r} \Big _{r=0} = \bar{j}$	
Mass conservation in the electrolyte	$\frac{\partial \bar{c}_e}{\partial r} \Big _{x=0} = 0$	Equation 19
	$\frac{\partial \bar{c}_e}{\partial r} \Big _{x=L} = 0$	

The system of non-linear and coupled partial differential equations in the P2D model can be solved using finite difference (FDM), finite element (FEM), and finite volume methods.³³ When solved numerically, these systems of equations are discretized in temporal and spatial coordinates and the resulting equations are solved simultaneously. As expected, the concentration profile of the lithium-ions in the solid electrode and liquid electrolyte regions are obtained as well as the potential profile in the solid electrode regions. The

cell terminal voltage (V_{cell}) is simply the difference of the potentials at either end of the battery cell. It is common practice to set the potential at the negative end to zero. Mathematically, the cell terminal voltage is defined as follows.³³

$$V_{cell}(t) = \phi_s(L^{tot}, t) - \phi_s(0, t) = \phi_s(L^{tot}, t) \quad \text{Equation 20}$$

Also, the state-of-charge of the battery cell can be calculated from either the state-of-charge of the positive or negative electrodes.

$$SOC_{cell} = \frac{\theta_n - \theta_{n,0\%}}{\theta_{n,100\%} - \theta_{n,0\%}} = \frac{\theta_p - \theta_{p,0\%}}{\theta_{p,100\%} - \theta_{p,0\%}} \quad \text{Equation 21}$$

Experimentally, the P2D model has been verified under various operating conditions (e.g., different operating currents and temperatures).^{35–40} For instance, Doyle *et al.* successfully demonstrated the accuracy of the model's prediction of cell potential and capacity at various C-rates (charge and discharge C-rates vary from 0.175 mA/cm² to 7mA/cm²) using lithium manganese oxide active material as the positive electrode and carbon as the negative electrode.³⁵ In other research works, model agreement with experimental results on battery cells with different positive electrode chemistries (including lithium manganese oxide, lithium cobalt oxide, and lithium iron phosphate), electrode and electrolyte thickness up to a high C-rate.^{37–39} Christensen *et al.* also demonstrated good model agreement with the experimental results using lithium titanate spinel negative electrode (an alternative chemistry for graphite).⁴⁰

2.5 Simplified Continuum-Scale and Reduced Order Models

ECM-based models are currently extensively used in battery management systems (BMS) applications due to their ease of implementation, low computational requirements, and low simulation times.²³ ECM models allow for individual battery cell and battery pack states, such as SOC, which cannot be measured directly.⁴¹ However, ECMs, unlike physics-based models such as P2D models, fail to provide the physical insights occurring in a battery cell.²³ For this reason amongst others, physics-based continuum-scale models, such

as the P2D model, have higher fidelity and accuracy when measuring the overpotentials compared to the ones measured using ECMs. Furthermore, it is fairly easier to implement other macroscale phenomena that affect the electrochemical performance, such as those resulting from the battery degradation (for example the increase in battery cell internal resistance from the growth of solid-electrolyte interface (SEI) during battery charge-discharge operations) to the physics-based models (Figure 7A).²¹ To utilize the physics-based models (PBM) for practical BMS applications, researchers have either (1) simplified the P2D models by utilizing simplifying model assumptions and/or (2) employed efficient and simpler numerical schemes for P2D's underlying PDEs (while considering their initial and boundary conditions). This subsection provides a brief overview of the research works that have employed these strategies.

The single particle model (SPM) for lithium-ion batteries, introduced by *Zhang et al.*, simplifies the P2D model majorly by having two assumptions (1) the local lithium-ion concentrations and potential in the electrolyte are constant, and (2) the flux of intercalating lithium-ions between the electrode and the electrolyte is constant. Furthermore, the electrodes of the cell are modeled as perfect spheres.⁴² Hence, the model uses the PDE pertaining to the lithium-ion concentration variation in the solid electrode region for both the positive and negative electrode regions (Equation 8)(Figure 7B).^{42,43} Moreover, there is an algebraic equation for the calculation of the cell terminal voltage that includes the polarization contributions of the slow lithium-ion diffusivity in the solid electrode regions and a constant aggregate internal cell resistance. While the accuracy of SPM has been demonstrated in various LIB chemistries, it fails under high charge and discharge rates. Therefore, it shows accuracy for charge/discharge currents up to 0.5C and can be used for applications that utilize application currents.^{43,44} The SPM inaccuracies at higher C rates can be attributed to the fact that it ignores the resulting polarizations from the electrolyte dynamics during the battery cell operations. Moura *et. al.* extended the single-particle model to include the electrolyte contributions, and this model is referred to as the enhanced single-particle model (ESPM) in this thesis. Their proposed model leads to the PDEs for the lithium-ion concentrations in the solid electrode region and liquid electrolyte region (Equation 8 and Equation 14). Again, an algebraic equation for the cell terminal

voltage is obtained that includes the polarization contributions from the factors in the SPM as well as those resulting from the lithium-ion diffusivity in the liquid electrolyte regions.⁴⁵

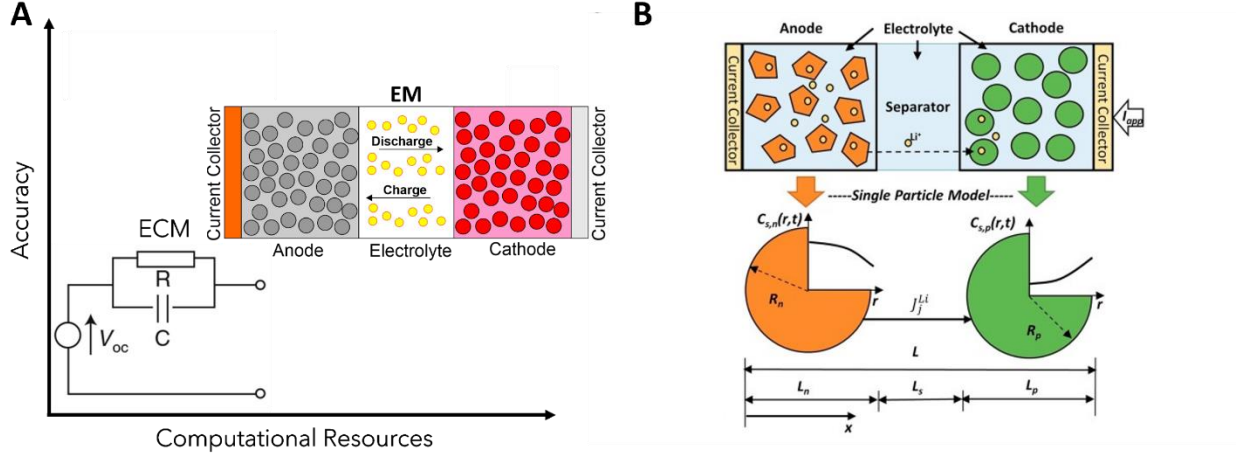


Figure 7: (A) The computation requirements and model accuracy of phenomenological battery cell-level equivalent circuit models (ECM) and physics-based continuum-level electrochemical models (EM).⁴⁶ (B)

The coordinate systems and governing PDEs of the single particle model (SPM).⁴⁷

Another approach used in the literature is to reduce the order of all or some of the PDEs outlined in the previous sections using mathematical approximations. The resultant equations, also known as reduced-order models (ROMs), are in the form of ordinary differential equations or algebraic equations. For instance, the PDEs on the P2D models can be represented using the state-space representation.⁴⁸ This representation, as represented below, consists of state and output equations and it gives the final equations in the form of algebraic equations that can calculate the output at discrete spatial and temporal points. For a linear time-invariant system, the discrete, the state-space representation is given by Equation 22 and Equation 23.^{48,49}

$$\mathbf{x}[k+1] = \mathbf{A} * \mathbf{x}[k] + \mathbf{B} * \mathbf{u} \quad \text{Equation 22}$$

$$\mathbf{y}[k+1] = \mathbf{C} * \mathbf{x}[k] + \mathbf{D} * \mathbf{u} \quad \text{Equation 23}$$

Where A, B, C, and D are the matrixes and are named state-transition matrix, input matrix, output matrix, and feedthrough matrix, respectively. Moreover, u, x, and y are the state, input, and output vectors, respectively. The P2D models can be expressed in the state-space representation through the discrete-time realization algorithms (DRA).⁴⁸ First, the P2D equations are linearized (wherever applicable) using the Taylor expansion series and then, the Laplace transfer functions of the linearized equations are evaluated. During the determination of transfer functions, transfers of the boundary conditions are also substituted to obtain a unique solution, and the integrator poles of the functions are eliminated using debiased variables.^{48,50–52} The pulse response of the transfer functions is computed and discretized by considering data points at specific time steps. The DRA takes advantage of the features of the Markov parameters and the Hankel matrices of the Markov parameters. Markov parameters are the response of the state-space system from a pulse signal and the determination of state-space matrices from a set of Markov parameters is called a “state-space realization problem”. The determined state-space matrices are not unique however all determined state-space matrices have similar inputs to output mapping^{48,51,53} The Markov parameter for a discrete-time is given by^{48,53}

$$\mathbf{g}_k = \begin{cases} \mathbf{D}, & \text{when } k = 0; \\ \mathbf{CA}^{k-1}\mathbf{B} & \text{when } k > 0 \end{cases} \quad \text{Equation 24}$$

Hence, a set of pulse responses from a state-space system would be $\{\mathbf{D}, \mathbf{CB}, \mathbf{CAB}, \mathbf{CA}^2\mathbf{B}, \dots\}$. Using the pulse response of the transfer function, the responses of the inverse transforms are also determined. Hence, the time series of pulse response are obtained and can be used to construct the Hankel matrix, which is factorized into the so-named observability and controllability matrix to obtain the A, B,C, and D (state-transition, input, output, and feedthrough matrices).^{48,51,53,54} For instance, Lee *et al.* used single-value decomposition to factorize the Hankel matrices. The Hankel (RHS of the equation below),

observability (first vector on the RHS), and controllability matrices (second vector on the RHS) are given below.⁴⁸

$$\begin{bmatrix} C \\ CA \\ CA^2 \\ \dots \\ CA^{n-1} \end{bmatrix} [B \quad AB \quad A^2B \quad \dots \quad A^{n-1}B] = \begin{bmatrix} CB & \dots & CA^{n-1}B \\ \vdots & \ddots & \vdots \\ CA^{n-1}B & \dots & CA^{2n-2}B \end{bmatrix} \quad \text{Equation 25}$$

The discrete state-space systems can be used to measure the electrode-electrolyte molar flux, lithium-ion concentrations in the solid-electrolyte interface and the electrolyte, and the potentials at the solid and electrolyte at discrete spatial and temporal points. Furthermore, these systems can model the above-mentioned battery cell variables with high accuracy when compared with the ones calculated from the model containing the original PDEs.^{51,54}

Other simpler approaches for ROM have been suggested in the literature, such as the Eigen expansion method and the PDE pertaining to the lithium concentration in the solid electrode region. The boundary conditions of this PDE are formulated so that it can be expanded using Fourier series and the resulting ODE of the lithium-ion concentration at the electrode particle's surface as a function of time is obtained. This ROM is particularly useful for the single particle model.⁴³ Another ROM approach is to formulate the PDE so that it gives a resulting ODE of the volume-averaged quantity of that variable.⁵⁵ The equations of these ROM methods have been described in more detail in 5.

2.6 Overview of the Application of Artificial Intelligence within Electric Vehicle Applications

Recently, Artificial intelligence (AI), which is defined as algorithms that support models aimed at mimicking natural thinking, perception, and action⁵⁶, has seen industrial and academic applications in the field of EVs and related infrastructures, such as in EV battery design and management, charging stations, and even smart grid⁵⁶⁻⁵⁹. Within the review's context, AI algorithms encompassing fields of machine learning and computational intelligence are considered. Depending on the problem, these AI algorithms

can outperform classical rule-based systems (also called expert systems), which use human knowledge to define rules within a system, due to their ability to find unusual trends and simpler implementation^{60,61}. Several attractive advantages of AI application include (1) EV cost reduction through optimal battery material design and manufacturing^{62,63}, (2) accurate range estimation to mitigate EV consumer range anxiety by predicting future driving conditions⁶⁴, (3) better EV energy consumption than traditional controls using AI controls for the EV auxiliary systems⁶⁵, (4) a potential for increased road safety and optimal traffic flow through connected and autonomous driving^{58,66}, and (5) a thorough and efficient modeling approach for optimal location and resource allocation for electric vehicle charging stations (EVCS) and energy scheduling for EV interaction with the smart grid⁶⁷.

For a better understanding of this topic, the AI techniques mainly used in EVs and related infrastructures are summarized and divided into machine learning (ML) and computational intelligence (CI), as shown in Figure 8. To be specific, ML models are very adept at finding relations and trends between inputs and outputs based on previous observations, and they require training on a previous dataset. The ML used in EVs and related infrastructures can be roughly classified into supervised, unsupervised, and reinforcement learning (RL). Supervised and unsupervised learning have been researched and applied in areas within EV and its infrastructure where large datasets are available or can be created, such as in EV battery state estimation and discovery of new materials for the EV batteries^{68,69}. The basic concept of them are displayed in Figure 8A-C, in which the DL involves the use of architectures involving neural networks (NN) with more than one hidden layer.⁷⁰ For instance, CNN, commonly used for image processing, as successive layers of convolutional and pooling layers that are connected to a fully connected neural network. Convolutional layers filter the image based on high-level image features while the pooling layer compresses the image to reduce its size for ease of handling (Figure 8B). Meanwhile, RL aims to learn the best course of action on its own through trial and error, the agent interacts with its environment through actions and is rewarded accordingly (Figure 8A). The agent interacts with the environment and gains a reward accordingly

based on its action. Through iterative learning, the agent learns and adopts a policy that aims to maximize its reward.⁷¹

Aside from ML, CI algorithms as shown in Figure 8D-E, are commonly used for solving search, optimization, and complex problems⁷². Within the context of EVs, CI algorithms are instrumental in solving complex dynamic optimization problems, such as the optimization of control systems within EVs, optimal placement of EVCS, and integration of EV infrastructure with the smart grid ⁷³⁻⁷⁵. As an illustration of the CI algorithm, consider particle swarm optimization (PSO) and genetic algorithm (GA). PSO, commonly used for search and optimization engineering problems, has a swarm of particles that search in the solution space. The particles communicate the particle's personal best and global best to find the solution (Figure 8D). Although the current EV market has not yet fully embraced AI, the number of related academic publications, patent applications, and the production scale have increased rapidly in the past decade. An overview specifically of the roles of AI in EVs and their infrastructure is rare but very important to accelerate the progress of R&D and mass industrial application and commercialization.

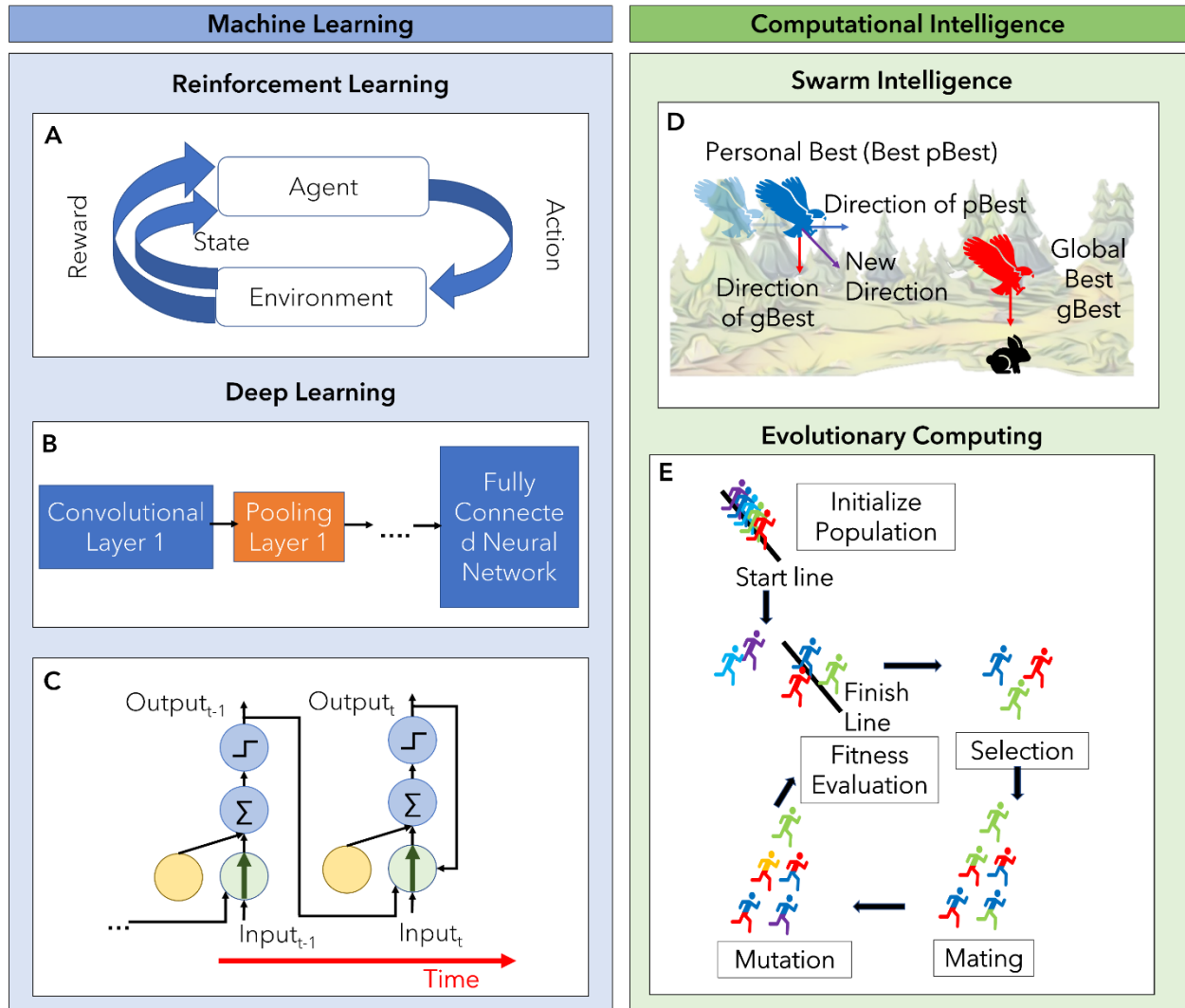


Figure 8: (A) Process flow for reinforcement learning. (B) Commonly used architecture for convolution neural networks (CNN). (C) Neural network of recurrent neural networks (RNN). RNN, commonly used for time series, has the output of the neural network at (t-1) time-step as the input of the at t time-step. The image shows a single RNN unrolled in time. (D) Depiction of the process flow of particle swarm intelligence (PSO). (E) Depiction of the process flow of genetic algorithm (GA). GA, commonly used for search and optimization engineering problems, has an initial population of solutions. During each iterative process, the population undergoes mutation and crossover operations to find the solution.

2.7 Physics-Based Continuum-Scale Battery Models and Machine Learning Models for Battery Management System Applications

2.7.1 Continuum Model Parameter Estimations

The material, geometrical, and physical parameters used in the electrochemical models can be determined experimentally or estimated using mathematical search algorithms. The experimental techniques, while accurate, are usually destructive, time- and capital-intensive. CI-based search algorithms, including GA^{76,77} and PSO⁷⁸, have been used in the literature to estimate the model parameters from experimental data. This approach for parameter estimation (PE) from experimental data is often termed an “inverse method” in literature.^{76,79} Compared to deterministic techniques for parameter estimations, CI-based algorithms are less prone to converge to a local minimum and, consequently, can provide a more accurate solution. This conclusion was reached by Colaco *et al.* when they compared CI-based search algorithms and deterministic techniques (e.g., Newton-Raphson method and Levenberg-Marquardt Method) in heat-transfer problems.⁸⁰ Moreover, PSO can lead to a faster solution convergence than GA because it allows for a higher solution exploration and diversity in the solution space.⁷⁸

The potential values with respect to time during battery charge and discharge are used as experimental data and compared with the simulated results obtained from the EMs. The accuracy of the results is dependent on the sensitivity of the model parameters to the potential.^{76,81,82} For instance, Santhanagopalan *et al.* mentioned that the diffusion coefficient of the negative electrode, for a LCO battery, has little effect on the potential voltage and, hence, it was kept constant during their PE procedure.⁸¹ Interestingly, Jokar *et al.* demonstrated that the sensitivity of each model parameter is dependent on the discharge rate and regions of the discharge curves (Figure 4B).⁷⁶ The overall goal of the inverse method is to minimize the difference between the experimental results and the ones obtained from the P2D model and ordinary least-square methods are used as the objective function.^{76,81} A set of initial model parameters (e.g., the electrode lithium initial and maximum electrode lithium concentrations, diffusion constant, and rate constant) is formulated (other model parameters are obtained from the relevant literature, datasheets, or experimentations). For

experimental data containing M charge and discharge cycles and N parameters to be estimated, the objective function (S) along with the optimization problem are formulated (vector form) as⁷⁶

$$S(\mathbf{P}) = \sum_{m=1}^M \sum_{n=1}^N \left(\mathbf{V}_{cell,m} - \mathbf{V}_{cell}(\mathbf{P}) \right)^T \left(\mathbf{V}_{cell,m} - \mathbf{V}_{cell}(\mathbf{P}) \right) \quad \text{Equation 26}$$

$$\min S = S(\mathbf{P}) \text{ subject to } P_{j,low} \leq P_j \leq P_{j,high}$$

Where V and P are the potential and parameter vectors. All estimated parameters are confined within bounds (represented by P_{low} and P_{high}). Furthermore, simplified EM models are employed to reduce the computational time for simulations. An example of such simplification can be that the flux of lithium ions in the electrode-electrolyte interface can be assumed to be constant or lithium-ion concentrations in the electrolyte can be assumed to be constant or a linear function.^{76,78} The search algorithm is run iteratively until the convergence criteria are met.

Recently, ML-based algorithms, including NN^{79,83}, RNN⁸⁴, and CNN⁸⁵ have also been utilized for PE and the methodology of this approach is illustrated in Figure 4C. Quite simply, the ML models are trained on various model parameters to output the cell's potential at various charging and discharging rates. EM is typically used to generate the training data.⁷⁹ Compared to CI-based approaches, the ML-based approach requires significant time for algorithm hyper-tuning and model training. However, the PE requires much lower computational time once the ML model is trained compared to the CI-based approach.⁷⁹

2.7.2 Electric Vehicles Range Estimations

Besides battery management technologies reviewed above, range optimization during EV driving can tremendously increase energy efficiency and save driving time. In general, range estimation (RE) is a step to achieve EV range optimization. As one of the most important research areas of EVs today, an accurate RE can largely mitigate 'range anxiety' faced by EV drivers due to limited driving range^{86,87}. EV drivers can make effective driving, parking, and charging decisions, and participate in more vehicle-to-grid (V2G)

charging. However, conventional RE methods are not always accurate mainly due to their negligence of some dynamically changing external and operational conditions^{86,88,89}. For example, the range estimators in Tesla's Model S use the energy consumption of the past few miles to estimate the future available range, without considering changes in environment, driving conditions, and behaviors.⁸⁶

While the range of commercially available EVs has been improving steadily in recent years, range anxiety can still influence potential EV buyers^{90,91}. EV range refers to the achievable driving distance before the need for an EV battery pack recharging or battery swap⁹². Various internal and external factors can affect the EV range. Internal factors that are inherent to the EV include the battery pack's terminal voltage, battery pack capacity, and the EV's exterior aerodynamic features. Also, external factors such as the driver's driving style, road conditions, and weather conditions can impact the EV range^{93,94}. Typically, standard drive cycles, such as highway fuel economy test (HWFET) and urban dynamometer drive cycles (UDDS), are used to compare the vehicular range and fuel economy^{95,96}. Research works have focused on estimating the range estimations using mathematical models that simulate the energy requirement of an EV during a trip. Hayes et al utilized the EV drivetrain parameters of the Nissan Leaf (model year of 2011) and Tesla Roadster (model year of 2008) into a simplified drivetrain model for their fuel consumption and range estimations⁹⁷. Shibata and Nakagawa outlined the EV energy and power consumption model based on the forces acting on the EV during its trip with the inclusion of the effects of air-conditioning⁹⁸. Abousleiman *et. al.* also proposed a similar EV model that considers the forces experienced by the EV from rolling resistance, road grade (gravity forces from the elevation on the roads), air resistance, and accelerating forces. Fiori *et. al.* further the effects of braking regeneration on the EV dynamical model⁹⁹.

While the above-mentioned research works introduced the EV dynamical model, they did not apply basic approximations to the LIBs. Within the LIB battery cell, there are various non-linear electrochemical phenomena at play during the battery cell charge or discharge operations. For instance, the LIBs display a non-linear relationship between the open-circuit voltage (OCV) and its state-of-charge (SOC)²³. The OCV refers to the thermodynamic terminal voltage at a given temperature. Moreover, there is electrochemical

polarization (i.e., the deviation of the battery cell terminal voltage from its OCV) during charge-discharge operations (also during the rest immediately after the charge-discharge) from various phenomena such as from the slow lithium-ion diffusivity in the solid electrode region and liquid electrolyte region. Phenomenological models, such as equivalent circuit models (ECM), and continuum-scale physics-based models (PBM) have been used to model such phenomena³³. Furthermore, these battery models can be incorporated with the thermal and degradation models^{43,100}. Barcelona et. al. applied ECM models to the standard drive cycles for EV range estimations¹⁰¹. When compared to PBM models, ECM models have a lesser number of model parameters, generally have faster solution times, and are easier to implement. They are easier to implement because of their simpler mathematical formulation.²³

Compared to traditional RE methods, AI can provide opportunities for accurate RE by mapping complex interactions between RE and factors affecting it. Furthermore, AI algorithms, including ML, can be used to accurately predict future environmental and driving conditions based on past and current conditions. These future predicted conditions can be used for a more accurate RE. AI algorithms have been used for RE by directly using environmental and driving behavior historical data^{102,103}, estimating the EV battery energy or power consumption^{89,104–107}, and identifying driving conditions and behaviors^{89,108,109}. For ML training, real-time historic discharge EV data is processed to extract relevant battery, vehicle, and external parameters (column Parameters in Table 2) and to remove missing and erroneous data. Moreover, the EV historic data can be fused with historic weather, traffic, and road conditions to include external parameters, which serves to increase the RE estimation accuracy through ML training^{102–104}. Parameters that have the most impact on RE, such as battery SOC and external temperature, can be estimated using correlations and are considered to reduce ML model complexity and training time.¹⁰² ML model can be trained to output EV range directly, or the EV future energy or power consumption.

2.7.3 Fault Detection

Although rare, a fire hazard from a thermal runaway in a LIB battery pack system poses a serious safety hazard. Predicting faults in a battery pack (in other words, early fault detection) is a desired feature of a

BMS, especially in EV applications. This safety feature enables the user and manufacturer to take early corrective actions before the predicted occurrence of the incident.⁴⁶ Extreme battery pack operating conditions (over-charge and discharge, extreme ambient temperature), improper electrical connection, and/or mechanical damage (to the individual cells and modules) can be the cause of faults in the battery pack.^{110,111} Simply measuring voltage across the battery load or observing the mean and standard deviation of a battery terminal voltage has been applied for battery fault determination applications, such as vehicle battery charging systems.^{112–115} However, due to the complex and non-linear potential behavior of a LIB, such simple detection algorithms cannot be implemented for EV applications.⁴⁶

ML-based techniques, especially deep learning algorithms, have been researched for early battery fault detection by training ML-algorithms to identify abnormal voltage and surface temperature behaviors, which are indicative of early signs of battery fault.^{110,111,116–118} Compared to fault detection techniques based on physical models, DL techniques are easier to implement since they do not require expertise in underlying physical models and tedious model parameter identification processes.¹¹⁹ As an example of research work, Yao *et al.* demonstrated that training a class of NN, general regression neural network, rolling variance on a historic dataset of a battery module containing 4 cells, can detect battery fault with an accuracy of 95 %.¹¹⁰ However, the voltage data had to be filtered from the sensor and vibration noise using waveform transforms and inverse, adding another computation step to fault identification. Another case in point is the work of Ojo *et al.*, where the researchers used a long-term memory (LSTM) neural network to calculate the cell's surface temperature from its voltage, current, SOC, and a data point from its historical surface temperature. The residual from the model and the actual cell surface temperature were compared to identify a faulty cell. The trained model successfully identified the time of fault in a cell, where the fault was induced by heating the battery using a heating pad.¹¹¹ In both works, the data was obtained from a workbench consisting of a few cells in series and not from an actual EV battery pack. Moreover, the fault was artificially introduced to the cells, and, therefore, it can differ from the nature of the fault that occurs in real-life EV operations. In contrast, Hong *et al.* and Li *et al.* used LSTM for voltage predictions in battery fault identification

applications using real-life EV BMS datasets.^{120,121} Li et al. trained LSTM on a dataset consisting of real-life BMS data from nine EVs to accurately predict the battery pack voltage values under normal non-fault circumstances. The final voltage value is weighted appropriately using predictions from LSTM and equivalent circuit model (ECM) to improve model accuracy. When the predicted voltage value is compared with the actual voltage for an EV with a known fault, the model is able to accurately identify faults based on a preset threshold value.¹²⁰ In all of the works mentioned above, the measurable value (voltage or surface temperature) has to be predicted in real-time, which requires considerable BMS storage and computational resources. Therefore, the practical application of the above techniques might be severely hindered due to the computational capabilities of the current BMS in EVs. Furthermore, LSTM has a complex neural structure and can suffer from unstable gradients, both of which can adversely impact its practical application for fault detection on a large scale.^{122,123}

3 Range Estimations

This section has been adopted from the following manuscript research articles.

- Moin Ahmed, Zhiyu Mao, Yun Zheng, Tao Chen, and Zhongwei Chen, “Electric Vehicle Range Estimations using Regression Techniques”, World Electric Vehicle Journal, 2021
- Moin Ahmed, Zhiyu Mao, Aiping Yu, Michael Fowler, and Zhongwei Chen. “User-Specific Range Estimations Using Application-Scale Electric Vehicle and Continuum-Scale Battery Cell Electrochemical Models.” Under Submission.

3.1 Chapter Objective

A methodology of the range estimation of electric vehicles that use lithium-ion battery packs was attempted in the works in this chapter. Amongst the functions of battery management systems for electric vehicles is to interface useful information to the user. In the following works, first, the important parameters of the electric vehicle’s battery pack, drive train, and performance specifications that affect its range are identified using a dataset of commercially available electric vehicles using self-learning regression techniques. Later, a more accurate range estimation methodology employing models for vehicle dynamics and battery cells is used for Tesla Model 3. This methodology for intended for battery management systems that interface with cloud-based systems. All of the models and simulations were performed using inhouse code using Python programming language.

3.2 Chapter Introduction

Recent years have seen a governmental and industrial push toward the mass adoption of electric vehicles (EVs) due to their various benefits, including lower reliance on unsustainable and non-renewable fossil fuels^{124–126}. Reduced global reliance on fossil fuels for transportation could mitigate some of their disastrous effects, including greenhouse gas emissions (GHG) and geopolitical tensions¹²⁷. However, it is important to note that GHG reduction from EV mass adoption is only possible when these electrified means of transport are incorporated into a renewable and sustainable energy grid^{125,126,128}.

However, despite their numerous benefits and the improvements in lithium-ion battery (LIB) technology, EVs still generally suffer from a lower range than conventional internal combustion engines, long charging times, and limited fast-charging infrastructure in some geographical areas¹²⁹. In combination, these limitations can induce “range anxiety” for EV users; hence, EV range estimations are an ongoing academic and industrial research area⁴⁶. EV range is described as the distance an EV can travel before recharging or battery swapping⁹². The manufacturer-reported vehicle range is measured by subjecting the vehicle to a controlled drive cycle. Commonly used drive cycles for EV range estimation include multiple cycles of the urban dynamometer driving schedule (UDDS), the highway fuel economy test driving schedule (HWFET), the new European drive cycle (NEDC), and the worldwide harmonized light vehicle test procedure (WLTP)^{95,96}. Since drive cycle protocols for multiple cycles of UDDS and HWFET are recommended by the U.S. Environmental Protection Agency (EPA), this drive cycle is referred to as EPA range throughout this work. EV design and performance specifications, external climate and road conditions, and consumer driving habits are critical parameters affecting EV range^{93,94}. EV design and performance specifications can include battery specifications, motor efficiency, curb weight, and vehicular body aerodynamics.

Many relevant works on EV range estimation focus on determining the remaining EV range during a driving trip using physical or empirical models^{97–99,130–132}. These models calculate the energy consumption of an EV based on the external conditions (e.g., road elevation and wind velocity) and EV parameters (e.g., weight, motor, braking efficiencies, etc.) and then determine the remaining EV range^{97–99,130–132}. Moreover, for more accurate EV range estimations, if the velocity–time series of an EV is known (based on the drive cycle), the required EV energy is then obtained for each time point during the EV trip. In 2011, Hayes et al. used the simplified drive train model for the range estimation of a Tesla roadster and a Nissan Leaf subjected to the EPA-based drive cycle. Their model calculates fuel economy to estimate the EV range for a given drive cycle. The authors used the published vehicle parameters for unknown parameters, while assumptions and estimations were made for the rest of the model parameters⁹⁷. Shibata et al. developed an EV power consumption model that included the energy consumption of HVAC and other auxiliary systems

during an EV trip. Furthermore, coulomb counting with the Kalman filter was used to estimate the remaining battery capacity at any point in the drive cycle. The authors of this work explored the effects of different drive cycles on EV power consumption⁹⁸. Abousleiman et al. used a similar model to compare model predictions with the energy consumption of a 2013 Fiat 500c driven on different routes. Depending on the route, the authors reported model discrepancies in the range of 0.66–2.9%¹³². While previously reported works assumed constant braking power regeneration, Fiori et al. improved on the power consumption model by incorporating the effects of EV deceleration on the braking power regeneration. In all the above works, the velocity–time characteristics and extensive vehicle parameters were required, and several model assumptions, such as constant external wind and road-related resistances, were made⁹⁹.

These EV energy consumption models can be improved by simulating EV battery pack voltage and capacity characteristics using various battery models, such as equivalent circuit models (ECM)^{101,132}. These battery models require their own model parameters. For instance, ECMs, which simulate the voltage and SOC characteristics of the battery from the drive current using an electric circuit, require parameters pertaining to the values of the circuit components¹⁰¹. For instance, Barcellona et al. used an ECM to simulate the remaining battery capacity and battery pack voltage during an EV trip while using the EV power consumption model. While these models can be implemented for real-time EV range estimation during a trip, these models are not easy to implement for consumers and manufacturers. EV driving range is an important decision factor for consumers when purchasing an EV¹³³. This work attempts to find the relationships between the most relevant EV design parameters and marketed vehicle range using simple regression models¹³³.

This chapter focuses on data-driven regression modeling of the impacts of EV characteristics, battery, and performance specifications on vehicle range. The data used for regression model training and testing included the specifications of commercial EVs; these data were collected using web-scraping and text-mining techniques. The first objective of this study was to ascertain significant parameters that affect EV range. To this end, correlations between range and other parameters were used. The second objective was

to train the regression model to predict EV range from significant parameters. The correlations and the simple equation obtained from the regression-trained model can be used by EV consumers, designers, and manufacturers to ascertain how different options available (such as an EV's top speed, body style, etc.) can impact its overall range. Furthermore, Vehicle dynamic model is used that estimates the battery pack energy requirements from the drive cycles and continuum-scale single particle model (SPM) is used as the battery cell model.

The subsequent sections overview the chapter objective, modeling approach and relevant equations, followed by the results and discussions. Finally, the conclusions section concludes this research work.

3.3 Main Contributions

- The determination of significant parameters that contribute to the EV range using the data-driven linear regression model. For this purpose, a dataset of commercially available was extracted and this dataset contains each EV's model, battery pack, drivetrain, and performance specifications. Various regression models were trained and tested, and a multivariable regression model was proposed.
- For the range determination of a specific EV, the vehicle dynamics model in conjunction with the battery cell model was used for the Tesla Model 3. The vehicle dynamics model was used to estimate the EV load during its drive cycle and ambient condition. Furthermore, a single particle model with the thermal model was used to simulate the battery cell terminal voltage and SOC during the EV load. The battery cell electrochemical parameters were obtained from battery cycling tests that were performed under various loads and temperatures. The obtained parameters for the battery cell were used in the battery cell simulations for EV range estimations.

3.4 Methodology

3.4.1 Data Collection and Preprocessing

Figure 9 below outlines the flowchart of the range estimation workflow used in this work.

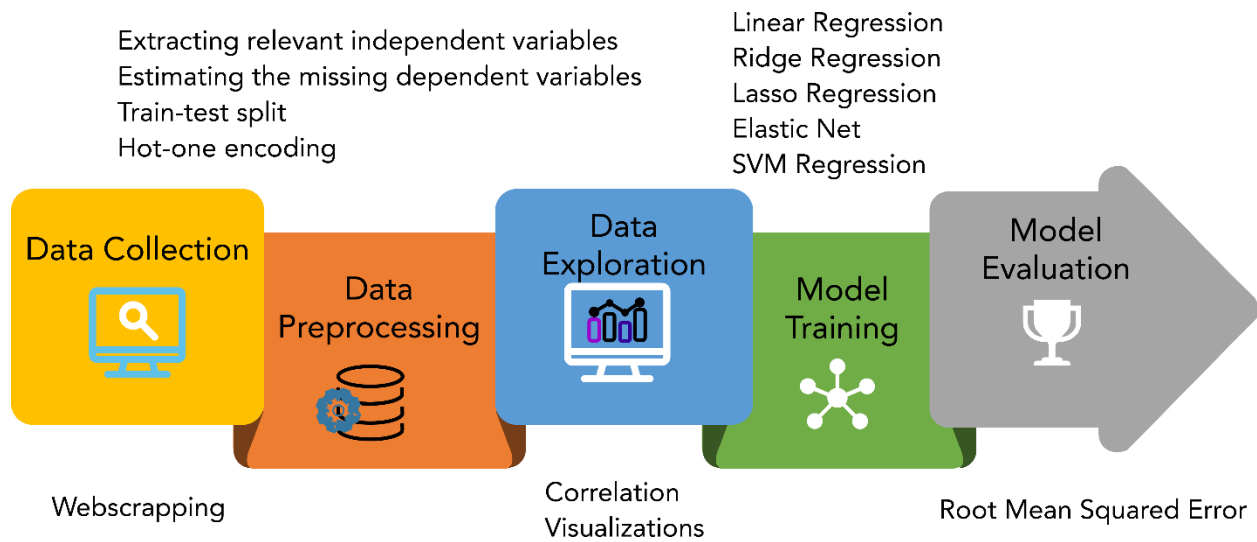


Figure 9: Flowchart of regression process.

The electric vehicle (EV) specification data were collected from a relevant website (EV Specifications) [23] through web-mining (using custom web-mining code in Python). The model year of the EVs ranges from 2008 to 2021. Data collected included the EV model, battery, dimensions, and performance specifications. At the time of analysis, the website (mentioned above) was missing information on Volkswagen's e-Gold 2018 model and, hence, this EV information was extracted from another website (Edmunds)¹³⁴. At this point, there were 317 EVs in the extracted dataset. Some of the information contained in the dataset (columns names in the dataset are referred to as the features in this work) are outlined in Table S1.

Table 2: Information Contained in Electric Vehicle (EV) Dataset

Feature Category	Feature
Model Specifications	Name
	Year
	Market (for e.g., China, North America, Europe)
	Price
	Trim (if applicable)
	Alias (if applicable)
Battery Pack	Voltage
	Number of Battery Modules

	Weight
	Volume
	Cooling System/Mechanism (air convection, water-based cooling, and active thermal management) *
	Battery Type as Advertised (lithium-ion, lithium polymer)
	Energy Density
Performance Specifications	Top Speed
	Acceleration
	Range
	Curb Weight
	Payload
	Gross Vehicle Weight Rating (GVWR)

For simplicity in further analysis, only the features in the dataset that have a maximal effect on EV range and were not sparse were considered. These features pertain to EV model information, battery characteristics, and performance metrics (Figure 10). The range estimated by multiple UDDS and HWFET ranges is referred to as the EPA range in this article.

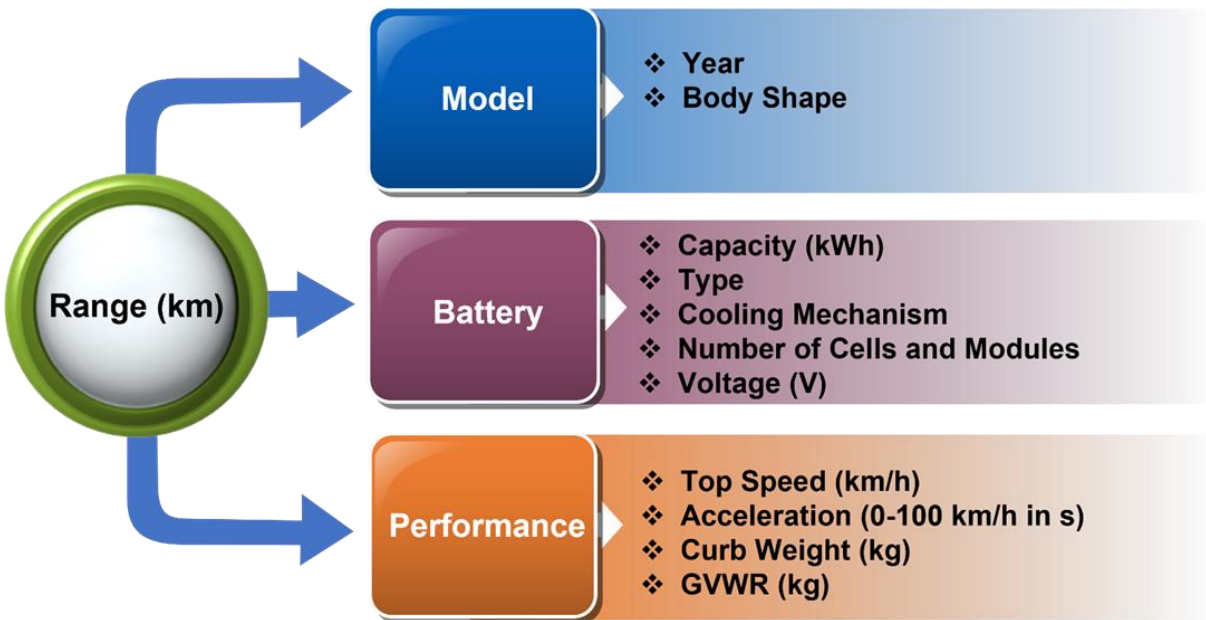


Figure 10: EV model, battery, and performance features considered from the dataset.

For many vehicles, the EV range was expressed in either EPA, NEDC or WLTP with several missing EPA range values. Therefore, EPA ranges of these vehicles were estimated using the WLTP and NEDC ranges using normal linear regression along with bootstrapping (as described in the section below). Bootstrapping is a resampling technique and, within the linear regression application, can be used for a more realistic linear regression parameter estimation in case of missing data.

Once the missing EPA ranges were estimated, the correlation between the features and the EPA range was determined using the Pearson correlation coefficient using the following equation (Equation 27).¹³⁵

$$r_{xy} = \frac{\sum(x_i - \bar{x}) \sum(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2} \sqrt{\sum(y_i - \bar{y})^2}} \quad \text{Equation 27}$$

In our case, x and y refer to the relevant independent variables and EV range, respectively. x_i and y_i denote the individual data-points while $\bar{x}$ and $\bar{y}$ denotes their mean values.

Next, a randomized train-test split of 80% - 20% was performed on the dataset. After further data analysis using visualizations, the independent variables that did not show much effect on the EV range were discarded. Furthermore, features containing qualitative information (type of battery, EV body type, and battery cooling system) were plotted for distribution and categories within a column were merged for simplicity. Furthermore, hot-code encoding was also performed on these qualitative features.

3.4.2 Linear Regression Parameters using Bootstrapping

Let $X = [x_1, \dots, x_n]^T$ and $y = [y_1, \dots, y_n]^T$ as the vector of regressor and response variables, respectively, where n is the total number of observations. In our case, x is the vector of EV range observations in WLTP (or NEDC), and y is the vector of EV range observations in EPA. For the observations where both regressor and response variables are present (where the corresponding $(k \times 1)$ vectors are denoted by X_{comp} and y_{comp}), the linear regression parameter (θ) in $y_{comp} = X_{comp} \theta + e$ is calculated using Equation 28.

$$\theta = (X_{comp}^T X_{comp})^{-1} X_{comp} y_{comp} \quad \text{Equation 28}$$

The values of y_{comp} were predicted using the relationship $y_{comp} = X_{comp}\theta$ and is denoted by $\hat{y}_{comp}$. Subsequently, the residual, $e = y_{com} - y_{comp}$, was computed for k observations. It should be noted that there are $(n-k)$ observations where the response variable (EPA range) is missing. Each of observations, with missing response variable, was assigned a random calculated residual with replacement. From the assigned residual and the predicted value, the actual value of response variables of the missing data was calculated. The bootstrapped linear regression parameter (θ_b) was calculated for all the observations using Equation 29.

$$\theta_b = (X^T X)^{-1} X y \quad \text{Equation 29}$$

The above steps were repeated 1000 times and the average bootstrapped linear regression parameters was calculated for each iteration.

3.4.3 Regression Techniques

Regression analysis is a simple yet versatile tool for exploring and mapping the relationship between variables. In this work, a multiple linear regression model is used to model the EPA range from multiple factors that affect it. Furthermore, the regularized linear regression models are utilized and explored to prevent data over-fitting and are ridge regression, lasso regression, and elastic net.¹³⁶

Support vector machines (SVM), another popular machine learning (ML) technique, can perform classification, regression, and outlier detection. Moreover, they are applicable on a small and medium dataset. When used for classification problems, the objective of the SVM is to separate different classes with the highest possible difference between them. However, when SVM are used for regression problems, the objective changes to achieve the lowest possible difference between different classes¹³⁷. In this work, linear regression and SVM were explored mainly due to their suitability on small datasets. Below, a mathematical overview of linear regression and SVM is presented.

3.4.3.1 Linear Regression

A linear regression model was trained to predict the EV EPA range from independent variables and can be concisely represented as follows.

$$\hat{y} = \boldsymbol{\theta} \cdot \mathbf{x} \quad \text{Equation 30}$$

Where $\hat{y}$ denotes the model predicted value while $\boldsymbol{\theta}$ and $\mathbf{x}$ denote the model's parameter vector and instance's feature vector. The normal equation used to solve for $\boldsymbol{\theta}$ is given by

$$\hat{\boldsymbol{\theta}} = (\mathbf{x}^T \mathbf{x})^{-1} \mathbf{x}^T \mathbf{y} \quad \text{Equation 31}$$

The mean squared error was used as the cost function for the linear regression and is expressed as

$$MSE = \frac{1}{m} \sum_{i=1}^m (\boldsymbol{\theta}^T \mathbf{x}^i - y^i)^2 \quad \text{Equation 32}$$

Where m is the total number of datapoints. In order to prevent the possibility of over-fitting the model, regularized forms of linear regression were also trained. These forms include ridge, lasso, and elastic regressions. The cost functions for ridge, lasso, and elastic regressions are presented below in order

$$\begin{aligned} J(\boldsymbol{\theta}) &= MSE(\boldsymbol{\theta}) + \alpha \left(\frac{1}{2} \right) \sum_{i=1}^n \theta_i^2 \\ J(\boldsymbol{\theta}) &= MSE(\boldsymbol{\theta}) + \alpha \sum_{i=1}^n |\theta_i| \\ J(\boldsymbol{\theta}) &= MSE(\boldsymbol{\theta}) + r \alpha \sum_{i=1}^n |\theta_i| + \alpha \left(\frac{1-r}{2} \right) \sum_{i=1}^n \theta_i^2 \end{aligned} \quad \text{Equation 33}$$

Where α is the regularization parameter used for ridge, lasso, and elastic net regressions and r is the additional mix-ratio parameter used in elastic net regression.

3.4.3.2 Support Vector Machines

Support Vector Machines (SVM) can be used for regression problems and are suitable for small sample sizes¹³⁸. The mapping function of a dataset $\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ is expressed as given below.^{138,139}

$$f(x) = \sum_{i=1}^n w_i \phi(x_i) + b \quad \text{Equation 34}$$

Where w_i and b , the weight vector and a constant, respectively, are found by first defining the following optimization problem. Also, the function, $\phi(x_i)$, denotes the mapping (can be non-linear for generality) in the feature space. This optimization problem ensures that the function above, $f(x)$, is as flat as possible.^{138,140}

$$\begin{aligned} & \text{Min } \frac{1}{2} \|w\|^2 \\ & \text{s. t. } \begin{cases} y_i - \phi(w, x_i) - b \leq \varepsilon \\ \phi(w, x_i) + b_i - y \leq \varepsilon \end{cases} \end{aligned} \quad \text{Equation 35}$$

By adding slack variables (ξ_i and ξ_i^*), the mathematical representation of the optimization problem is represented below.

$$\begin{aligned} & \text{Min } \frac{1}{2} \|w\|^2 + \sum_{i=1}^n C_i (\xi_i + \xi_i^*) \\ & \text{s. t. } \begin{cases} y_i - \phi(w, x_i) - b \leq \varepsilon + \xi_i \\ \phi(w, x_i) + b_i - y \leq \varepsilon + \xi_i^* \\ \xi_i \xi_i^* \geq 0 \end{cases} \end{aligned} \quad \text{Equation 36}$$

The solution of this problem is expressed as given below.

$$f(x) = \sum_{i=1}^n (\alpha_i - \alpha_i^*) K(x_i, x) + b \quad \text{Equation 37}$$

Where the variables, α_i and α_i^* , are the Lagrange multipliers and the function, K , is the so-called kernel function. The radial basis function is used for the kernel function in this work.

3.4.4 Model Training

The models were trained using the LinearRegression and SVC classes in python's SciKit Learn package¹⁴¹. A randomized train-test split of 80% - 20% was performed on the dataset and hot-code encoding was also performed on qualitative features (battery cooling mechanisms, type of lithium-ion battery, and body style). Due to the small size of the dataset, the hyperparameters of the regression learning techniques were kept constant, and, hence, the dataset was not split for validation.

3.4.5 Model Evaluation

Root mean squared error (RMSE) was used to evaluate the model and is expressed by¹⁴²

$$RMSE = \frac{\sqrt{\sum_{i=1}^m \hat{y}_i - y_i}}{m} \quad \text{Equation 38}$$

Where $\hat{y}_i$ and y_i denote the model predicted and actual values while m is the total number of observations/datapoints.

3.4.6 Vehicle Dynamics Model

The purpose of the vehicle dynamics model, presented here, is to simulate the EV load at each time step. This EV load is later used by the battery models to simulate the current required by the battery cell and the resulting change in the battery SOC. The vehicular dynamic model described here requires the drive cycles and the vehicle parameters. The drive cycle is the time series of the desired vehicle velocity (Figure 11) urban dynamometer driving schedule (UDDS) (Figure 11) and highway fuel efficiency (HWFET) drive cycles are considered in this work. UDDS and HWFET drive cycles were originally defined by the United States Environmental Protection Agency to represent the urban and highway driving cycles, respectively¹⁴³. As can be observed from the speed profile of these drive cycles (Figure 11), the speed of the vehicle is

usually above and below 65 km/h for the HWFET and UDDS respectively. Meanwhile, the vehicle parameters are broadly related to the specifications of the vehicle's battery pack, motor, wheel, and drivetrain (Table 3). At each time step, the desired acceleration, accelerating force, aerodynamic force, rolling force, and grade force (Equation 40-Equation 44).¹⁴⁴

$$a_{desired} = \frac{v_{desired} - v_{actual}}{\Delta t} \quad \text{Equation 39}$$

$$F_{acceleration} = \frac{(J_{motor} + J_{gearbox}) * N^2 + N_{wheels} * J_{wheel}}{R_{wheel}} \quad \text{Equation 40}$$

$$* a_{desired} \quad \text{Equation 41}$$

$$N = \frac{RPM_{motor}}{RPM_{wheel}}$$

$$F_{aerodynamics} = \frac{1}{2} (\rho_{air} * A_{frontal} * F_{aerodynamics} * C_d * v[k-1])^2 \quad \text{Equation 42}$$

$$F_{rolling} = C_r * m_{max} * g \quad \text{Equation 43}$$

$$F_{grade} = m_{max} * g * \sin(\theta) \quad \text{Equation 44}$$

The desired motor torque is then computed. Note that the F_{brake} is the brake force that represents the friction force from other contributions and is considered constant.

$$\tau_{desired} = (F_{acceleration} + F_{aerodynamics} + F_{rolling} + F_{grade} + F_{brake}) * \frac{r_{wheel}}{N} \quad \text{Equation 45}$$

The AC induction motor of the EV has a rated RPM, RPM_{rated} , below which it is able to deliver a constant torque. If the required torque is higher than its rated RPM, it is able to deliver constant motor power. Furthermore, the motor has a maximum deliverable speed, RPM_{max} . The actual force, and acceleration are computed as follows (Equation 46 and Equation 50).¹⁴⁴

$$F_{actual} = \tau_{limited} * \frac{N}{r_{wheel}} - F_{aero} - F_{rolling} - F_{grade} - F_{brake} \quad \text{Equation 46}$$

$$a_{actual} = \frac{F_{actual}}{m_{equivalent}} \quad \text{Equation 47}$$

$$v_{test} = v_{actual}[k - 1] + a_{actual} * \Delta t \quad \text{Equation 48}$$

$$v_{motor} = \frac{v_{test} * N * 60}{2 * \pi * r_{wheel}} \quad \text{Equation 49}$$

$$v_{actual} = \frac{v_{limited} * 2 * \pi * r_{wheel}}{60 * N} \quad \text{Equation 50}$$

Finally, the motor power and the battery pack current as computed as follows.¹⁴⁴

$$P_{motor} = 2 * \pi * \left(\frac{RPM_{motor} + RPM_{motor}}{2} \right) \quad \text{Equation 51}$$

$$P_{battery} = P_{overhead} + \frac{P_{motor}}{\varepsilon_{drivetrain}} \quad \text{Equation 52}$$

$$I_{pack} = P_{battery} * \frac{1000}{V_{nom,pack}} \quad \text{Equation 53}$$

At each time step, the desired acceleration and resulting forces on the EV are calculated from the desired user speed. Then, the actual forces and acceleration are determined from the limiting motor and drivetrain characteristics. Overall, the vehicle dynamics model presented here has the following model assumptions.

- The energy and power requirements from the load are supplied by the battery pack at each time step.¹⁴³
- The gearbox ratio is fixed to a constant value. Most commercially available EVs have a single-gear ratio since the powertrain is connected to the wheel axle through a single gear ratio.¹⁴⁵ This gear ratio for the Tesla Model 3 is reported to be around 9 in the literature.¹⁴⁶
- Brake force, which represents the contributions from other sources (e.g., mechanical resistance sources in the EV's brake pads¹⁴³), is considered constant.
- A road grade of zero was assumed for the duration of the EV's drive cycles. Furthermore, an ambient temperature of 25°C was assumed.

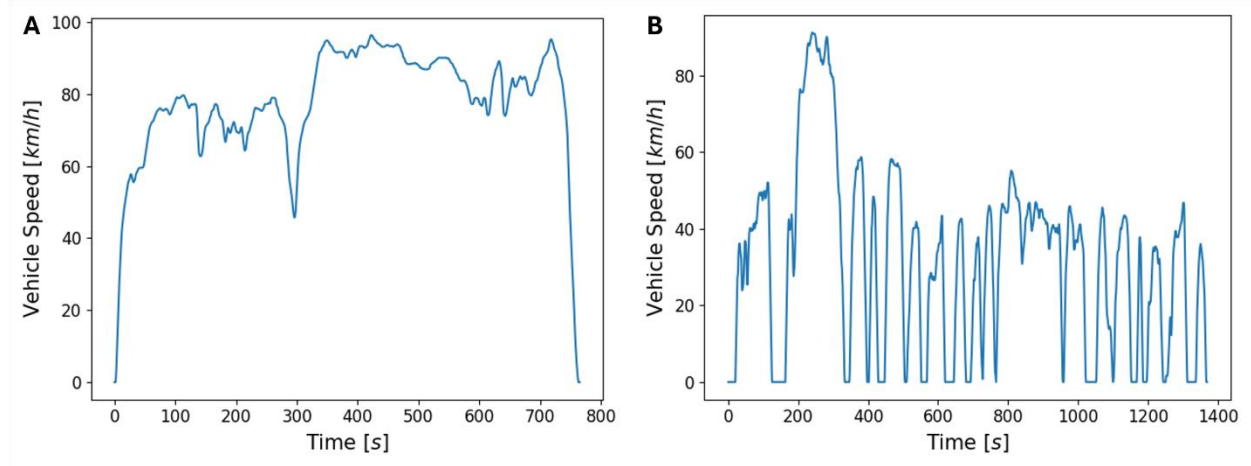


Figure 11: (A) HWFET and (B) UDDS drive-cycles

Table 3: Parameters for the Vehicle Dynamics Simulations

Classification	Name	Value
Basic Vehicle	Model Name	Model 3
	Year	2021
	Manufacturer	Tesla
	Trim	Standard Range Plus
	Positive Electrode Chemistry	NCA
Battery cell	Manufacturer	Panasonic
	Format	2170
	Capacity [$A\ hr$]	4.5
	Mass [g]	68
	$V_{\max}$ [V]	4.2
	$V_{\min}$ [V]	2.5
Battery Module	N_s	31

	N_p	24
	Overhead Mass [%]	0.5
Battery Pack	$N_{\text{module, series}}$	4
	Overhead mass [%]	0.33
	Efficiency	0.96
Motor	Type	Permanent magnet synchronous
	L_{max}	330
	$\text{RPM}_{\text{rated}}$	8000
	RPM_{max}	15000
	Efficiency	0.95
	Inertia [kg/m^2]	0.2
Drive Train	Inverter Efficiency	0.94
	Fraction of Regeneration Torque	0.9
	Gear Ratio	9
	Gear Inertia	0.05
	Efficiency	0.93
Vehicle	Drag Coefficient	0.23
	Frontal Area [m^2]	2.26
	Mass [kg]	1829.99878
	Payload Capacity [kg]	471
	Overhead Power [W]	200

3.4.7 Battery Cell Model

Single particle model (SPM) is used to model the cell terminal potential in this work. During the lithium-ion battery (LIB) discharge operations, there is a movement of lithium-ions from the interstitial sites in the

electrochemically active region of the positive electrode, through the liquid electrolyte region, to the interstitial sites in the negative electrode region ⁷. The single particle model (SPM) is used as the continuum-scale battery model for the battery cells in this work. In SPM, the lithium-ion flux through the electrode region is assumed to be constant and is calculated using the expression below.⁴³

$$j_l = \frac{I_{app}}{FS_l} \quad \text{Equation 54}$$

The electrochemical active area of an electrode, S_l , is further expressed as follows.

$$S_l = \frac{3 * \epsilon_l * Vol_l}{R_l} \quad \text{Equation 55}$$

The electrodes are modeled using perfectly spherical particles and lithium-ion diffusion across these electrode particles is assumed to be driven by Fick's law (i.e., the flux is opposite the concentration gradient, $\mathbf{j} = -\nabla c$). The partial differential equation (PDE) that represents the temporal and spatial concentration variation is presented below (Equation 56), along with the boundary (BC) (Equation 57 and Equation 58) and initial condition (IC) (Equation 59) ^{28,33,43}.

$$\frac{\partial c_{s,l}}{\partial t} = \frac{1}{r^2} \frac{\partial \left(D_{s,l} r^2 \left(\frac{\partial c_{s,l}}{\partial r} \right) \right)}{\partial r} \quad \text{Equation 56}$$

$$\left. \frac{\partial c_{s,l}}{\partial r} \right|_{r=0} = 0 \quad \text{Equation 57}$$

$$D_{s,l} \left. \frac{\partial c_{s,l}}{\partial r} \right|_{r=R_l} = -j_l \quad \text{Equation 58}$$

$$c_{s,l}(r, t = 0) = c_{s,l}^{init} \quad \text{Equation 59}$$

As mentioned earlier, SPM is used to model the cell terminal voltage. The model assumptions of the SPM, about the calculation of cell terminal voltage, are summarized in the list below.

- The lithium-ion flux (denoted by the variable j_l) entering or exiting the negative and positive electrodes is constant throughout the negative and positive electrode regions, respectively.⁴³ Note, that this flux is calculated by Equation 54.
- The overpotential from the concentration and migration effects in the electrolyte during battery cell charge/discharge operations are ignored.⁴³ These contributions along with the other overpotential contributions (e.g., the electric resistances in the current collector and external connections) are coupled in the bulk resistance terms. For low charge and discharge C-rates (such as in the EV applications) the overpotential from the electrolytes dynamics can be ignored.^{43,44}
- The anodic and cathodic apparent transfer co-efficient (α_a and α_c) as introduced in the Butler-Volmer equation (refer to Equation 4) are assumed to be equal and equal to 0.5.^{19,43,45}
- The resulting overpotential, loss of lithium-ion, and battery cell internal resistance change from the side reactions during battery cell charge and discharge operations are ignored.

Furthermore, the lumped thermal model is applied to calculate the thermal evolution of the battery cell temperature. The energy balance is represented by the following ordinary differential equation.

$$\rho C_p Vol_{cell} \frac{dT}{dt} = IT \left[\frac{\partial U_p}{\partial t}(soc_{p,surf}) - \frac{\partial U_n}{\partial t}(soc_{n,surf}) \right] + I * (V_{cell} - (U_p(soc_{p,surf}) - U_n(soc_{n,surf}))) + hA_{cell}(T_{cell} - T_{amb}) \quad \text{Equation 60}$$

In the above equation, T represents the temperature of the battery cell. Moreover, h , C_p , and A_{cell} represent the battery cell heat transfer coefficient, specific heat capacity, and the surface area, respectively. Also, in the above equation, the three terms on the RHS represent the thermal contributions from the reversible electrode reaction entropy change, irreversible electrode polarizations, and the heat transfer from the

surroundings. In the above thermal model, the spatial temperature profile throughout the battery cell was ignored in favor of bulk cell temperature.

The battery pack current as obtained from the vehicle dynamics model is assumed equally distributed amongst the parallel connected battery cells. Furthermore, it is assumed that all the battery cells in the battery pack have equal capacity and internal resistance in this work. The EV range was estimated by running the battery cell model through repeated cycles of UDDS and HWFET until the minimum battery cell terminal voltage was reached. An average weight of 45% and 55% was used for the HWFET and UDDS drive cycles for the EV range estimations¹⁴⁷.

3.4.8 Battery Cell Parameter Estimations using Genetic Algorithm

The parameters for the ESPM are estimated using the genetic algorithm (GA). For the parameter estimation using GA, the range of expected parameter values of the relevant parameters are inputted into the GA, and the initial population is generated with a set of parameters randomly selected using the normal distribution. The simulations are run on all of these parameter sets, and the fitness of these results is evaluated using the mean-squared error (MSE) as represented below.

$$MSE = \frac{\sum (V_{sim} - V_{actual})^2}{N} \quad \text{Equation 61}$$

In the above expression, V_{sim} and V_{actual} represent the cell terminal voltage as obtained from the ESPM simulation and experiments, respectively. Meanwhile, N represents the number of datapoints. The parameter sets are ranked based on their fitness. The elite parameter set is retained for the next generation, while the cross-over operation is performed on the parent members.

$$\alpha_i = \alpha_i \beta + \alpha_i (1 - \beta) \quad \text{Equation 62}$$

3.4.9 Experimental

The experimental dataset for this work was obtained from the open-source dataset and a detailed experimental description is explained in the relevant citation ¹⁴⁸. The positive electrode chemistry of the battery cells in the subject were lithium-nickel-cobalt-aluminum oxides (NCA). Briefly, the 4.5 Ahr battery cells from Tesla/Panasonic with the cylindrical-2170 form factor (21 cm in diameter and 70 cm in height) were discharged using an Arbin cell cycler with voltage and current control accuracy of ± 2 mV and ± 24 mA, respectively. The battery cells were exposed to controlled ambient temperature using the Envirotronics SH16C environmental thermal chamber. The drive-cycles considered in this work consist of (1) the constant current charge and discharge at low C-rate (0.05C) and higher C-rates (0.33C and 0.5C) at different temperatures, (2) hybrid pulse power characterization (HPPC), and (3) custom discharge ¹⁴⁸.

3.5 Results and Discussions

The missing values of the EPA range (EPA range refers to the range estimated from multiple cycles of UDDS and HWFET drive cycles) were estimated using the WLTP and NEDC range values. Figure 3 shows EPA plotted against WLTP and NEDC. The Pearson correlation coefficient between EPA and WLTP is 0.96, while it is 0.98 between EPA and NEDC. In the dataset, out of 317 entries, only 176 entries had an EPA range. In order to get estimates for the missing EPA range, the bootstrapped linear regression parameters (Figure 12A and B) between EPA and WLTP, and EPA and NEDC were used. The bootstrap method was performed for 2000 iterations until the bootstrapped linear regression parameters (coefficient and the y-intercept) had stabilized (Figure 12C and D).

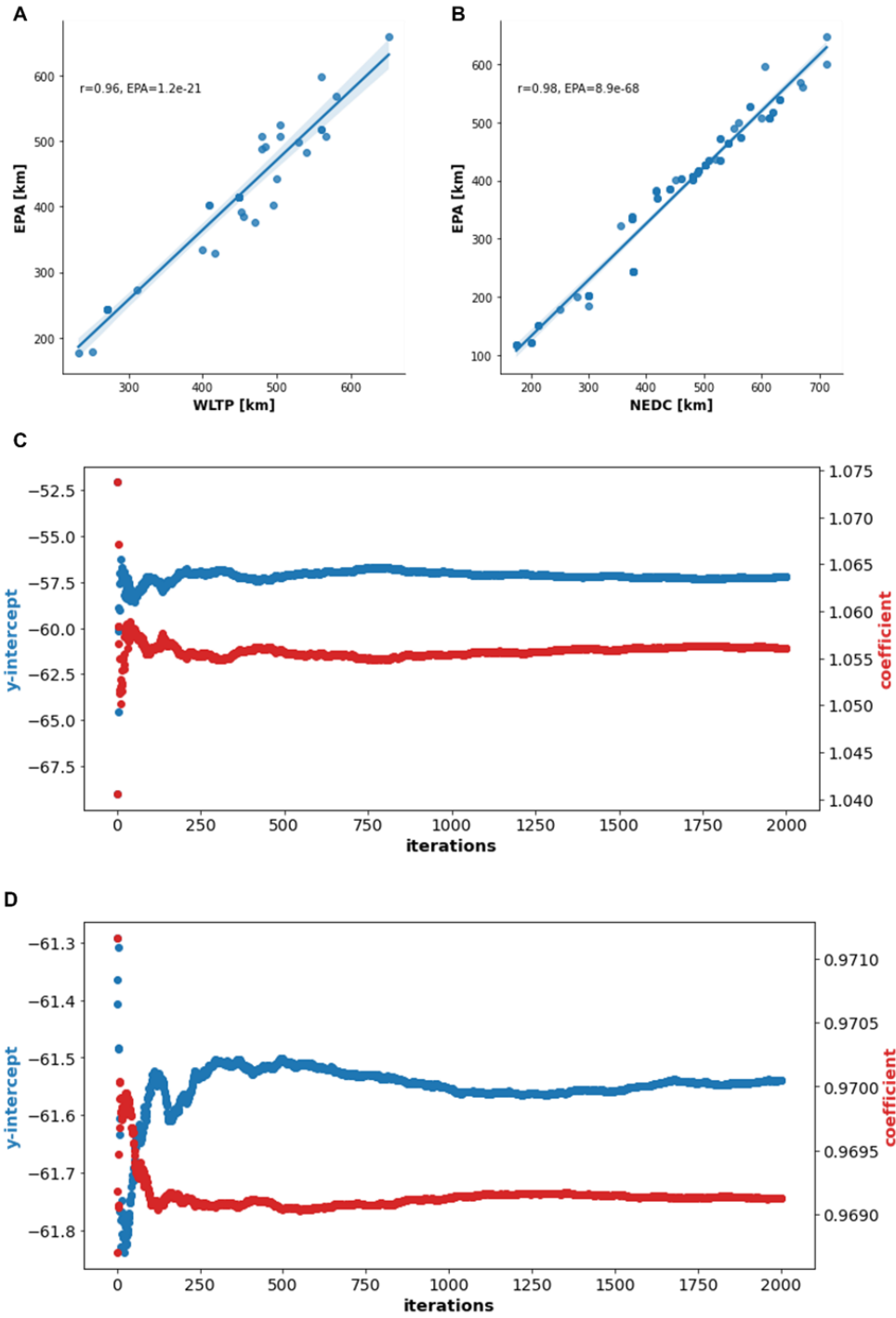


Figure 12: A plot, along with the line of best fit and Pearson correlation coefficient between (A) EPA and WLTP (B) EPA and NEDC. All units are in km. Furthermore, the bootstrapped linear regression parameters for each iteration are presented for (C) EPA and WLTP and (D) EPA and NEDC.

The relationship between the range are given below.

$$EPA\ range = -34.721 + 1.049 * WLTP\ range$$

Equation 63

$$EPA\ range = -38.9802 + 0.9703 * NEDC\ range$$

Where the ranges are in units of miles.

Next, the independent variables which are considered to have the highest impact on EPA range were retained (see Methodology section for specific variables considered), while other variables were removed for further consideration.

This reduction was made to simplify the analysis. Table 1 summarizes the correlations between the independent variables and the dependent variable (EPA range). Battery capacity, top speed, and curb weight seem to show a strong positive correlation to EPA range, while acceleration shows a strong negative correlation. This negative correlation can be explained by the physics-based energy consumption models of an EV. When all operating conditions and vehicle parameters are held constant, higher acceleration requires higher power output from the battery pack that drains the battery state-of-charge faster^{99,132,143}.

Table 4: Correlations between independent variables and EPA.

Variables	Correlations
Battery Capacity (kWh)	0.903860
Top Speed (mph)	0.794916
Curb Weight (lb)	0.700247
Number of battery cells	0.689241
GVWR (lb)	0.660293
Model Year	0.262226
Voltage (V)	0.187742
Number of battery modules	0.041696
Acceleration from 0 to 100 km/h (s)	-0.839825

Figure 4 shows the plots for all the independent variables against EPA. While model year seems to have a low impact on the EPA from the correlation (Table 4) and plot (Figure 13A), it was kept for further analysis because it is well-known that the range of EVs has been steadily increasing over the years ¹⁴⁹. This low correlation and unclear linear trend might arise from differences in EPA range between different EV body shapes (Figure 14A). Since batteries provide the energy source for the EV propulsion system and other auxiliary systems, the positive linear relationship between the battery capacity and range is expected and demonstrated by the high correlation in Table 1 and Figure 13B.

Usually, parameters for battery design, e.g., number of battery cells and modules, correlate with single-cell capacity and voltage to obtain battery pack capacity (kilowatt-hours) for EV powertrain requirement. In this work, instead of single-cell specification for each EV model, battery capacity is provided as an entire battery pack kilowatt-hour in EV level. Therefore, the number of cells, modules, and voltage would show an uncertain correlation to battery capacity and have a low correlation to EPA and uneven distribution, as Table 4 and Figure 13C-E show separately. These parameters were not considered in further calculation and analysis.

Generally, higher battery capacity delivers higher energy. It is reasonable to expect that EVs with an increased battery capacity will have a higher top speed and acceleration, as Figure S1A-B shows, and further supported by a higher EPA as indicated in Table 4 and Figure 13F-G. Moreover, The EV with higher battery capacity also has higher curb weight and GVWR, as Figures S1C-D show separately, which explains the positive correlations between the EPA and both curb weight and GVWR (Table 4 and Figure 13H-1).

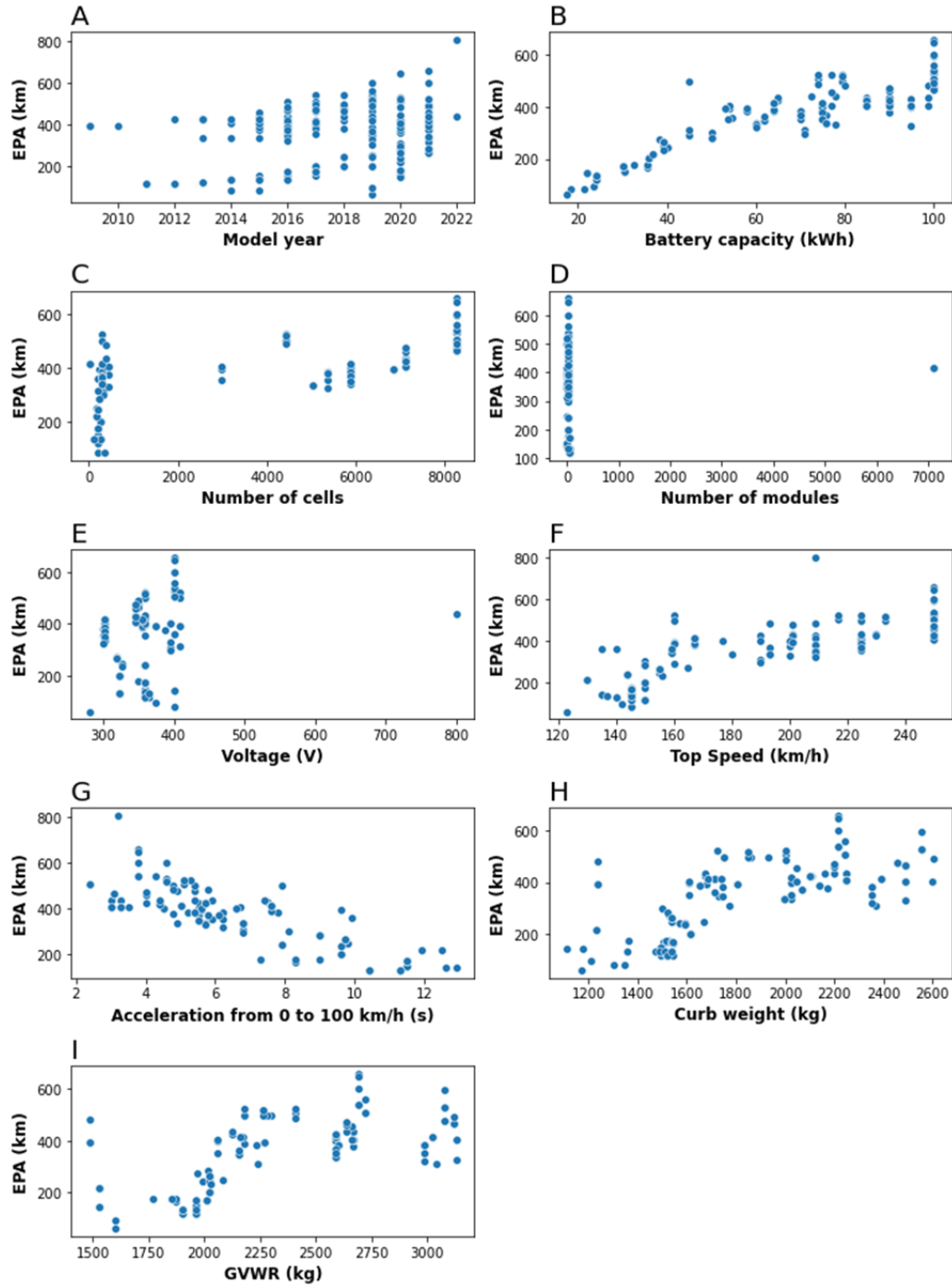


Figure 13: Plots of independent variables against EPA range, where the independents variables are (a) model year (b) battery capacity (in kWh), (c) number of battery cells in an EV, (d) number of battery modules, (e) EV battery pack voltage, (f) EV top speed (in km/h), (g) EV acceleration from 0 to 100 km/h (in s), (h) EV curb weight (in kg), and (I) Gross Vehicle Weight Rating (in kg).

In order to understand the impact of EV body shape on EPA, plots of various independent variables are plotted against EPA with data points colored differently based on body shape (Figure 14). A general trend, where the sedan has the highest and Hatchback has the lowest EPA, is observed. Sedan generally has a higher battery capacity than Hatchback (Figure S1B-D and Figure 6B), which might explain the EPA range difference. Furthermore, Hatchbacks experience more drag forces from its increased weight compared to Sedan (weight difference of Hatchback and Sedan is shown in Figure 6E) when external conditions are held constant¹⁵⁰, and, thus, additional battery capacity is utilized to overcome this drag.

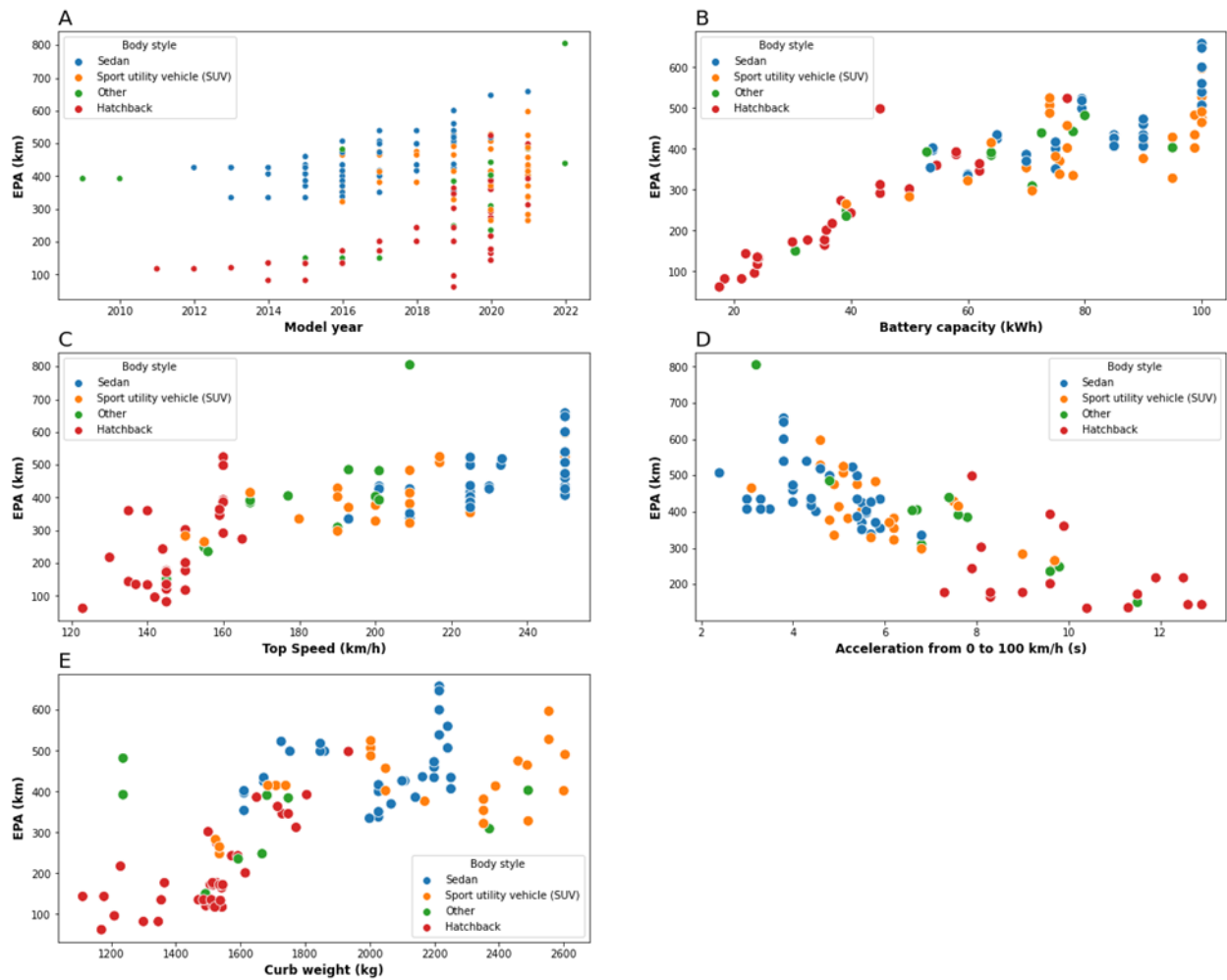


Figure 14: Scatter plot of independent variables with EPA with scatter points colored differently based on EV body style.

Figure 15 presents the distributions (histogram) and the impact on EPA (box plot) for qualitative information. It should be noted that some categories within these features were merged to even out the distributions. For example, there were nine categories of battery cooling mechanisms, eight of which were merged into a category 'Other'. Similarly, eight categories of EV body types (as presented in Figure 15) were condensed into four categories. From the box plot, it can be seen that each category has a noticeable difference in mean and range of EPA range values. Consistent with previous observation in Figure 5 above, the median and interquartile range (IQC) for the EV range with respect to the body style follows the order: $IQC(\text{Sedan}) > IQC(\text{SUV}) > IQC(\text{Other}) > IQC(\text{Hatchback})$ as observed in Figure 6A. EV with a water-based cooling mechanism has a higher median and IQC than other circulation mechanisms, which consist mainly of air-based cooling systems (Figure 15B). This thermal observation is consistent with the literature¹⁵⁰. Furthermore, given that the IQC for the EVs using lithium-ion and lithium-polymer battery is roughly the same and the data is unbalanced (i.e., much greater data points for lithium-ion batteries than that for lithium-polymer), it is difficult to ascertain the effect of different lithium-ion battery types on the EV range from Figure 15C alone. However, from the linear regression equation (Equation S6), there is a detrimental effect of different using lithium-ion batteries on EV range compared to lithium-polymer batteries.

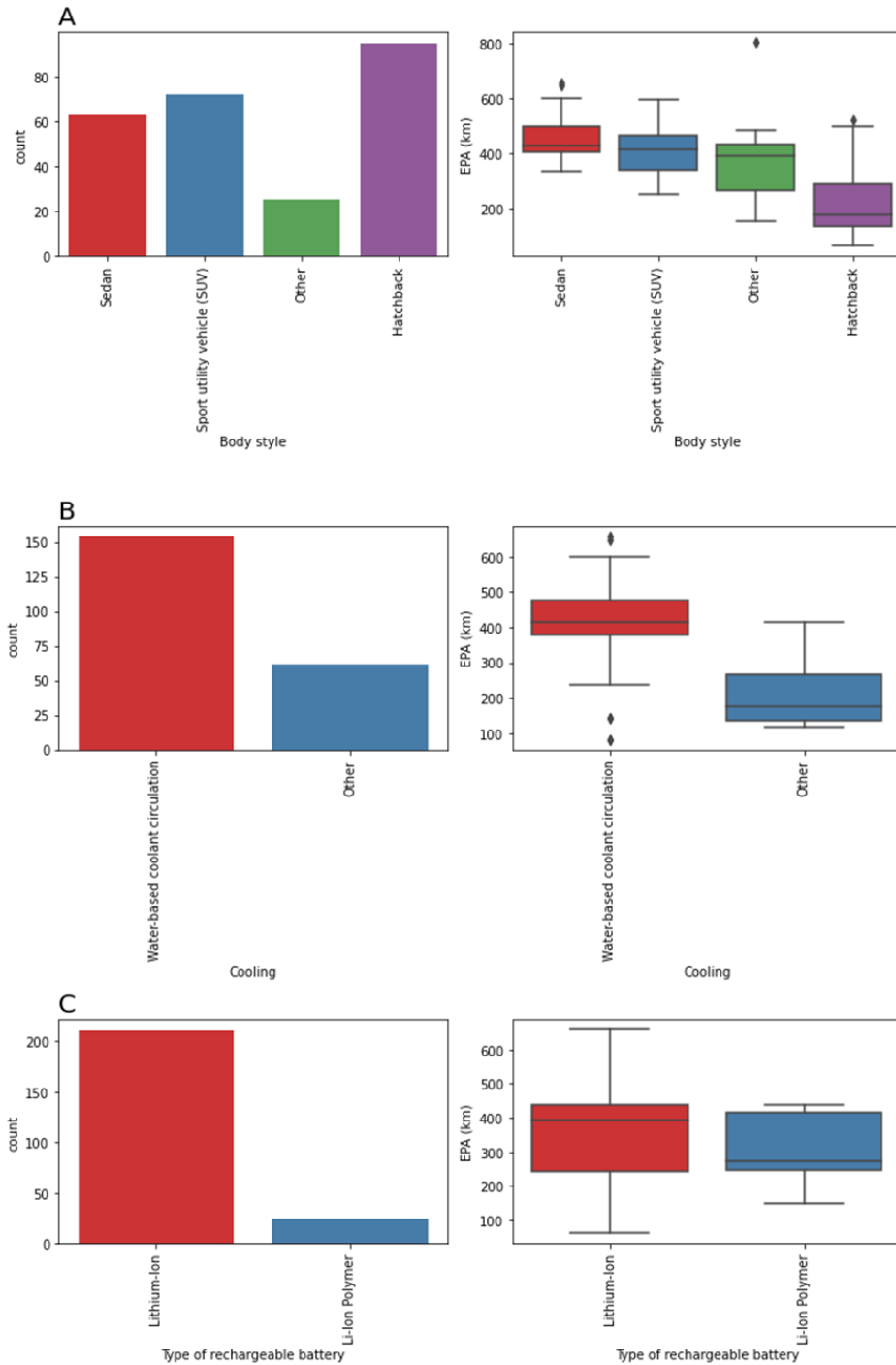


Figure 15: Histogram and boxplots of (A) EV body style, (B) battery cooling mechanism, and (C) type of lithium-ion battery used. The histogram plots the frequency (referred to as the count) of datapoints against the qualitative independent variables. Boxplot displays the average and quartiles of the EPA range against the qualitative independent variables.

Based on the findings from data exploration, the independent variables, such as battery capacity, model year, top speed, acceleration, curb weight, GVWR, body style, battery cooling mechanism, and type of lithium battery, were used for model training and evaluation. The models were trained on the training set, and the features were preprocessed as outlined in the methods section. Table 5 shows the model parameters used in this work.

Table 5: The ML model parameters.

Model	Parameters
Normal Linear Regression	-
Ridge Regression	$\alpha = 1$
Lasso Regression	$\alpha = 0.1$
Elastic Net	$\alpha = 0.1$

Table 7 shows the RMSE values obtained by model evaluation on the test set. It can be seen that SVM regression demonstrates the lowest RMSE of 31.428 km. The mean EPA range on the test set was around 364.5 km, and an error of 31.428 km equates to around 8.6 % accuracy. Interestingly, the RMSE increases to 32.13 km when regularization in the SVM model increases ($C = 100$). While RMSE was considered as the evaluation metrics in this work, other metrics (i.e., mean absolute error (MAE), mean absolute percent error (MAPE), and R-squared (R^2) error) were also calculated and their equations are given below.

$$MAE = \frac{1}{m} \sum_{i=1}^m |\hat{y}_i - y_i|$$

$$MAPE = \frac{1}{m} \sum_{i=1}^m \left| \frac{\hat{y}_i - y_i}{\hat{y}_i} \right|$$

Equation 64

$$R^2 = 1 - \frac{\sum_{i=1}^m \hat{y}_i - y_i}{\sum_{i=1}^m \bar{y} - y_i}$$

Furthermore, the results are tabulated below.

Table 6: Model evaluations from MAE, MAPE, and R²

Model	MAE	MAPE	R ²
Normal Linear Regression	23.379	0.067	0.932
Ridge Regression	15.542	0.072	0.930
Lasso Regression	23.570	0.067	0.932
Elastic Net	29.674	0.088	0.913
SVM Regression	22.367	0.062	0.939

Similar to the trend observed in the RMSE model evaluations, SVM regression outperforms other regression methods in MAE and MAPE evaluations (Table 6). The R² error is a measure of the portion of variability in the dependent variable predicted from the independent variable and its value close to 1 indicates a good fit between the model and data^{151,152}. The R² value was almost 0.9 (Table 6) suggesting that the models can predict the variability of the EV range from the independent variables.

Table 7: RMSE from model evaluations.

Model	RMSE
Normal Linear Regression	33.200
Ridge Regression	33.540
Lasso Regression	33.084
Elastic Net	37.423
SVM Regression	31.428

The equation for estimating the EPA range of the EV using linear regression (without standardization with RMSE of 33.20 km) is given below.

$$\begin{aligned}
EPA [miles] = & -37562 + 18.6 * Model Year + 3.8 \\
& * Battery capacity [kWh] + 0.08 \\
& * Top Speed [mph] + 10.2 \\
& * Acceleration (0 to 100 kmh^{-1})[s] + 0.22 \quad \text{Equation 65} \\
& * Curb Weight [lb] - 0.19 * GVWR [lb] + a \\
& * Battery Type + b * Cooling Mechanism + c \\
& * Body Style
\end{aligned}$$

Where a = 12.6 when lithium polymer is used and is -12.6 when lithium-ion batteries are used. Similarly, b = 28.9 or -28 when water-based cooling or other cooling mechanism are used, respectively. Moreover, c = -5.1, -60.2, 51.8 or 13.5 when Hatchback, other, sedan or SUV are considered, respectively.

It should be noted that while the proposed linear regression equation gives a higher error than estimating the EV range using a power consumption model in the literature (8.6 % vs. approximately 1 %), it requires significantly less parameters. This regression equation requires eight EV parameters that can be easily found in the literature or the EV manual while the energy consumption model requires more than 20 EV model-specific parameters^{98,99,132}.

The motor speed, demand from the battery pack, and the battery pack current from the vehicle dynamics model on various electric vehicles (EV) under a 1 cycle of highway fuel economy test cycle (HWFET) and urban dynamometer driving schedule (UDDS) are presented in the Figure below. It is assumed that the battery pack is able to deliver the power and current required from it. While the main focus of this research is on the EV with battery pack containing Tesla's lithium nickel cobalt aluminum oxide (NCA) battery cells, EVs with other battery chemistries are included in the figure below for the sake of comparison. The vehicle power demand for the battery pack for the drive cycles is shown in the plot below. The peak power demand for the Tesla Model 3 ranges from -60 kW to 60 kW for both of the considered drive cycles.

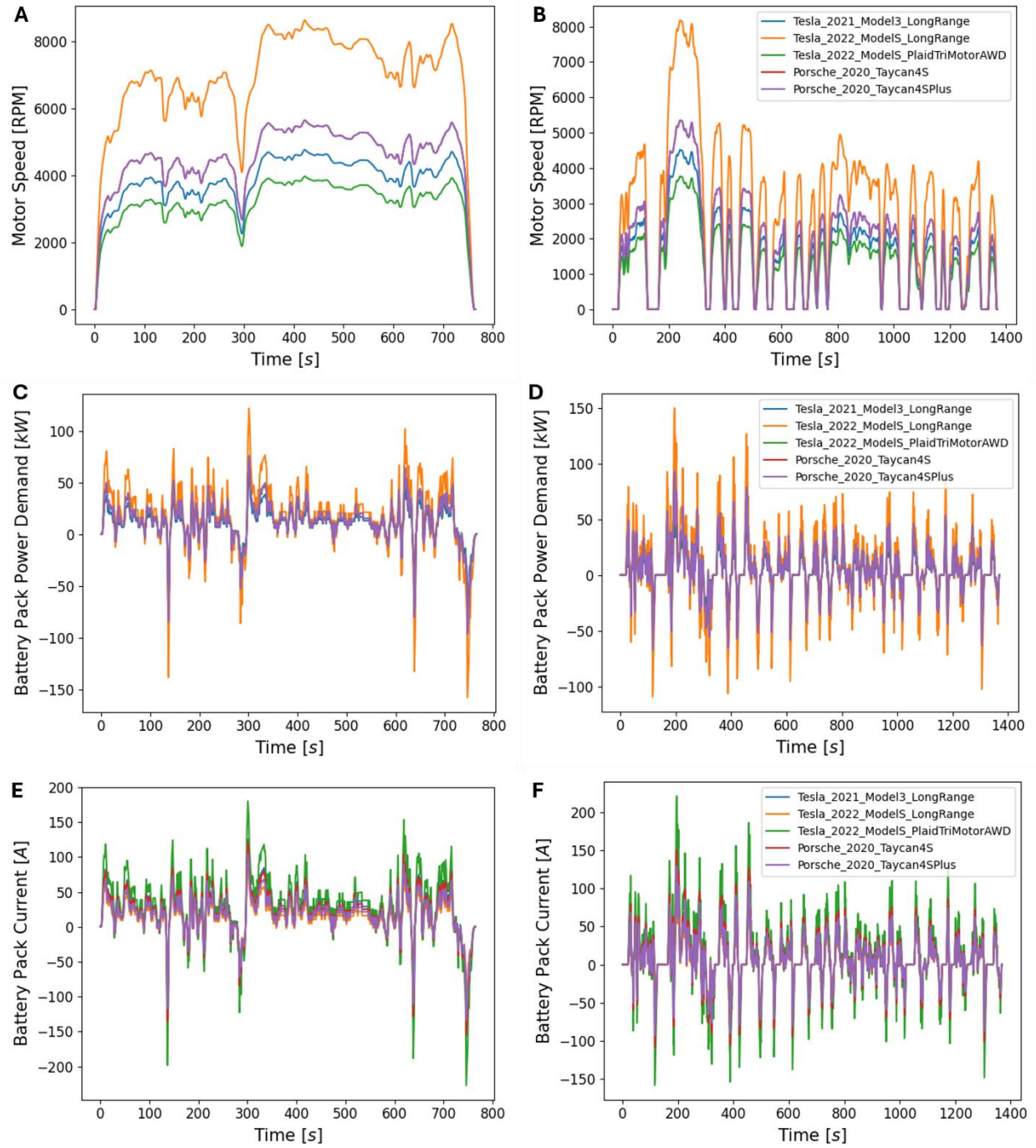


Figure 16: Resultant variables calculated from the vehicle dynamics model. The top (Figure (A) and (B)), middle (Figure (C) and (D)), and bottom row (Figure (E) and (F)) represent the calculated motor speed, required battery pack power, required battery pack current for HWFET and UDDS drive cycles.

Battery cell parameters were obtained from the relevant literature, while the remaining parameters were estimated using the genetic algorithm (GA). The functions for the electrode (for both positive and negative

electrode) open-circuit potential (OCP) and the change of OCP w.r.t to temperature (i.e., $\frac{dU_j}{dT}$) are expressed and plotted in sections S1 and S2, respectively. In order to calculate the change in OCP w.r.t temperature, the battery cell terminal potential at low discharge (i.e., 1/20 C) at various temperatures (i.e., 10°C, 25°C, and 40°C) was assumed to be the OCP of that electrode at that temperature (Figure S2A). Then, a polynomial was fitted through those data points. Centered finite difference was used to calculate the change of battery cell's change in OCP w.r.t to temperature at various SOC (Figure S2B). Finally, the positive's electrode changes in OCP w.r.t to temperature was calculated from the that of the battery cell and the negative electrode⁴³. The parameters used for conducting thermal single particle model are tabulated below. Parameters, including the particle radius size, lithium diffusivity in electrode, thermal activation energies for diffusivity and charge-transfer reactions, were estimated using the GA and the experimental constant current discharge curves at various discharge rates and temperatures. The results from the obtained optimized parameters are shown in Figure 17.

Table 8: Parameters for the battery cell model.

Symbol	Positive Electrode	Negative Electrode	Unit
L_j	128	250	μm
A_j	0.0547	0.0547	mm^2
$c_{s,j,max}$	63104	31833	mol m^{-3}
R_j	4.529*	3.0161*	μm
$k_{j,ref}$	$6.67 \cdot 10^{-11}$	$1.76 \cdot 10^{-11}$	$\text{m}^2 \text{mol}^{-0.5} \text{s}^{-1}$
$Ea_{r,j}$	51555*	18798*	$\text{m}^2 \text{s}^{-1}$
$D_{s,j,ref}$	$1.45 \cdot 10^{-14}$ *	$3.92 \cdot 10^{-12}$ *	$\text{m}^2 \text{s}^{-1}$
$Ea_{D,j}$	2537*	3693*	$\text{m}^2 \text{s}^{-1}$
c_e		1000	mol m^{-3}
ρ		1626	kg m^{-3}
Vol_{cell}		$2.425 \cdot 10^{-8}$	m^3
C_p		750	J K^{-1}
T_{ref}		273.15	K
h		1250	$\text{W m}^{-2} \text{K}^{-1}$
A_{cell}		$2.424 \cdot 10^{-05}$	m^2

* Parameters were estimated using GA

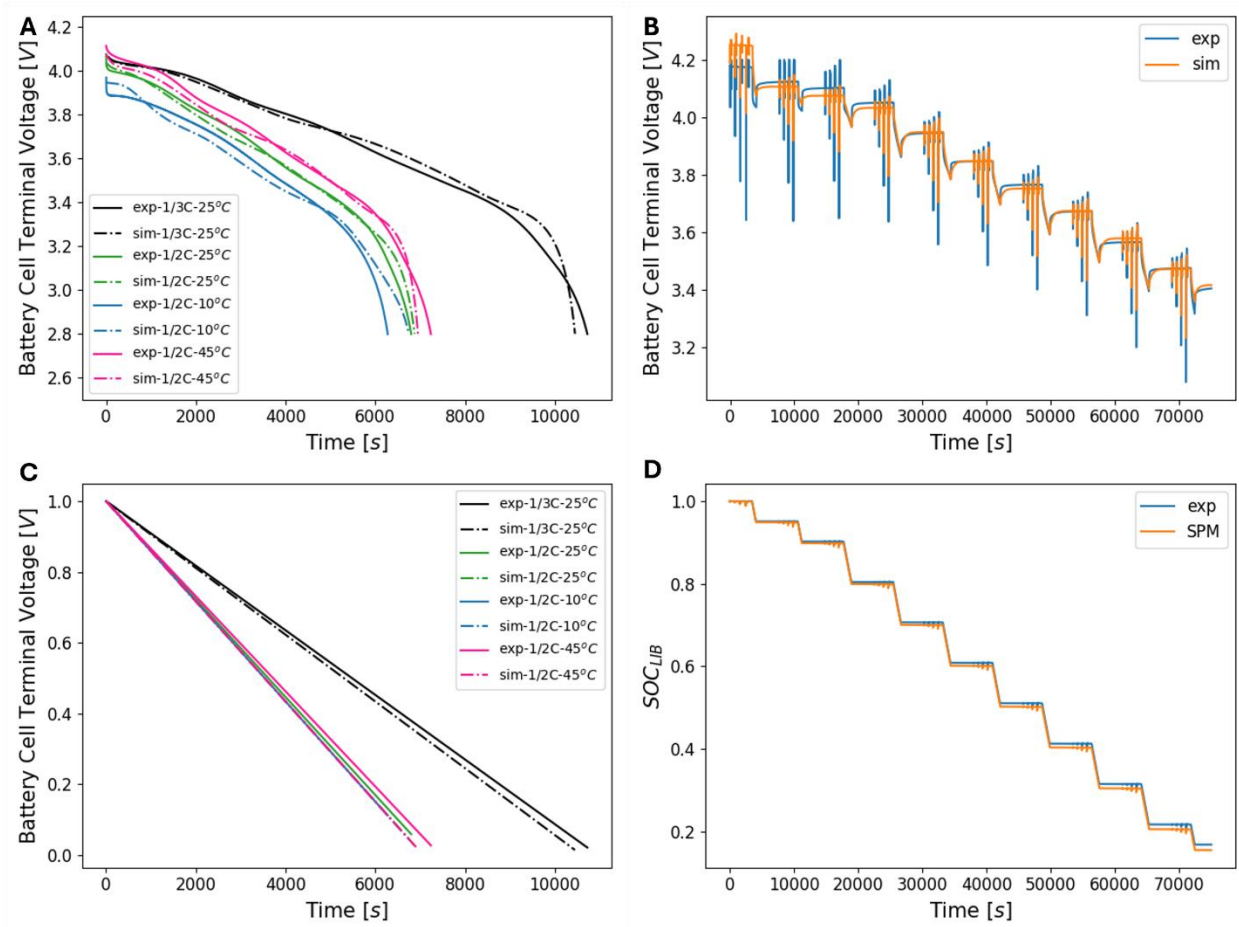


Figure 17: (A) The experimental and simulated discharge profile at two C-rates (1/3 C and 1/2 C) and various temperatures (10°C, 25°C, and 40°C). (B) Experimental and simulated HPPC discharge profile. The experimental and simulated battery cell SOC for the (C) constant current and (D) HPPC discharge profiles.

Finally, the range of the EVs were estimated using the application-level vehicle dynamics model and the continuum-scale thermal single particle. For this purpose, the vehicle dynamics model was run on cycles of UDDS and HWFET separately and the required battery pack current was obtained. Subsequently, the current through the battery cell was obtained based on the series and parallel connections of the battery cell within the EV's battery modules and the number of series connected battery modules. The thermal single particle model was run on the obtained battery cell current profile until the minimum terminal voltage was

reached during the simulation. It was assumed that the parameters of the battery cell within the battery pack exhibit similar parameters. The terminal voltage during a cycle HWFET and UDDS profile are shown in the figure below.

3.6 Conclusions

EVs have several potential merits, such as low-environment impact and low operating cost, but they suffer from a lower range than conventional vehicles. Estimation and interface of remaining range by the battery management system can help alleviate the consumer range anxiety. Among the range estimation strategies, prior literature has utilized vehicle dynamics models with and without conjunction with battery cell models. However, these utilized battery cell models use empirical models, such as equivalent circuit models. While these empirical models are simpler to implement and require low computational resources, they are less accurate in measuring the overpotentials during the battery cell charge/discharge operations. This work instead uses a physics-based model in conjunction with the thermal model to model the battery cell behavior. As such, this work consists of two separate projects (detailed separately below); (1) the determination of EV parameters that affect its range and (2) the application of the vehicle dynamics model with the continuum-scale thermal battery cell model. For the latter, the Tesla Model 3 was used as the test subject, whereby the battery model parameters were estimated from experimental cycling data.

First data-driven technique was used to explore the EV design specifications that influence its range. For this purpose, a detailed dataset for commercial EV models manufactured from 2008 to 2021 was generated by extracting information from the internet using web-scraping techniques. Through correlation, it was observed that relevant EV parameters, including battery capacity, battery voltage, top speed, and curb weight, seem to have the most significant effect on the EV range. Battery capacity, top speed, and curb weight showed a strong positive correlation to the range, while acceleration showed a strong negative correlation. Moreover, newer EV model years and higher battery capacity seem to impact its EPA range positively. Also, EVs with water-based cooling mechanisms have a higher EPA range than those with other cooling mechanisms. Furthermore, EVs with body-style sedans and SUVs have a higher EPA range when

all other designs, battery, and performance specifications are kept constant. Various self-learning linear regression and SVR machine-learning techniques were employed for model training and testing. Normal linear regression models seem to have the lowest RMSE of 31.428 km. Furthermore, a multi-linear regression model equation that related EV's EPA range from its battery, design, and performance specifications was proposed.

Afterward, EV application- and battery cell continuum-scale modeling approaches were combined to estimate the range of Tesla's Model 3 (model year 2021). This EV uses lithium-nickel-cobalt-aluminum oxide (NCA) chemistry for its positive electrode and the battery cell is in the cylindrical-2170 battery form factor. The drive cycles considered in this study were the highway fuel economy test cycle (HWFET) and urban dynamometer drive schedule (UDDS) since these are used for the reported range by the United States Environmental Protection Agency (EPA). The output power and current were calculated using the vehicle's drivetrain and battery pack specifications. Assuming that the battery cells have consistent capacity and internal resistance, the current required through a battery cell was calculated and then inputted into the thermal single-particle model. The thermal and material parameters required by the thermal single-particle model were obtained either from the literature or were successfully estimated using a genetic algorithm (GA). The simulation results from the final obtained parameters were tested against the ones obtained from experiments and a good agreement between the two was observed. Finally, the battery cell was subjected to the HWFET and UDDS drive cycles until its minimum cell terminal voltage was obtained, and its range was estimated using the EPA range criteria. The estimated range has a relative error of 0.081 compared with the reported EPA range. The proposed methodology in this work can be applied by an EV's BMS for estimating the range of a battery pack for a user-specific drive cycle and external condition.

3.7 Supporting Information

3.7.1.1 Section 1: Electrode Open-Circuit Potentials

The following functions were used to represent the relationship between the respective electrode's state-of-charge (SOC) and its open-circuit potential (OCP) at room temperature (Equation 66 and Equation 67).

$$\begin{aligned}
 U_p = & 8.88227776 * 10^{-3} * soc_p^{11} - 5.02049507 * 10^4 * soc_p^{10} \\
 & + 1.23757279 * 10^5 * soc_p^9 - 1.74782396 * 10^5 * soc_p^8 \\
 & + 1.56299049 * 10^5 * soc_p^7 - 9.23629992 * 10^4 * soc_p^6 \\
 & + 3.65421073 * 10^4 * soc_p^5 - 9.58957391 * 10^3 * soc_p^4 \\
 & + 1.61221861 * 10^3 * soc_p^3 - 1.60717828 * 10^3 * soc_p^2 \\
 & + 7.24113978 * soc_p + 4.05419291
 \end{aligned}
 \tag{Equation 66}$$

$$\begin{aligned}
 U_n = & 0.13966 + 0.68920 * \exp(-49.20361 * soc_n) + 0.41903 \\
 & * \exp(-254.40067 * soc) \\
 & - \exp(49.97886 * soc_n - 43.3788) - 0.028221 \\
 & * \arctan(22.52300 * soc_n - 3.65328) - 0.01308 \\
 & * \arctan(28.34801 * soc_n - 13.43960)
 \end{aligned}
 \tag{Equation 67}$$

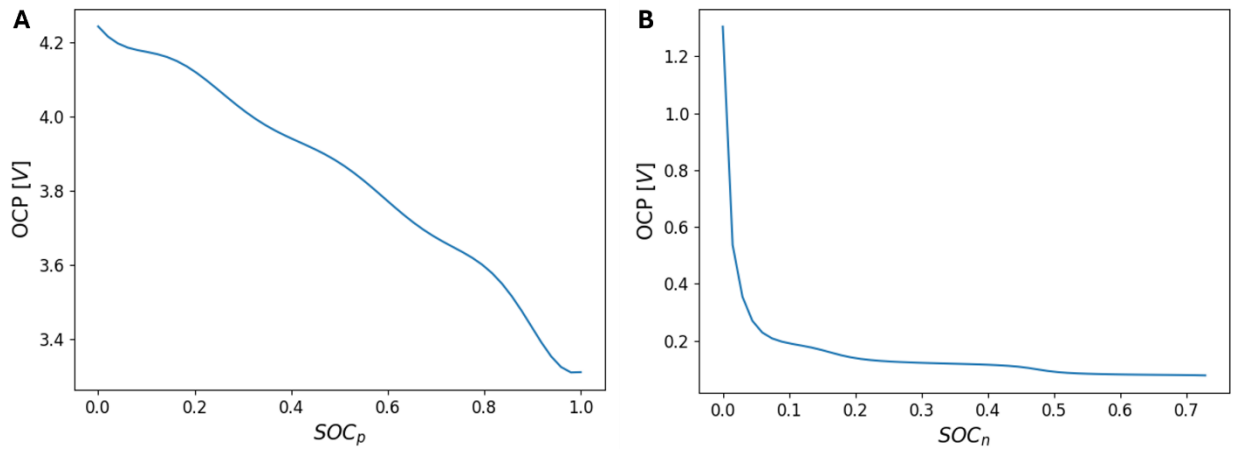


Figure 18: Open-circuit potential at room temperature for (A) positive and (B) negative electrodes.

3.7.1.2 Section S2: The change of open-circuit potential with respect to temperature

The following functions represent the change in electrode's open-circuit potential with respect to temperature.

$$\begin{aligned} \frac{dU_p}{dT} = & -36.2797735 * soc_p^{11} - 114.845612 * soc_p^{10} + -61.0391959 \\ & * soc_p^9 - 216.345537 * soc_p^8 + 439.70208 * soc_p^7 \\ & - 378.213227 * soc_p^6 + 178.64472 soc_p^5 - 47.5684336 \\ & * soc_p^4 + 6.63342923 * soc_p^3 - 0.37800454 * soc_p^2 \\ & - 0.00453340223 * soc_p + 0.00238263722 \end{aligned} \quad \text{Equation 68}$$

$$\begin{aligned} num = & 0.00527 + 3.29927 * soc_n - 91.79326 * soc_n^2 + 1004.9110 * soc_n^3 \\ & - 5812.27813 * soc_n^4 + 19329.7549 * soc_n^5 \\ & - 37147.89470 * soc_n^6 + 38379.18127 * soc_n^7 \\ & - 16515.05308 * soc_n^8 \end{aligned}$$

$$\begin{aligned} dem = & 1 - 48.09287 * soc_n + 1017.23480 * soc_n^2 - 10481.80419 * soc_n^3 \\ & + 59431.30001 * soc_n^4 - 195881.64880 * soc_n^5 \\ & + 374577.31520 * soc_n^6 - 385821.16070 * soc_n^7 \\ & + 165705.85970 * soc_n^8 \end{aligned} \quad \text{Equation 69}$$

$$\frac{dU_n}{dT} = \frac{dem}{num} * 1 * 10^{-3}$$

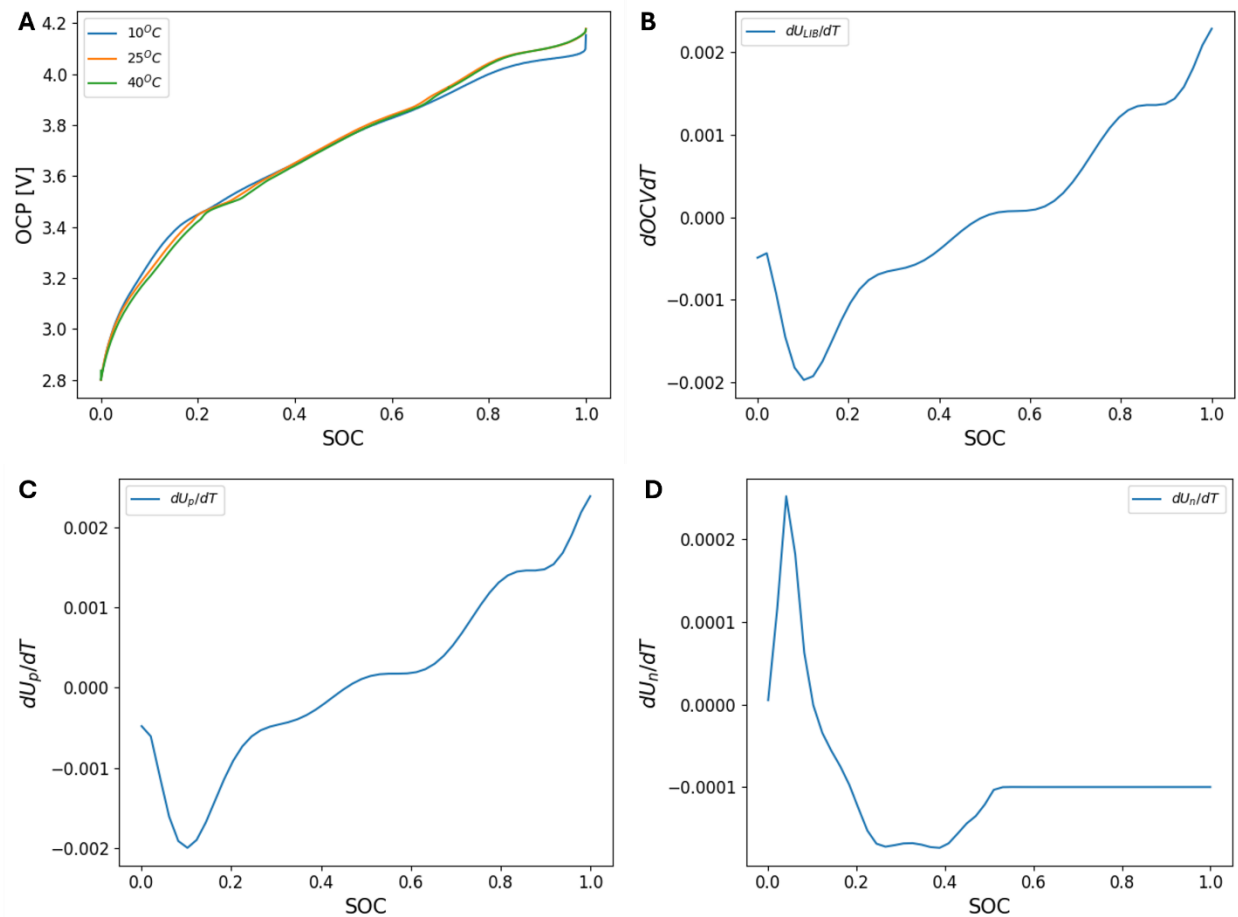


Figure 19: (A) The battery cell terminal potential at various temperatures at 0.05 C. (B) The change of the battery cell's open-circuit potential with respect to temperature. Open-circuit potential of the (C) positive and (D) negative electrode with respect to temperature.

3.7.1.3 Section S3: Absolute error between on the experimental and simulation results

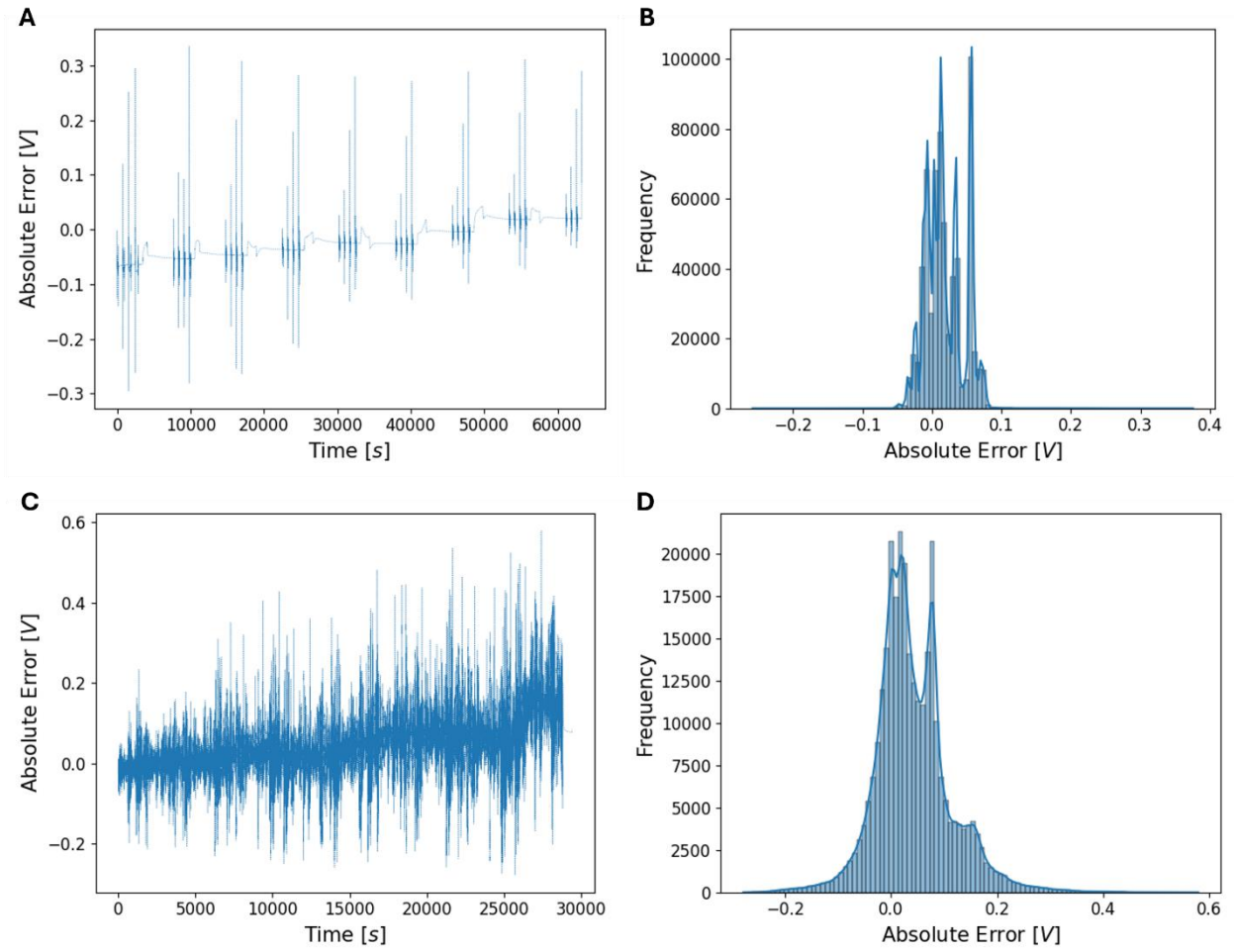


Figure 20: (A) The absolute error between the simulated and experimental battery cell terminal voltage and (B) its distribution for HPPC cycle. (C) The absolute error between the simulated and experimental battery cell terminal voltage and (D) its distribution for the custom discharge cycle.

4 Fault Detection Using Machine Learning Techniques

This section has been adapted from the manuscript research article “Moin Ahmed, Xingling Zhang, Zhiyu Mao, Tao Chen, Zhongwei Chen, “Fault Detection in Lithium-ion Battery Pack of Electric Vehicle using Auto-Encoder”. Journal of Electrochemical Society [Manuscript under Submission]”.

4.1 Chapter Objective

Fault detection of lithium-ion battery packs for electric vehicles was attempted in the work of this chapter. A dataset from the battery management system of 100 electric vehicles that were under operation for a year was used. Faults from the battery pack, while uncommon, are a major safety concern that can lead to thermal runaway in severe cases. A machine learning model, based on an autoencoder, was trained and tested on the dataset for the detection of abnormal sensor reading from the battery management system. The methodology was intended for the battery management system that interfaces with the cloud.

4.2 Chapter Introduction

A solution to mitigate the contribution of greenhouse gas emissions (GHG) from the global transport sector is through its electrification, for instance, in the form of electric vehicles (EV) ^{153,154}. Lithium-ion batteries (LIBs) are majorly employed in EVs for energy storage, and battery cells are connected in series and parallel combinations to meet the desired voltage, current, energy, and power specifications (Figure 21A illustrates an example EV with a battery pack) ^{155,156}. As an example, the battery pack in Nissan Leaf (the model year of 2011) houses two parallel combinations of 96 cells in series for a total of 192 cells ¹⁵⁷. Despite its high performance, LIB poses safety hazards, including thermal runaway, particularly when operated outside of safe operating conditions (SOA) ¹⁵⁸. In EVs, fire incidents from the sustained abuse of the battery pack have been reported; therefore, early battery fault detection becomes a crucial way of battery fault-induced accidents ¹⁵⁹.

The battery management system (BMS) manages the EV battery pack, primarily to ensure EV user safety and maintain its performance. The functions of the BMS include but are not limited to (a) measurement of

battery pack variables through sensors, (b) protection of the battery pack from abuse, (c) evaluation of important battery pack state variables (including battery pack state of charge (SOC)), and (d) interfacing/communication of important information (e.g., SOC and remaining EV range) with the external user. The BMS can measure the cell voltages, cell surface temperatures, and the pack current (Figure 21B) ¹⁶⁰. While the onboard BMS in an EV can perform basic calculations and can have limited storage capacity, it can be combined with cloud computing to improve its computational and storage capabilities ¹⁶¹. The BMS's measured data (e.g., cell voltages, temperatures, and pack current) can be transmitted and stored to cloud computing. In addition, advanced algorithms can be performed on this collected historic BMS dataset, which opens the opportunity for performing sophisticated fault detection algorithms.

Recently, deep learning (DL) algorithms have been employed in the literature for fault detection ^{46,120,121,162–164}. Compared to fault detection techniques based on physical models, DL techniques are easier to implement since they do not require expertise in underlying physical models and tedious model parameter identification processes ¹⁶³. Moreover, the research works can be classified depending on the analyzed battery measurement values, namely voltage ^{120,162} or cell surface temperature ¹⁶³. Yao *et al.* utilized general regression neural networks (GRNN), a neural network (NN) suitable for non-linear mappings, to identify different fault levels (no, slight, medium, or severe fault) from different voltage-based inputs. These model inputs include the voltage values, inter-cell covariances, intra-cell variances, and inter-cell voltage differences. The researchers reported a model error of only 0.425 % ¹⁶². However, the voltage data had to be filtered from sensor and vibration noise using waveform transforms and inverse, adding another computation step to fault identification. Another case in point is the work from Ojo *et al.*, where the researchers used long-short term memory (LSTM) neural network to calculate the cell's surface temperature from its voltage, current, SOC, and a data point from its historical surface temperature. The residual from the model and the actual cell surface temperature were compared to identify a faulty cell. The trained model was successfully identified the time of fault in a cell, where the fault was induced by heating the battery

using a heating pad¹⁶³. In both works, the data was obtained from a workbench consisting of a few cells in series and not from an actual EV battery pack. Moreover, the fault was artificially introduced to the cells, and, therefore, it can differ from the nature of the fault that occurs in real-life EV operations.

In contrast, Hong et al. and Li et al. used LSTM for voltage predictions in battery fault identification applications using real-life EV BMS datasets^{120,121}. Li *et al.* trained LSTM on a dataset consisting of real-life BMS data from nine EVs to accurately predict the battery pack voltage values under normal non-fault circumstances. The final voltage value is weighted appropriately using predictions from LSTM and equivalent circuit model (ECM) to improve model accuracy. When the predicted voltage value is compared with actual voltage for an EV with a known fault, the model was able to accurately identify faults based on a preset threshold value¹²⁰. In all of the works mentioned above, the measurable value (voltage or surface temperature) has to be predicted in real-time, which requires considerate BMS storage and computational resources. Therefore, the practical application of the above techniques might be severely hindered due to the computational capabilities of the current BMS in EVs. Furthermore, LSTM has a complex neural structure and can suffer from unstable gradients, both of which can adversely impact its practical application for fault detection on a large scale^{122,123}.

In this work, an autoencoder was trained and tested on a dataset of real-world hundred EVs. In contrast to voltage or temperature prediction, the autoencoder learns the general temperature variation trend amongst cells of a normally functioning battery pack. This autoencoder, which employed a relatively simple and easily adaptable neural network architecture, successfully differentiated a faulty EV from the normal functioning ones. This approach is well suited for large-scale BMS applications with additional cloud storage and computational resources, where periodic BMS data analysis can point to a faulty battery pack. Moreover, this approach is ideal for unbalanced data, where there is more likely to be data from normal functioning EVs than from faulty EVs. A cloud-based architecture is suggested that, theoretically, actively learns from on-coming new data and adjusts its model parameter and threshold accordingly. Accordingly, the contributions of this work are as follows:

- (1) An artificial-neural-network-based autoencoder was trained and tested for fault detection on a real-world dataset of 100 EV BMS dataset for fault detection. For each EV, the BMS dataset contained a time-series dataset of 96 terminal voltage values and 48 temperature probes.
- (2) Statistical variation coefficient was calculated on terminal voltage and temperature sensor readings, individually, to compare the suitability of either the terminal voltage or the temperature sensor for early fault detection. Furthermore, the trained autoencoder was also tested on the terminal voltage and the temperature sensor individually to confirm the suitability of the relevant sensor for fault detection.
- (3) The threshold mean squared error (MSE) value between the model prediction and the actual sensor readings was identified based on the percentile number that differentiates the MSE of the normally functioning EVs from the MSE of the faulty EV in the dataset.

The remainder of this article is structured as follows. Section 2 provides an overview of the methodology, including an overview of the BMS data preprocessing, variation coefficient calculations, autoencoder training and testing, and the determination of the MSE threshold. Section 3 discusses the results from the variation calculations and the outputs from the autoencoder on the terminal voltage and terminal voltage. Moreover, the results leading to the determination of the MSE threshold as well as a potential deployment strategy of the autoencoder on the cloud BMS server are presented in this section. Section 4 concludes the article with concluding remarks.

4.3 Main Contributions

This project uses a machine-learning-based approach for battery pack fault detection on real-world commercial datasets. The occurrence of severe battery pack faults in commercial EVs is not as common as the normally functioning one.¹⁶⁵ Therefore, existing commercial BMS datasets can disproportionately have more instances of normally functioning EVs than the faulty ones. Here, artificial-neural-based autoencoder is investigated for abnormal pattern in the BMS based on cell terminal voltage and temperature. In the

literature, autoencoders have been investigated for abnormal event detection in applications with unbalanced datasets such as cyberattacks, videos, and credit card fraud detection.^{166–168} Moreover, autoencoder was trained and tested separately on cell terminal voltage and temperature sensor readings in the battery pack. By testing both readings, this work was able to identify whether cell terminal voltage variation or thermal sensor reading in the battery pack is more viable for fault detection in the trained dataset.

4.4 Methodology

4.4.1 Data Acquisition and Preprocessing

A dataset containing battery (e.g., time series and statistics of cell voltages and temperatures) and EV (e.g., time series of vehicle speed and distance) information from the BMS of 100 EVs was obtained from an EV manufacturer. The relevant features include the time series of cell voltages and temperatures in this research work. The data were sampled at a regular time interval of 50 seconds to reduce the data file size. The dataset had the voltage values for 96 cells and temperature readings from 48 temperature probes, which were optimally placed in the battery pack. It is also worth mentioning that the EV batteries analyzed in this research work had lithium nickel manganese cobalt oxide (NMC) chemistry.

The dataset had ninety-nine normal functioning EVs with no apparent signs of a battery fault (referred to as normally functioning EVs in this work) and an EV with a faulty battery (referred to as “faulty EV” in this work) that had undergone thermal runaway (Figure 21C). Therefore, when comparing the faulty EVs from non-faulty ones, the dataset is unbalanced. Voltage or temperature values from fifty and twenty-five of the normally functioning EVs were used for model training and validation, respectively. After extracting cell voltages or temperatures (also referred to as the feature vector in this work) for each time step from the dataset, any time-point with missing values were removed.

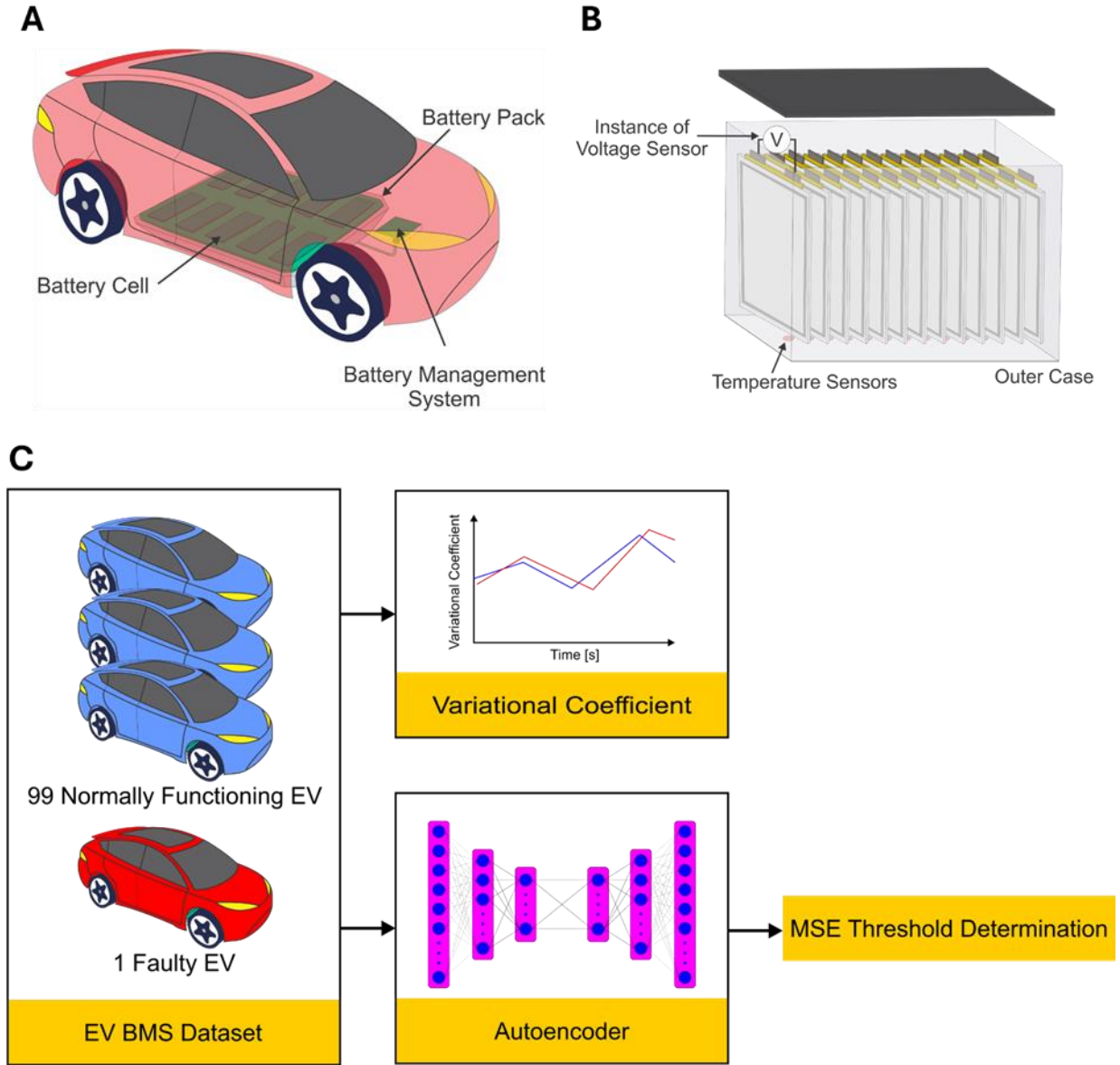


Figure 21: (a) An example illustration an EV with a battery pack, comprising of battery cells, and connected to a BMS, (b) an example of a battery pack with terminal voltage and temperature sensors, (c) block diagram portraying the dataset and the methodology in this work.

4.4.2 Variation Coefficient

As a preliminary step, a measure of the variation of voltage and temperature values across the battery cells in a faulty battery was calculated using the method suggested by Lu *et al.*¹⁶⁹. The variation coefficient, as used in this research work, is the measure of dispersion of voltage or temperature sensor readings at a given

time point. A higher variation coefficient is indicative of higher dispersion in the sensor readings and, consequently, a potential fault. This indication is due to the fact that, under normal circumstances, battery cells in EV battery packs should have consistent terminal voltage or surface temperature. The relevant equations (Equation 70-Equation 81) for voltage variation are given below:

$$V_{avg,t_n} = \frac{1}{n_{total}} \sum_{i_c=0}^{n_{total}} V_{i_c,t_n}, \quad i_c = 1, 2, \dots, n_{total} \quad \text{Equation 70}$$

$$V_{avg} = \frac{1}{N} \sum_{t_n=0}^N V_{avg,t_n}, \quad t = 1, 2, \dots, N \quad \text{Equation 71}$$

$$V_{max,t_n} = \max \{V_{1,t_n}, V_{2,t_n}, V_{3,t_n} \dots V_{n_{total},t_n}\} \quad \text{Equation 72}$$

$$V_{min,t_n} = \min \{V_{1,t_n}, V_{2,t_n}, V_{3,t_n} \dots V_{n_{total},t_n}\} \quad \text{Equation 73}$$

$$\sigma_V = \sqrt{\frac{1}{k} \sum_{t=0}^N (V_{max,t_n} - V_{min,t_n})^2} \quad \text{Equation 74}$$

$$\delta_V = \frac{\sigma_V}{V_{avg}} \quad \text{Equation 75}$$

Where n_{total} , N , and i_c represent the total number of measurement points at a certain time step (t_n), the total number of time steps, and a particular measurement point, respectively. Meanwhile, the variables V_{avg,t_n} and V_{avg} represent the average voltage value at a particular time step (t_n) and the average voltage of all measurement points at all time steps. After calculating the maximum and minimum voltages at a time step, given by equations 3 and 4, the root-mean square error (σ_V) between them is calculated (equation 5). Finally, the variation coefficient (δ_V) is calculated, where a high variation coefficient indicates a high variation between measurement points at a certain time step¹⁶⁹.

Similarly, the variation coefficient for temperature was also calculated, using equations 1-6, with the apparent replacement of temperature measurements instead of voltage ones¹⁶⁹. It should be noted that the variation coefficient, as defined above, considers the variation between the smallest and the largest measurement points at a time step.

4.4.3 Autoencoder

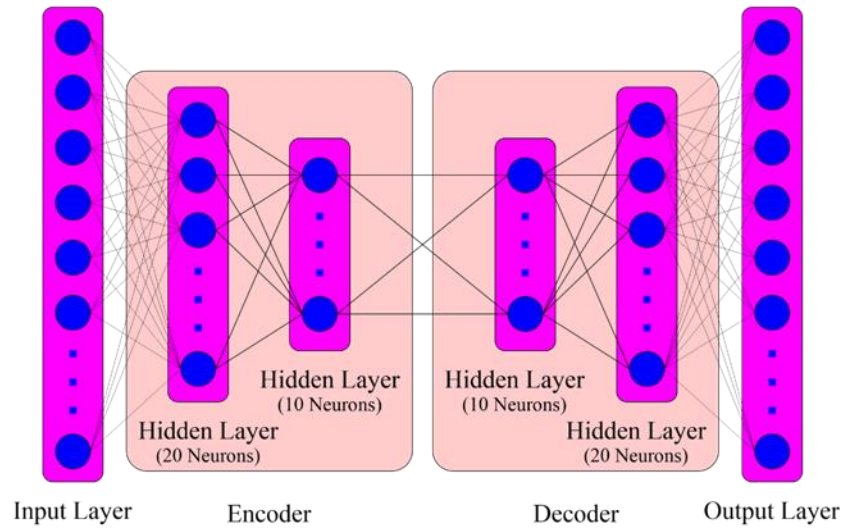
An autoencoder is an artificial neural network (ANN) that essentially outputs an approximation of the input. It comprises two ANN-based operators: (1) encoder and (2) decoder (Figure 22A). The input vector passes through the encoder, which outputs its low-dimensional tensor representation, usually a vector (known as the latent vector). The decoder then decodes this low-dimensional representation to give the final output. Autoencoders have found vast application areas, including noise filtering, feature detection, regenerative models, and anomaly detection^{170–172}. Regarding anomaly detection, it is assumed that the autoencoder learns the latent representation of the normal samples and will be able to identify samples not conforming to normal samples as an anomaly¹⁷². The assumptions for autoencoder for the application of autoencoder for fault detection in lithium-ion battery pack in this work are listed below.

- As mentioned in the preceding paragraph, it was assumed that the autoencoder learns the latent representation of the variation in the cell terminal voltage and temperature reading normally functioning battery pack. This assumption is validated later in the Results and Discussion section.
- The variation in the cell terminal voltage and temperature readings are good indicators of the battery pack fault. This assumption is based on the prior literature works.^{162,163} For instance, *Ojo et al.* trained a machine learning model, specifically a recurrent neural network, for battery fault detection based on the time-series variation of battery cell voltage, pack current, and battery cell state-of-charge.¹⁶³
- Battery pack faults (and therefore its subsequent fault detection) from external electrical and mechanical connections are ignored in this work. For instance, the vibrations and/or mechanical defects from a collision can cause electrically disconnected battery cells or modules.¹⁷³

The autoencoder structure used in this research is shown below (Figure 22A, Table 9). Two separate autoencoders were trained on either voltage or temperature measurements. The input layer would take a vector of 96 voltage measurements or 48 temperature readings in either case. The neurons in the hidden layers of the encoder had a dimension smaller than the input vector to constrain the network to facilitate

generalized input learning¹⁷⁰. Furthermore, as is the case in general applications of autoencoder, its encoder and decoder have a symmetric shape¹⁷¹.

A



B

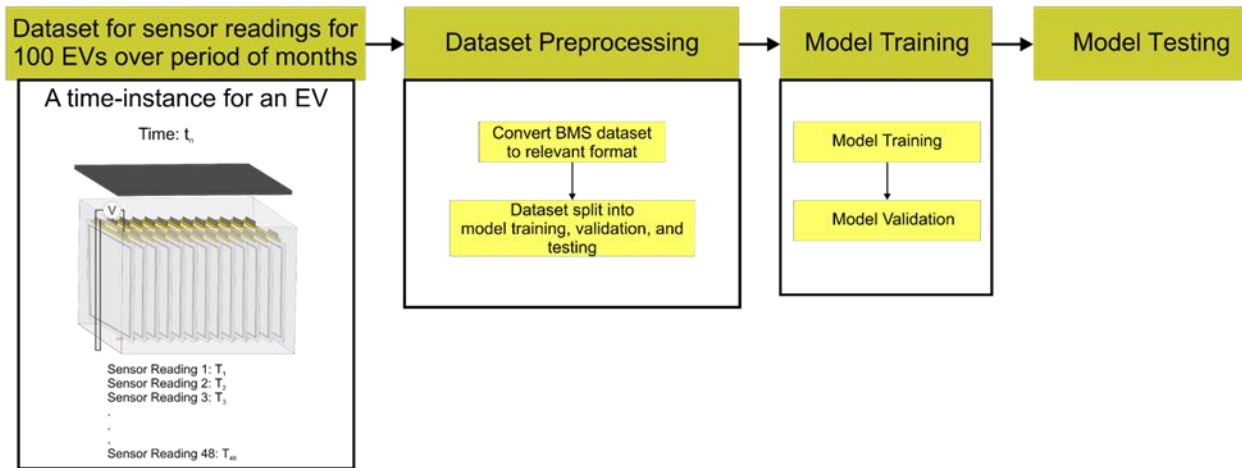


Figure 22: (a) The proposed autoencoder structure. (b) Autoencoder model training process.

As mentioned above, the encoder uses the input vector, $\mathbf{x}$, and transforms it into a low-dimensional latent representation, $\mathbf{z}$, where $\mathbf{z} = f(\mathbf{x})$. The non-linear function is the neural network in this case. The decoder

outputs the closest representation of the input, y . During model training, mean-squared error (MSE) was used as shown below:

$$\mathcal{L}(\mathbf{x}, \mathbf{y}) = \frac{1}{m} \sum_{i=1}^m (x_i - y_i)^2 \quad \text{Equation 76}$$

Moreover, after every training epoch, Adam optimizer was used to readjust the neural network weights and biases, and 100 epochs were used. Model training and validation were performed on time-series voltage or temperature dataset of 50 and 25 normally functioning EV, each chosen randomly from the choices of 99 normally functioning EV. The trained model was then tested on remaining normally functioning EV and the faulty EV.

Table 9: Autoencoder Specifications

Input Layer	
No. of Neurons	70 (for terminal voltage readings) or 48 (for temperature readings)
Encoder	
No. of Layers	2
No. of Neurons in the first layer	20
No. of Neurons in the second layer	10
Decoder	
No. of Layers	2
No. of Neurons in the first layer	10
No. of Neurons in the second layer	20
Output Layer	
No. of Neurons	70 (for terminal voltage readings) or 48 (for temperature readings)
Model Training	
Optimizer	Adam
Loss Function	Mean Squared Error
Epochs	100

4.4.4 Threshold Determination

The inputs (voltage or temperature measurements) were passed through the respective trained autoencoder, and the outputs of the autoencoder were compared with the actual measurements using MSE (Equation 76). It is assumed that the trained autoencoder will correctly identify the anomalies values. Therefore, in the case of anomalies data, the disagreement between the actual measurement and model output will be relatively high and, consequently, a relatively higher MSE.

In particular, the faulty EV will be expected to have a higher MSE. To observe and compare this high MSE with that from other normal functioning EVs, a distribution of threshold MSE is plotted in a histogram. Various threshold based on MSE percentile is calculated for this plot. A percentile that is effectively able to differentiate between the faulty and normally functioning EV is subsequently chosen.

4.5 Results and Discussion

The variation coefficient for both voltage and temperature values for one of a normal and faulty EVs was calculated to roughly understand the degree of variation between the voltage and temperature measurement points at time steps (Figure 23). It should be emphasized that this coefficient takes into account only the root-mean square error between the smallest and the largest measurement points¹⁶⁹. The faulty EV experienced a thermal runaway in a relatively short period of its life and, hence, has fewer data points than the normal EV. It is interesting to note that the faulty EV experienced relatively high voltage and temperature variation coefficients towards the end of its life. Furthermore, it can be observed that the temperature variation coefficients are generally higher in the number of occurrences and relative intensity (if the voltage variation coefficient towards the end of the EV life is discarded) than the voltage variation coefficients. This observation roughly, though not conclusively, indicates that the variation in temperature between cells in a battery pack can be a better metric for battery pack fault identification. This is later confirmed by comparing the results from the autoencoder on voltage and temperature data points.

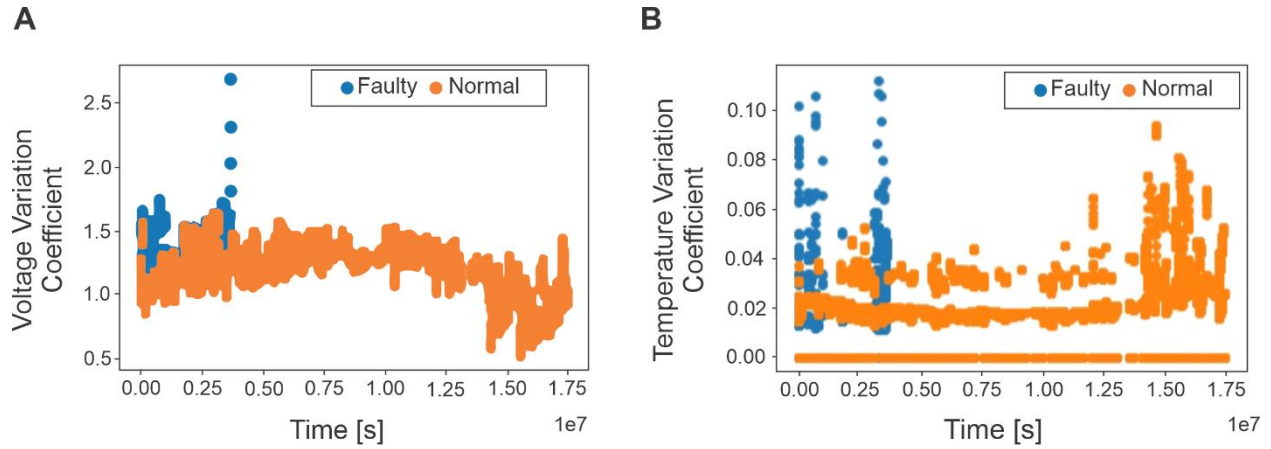


Figure 23: The variation coefficient of (a) voltage and (b) temperature of a normally functioning EV and a faulty EV, referred to as “normal” and “faulty” in the figure legends.

First, an autoencoder on the voltage values was trained using a large number of training epochs (100 epochs), with the early stopping criteria based on the convergence of validation error. This epoch was chosen to ensure that the model has enough training time while preventing overfitting. The input from normal and faulty EV was passed through the trained autoencoder for each time step, and the MSE between the actual voltages and the autoencoder output was calculated. Figure 24 shows the MSE of the faulty and representative normally functioning EV. There seems to be no distinguishing feature between the MSE of the faulty car and the normal functioning EV. This observation was consistent when observing MSE of other normal functioning EVs.

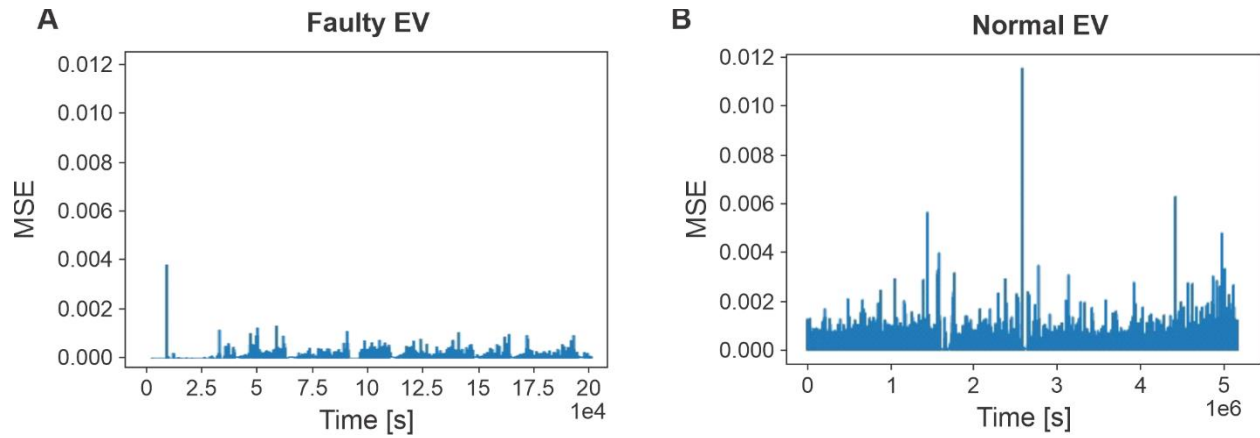


Figure 24: The MSE from the temperature autoencoder for (a) faulty and (b) a representative normal EV.

The MSE distribution of all EVs was plotted in a histogram to effectively visualize the MSE of the faulty car and all normally functioning EVs (Figure 25). Different MSE values for each individual EV was used for the distributions and it can be seen that none of the MSE threshold is suitable to distinguish faulty EV from a normally functioning one based on our autoencoder.

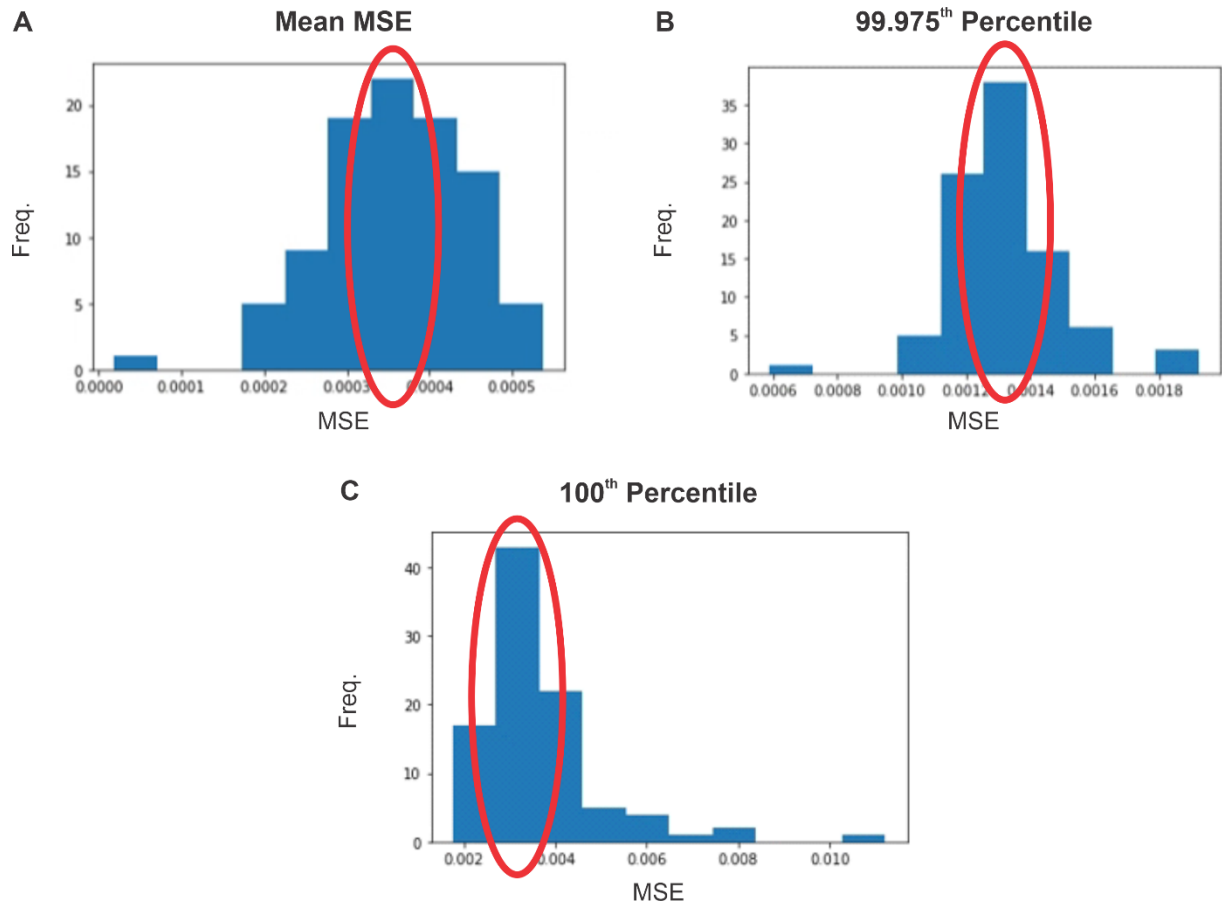


Figure 25: MSE distributions with different individual EV MSE values based on (a) mean, (b) 99.975th percentile, and (c) 100th percentile. The red outline indicates the histogram bar with the faulty EV.

Similarly, a separate autoencoder was trained using just the temperature measurements at each time step using training and validation data from normally functioning data. Figure 26(a) – (b) shows the commonly observed MSE pattern for normally functioning EVs. Most of the EVs displayed a relatively flat MSE with little to no peaks as in Figure 25A, with some major peaks as in Figure 25B. However, the MSE-time curve shows higher MSE intensities and frequencies (Figure 25C). Moreover, this difference is also evident when the distributions of MSE of all normally functioning EVs and the faulty EV is plotted as a histogram (Figure 26D and E). This observation suggests that the battery pack of the faulty EV has shown high-temperature abnormalities and, consequently, higher variations. Moreover, a similar trend of high and frequent temperature variation between the autoencoder's output and actual temperature measurements in the faulty EV when compared to all of the normally functioning EV is observed (Figure 26F). The variation in

temperature sensor reading between the battery cell in a battery pack were evident from the calculation of the variation coefficient (Figure 23). The variation coefficient is based on the statistical variation of the battery cell distributions in the battery cell. The trained autoencoder may be able to learn and generalize the distribution in terms of variation and other metrics of distribution, such as the mean and skewness. The exact cause of the autoencoder's ability to detect fault requires further investigation.

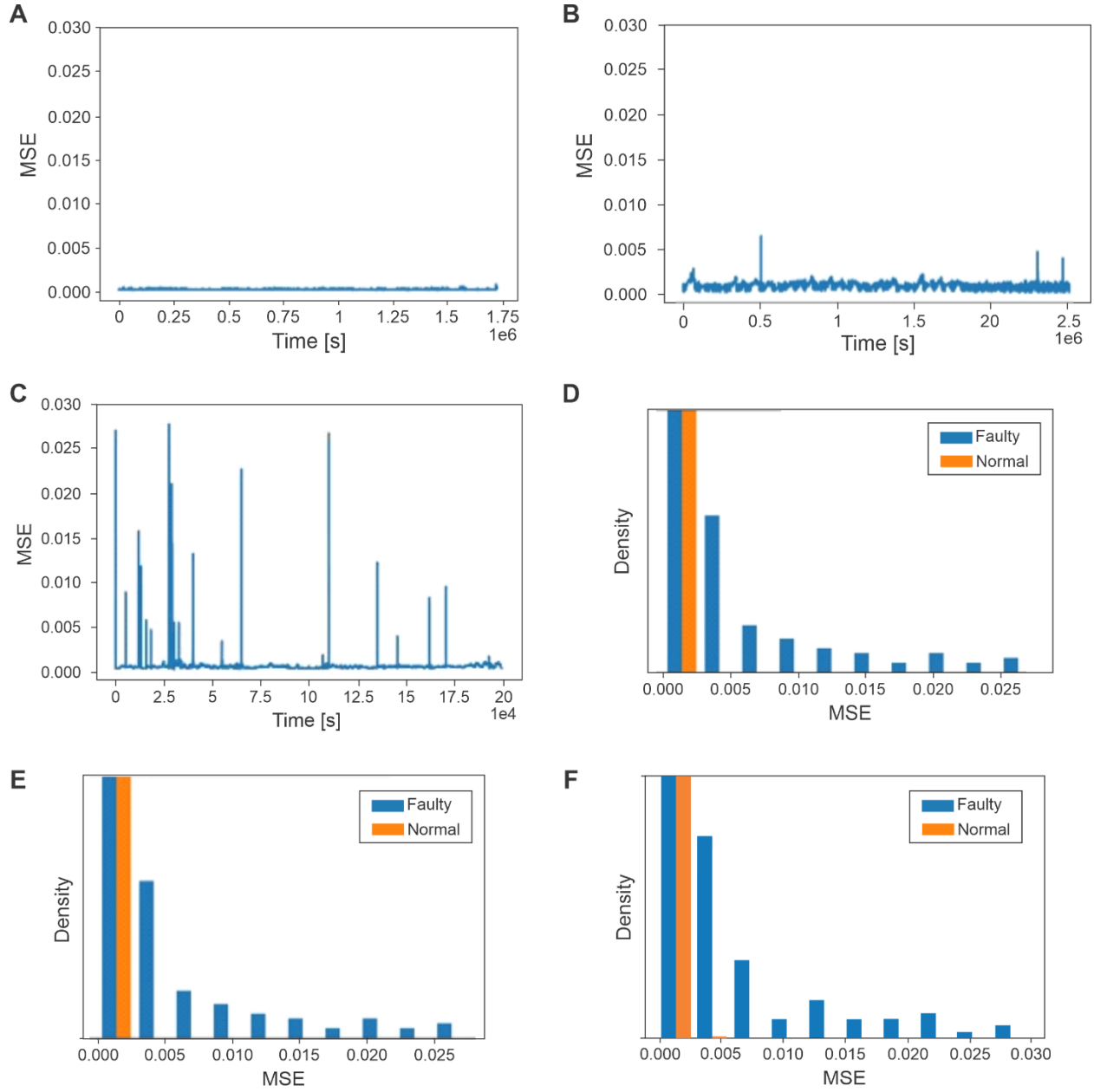


Figure 26: The MSE from the temperature autoencoder for (a) a normal functioning EV (EV1), (b) another normal functioning (EV2), and (c) the faulty EV. The distribution of MSEs for (d) EV1, (e) EV2, and (f) all normally functioning EVs compared to the faulty EV.

Finally, the temperature MSE distributions of the autoencoder's output and the actual measurements for all EVs was plotted (Figure 27). In this distribution, mean MSE (Figure 27A) and the 99.975th percentile (Figure 27B) were determined for all EVs and plotted as a histogram. It is clearly evident that the 99.975th MSE percentile can be used to differentiate the normal functioning EVs from the faulty ones. Hence, based

on our dataset, a MSE value of 0.004 can be a plausible threshold value for separating normal functioning and faulty EVs.

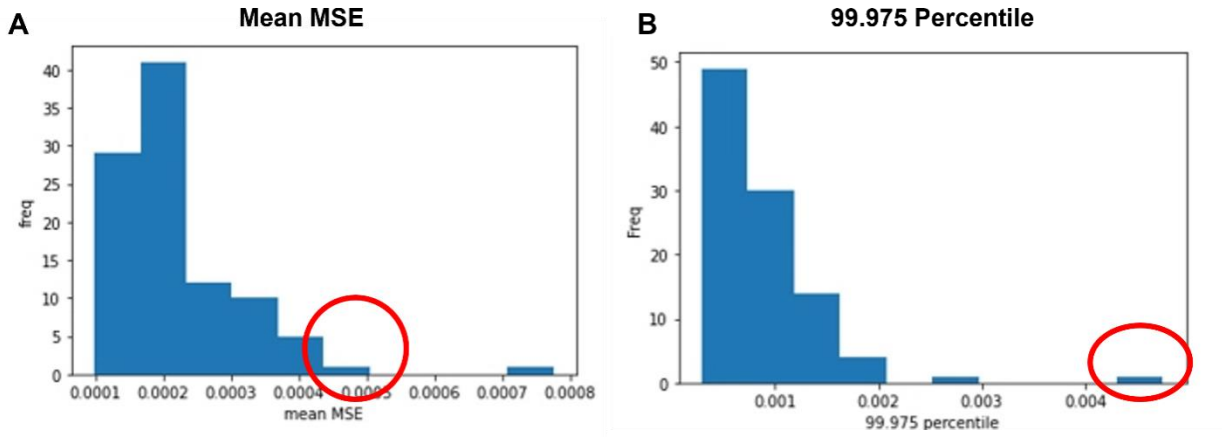


Figure 27: MSE distributions with different individual EV MSE values based on the (a) mean and (b) 99.975th percentile. The red outline indicates the histogram bar with the faulty EV. These plots help identify the threshold value.

Based on the findings of this work, a process flow is suggested for battery pack fault identification of cloud-based BMS in EV applications (Figure 28). The BMS data is pushed to the cloud storage for long-term data storage. At every interval (t_a) data is pushed to the computational server for data preprocessing and, subsequent, analysis using trained auto-encoder. Similarly, at another interval (t_b , where $t_b \gg t_a$), the autoencoder is retrained using the past data and a new threshold can be re-evaluated. While the proposed process can handle 100 EVs (model training and analysis) with ease, it needs to be determined if this process flow and the autoencoder model can sustain a large-scale commercial EV application, with millions of operating EVs.

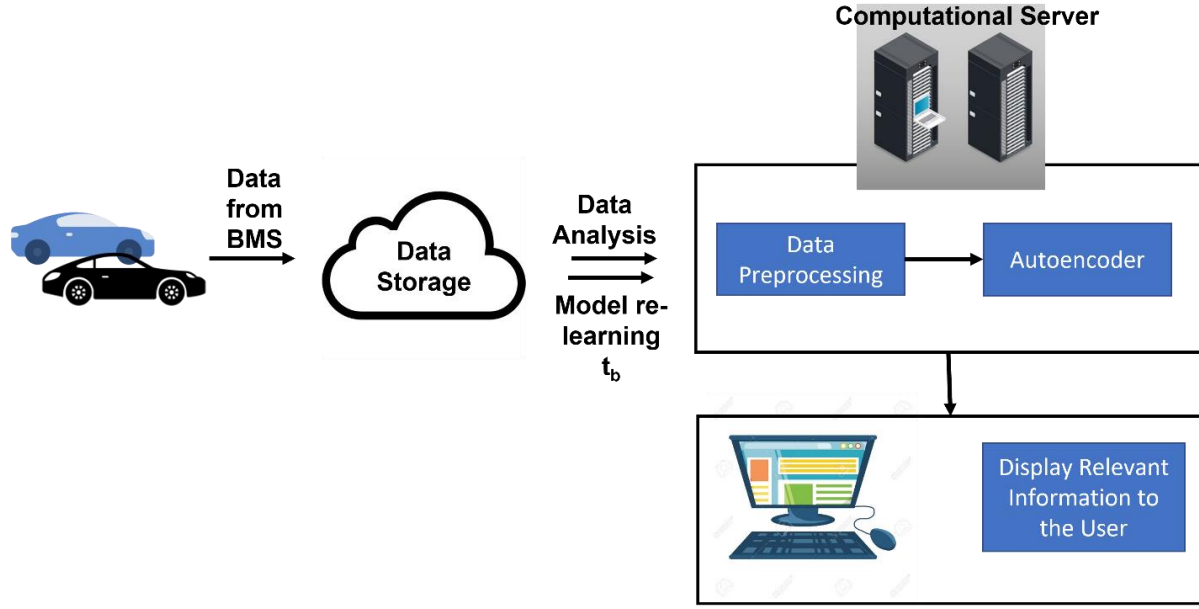


Figure 28: Recommended process flow for battery pack fault detection for cloud-based BMS in EV applications.

4.6 Conclusions

Artificial neural network (ANN)-based autoencoders have been utilized in this work for early fault detection in electric vehicles (EV) battery packs. The battery pack of hybrids and EVs are composed of series and parallel combinations of lithium-ion battery (LIB) cells. There is a user safety risk with battery packs since an occurrence of a fault and potential thermal runaway of a battery cell can propagate across the entire battery cell and can cause the battery pack to explode and/or catch fire. Prior literature works have leveraged the variation in BMS sensor readings for potential fault detection in their methodology including the use of machine learning algorithms. Since the predictions of the trained model depends heavily on the quality of the training dataset, prior literature has trained the models on balanced datasets. Balanced dataset, here, refers to equal datapoints for battery pack with and without faulty battery cells. However, in practice, especially for commercial dataset, datasets are unbalanced since the occurrence of faulty battery cell is uncommon.

In this work, a real-world dataset from the battery management systems (BMS) of a hundred EV battery packs was used for artificial neural network (ANN) based autoencoder model training, validation, and testing. In addition, the unbalanced nature of the dataset (more normally functioning EV compared with the faulty ones) makes the use of an autoencoder as an anomaly detector a suitable candidate. Among the features in the dataset, a time series of voltage and temperature values from each cell and temperature probe, respectively, was used. It was assumed that after model training on normal functioning EV, the autoencoder would be able to detect non-conforming inputs as an anomaly. After data preprocessing, two separate autoencoders were trained on voltages and temperature measurements, respectively. Variation coefficient calculations and the model predictions from the autoencoder demonstrated the use of a thermal sensor as a better indicator of battery pack fault. Finally, a suitable threshold was obtained by plotting the 99.975th percentile of MSE for each EV from a temperature-based autoencoder. This threshold distinguished between the normal functioning EVs and the faulty ones.

The proposed use of an autoencoder for fault detection in the battery pack is able to process the sensor values from each battery cell since it takes the inputs from all the battery cells. Hence this proposed methodology is practical for applications with large LIB battery packs. Furthermore, a trained model is able to quickly perform the MSE calculations and is easier to implement compared to a physics-based battery model. Theoretically, the model can be retrained periodically with the influx of new EV data, which can make the model predictions more robust. The inclusion of more data allows the model to learn battery cell temperature sensor reading patterns across the battery cells in a pack in a variety of user drive patterns and external conditions. The proposed model is particularly suitable for cloud-based BMS systems whereby the incoming new data can be incorporated for model training.

5 Comparative Analysis of Computational Times of Lithium-Ion Battery Management Solvers and Battery Models Under Different Programming Languages and Computing Architectures for States Estimation

This section has been adapted from the manuscript “Moin Ahmed, Zhiyu Mao, Zhongwei Chen. Comparative Analysis of Computational Times of Lithium-Ion Battery Management Solvers and Battery Models Under Different Programming Languages and Computing Architectures.” Batteries. [Manuscript Accepted].

5.1 Chapter Objective

The work presented in this chapter attempted to verify the feasibility of using physics-based battery cell models for the state estimations of battery cells in embedded and cloud systems. Currently, the state estimations are predominantly performed by phenomenological models, such as the equivalent circuit models due to their relative ease of implementation and lower computational requirements. Future embedded battery management systems will require more accurate and robust physics-based models. Various reduced-order solvers and battery models were tested under different programming languages and hardware under different cycling conditions. All the models were coded inhouse using C++ and Python programming languages that are intended to be easily implemented and installed even in microprocessors of the embedded systems.

5.2 Chapter Introduction

At present, battery cells comprising lithium-ion batteries (LIBs) are primarily used in the battery packs of consumer electronics, electrified vehicles, and renewable energy generation plants^{12,174,175}. LIBs chemistries, containing a lithium transition metal oxide positive electrode and graphite negative electrode, offer excellent cycling life, high specific energy density, low memory effects, and low self-discharge rate^{8,9,176}. Inside the LIB-based battery pack, the cells are electrically connected in series and parallel to give the desired pack voltage and capacity (Figure 29A). The battery pack is connected to a battery management

system (BMS), which serves various functions (Figure 29A). These functions include but are not limited to (1) sensing for voltage, current, and temperature, (2) protection against extreme conditions such as excessive currents, under and high voltage limits, etc., (3) interface with the user on helpful information such as charge control and range estimation and (4) battery state estimation for performance management and diagnostics^{16,177}. Meanwhile, the battery cell within the battery pack can come in various form factors (Figure 29B). Various battery models, including cell-scale equivalent-circuit model (ECM) and continuum-scale physics-based model (PBM), are suitable for the battery cell state and health estimations for the BMS applications. From the mentioned models, ECM models are still standard in industrial BMS applications, majorly due to their ease of implementation and mathematical simplicity^{7,178}.

Both ECM and continuum-scale models PBM estimate the extent of various battery cell polarizations during its operations. ECM are considered phenomenological models since they use electric circuits to model the observable battery cell polarizations, such as polarizations that stem from the battery cell internal resistance, diffusional voltages, hysteresis effects, etc., and the model parameters are chosen to fit the experimental data²³. PBMs, in particular pseudo-two-dimensional (P2D), on the other hand, use the concepts of mass transport (in the solid electrode region and the liquid electrolyte region), charge conservation (in the solid electrode region and the liquid electrolyte region), and charge-transfer reactions (electrode-electrolyte interface) to model various battery cell polarization^{28,33}. Hence, the PBM model parameters are based on the battery material's electrochemical, physical, and mechanical properties. PBMs are considered more accurate in predicting the battery states than the ECM, and battery cell degradation models are more accessible to integrate with the PBMs than the ECMs^{19,21,178}. Moreover, the solution of the relevant PBM equations leads to the spatial and temporal distribution of the lithium-ion concentrations and potentials across the battery cell as the lithium-ion flux at the electrode-electrolyte interface that cannot be obtained from traditional ECM models. This added information allows for the estimation of thermal evolution from various sources across the battery cell (e.g., thermal evolution from Joule heating due to lithium diffusivity in the electrolyte and electrode regions, etc.,) a thermal model across the battery cell. An

accurate thermal evolution estimation is an important safety and performance feature for the thermal management of the battery packs.^{119,179,180} However, PBM are mathematically more involved since they generally include a set of coupled partial differential equations (PDEs) (Figure 29C)^{28,33,43,178,181}. These sets of coupled PDEs contain a more significant number of model parameters and require elaborate numerical solution schemes (compared to ECMs). An extensive set of equations are formed due to numerical solution schemes that must be solved at each time step. Consequently, compared to ECMs, PBMs usually have a higher computational demand and require a longer solution time. Moreover, implementing the Kalman filter for practical BMS applications is more cumbersome than that needed for ECM due to PBM's increased mathematical complexity.

In the literature, two main strategies have been employed to simplify the P2D's implementation and/or reduce its solution time for BMS applications, namely (1) simplification of model equations^{43,45} and (2) improvement of the solution schemes^{178,181}. Guo *et al.* introduced the single particle model (SPM) to calculate the lithium-ion battery cell terminal voltage. The researchers assumed a constant lithium-ion flux throughout the positive and negative electrode regions during the battery cell operations, negligible battery cell polarization contributions from the electrolyte dynamics, and continuous electrolyte solution resistance. Also, the researchers assumed that the lithium concentration profile in each of the electrodes could be approximated from that of a single spherical particle for that electrode⁴³. SPM is in good agreement with the experimental data for low charge/discharge rates (approximately $< 0.5C$) when the polarizations from the electrolyte dynamics are very low¹⁸². Later, Moura *et al.* extended the SPM with electrolyte dynamics (ESPM) for higher charge/discharge rates without resorting to the P2D model⁴⁵. The ESPM demonstrated a stronger rapport with the experimental data for up to $2C$ ⁴⁵. On another front, a set of resultant equations from the numerical solution scheme of a coupled set of PDEs can be solved iteratively until convergence or simultaneously^{178,181}. Han *et al.* proposed a finite volume method (FVM) based numerically efficient algorithm for solving the P2D equation, whereby the authors iteratively solved a few equations (specifically the equations about the electrode and electrolyte potentials) and checked for their convergence from the

results of another electrochemical model. The remaining equations were solved once the first converged, ensuring a faster solution time ¹⁷⁸.

Specific to the BMS applications, ECM and PBM model equations are utilized for the battery cell state-of-charge (SOC) and cell terminal voltage. It is worthwhile to note that typically, in PBM schemes mentioned above, a 1D cartesian coordinate system is used for the spatial distribution of the lithium-ion concentrations in the liquid electrolyte region and the potential profile across the liquid electrolyte and solid electrode regions across the thickness of a battery cell. Meanwhile, the spatial distribution of the lithium-ion across a solid electrode region uses a 1D spherical coordinate system. Hence, for the large format battery cells (e.g. the battery cell form factors include cylindrical, pouch, and prismatic battery cell formats), the PBM modeling approach alone does not include the cell-level effects of the cell-level parameters (such as the electrode dimensions, dimensions of the electrical contact tabs protruding from the electrode, 3D placements of the cooling system to the electrodes, etc.) on the macroscopic processes (such as the electric conductivity and thermal evolution).¹⁸³ Multi scale multi domain (MSMD) modeling framework, as introduced by the National Renewable Energy labs, tries to address this limitation in a computationally efficient manner. In MSMD, different length scales from cell-level to macroscopic levels are separated into separate domains with separate relevant models and coordinate systems. For instance, there can be domains for an electrode particle, the entire electrode, and the battery cell. Relevant variables are solved in their respective domain and these solved variables (after relevant area and volume averaging) are passed onto the next domain. As an example, the lithium-ion flux during the charge transfer at the electrode particle-electrolyte interface is calculated at the particle domain and then transferred to the electrode domain after averaging over the electrode-electrolyte interfacial area. This areal flux is transferred to the cell domain after volume averaging over the specific electrode plate area. Furthermore, MSMD allows flexibility in the selection of the model and solver schemes for each of the domains.^{184,185} For instance, the versatility of the model selection in MSMD was leveraged by Margi et al. where computational fluid dynamics was used for the determination of the spatial thermal distribution across a prismatic battery cell.¹⁸⁵ Due to its ability to

combine different length scales, MSDS have been studied in the literature to understand the effects of cell-level parameters on the inhomogeneous thermal evolution and electrode state-of-charge across the electrode.^{179,183,184,186} *Kim et al.* utilized MSDS to study the effects of the different stacked cell designs on the current density, SOC, and thermal evolution across the electrodes. Interestingly insignificant variations in the cell terminal voltage were observed though there were inhomogeneities in the cell kinetics, charging, and thermal evolution across the electrodes.¹⁸⁴ Moreover, by leveraging the thermal models at different scales, researchers have evaluated different battery thermal management systems using MSDS.^{179,183,186} In their approach, Schmidt et al. extended the P2D model by incorporating the model that includes the particle size distribution across the electrode.¹⁸³ Kim et al. added a 3D model in the cell domain and demonstrated a 3D profile of the current density, and thermal evolution across the electrodes.¹⁸⁴ While the MSDS framework adds model accuracy to the cell terminal voltage and SOC, it requires higher computation resources than the PBM, particularly from the inclusion of the model equations at the cell domain. The solution of the model equations requires creation of two- or three-dimensional mesh and solution at each spatial node for every time step.^{183,187} Various reduced order methodologies have been implemented with MSDS which have resulted in reduced solution times.¹⁸⁴

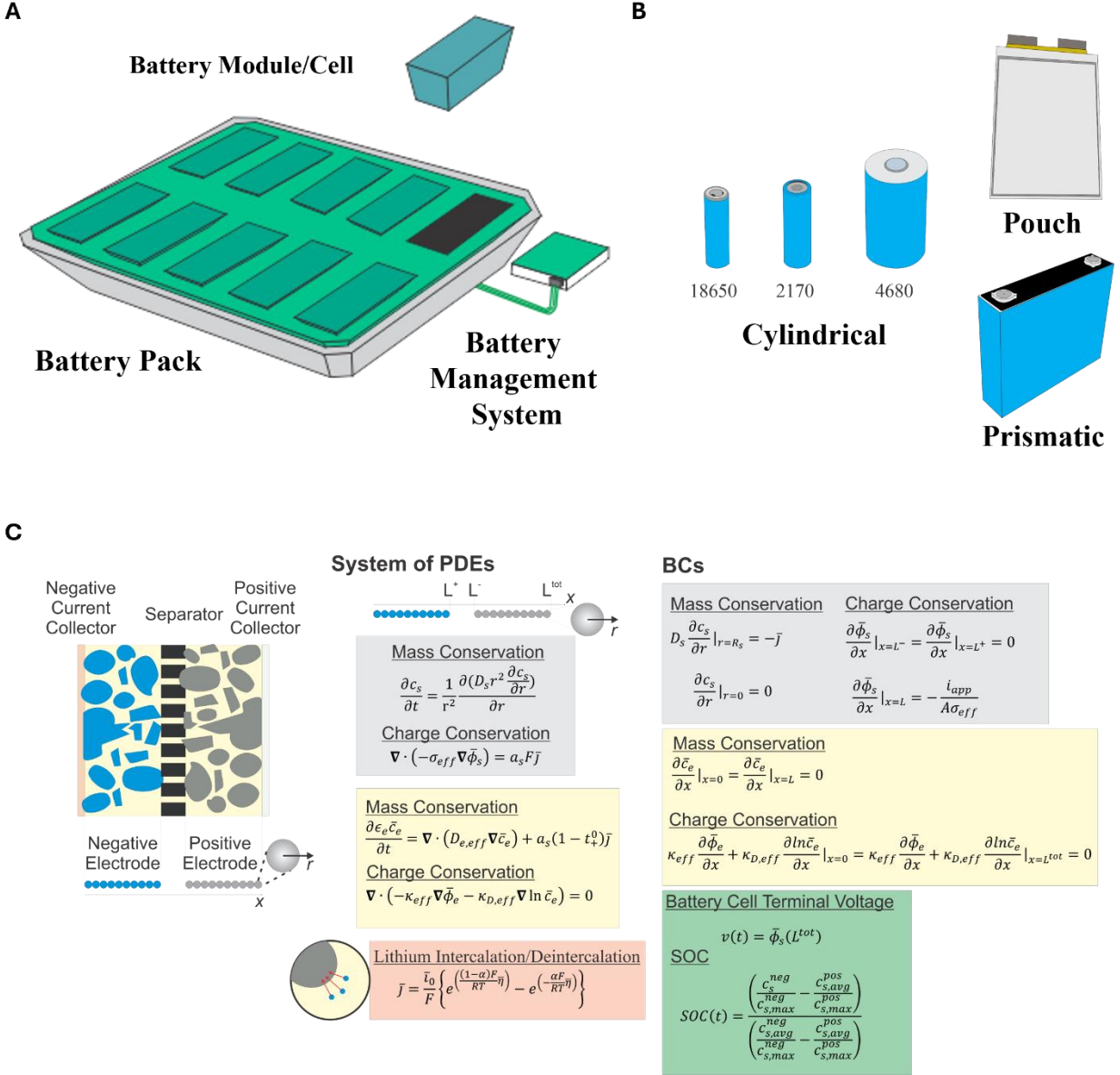


Figure 29: (A) Battery pack, comprising various battery cells, connected to a battery management system (B) Various form factors of commercial lithium-ion batteries used in battery packs. (C) coordinate systems, a system of partial differential equations (PDEs), and their respective boundary conditions.

This work compared various solvers and battery models based on their solution times and computational requirements on different architectures and programming languages. This research novelty is outlined below.

This work compared various solvers and battery models based on their solution times and computational requirements on different architectures and programming languages. This research novelty is outlined below. Practical industrial application of PBMs in the BMS (including PBM application for the battery cell/pack SOC estimations in the BMS) requires the quantification of the PBM computational load in the current BMS hardware. The performance and the computing speed of various numerical solvers for solving the PDE for the lithium-ion concentration in the solid electrode and liquid electrolyte region are analyzed. These solvers have been programmed in the C++ and Python programming languages and tested on various computing hardware whose computing specifications represent personal computing, cloud, and embedded systems. For instance, the embedded system, used in this work, uses a 1.2 GHz Broadcom BCM2837 processor with 1 GB RAM. C++ and Python are the programming languages that are used for programming microcontrollers and microprocessors (microcontrollers and microprocessors are used for the embedded systems including BMS).^{188–190} Furthermore, the performance, in terms of the solution times, of various reduced-order PBM for the battery cells was compared using different programming languages and computing hardware. These battery models include the SPM and the ESPM models. As mentioned above, the single particle model is suitable for battery cell applications where the charge/discharge currents are ideally below 0.5 C. At the same time, the ESPM can be used for applications that require currents up to 2C. The electrode solvers and the battery models are simulated under various cycling steps, including charge, discharge, discharge-rest, charge-rest, and hybrid pulse power characterization (HPPC). The physical and electrochemical model parameters used in this work have been borrowed from other literature works.^{43,181} The physical and electrochemical model parameters are representative of an instance of a prismatic lithium-ion battery cell that is comprised of lithium cobalt oxide (LCO) chemistry. Note, that the C++ and Python codebase was programmed in house with similar class structure with minimal use of external packages and libraries. The codebase along with the reduced order model allowed for the simulations even in embedded systems with very limited memory and computational resources.

In terms of the structure of this chapter, the subsequent methodology section overviews the various LIB battery solver and model equations, the architecture of the applied codebase, and the specifications of the utilized hardware. The following results and discussion section presents the results. Finally, this work's summary and its significant findings are summarized in the conclusion section.

5.3 **Main Contributions**

The contributions and novelty of this chapter, as alluded to in the 5.2 Chapter Introduction, have been summarized below.

- Various reduced order solvers for the PDEs about the lithium-ion concentration in the solid electrode and liquid electrolyte region have been compared in this work. These solvers were tested in different programming languages (C++ and Python) and hardware (embedded, cloud, and personal computing). These solvers are used in various physics-based battery cell models. These solvers were tested in static and dynamic battery cycling.
- Various simplified PBMs, namely SPM, ESPM, and reduced-order P2D, were tested in different programming languages and hardware. The thermal model was used in conjunction with the PBM battery cell models.
- The feasibility of probabilistic inference algorithms (specifically non-linear Kalman filter) with PBM was investigated in this work.

It should be noted that Python and C++ were chosen due to their popularity, particularly in embedded systems.^{188,190} The codebase was programmed in-house and care was taken to ensure minimal use of external libraries and packages. Also, the Python and C++ codebase was programmed using a similar class structure as detailed in Section 5.4.6: Programming Language. Broadly speaking, Python and C++ are regarded as interpreted and compiled languages, respectively. Moreover, as opposed to Python, the datatypes in C++ are strictly typed which makes C++ have less computational and memory overhead. Due to their mentioned differences, it is expected that the C++ codebase will have a faster solution time. Many microcontrollers used in the BMS embedded systems, such as the STM32 lineup from STMicroelectronics,

can only be coded in C/C++ programming languages.^{191–193} However, Python's ease of use and availability of external libraries for numerical computations, graphical user interface, and cloud interface makes it suitable for cloud-based BMS systems.^{194,195} In order to implement SOC and SOH estimations using PBM for the future BMS systems, it is important to evaluate the performance of both programming languages in the existing embedded and cloud hardware.

5.4 Methodology

5.4.1 Electrode Concentration Solver

The diffusion of lithium in the solid electrode region is assumed to be driven by the concentration gradient and is given by the following partial differential equation (PDE) and boundary conditions.

$$\frac{\partial c_{s,j}}{\partial t} = \frac{1}{r^2} \frac{\partial \left(D_{s,j} r^2 \left(\frac{\partial c_{s,j}}{\partial r} \right) \right)}{\partial r} \quad \text{Equation 77}$$

$$\frac{\partial c_s}{\partial r} \Big|_{r=0} = 0 \quad \text{Equation 78}$$

$$D_s \frac{\partial c_s}{\partial r} \Big|_{r=R_s} = -j_s \quad \text{Equation 79}$$

In the above equations, $c_{s,j}$ [mol/s²] refers to the lithium-ion concentration in the solid electrode particle for the electrode k (k refers to either the positive or negative electrodes). Likewise, $D_{s,k}$ [m²/s] represents the diffusivity of lithium-ions in the solid electrode particles.

The initial condition, defined below, is known by the user.

$$c_{s,k}(r, t = 0) = c_{s,k}^{init} \quad \text{Equation 80}$$

$c_{s,j}^{init}$ represents the initial lithium-ion concentration in the solid electrode particles.

Before proceeding, the expanded version of the above PDE (Equation 77) is given below and will be used later during the derivation of the relevant solvers.

$$\frac{\partial c_{s,j}}{\partial t} = \frac{D_{s,j}}{r^2} \frac{\partial}{\partial r} \left(r^2 \frac{\partial c_{s,j}}{\partial r} \right) = D_{s,j} \frac{\partial^2 c_{s,j}}{\partial r^2} + 2 \frac{D_{s,j}}{r} \frac{\partial c_{s,j}}{\partial r} \quad \text{Equation 81}$$

Moreover, the scaled version of the above PDE, along with the scale boundary conditions, is also presented below for convenience. The scaled concentration, x_j , and the radius variables, R_j , are obtained by scaling them with respect to the maximum concentration, $c_{s,j}^{max}$, and the electrode radius, R_j , respectively (i.e., $x_j = \frac{c_{s,j}}{c_{s,j}^{max}}$ and $\bar{r}_j = r_j/R_j$).

$$\frac{\partial x_j}{\partial t} = \frac{D_{s,j}}{R_j^2} \frac{1}{\bar{r}_j^2} \frac{\partial}{\partial \bar{r}_j} \left(\bar{r}_j^2 \frac{\partial x_j}{\partial \bar{r}_j} \right)$$

$$x_j(\bar{r}, t = 0) = x_j^{init}$$

$$\frac{\partial x_j}{\partial r} \Big|_{r=0} = 0$$

$$\frac{\partial x_j}{\partial r} \Big|_{r=1} = \frac{J_j R_j}{c_{s,max} D_{s,j}(t)} = \delta_j(t)$$

The above PDE (Equation 77) was solved numerically using the implicit Crank-Nicolson method, Eigen expansion method, and reduced order polynomial formulation (both two-order polynomial and higher order polynomial), and relevant solver equations are summarized in the following subsections.

5.4.1.1 Crank-Nicolson Approximations

This section mentions the relevant equations on the use of an implicit numerical scheme for lithium concentration PDE (Equation 77), the Crank-Nicolson (CN) scheme, which has a second-order accuracy in time¹⁹⁶. Here, the average of the backward and forward time difference is applied to the time derivative of

the diffusion PDE, while the centered-finite difference is applied to the spatial derivative. The following scheme is obtained after temporal and spatial discretization. Note that the spatial and temporal points are denoted by i and k , respectively, where $i = \{1, 2, \dots, N\}$ and $k = \{1, 2, \dots, K\}$.

$$\begin{aligned} \frac{c_i^{k+1} - c_i^k}{\Delta t} = & \frac{1}{2} \left(D_{s,j} \frac{c_{i+1}^{k+1} - 2c_i^{k+1} + c_{i-1}^{k+1}}{\Delta r_i^2} + \frac{D_{s,j}}{r_i} \frac{c_{i+1}^{k+1} - c_{i-1}^{k+1}}{\Delta r_i} + D_{s,j} \frac{c_{i+1}^k - 2c_i^k + c_{i-1}^k}{\Delta r_i^2} \right. \\ & \left. + \frac{D_{s,j}}{r_i} \frac{c_{i+1}^k - c_{i-1}^k}{\Delta r_i} \right) \end{aligned}$$

The rearrangement of the above expression leads to the following.

$$[1 + A]c_i^{k+1} - \left[\frac{A}{2} + \frac{B}{r_i} \right] c_{i+1}^{k+1} - \left[\frac{A}{2} - \frac{B}{r_i} \right] c_{i-1}^{k+1} = [1 + A]c_i^k + \left[\frac{A}{2} + \frac{B}{r_i} \right] c_{i+1}^k + \left[\frac{A}{2} - \frac{B}{r_i} \right] c_{i-1}^k$$

Furthermore, L'Hôpital's is applied to prevent the occurrence of singularity at the boundary condition, B.C. 1, when $r = 0$ ¹⁹⁷. The following are the results of discretization to the boundary conditions, B.C. 1 and B.C. 2.

$$(1 + 3A)c_1^{k+1} - 3Ac_2^{k+1} = (1 - 3A)c_1^k + 3Ac_2^k$$

$$(1 + A)c_N^{k+1} - Ac_{N-1}^{k+1} = (1 - A)c_N^k - \left(A - \frac{B}{R_j} \right) \left(\frac{2\Delta r J_j}{D_{s,j}} \right) + Ac_{N-1}^k$$

The above set of equations for each of the spatial nodal points can be described in a matrix form as follows.

$$\begin{aligned}
& \begin{bmatrix} 1+3A & -3A & 0 & \cdots & 0 \\ & \ddots & & & \\ & -\frac{A}{2} + \frac{B}{r_i} & 1+A & \frac{A}{2} + \frac{B}{r_i} & \\ & & & \ddots & \\ 0 & 0 & \cdots & -A & 1+A \end{bmatrix} \begin{bmatrix} c_1^{k+1} \\ \vdots \\ c_N^{k+1} \end{bmatrix} \\
&= \begin{bmatrix} 1+3A & -3A & \cdots & 0 \\ & \ddots & & \\ & \frac{A}{2} - \frac{B}{r_i} & 1+A & \frac{A}{2} + \frac{B}{r_i} \\ & & & \ddots \\ 0 & \cdots & -A & 1+A \end{bmatrix} \begin{bmatrix} c_1^k \\ \vdots \\ c_N^k \end{bmatrix} \\
&- \begin{bmatrix} 0 \\ \vdots \\ \left(A + \frac{B}{R} \right) \left(\frac{2\Delta r J_j}{D_{s,j}} \right) \end{bmatrix}
\end{aligned} \tag{Equation 82}$$

The above expression can be solved by matrix multiplication of it with the inverse of the matrix in the RHS. Also, the matrix contains a tridiagonal matrix which can be solved using Thomas, also known as the tridiagonal matrix (TDMA), in the case of the diagonally dominant matrix¹⁹⁸.

5.4.1.2 Eigen Expansion Method

Guo *et al.* utilized the Eigen expansion method to solve the PDE about lithium concentration in the solid electrode region profile (Equation 77), which resulted in a set of algebraic and ordinary differential (ODE) equations⁴³. When the boundary conditions are made homogeneous and the resultant variables are solved, the authors obtained the following expressions.

$$\begin{aligned}
\frac{du_{k,j}(t)}{dt} &= -\frac{\lambda_k^2 D_{s,p}(t)}{R_p^2} u_{k,p}(t) + \frac{2D_{s,p}(t)}{R_p^2} \delta_p(t) \\
u(0) &= 0 \\
x_j^{surf} &= x_{ini,j} + \frac{1}{5} \delta_j(t) + 3 \int_0^t \frac{D_{s,j}(\tau)}{R_j^2} \delta_j(\tau) dt + \sum_{k=1}^N \left[u_{j,k} - \frac{2\delta_j(t)}{\lambda_k^2} \right]
\end{aligned}$$

Where δ_j represents the scaled lithium-ion flux.

5.4.1.3 Reduced Order Polynomial Approximation

A volume-averaged quantity after application of volume averaging over a spherical coordinate is given by⁵⁵

$$\bar{f} = \frac{3}{r^3} \int_{r=0}^{r_k} r^2 f(r) dr$$

After the application of volume averaging quantity to LHS and the RHS of the Eq. 1 above, the following ordinary differential equation (ODE) is obtained.

$$\frac{\partial \bar{c}_{s,k}}{\partial t} = -3 \frac{j_k}{R_k} \quad \text{Equation 83}$$

With the initial condition

$$c_s(r, t = 0) = c_s^0$$

Furthermore, the lithium-ion concentration profile across the radial coordinates is approximated using a polynomial expression^{55,199}

- Two-order polynomial

$$\frac{\partial \bar{c}_{s,k}}{\partial t} = -3 \frac{j_k}{R_k}$$

$$c_s^{surf} = \bar{c}_{s,k} - \frac{R_k j_k}{D_s 5}$$

- Higher-order polynomial

$$\frac{\partial \bar{c}_{s,k}}{\partial t} = -3 \frac{j_k}{R_k}$$

$$\frac{d\bar{c}_{skr}}{dt} + \frac{30D_s}{R_s^2} \bar{c}_{skr} = -\frac{45}{2R_k^2} j_k$$

$$c_s^{surf} = \bar{c}_{s,k} - \frac{R_k}{35D_s} j_k + \frac{8R_k}{35} \bar{c}_{skr}$$

5.4.2 Electrolyte Concentration Solver

The mass transport of the lithium-ion within the electrolyte region across the battery cell is assumed to be driven by the concentration gradient and migration. The continuum-scale PDE, along with boundary and initial conditions, are given below ^{28,33,178}.

$$\frac{\partial \epsilon_e c_e}{\partial t} = \nabla \cdot (D_e \nabla c_e) + a_s(1 - t_+^0)j \quad \text{Equation 84}$$

$$\frac{\partial c_e}{\partial x} \Big|_{x=0} = \frac{\partial c_e}{\partial x} \Big|_{x=L^{tot}} = 0 \quad \text{Equation 85}$$

$$c \Big|_{t=0} = c_{e,init} \quad \text{Equation 86}$$

In the above equations, c_e and $c_{e,init}$ refers to the lithium-ion concentration and the initial lithium-ion concentration, respectively (with the units of mol/m^3).

5.4.2.1 Finite Element Formulation

The volume integral formulation for the above PDE, under a constant electrolyte volume fraction, is given below ^{178,199}.

$$\int \frac{\partial \epsilon_e c_e}{\partial t} dx = \int \frac{\partial}{\partial x} \left(D_e^{eff} \frac{\partial c_e}{\partial x} \right) dx + \int \frac{1 - t_+^0}{F} dx$$

In the matrix representation after the discretization of the above integral form and the consideration of the boundary conditions is given below ¹⁷⁸.

$$\begin{aligned}
& \begin{bmatrix} \epsilon_{e,1} + \frac{\Delta t}{2\Delta x_1} \left(\frac{D_{e,1}^{eff} + D_{e,2}^{eff}}{x_2 - x_1} \right) & -\frac{\Delta t}{2\Delta x_1} \left(\frac{D_{e,1}^{eff} + D_{e,2}^{eff}}{x_2 - x_1} \right) & 0 & \dots & 0 \\ \frac{\Delta t}{2\Delta x_2} \left(\frac{D_{e,1}^{eff} + D_{e,2}^{eff}}{x_2 - x_1} \right) & \epsilon_{e,2} + \frac{\Delta t}{2\Delta x_2} \left(\frac{D_{e,1}^{eff} + D_{e,2}^{eff}}{x_2 - x_1} + \frac{D_{e,2}^{eff} + D_{e,3}^{eff}}{x_3 - x_2} \right) & -\frac{\Delta t}{2\Delta x_2} \left(\frac{D_{e,2}^{eff} + D_{e,3}^{eff}}{x_3 - x_2} \right) & & \\ 0 & 0 & \dots & \ddots & \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} c_1^{k+1} \\ \vdots \\ c_N^{k+1} \end{bmatrix} \\
& = \begin{bmatrix} \epsilon_{e,1} c_1^k \\ \vdots \\ \epsilon_{e,N} c_N^k \end{bmatrix} + \frac{\Delta t(1 - t_0^+)}{F} \mathbf{J}_e
\end{aligned}$$

Equation 87

5.4.2.2 Reduced Order Polynomial Approximation

Similar to the methodology employed for the polynomial approximation the lithium concentrations in the solid electrode phase, approximations on the electrolyte spatial concentration distribution can be applied on the volume-averaged version of relevant PDE, boundary conditions, and initial conditions (Equation 84-Equation 87)⁵⁵. Once the resultant set of equations are rearranged and decoupled, we obtain the following equations for the electrolyte concentration distribution across the battery cell components⁵⁵.

$$c_{e,n}(x, t) = c_{lin} + \frac{q_{lin}(t)}{2L_n D_{e,n}} (L_n^2 - x^2) \quad \text{Equation 88}$$

$$c_{e,s}(x, t) = c_{lin} - \frac{q_{lin}(t)}{D_{e,s}} (x - L_n) + \left(\frac{q_{lin}(t) - q_{lip}(t)}{L_s D_{e,s}} \right) \frac{(x - L_n)^2}{2} \quad \text{Equation 89}$$

$$c_{e,p}(x, t) = c_{lip} + \frac{q_{lin}(t)}{2L_p D_{e,p}} (L_p^2 - (L_{cell} - x)^2) \quad \text{Equation 90}$$

The above equations rely on the time-variant quantities, c_{lip} , c_{lin} , q_{lip} , and q_{lin} . The algebraic expressions and ODEs that represent these quantities are expressed below⁵⁵.

$$c_{lin} = c_{lip} + \frac{L_s(q_{lin} + q_{lip})}{2D_{e,s}} \quad \text{Equation 91}$$

$$c_{lip} = c_{e,init} + \alpha_{in}q_{lin} + \alpha_{ip}q_{lip} \quad \text{Equation 92}$$

$$\frac{dq_{lin}}{dt} = \frac{1}{\mathcal{D}}(-B_2q_{lin} - A_2q_{lip} + A_3B_2j_n - A_2B_3j_p) \quad \text{Equation 93}$$

$$\frac{dq_{lip}}{dt} = \frac{1}{\mathcal{D}}(B_1q_{lin} - A_1q_{lip} + A_3B_1j_n - A_1B_3j_p) \quad \text{Equation 94}$$

Moreover, the intermediate variables and constants used in the above equations are expressed below ⁵⁵.

$$\alpha_{in} = \frac{\frac{L_n L_s \epsilon_n}{2D_{e,s}} + \frac{L_s^2 \epsilon_s}{6D_{e,s}} + \frac{L_n^2 \epsilon_n}{3D_{e,n}}}{L_n \epsilon_n + L_s \epsilon_s + L_p \epsilon_p} \quad \text{Equation 95}$$

$$\alpha_{ip} = \frac{\frac{L_p L_s \epsilon_p}{2D_{e,s}} + \frac{L_s^2 \epsilon_s}{6D_{e,s}} + \frac{L_p^2 \epsilon_p}{3D_{e,n}}}{L_n \epsilon_n + L_s \epsilon_s + L_p \epsilon_p} \quad \text{Equation 96}$$

$$A_1 = L_n \epsilon_n \alpha_n + \frac{L_s L_n \epsilon_n}{2D_{e,s}} + \frac{L_n^2 \epsilon_{e,n}}{3D_{e,n}} \quad \text{Equation 97}$$

$$A_2 = L_p \epsilon_{e,p} \alpha_{in} - \frac{L_p^2 \epsilon_{e,p}}{3D_{e,p}} \quad \text{Equation 98}$$

$$A_3 = (1 - t_+^0) a_n L_n \quad \text{Equation 99}$$

$$B_1 = L_p \epsilon_{e,p} \alpha_{in} \quad \text{Equation 100}$$

$$B_2 = L_p \epsilon_{e,n} \alpha_{ip} - \frac{L_p^2 \epsilon_{e,p}}{3D_{e,p}} \quad \text{Equation 101}$$

$$B_3 = (1 - t_+^0) a_p L_p \quad \text{Equation 102}$$

$$\mathfrak{D} = A_1 B_1 - A_2 B_1 \quad \text{Equation 103}$$

$$\begin{aligned}
\alpha_{in} &= \frac{\frac{L_n L_s \epsilon_n}{2D_{e,s}} + \frac{L_s^2 \epsilon_s}{6D_{e,s}} + \frac{L_n^2 \epsilon_n}{3D_{e,n}}}{L_n \epsilon_n + L_s \epsilon_s + L_p \epsilon_p} \alpha_{ip} = \frac{\frac{L_p L_s \epsilon_p}{2D_{e,s}} + \frac{L_s^2 \epsilon_s}{6D_{e,s}} + \frac{L_p^2 \epsilon_p}{3D_{e,n}}}{L_n \epsilon_n + L_s \epsilon_s + L_p \epsilon_p} A_1 = L_n \epsilon_n \alpha_n + \frac{L_s L_n \epsilon_n}{2D_{e,s}} + \frac{L_n^2 \epsilon_{e,n}}{3D_{e,n}} A_2 \\
\alpha_{ip} &= \frac{\frac{L_p L_s \epsilon_p}{2D_{e,s}} + \frac{L_s^2 \epsilon_s}{6D_{e,s}} + \frac{L_p^2 \epsilon_p}{3D_{e,n}}}{L_n \epsilon_n + L_s \epsilon_s + L_p \epsilon_p} A_1 = L_n \epsilon_n \alpha_n + \frac{L_s L_n \epsilon_n}{2D_{e,s}} + \frac{L_n^2 \epsilon_{e,n}}{3D_{e,n}} A_2 = L_p \epsilon_{e,p} \alpha_{in} - \frac{L_p^2 \epsilon_{e,p}}{3D_{e,p}} \text{The} \\
A_1 &= L_n \epsilon_n \alpha_n + \frac{L_s L_n \epsilon_n}{2D_{e,s}} + \frac{L_n^2 \epsilon_{e,n}}{3D_{e,n}} A_2 = L_p \epsilon_{e,p} \alpha_{in} - \frac{L_p^2 \epsilon_{e,p}}{3D_{e,p}} \text{The} \\
A_2 &= L_p \epsilon_{e,p} \alpha_{in} - \frac{L_p^2 \epsilon_{e,p}}{3D_{e,p}} \text{The} \\
&= L_p \epsilon_{e,p} \alpha_{in} - \frac{L_p^2 \epsilon_{e,p}}{3D_{e,p}} \text{The}
\end{aligned}$$

5.4.3 Battery Models

5.4.3.1 Single Particle Model

Single particle model (SPM) is the most widely used simplified model to the pseudo-two-dimensional (P2D) model due to its ease of implementation. It assumes that the lithium concentration profile across each of the electrodes can be approximated to that across a single spherical particle and the electrochemical polarization from the electrolyte dynamics is negligible (Figure 30A). Consequently, the PDE, that represents the diffusivity of the lithium in the solid electrode, needs to be solved for both positive and negative electrodes. These PDEs for both the electrodes are presented below.^{19,21,43}

$$\begin{aligned}
\frac{\partial c_{s,p}}{\partial t} &= \frac{1}{r^2} \frac{\partial \left(D_{s,p} r^2 \left(\frac{\partial c_{s,p}}{\partial r} \right) \right)}{\partial r} \\
\frac{\partial c_{s,n}}{\partial t} &= \frac{1}{r^2} \frac{\partial \left(D_{s,n} r^2 \left(\frac{\partial c_{s,n}}{\partial r} \right) \right)}{\partial r}
\end{aligned}$$

Both above PDEs are subject to the boundary conditions, B.C. 1 and B.C. 2. Furthermore, SPM assumes that the lithium flux across the electrode-electrolyte interface is constant and can be represented as given below⁴³.

$$J_p = \frac{I_{app}}{FS_p} = \frac{I_{app}R_p}{3F\epsilon_p Vol_p}$$

$$J_n = -\frac{I_{app}}{FS_n} = \frac{I_{app}R_n}{3F\epsilon_n Vol_n}$$

Where, i_{app} represents the applied current [A] across the battery cell terminal. Here, by convention, the charge and discharge currents are represented by positive and negative values, respectively. Moreover, J_p and J_n refers to the lithium-ion flux in the positive and negative electrode regions, respectively (in $mol\ m^{-2}\ s^{-1}$). To be clear, negative flux represents the lithium flux out of the particle, while the positive flux represents the lithium flux out the electrode particle. The sign in the above equation makes sense since the flux leaves a positive electrode and enters the negative electrode during charge and the opposite is true for discharge. S_p and S_n represent the total electrochemically active area of the positive and negative electrode, respectively (in the units of m^2). Meanwhile, ϵ_j and Vol_j represents the volume fraction of the electrochemically active region of the solid electrode and the total volume of the electrode.

When the polarization from the electrolyte solution is considered known and given (i.e., $\phi_{e,n} \big|_{x=0} - \phi_{e,p} \big|_{x=L} = I_{app}R_{int}$), the Butler-Volmer (BV) equation can be rearranged to solve for the battery cell terminal voltage and this leads to the following expression⁴³.

$$V_{cell} = OCP_p(SOC_p, T) - OCP_n(SOC_n, T) + \frac{2RT}{F} \operatorname{asinh}\left(\frac{m_p}{2}\right) + \frac{2RT}{F} \operatorname{asinh}\left(\frac{m_n}{2}\right) + I_{app}R_{int}$$

Equation 104

Where OCP_p and OCP_n represent the open-circuit potential of the positive and negative electrode, respectively. The result from the first and the second terms represents the thermodynamic battery cell potential at a given electrode SOC's and temperature, while that from the fourth and fifth terms represents the polarization from the lithium diffusivity in the solid electrode region. Finally, the final term is the polarization from the electrolyte solution polarization as mentioned previously.

Similar to the battery cell model assumptions mentioned in previously (refer to Section 3.4.7 Battery Cell Model), the assumptions pertaining to the single particle model are presented below.

- The lithium-ion flux (denoted by the variable j_l) entering or exiting the negative and positive electrodes is constant throughout the negative and positive electrode regions, respectively.⁴³
- The overpotential from the concentration and migration effects in the electrolyte during battery cell charge/discharge operations are ignored.⁴³ These contributions along with the other overpotential contributions (e.g., the electric resistances in the current collector and external connections) are coupled in the bulk resistance terms. For low charge and discharge C-rates (such as in the EV applications) the overpotential from the electrolytes dynamics can be ignored.^{43,44}
- The anodic and cathodic apparent transfer co-efficient (α_a and α_c) is equal to 0.5.^{19,43,45}
- The resulting overpotential, loss of lithium-ion, and battery cell internal resistance change from the side reactions during battery cell charge and discharge operations are ignored.

5.4.3.2 Enhanced Single Particle Model

Moura et al. incorporated the electrolyte dynamics into the SPM to alleviate its major limitation, which is its applicability for low applied currents ($< 0.5C$)⁴⁵. This model is referred to as the enhanced single particle model (ESPM) in this work. In this subsection, the model equations for the enhanced single particle model, as mentioned by Moura et al., are presented here with slight modifications⁴⁵. These modifications stem from the fact that the definition of applied current, I_{app} , defined here is different than that defined by Moura et al. The PDE representing the lithium-ion concentration in the electrolyte region was introduced above and its formulation in 1D space is given below.

$$\frac{\partial c_e}{\partial t} = \frac{\partial}{\partial x} \left(\frac{D_e}{\epsilon_{e,j}} \frac{\partial c_e}{\partial x} \right) + a_{s,j} (1 - t_c^+) J_j$$

Specifically, the above expression for the negative electrode, separator, and positive electrode is as follows.

$$\frac{\partial c_{e,p}}{\partial t} = \frac{\partial}{\partial x} \left(\frac{D_{e,p}}{\epsilon_{e,j}} \frac{\partial c_{e,p}}{\partial x} \right) + a_{s,p}(1 - t_c^+) J_p \quad \text{Equation 105}$$

$$\frac{\partial c_{e,sep}}{\partial t} = \frac{\partial}{\partial x} \left(\frac{D_{e,sep}}{\epsilon_{e,sep}} \frac{\partial c_{e,sep}}{\partial x} \right) \quad \text{Equation 106}$$

$$\frac{\partial c_{e,n}}{\partial t} = \frac{\partial}{\partial x} \left(\frac{D_{e,n}}{\epsilon_{e,n}} \frac{\partial c_{e,n}}{\partial x} \right) + a_{s,n}(1 - t_c^+) J_n \quad \text{Equation 107}$$

Finally, the cell terminal voltage from the enhanced single particle model is given by the following expression.

$$\begin{aligned} V_{cell} = & OCP_p - OCP_n + \frac{2RT}{F} \arcsin\left(\frac{m_p}{2}\right) \\ & + \frac{2RT}{F} \arcsin\left(\frac{m_n}{2}\right) + I_{app} R_{cell} \\ & + \frac{L_p + 2L_{sep} + L_n}{2\kappa} I_{app}(t) \\ & + k_{conc}(t) [\ln c_{e,p}(L_{cell}, t) - \ln c_{e,n}(0, t)] \end{aligned} \quad \text{Equation 108}$$

In the above equation, there are additional terms that stem from the resultant electrochemical polarization from the electrolyte dynamics compared to similar equation from the SPM (Equation 104). The following lists the model assumptions of the enhanced single particle model.

- The lithium-ion flux (denoted by the variable j_l) entering or exiting the negative and positive electrodes is constant throughout the negative and positive electrode regions, respectively.⁴³
- The anodic and cathodic apparent transfer co-efficient (α_a and α_c) is equal to 0.5.^{19,43,45}
- The resulting overpotential, loss of lithium-ion, and battery cell internal resistance change from the side reactions during battery cell charge and discharge operations are ignored.

5.4.3.3 Reduced Order Pseudo Two-Dimensional Model

A reduced order P2D is also used in this work due to its suitability as an embedded battery management system. The implementation of the reduced-order P2D model is similar to that found in the literature with some modifications.^{55,181,200,201} Similar to other works in the literature, it is assumed that the lithium-ion flux at the electrode-electrolyte interface is spatially consistent, and the lithium-ion flux is calculated using those utilized for SPM and ESPM models. The PDEs for the lithium concentrations in the solid electrode regions and the liquid electrolyte region are then solved using the polynomial approximation and FVM, respectively. Later, the cell terminal voltage is calculated using Equation 25, above. Then, the PDEs for the spatial potential distribution in the electrode and electrolyte regions are solved using the FVM. The PDEs for the spatial electrode distribution in the solid electrode and liquid electrolyte region are presented below, respectively.

$$\frac{\partial}{\partial x} \left(\sigma_{eff} \frac{\partial \phi}{\partial x} \right) = a_s F J_j \quad \text{Equation 109}$$

$$\frac{\partial}{\partial x} \left(\kappa_{eff} \frac{\partial \phi_e}{\partial x} \right) + \frac{\partial}{\partial x} \left(\kappa_{D,eff} \frac{\partial \ln c_e}{\partial x} \right) + a_s F J_j = 0 \quad \text{Equation 110}$$

Where,

$$\kappa_{D,eff} = \frac{2RT\kappa(t_+ - 1)}{F} \epsilon_e^{brugg} \quad \text{Equation 111}$$

In the above equations, the variables ϕ_s and ϕ_e represents the electrostatic potential in the solid electrode and liquid electrolyte regions, respectively. Furthermore, σ_{eff} and a_s represents the effective bulk ionic conductivity (calculated from the bulk ionic conductivity, $\sigma_{eff} = \sigma_s * \epsilon^{brugg}$) of the solid electrode

particles and specific interfacial surface area, respectively. Similarly, κ_{eff} is the effective electrolyte conductivity (where $\kappa_{eff} = \kappa * \epsilon^{brugg}$).

The following matrices for the positive and negative electrode, respectively, are obtained after the implementation of the finite volume equation and subsequent approximations.¹⁸¹

$$\begin{bmatrix} -1 & 1 & 0 & \cdots & 0 \\ 1 & -2 & 1 & & \\ & & & \ddots & \\ 0 & 0 & \cdots & 1 & -2 \end{bmatrix} \begin{bmatrix} \phi_{p,1} \\ \vdots \\ \phi_{p,N} \end{bmatrix} = \frac{\Delta x^2}{\sigma_{eff}} \begin{bmatrix} J_{p,1} \\ \vdots \\ J_{p,2} \end{bmatrix} - 2 \begin{bmatrix} 0 \\ \vdots \\ \phi_p|_{x=L^{TOT}} \end{bmatrix} \quad \text{Equation 112}$$

$$\begin{bmatrix} -3 & 1 & 0 & \cdots & 0 \\ 1 & -2 & 1 & & \\ & & & \ddots & \\ 0 & 0 & \cdots & 1 & -1 \end{bmatrix} \begin{bmatrix} \phi_{n,1} \\ \vdots \\ \phi_{n,N} \end{bmatrix} = \frac{\Delta x^2}{\sigma_{eff}} \begin{bmatrix} J_{n,1} \\ \vdots \\ J_{n,2} \end{bmatrix} - 2 \begin{bmatrix} \phi_n|_{x=0} \\ \vdots \\ 0 \end{bmatrix} \quad \text{Equation 113}$$

The following matrix is obtained for the spatial electrolyte potential distribution.¹⁸¹

$$\begin{bmatrix} \frac{\kappa_{eff,e,1}}{x_2 - x_1} & \frac{\kappa_{e,1+1/2}}{x_2 - x_1} & 0 & \cdots & 0 \\ \frac{\kappa_{eff,e,2}}{x_2 - x_1} & -\left(\frac{\kappa_{eff,2-\frac{1}{2}}}{x_2 - x_1} + \frac{\kappa_{eff,2+\frac{1}{2}}}{x_2 - x_1}\right) & \frac{\kappa_{eff,2+\frac{1}{2}}}{x_2 - x_1} & & \\ & & \ddots & & \\ 0 & 0 & \cdots & \frac{\kappa_{eff,e-\frac{1}{2}}}{x_{N_e} - x_{N_e-1}} & \frac{\kappa_{eff,e}}{x_{N_e} - x_{N_e-1}} \end{bmatrix} \quad \text{Equation 114}$$

$$= \begin{bmatrix} \frac{\kappa_{D,N_e+\frac{1}{2}}}{x_2 - x_1} \left(\frac{c_{e,2} - c_{e,1}}{c_{e,2} - c_{e,1}} \right) \\ \vdots \\ \frac{\kappa_{D,N_e-\frac{1}{2}}}{x_{N_e} - x_{N_e-1}} \left(\frac{c_{e,N_e} - c_{e,N_e-1}}{c_{e,N_e} - c_{e,N_e-1}} \right) \end{bmatrix} - a_s F j_j \Delta x$$

5.4.4 Energy Balance

In this work, the lumped thermal model is used where the spatial thermal distribution across the battery cell is neglected, and the battery cell thermal evolution is assumed to be consistent throughout the battery cell.

The ODE representing the relevant energy balance is described below⁴³.

$$\begin{aligned} \rho C_p Vol_{cell} \frac{dT}{dt} = IT \left[\frac{\partial U_p}{\partial t}(soc_{p,surf}) - \frac{\partial U_n}{\partial t}(soc_{n,surf}) \right] + I * (V_{cell} \\ - (U_p(soc_{p,surf}) - U_n(soc_{n,surf})) + hA_{cell}(T_{cell} \\ - T_{amb})) \end{aligned} \quad \text{Equation 115}$$

In this above equation, the three terms on the RHS represent the thermal contributions from the reversible electrode reaction entropy change, irreversible electrode polarizations, and the heat transfer from the surrounding (Figure 30B). Furthermore, the above equation is subjected to the following initial condition which states that the initial battery temperature at the start of any simulation is equal to the ambient temperature.

$$T \big|_{t=0} = T_{amb} \quad \text{Equation 116}$$

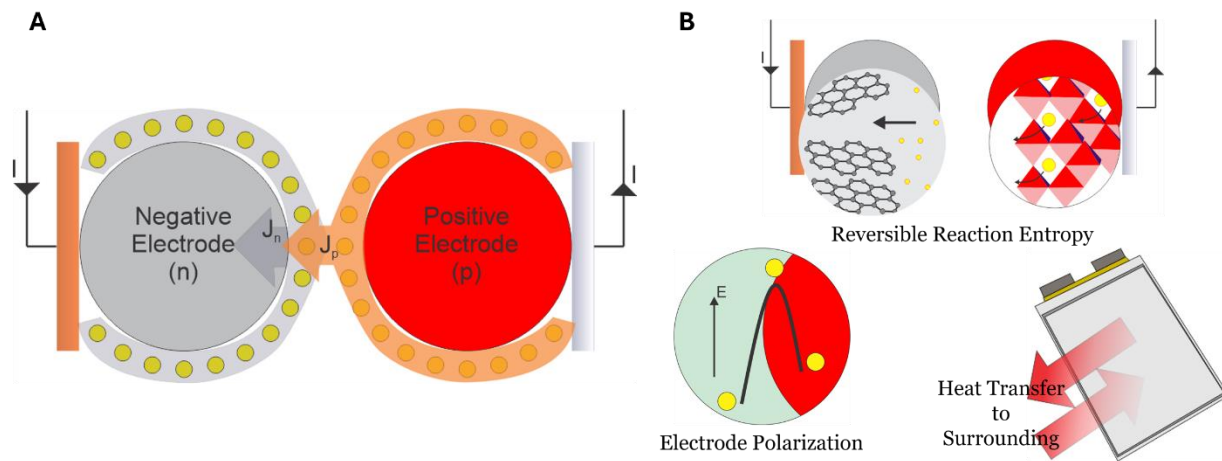


Figure 30: (A) Depiction of the single particle model and (B) various thermal sources considered in the lumped thermal model.

Table 10: Battery Cell Parameters

Symbol	Positive Electrode	Separator	Negative Electrode	Unit
L_j	70*	20*	73.5*	μm
S_j	1.1167*		0.7824*	mm^2
$c_{s,j,max}$	51410		31833	mol m^{-3}
R_j	8.5*		12.5*	μm
$k_{j,ref}$	$6.67 \cdot 10^{-11}$		$1.76 \cdot 10^{-11}$	$\text{m}^2 \text{mol}^{-0.5} \text{s}^{-1}$
$Ea_{r,j}$	58000*		28000*	$\text{m}^2 \text{s}^{-1}$
$D_{s,j,ref}$	$1 \cdot 10^{-14}$ *		$3.9 \cdot 10^{-14}$ *	$\text{m}^2 \text{s}^{-1}$
$Ea_{D,j}$	2900*		3500*	$\text{m}^2 \text{s}^{-1}$
κ_e	0.2875**			S m^{-1}
t_c^+	0.38**			
$c_{e,init}$	1000*			mol m^{-3}
ρ	1626*			kg m^{-3}
Vol_{cell}	$2.425 \cdot 10^{-8}$			m^3
C_p	750*			J K^{-1}
T_{ref}	298.15*			K
hA_{cell}	0.085*			$\text{W m}^{-2} \text{K}^{-1}$

* ref ⁴³**ref ¹⁸¹

5.4.5 Sigma-Point Non-Linear Kalman Filter

Research works have employed non-linear Kalman filters have been employed on the state-space representations of the continuum-scale battery models for battery cell states estimations in practical applications^{202,203}. Kalman filters, which comes under the domain of sequential probabilistic inference, find the efficient estimation of the system (for e.g., the SOC of positive and negative electrodes) using the prior estimates while considering the randomness of the system's process and sensor noises. Sigma-Point non-linear Kalman filter (SPKF) is suitable for non-linear models (as is the case of the battery models in this work) since (1) it does not require the calculation of the derivatives and (2) is samples the strategic points in the probability distribution function of the state and output equations²⁰⁴. The state space representation contains the state and output equations as defined below²⁰⁴.

$$\begin{aligned} \mathbf{x}_k &= f(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}) \\ \mathbf{y}_k &= h(\mathbf{x}_k, \mathbf{u}_k, \mathbf{v}_k) \end{aligned} \quad \text{Equation 117}$$

In the above, $\mathbf{x}_k$ and $\mathbf{y}_k$ represents the random vector containing the estimates of the states and outputs, respectively. Meanwhile, $\mathbf{u}_k$, $\mathbf{w}_k$, and $\mathbf{v}_k$ represent the random vectors for the inputs, process noise in the input signal, and sensor noise in output. Moreover, $f(\dots)$ and $h(\dots)$ represent the functions for the states and outputs, respectively.

For the practical state's estimations, algorithms pertaining to sequential probabilistic inference, such as SPKF, are critical and they can lead to increased computational resource utilization and solution times. Hence, in this work, SPKF was implemented on the isothermal single particle model to estimate its computational overload. The following sub-section provides the relevant SPKF equations.

5.4.5.1 Isothermal Single Particle Model

The random vectors for the states and outputs used in this work are defined as follows.

$$\begin{aligned} \mathbf{x}_k &= \begin{bmatrix} soc_p \\ soc_n \end{bmatrix} \\ \mathbf{y}_k &= [V_{cell}] \end{aligned} \quad \text{Equation 118}$$

The numerical schemes for the determination of the lithium concentration in the electrode particles, as delineated in the above sections, are used for the electrode's soc_j determination. Meanwhile, the equation for the terminal voltage from the single particle model (Equation 104) is used for the calculation of V_{cell} .

The SPKF is carried out in six steps, as outlines by Plett in his book on Battery Management Systems, and the relevant equations for these steps are as follows ²⁰⁴. Even before using the SPKF equations, the random vector and their covariance matrices for the states and noises are augmented.

$$\begin{aligned} \mathbf{x}_k^{aug} &= \begin{bmatrix} \mathbf{x}_k \\ \mathbf{w}_k \\ \mathbf{v}_k \end{bmatrix} \\ \Sigma_k^{aug} &= \begin{bmatrix} \Sigma_{\tilde{x},k} & 0 & 0 \\ 0 & \Sigma_{\tilde{w}} & 0 \\ 0 & 0 & \Sigma_{\tilde{v}} \end{bmatrix} \end{aligned} \quad \text{Equation 119}$$

Then the sigma -points using the augmented states and covariance matrices are generated which are stored in the following matrix. In the matrix below, the first column contains the state estimates. The subsequent L (where L denotes the dimension of the augmented matrix) columns contains the result of addition of augmented state vector with $\gamma \sqrt{\Sigma_k^{aug}}$. Similarly, the next L columns contain the result of the subtraction of augmented state vector with $\gamma \sqrt{\Sigma_k^{aug}}$. The resultant matrix has dimensions of $(L \times (2L+1))$. Note that in case of states being positive and negative SOC, L is equal to two ($L = 2$) and, hence, five sigma points are generated.

$$\mathbf{x}_{k-1}^{aug} = \left\{ \mathbf{x}_{k-1}^{aug}, \mathbf{x}_k^{aug} + \gamma \sqrt{\boldsymbol{\Sigma}_k^{aug}}, \mathbf{x}_k^{aug} + \gamma \sqrt{\boldsymbol{\Sigma}_k^{aug}} \right\} \quad \text{Equation 120}$$

The rows of the above augmented matrix are then split to create the matrices, $\mathbf{x}_{k-1}^x$, $\mathbf{x}_{k-1}^w$, and $\mathbf{x}_{k-1}^v$ for the states, system noise, and sensor noise respectively. These matrices are then used as follows.

Step 1: Preliminary states estimations, $\hat{\mathbf{x}}_k^-$

$$\begin{aligned} \hat{\mathbf{x}}_k^- &= \mathbb{E}[f(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}) | \mathbb{Y}_{k-1}] \\ &\approx \sum_0^p \alpha_i^{(m)} f(\mathbf{x}_{k-1}^{x,+}, u_{k-1}, \mathbf{x}_{k-1}^{w,+}) = \sum_0^p \alpha_i^m \mathbf{x}_{k,i}^{x,-} \end{aligned} \quad \text{Equation 121}$$

Since centered-difference Kalman filter (CDKF) (and further assuming Gaussian distribution) is used here the value of $\alpha_i^{(m)}$ is as follows.

$$\alpha_0^{(m)} = \frac{3-L}{3} \quad \text{Equation 122}$$

$$\alpha_i^{(m)} = \frac{1}{6} \quad \text{Equation 123}$$

Step 2: Preliminary covariance estimates, $\boldsymbol{\Sigma}_{\tilde{\mathbf{x}},k}^-$

$$\boldsymbol{\Sigma}_{\tilde{\mathbf{x}},k}^- = \sum_0^p \alpha_i^{(c)} \left((\tilde{\mathbf{x}}_{k,i}^{x,-}) (\tilde{\mathbf{x}}_{k,i}^{x,-})^T \right) \quad \text{Equation 124}$$

Again, for the Gaussian distribution on the CDKF, the following is obtained for the $\alpha_i^{(c)}$.

$$\alpha_i^{(c)} = \frac{3-L}{3} \quad \text{Equation 125}$$

$$\alpha_i^{(c)} = \frac{1}{6} \quad \text{Equation 126}$$

Step 3: Calculation of the model output

In our case, since there are five sigma points, the output is calculated five times on each sigma point before the final output estimation is performed as represented below.

$$\begin{aligned} \Sigma_{\tilde{x},k}^- &= \sum_0^p \alpha_i^{(c)} \left((\tilde{\mathbf{x}}_{k,i}^{x,-}) (\tilde{\mathbf{x}}_{k,i}^{x,-})^T \right) \\ \hat{y}_k &= \mathbb{E}[h(\mathbf{x}_k, \mathbf{u}_k, \mathbf{v}_k) | \mathbb{Y}_{k-1}] \approx \sum_0^p \alpha_i^{(m)} f(\mathbf{x}_{k-1}^{x,+}, \mathbf{u}_{k-1}, \mathbf{x}_{k-1}^{v,+}) \\ &= \sum_0^p \alpha_i^m \mathbf{y}_{k,i}^{x,-} \end{aligned} \quad \text{Equation 127}$$

Step 4: Estimate of the estimator gain matrix, $\mathbf{L}_k$, using the output sensor reading

$$\Sigma_{\tilde{y},k}^- = \sum_0^p \alpha_i^{(c)} (\mathbf{y}_{k,i} - \hat{y}_k) (\mathbf{y}_{k,i} - \hat{y}_k)^T \quad \text{Equation 128}$$

$$\Sigma_{\tilde{x}\tilde{y},k}^- = \sum_0^p \alpha_i^{(c)} (\mathbf{x}_{k,i}^x - \hat{x}_k) (\mathbf{y}_{k,i} - \hat{y}_k)^T \quad \text{Equation 129}$$

$$\mathbf{L}_k = \Sigma_{\tilde{x}\tilde{y},k}^- \Sigma_{\tilde{y},k}^{-1} \quad \text{Equation 130}$$

Step 5: State Estimation

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \mathbf{L}_k(\mathbf{y}_k - \hat{\mathbf{y}}_k) \quad \text{Equation 131}$$

Step 6: Covariance Estimation

$$\Sigma_{\tilde{\mathbf{x}},k}^+ = \Sigma_{\tilde{\mathbf{x}},k}^- - \mathbf{L}_k \Sigma_{\tilde{\mathbf{y}},k}^- \mathbf{L}_k^T \quad \text{Equation 132}$$

5.4.6 Programming Language

The code for this work was written inhouse in C++ and Python programming languages using the object-oriented programming (OOP) approach^{205,206}. Both programming languages have various inherent relative advantage and limitations. Amongst these pros and cons, C++ boosts a relatively higher computational performance, while Python programming language is relatively easier to learn and implement^{207,208}. OOP programming approach was utilized to ensure the modularity of electrochemical, thermal, and degradation models. The codebase architecture is depicted by a class diagram as specified by the unified modeling language (UML)²⁰⁹.

Various classes' names and their relationships are depicted in the conceptual class diagram. The BatteryCell, NElectrode, Electrolyte, and PElectrode classes (note that the NElectrode and PElectrode class inherits from the Electrode class, the class diagram of the ElectrodeClass is presented in) store the various parameters and functions pertaining to the battery cell, negative electrode, electrolyte, and positive electrode, respectively. Furthermore, the BatteryCell class contains an instance of the NElectrode, Electrolyte, and PElectrode. BaseCycler, and other subsequent inherited classes, store the information relevant to the battery cycling and contains functions to provide the relevant cycling current based on the cycling time and other relevant cycling information. The electrochemical PDE, mentioned above, as well as the thermal and degradation model processes their own solver Class that contains relevant methods to solve for the desired output as required. The instances of the BatteryCell, any relevant solver class(es), and

the relevant cyclers class are included as the instance in the battery cell models, including the one for the single particle model (SPM) and enhanced single particle model (ESPM).

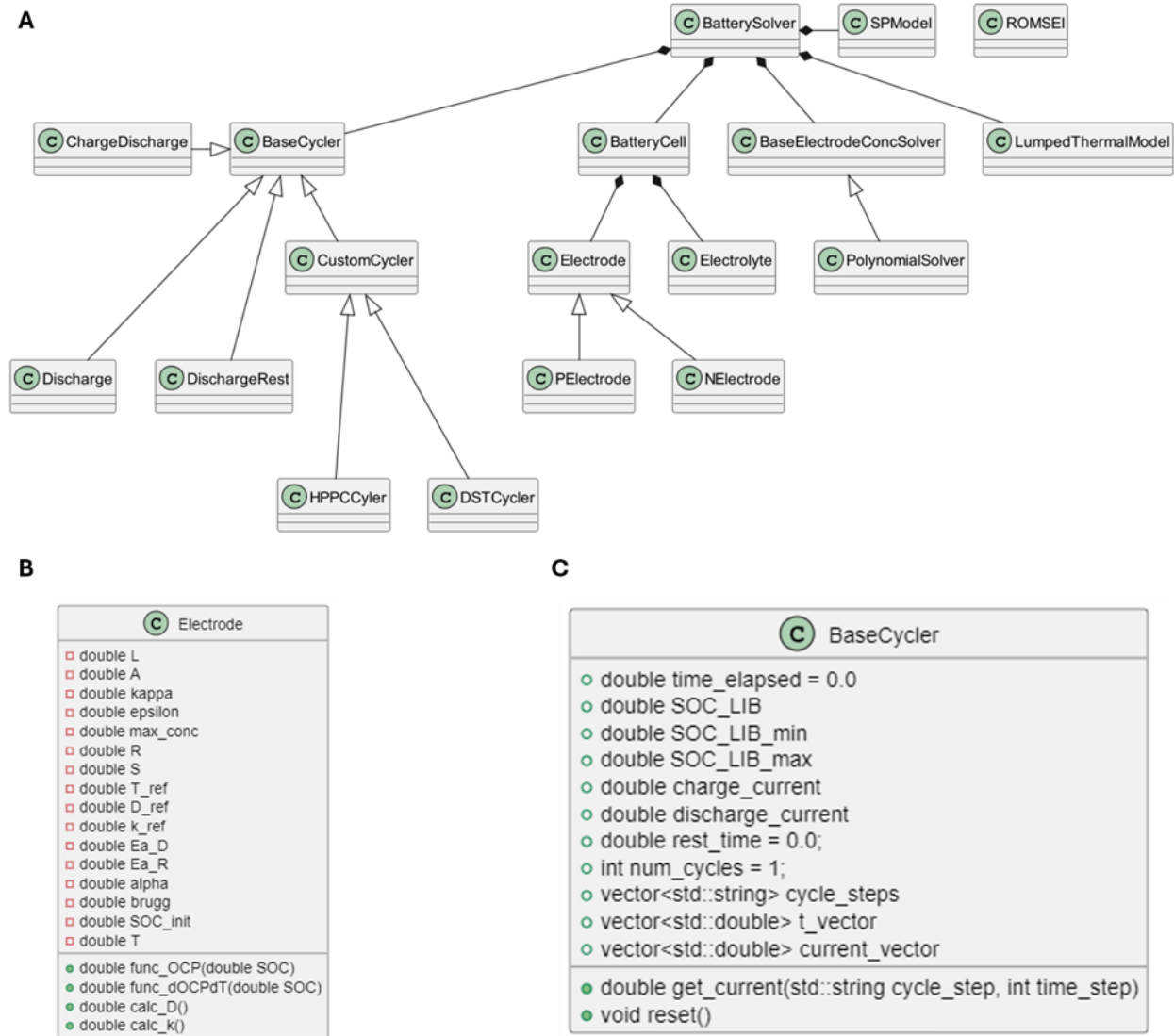


Figure 31: (A) Conceptual class diagram representing the codebase object-oriented architecture of the codebase. Class diagrams of the (B) Electrode, (C) BaseCycler

5.4.7 Hardware Specifications

The hardware specifications, specifically the central processing unit (CPU) and the random-access memory (RAM) specifications are listed in the table below. (Table 11). Raspberry Pi 3 model B was used as the embedded system in this work and its hardware specifications are listed accordingly.

Table 11: CPU and Memory Hardware Specifications used in this work.

	Embedded	Personal Computer	Cloud
CPU	Broadcom BCM2837 (1.2 GHz)	AMD Ryzen 7 6800H (3.2 GHz)	Intel(R) Xeon(R) CPU E5-2680 v3 (2.50GHz)
RAM	1 GB LPDDR2	16.0 GB	1 GB DDR3

5.5 Results and Discussions

The lithium-ion concentration in the solid electrode region was solved using various solvers, namely, Crank-Nicolson, eigen expansion method, and reduced-order polynomial approximations (second and higher order) on volume averaged quantities. The electrode state-of-charge (SOC) for the negative and the positive electrode was solved using the solvers mentioned above during various battery cycling steps, as depicted below (Figure 32). The solution time reduction of 46 % was observed when the Thomas algorithm (TDMA) was used instead of performing the matrix inversion for the Crank-Nicolson scheme in the Python programming language on the personal computer (PC) system. Also, the memory usage was at least 10 times lower for the Thomas algorithm than for the matrix inversion (refer to Table 12). Therefore, the Thomas algorithm is applied to the Crank-Nicolson scheme (given the matrix system is diagonally dominant) after that. For the Eigen Expansion method, the number of roots required for the process is set to 5. When compared to the other number of roots for the Eigen Expansion method ($N = 2, 5, 10, 20, 50, \text{ and } 100$), $N = 5$ seems to give a good compromise between accuracy and solution speed (refer to Figure 38). Interestingly, the solution time scales linearly with the number of roots Figure 38A and Figure 38D). Also, the second-order polynomial approximation was dropped since it could not capture the polarization after the change in the cycling step (Figure 39). Figure 40 and Figure 32 shows the electrode SOC as calculated from different solvers under static and dynamic cycling loads, respectively. Overall, the solvers agreed well regarding the different cycling steps. When only charge or discharge cycling steps are

considered, solvers for the expansion and volume averaging methods with higher polynomial approximation agree with the Crank-Nicolson method. The disagreement between the electrode's SOC, as calculated from the Crank-Nicolson method, and the other solvers are during the initial periods with the absolute error magnitude of 0.008 (Figure 41A and B). As for the HPPC cycling step, errors of similar magnitude were observed between these methods, with the highest errors occurring at the beginning of each pulse discharge step (Figure 41C and D).

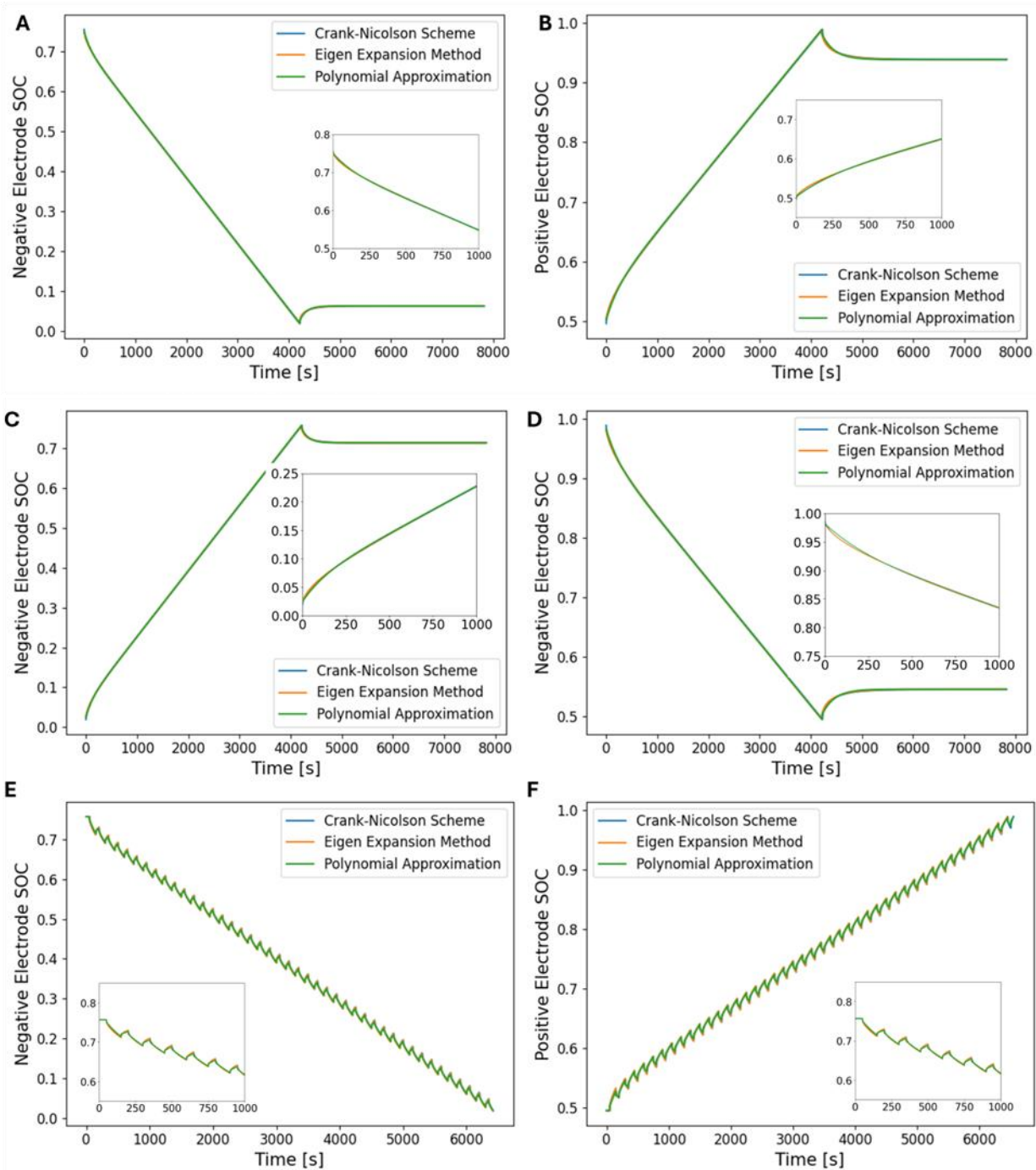


Figure 32: The solution of various solvers for the PDE about the lithium diffusivity in the solid electrode region for (A) negative electrode and (B) positive electrode under discharge-rest, (C) negative electrode and (D) positive electrode under charge-rest, and (E) negative electrode and (F) positive electrode under HPPC cycling steps.

The solver solution times of these solvers were compared in three computing hardware with different hardware specifications and two different programming languages (Figure 33). Generally, compared to the Crank-Nicolson method, eigen expansion and polynomial approximation on volume averaging methods are faster with an average factor (calculated by $\frac{avg. \text{ solution time}_{solver}}{avg. \text{ solution time}_{CN}}$) of about $2.1 * 10^{-1}$ and $1.36 * 10^{-3}$ for most cycling steps (except for HPPC) on the personal computing (PC) system. However, these factors were only $6.62 * 10^{-1}$ and $0.56 * 10^{-1}$ for the HPPC drive cycle for the respective solvers (Figure 33A) again on the PC system. This reduction in solution time for the HPPC drive cycle is most probably from the high number of set rest intervals, and consequently, all the solvers have to undergo these iterations. As expected, cloud and cloud-embedded hardware, which have lower performance specifications (in central processing units and memory) than the PC system, took longer than the PC system's solution times (Figure 33B and Figure 33C). Interestingly, while the Crank-Nicolson and Eigen Expansion method took an average of 1.37 and 1.36 times longer on the cloud, the polynomial approximation method took 2.78 times longer than the exact solvers' solution times on the PC system. Finally, when comparing programming languages on the same hardware (PC), Crank-Nicolson, eigen expansion and polynomial expansion methods proved to be, on average, 320.76, 158.07 and 18.23 times faster on C++, respectively, than on Python (Figure 33D). On the embedded systems, the Crank-Nicolson, eigen and the polynomial solver were over 130, 110 and 12 times faster on the C++ compared to the Python programming language, respectively.

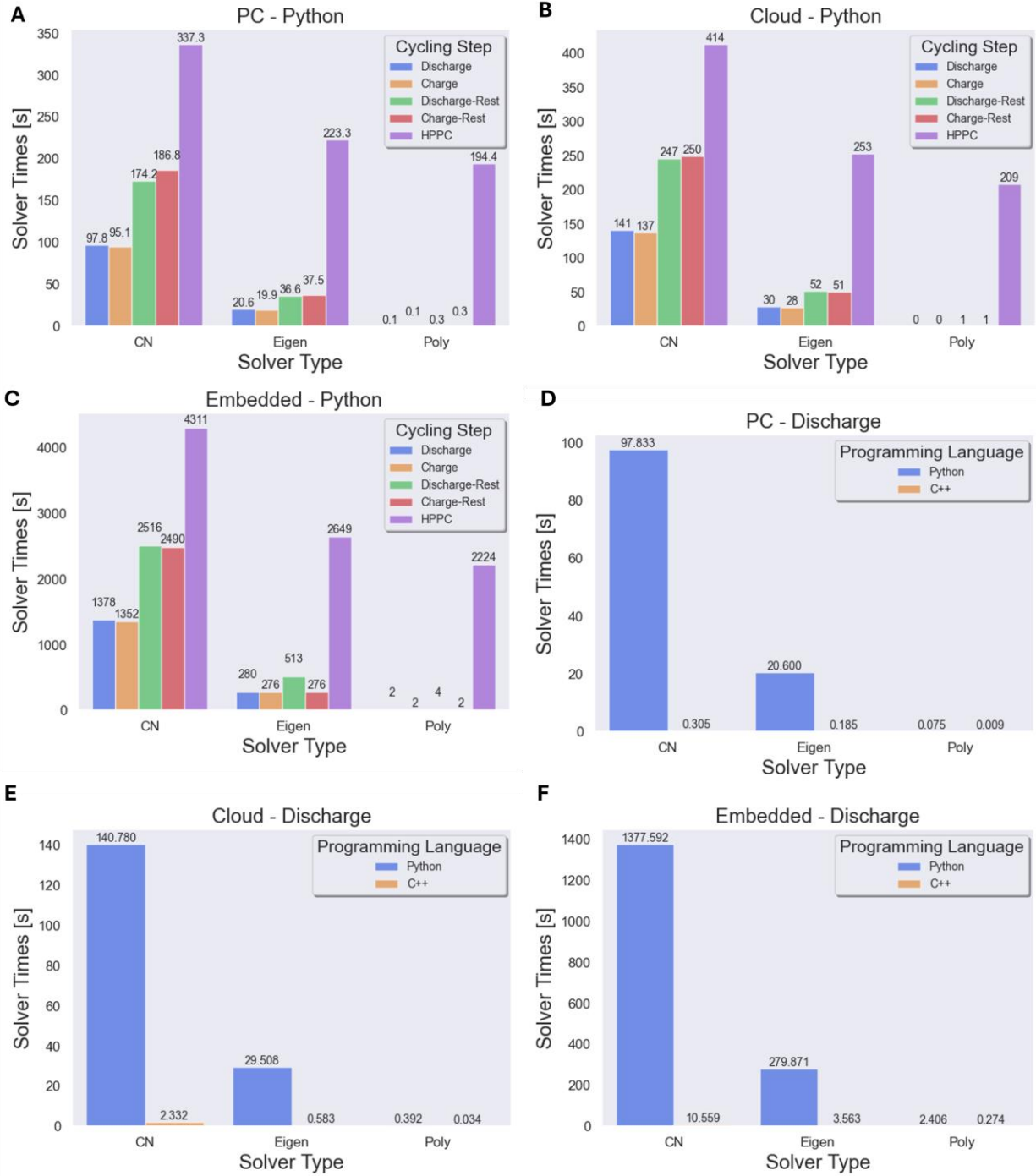


Figure 33: The solution times of various cycling steps in (A) PC, (B) cloud, and (C) embedded systems for the negative electrode. The solution times of the cycling steps performed in the Python and C++ programming language on the PC using (D) Eigen and (E) Polynomial solvers. (F) The solution times of the Polynomial solver on the embedded system are presented.

Since the enhanced single-particle model requires the solution for the PDE about the lithium-ion concentrations in the electrolyte regions (Equation 105–Equation 107) to be solved in each of its iterations, its solvers are tested in different computing hardware and programming languages. For testing purposes, the mentioned PDE is solved for a fixed number of iterations (the number of iterations is determined by the total time of the simulation). For testing purposes, the PDE, pertaining to the lithium-ion diffusivity in the liquid electrolyte region, is solved for each time step during the discharge cycle. For the solver based on the finite-volume method (FVM), the number of nodes is set to 10 for each of the negative electrode, separator, and positive electrode regions. Since there was a 20.5 % reduction in solution time when the Thomas algorithm was used instead of the regular matrix inversion (refer to Section S7), the Thomas algorithm is used throughout this section for the mentioned solver. However, it is to be noted that, given our simulation parameters, there was not a significant usage in the memory usage during the solution determination using matrix inversion or Thomas algorithm. The electrolyte concentration profile across the battery cell at various times, as measured by the FVM solver, is depicted in Figure 34A. In this work, the C++ programming language demonstrated at least a ten-fold (10X) decrease in solution times across all studied computing systems (Figure 34B). When the polynomial approximation on the average volume quantity is used, the solution time speeds up 115 times, with the actual solution time of 0.854s compared to 98.152s from the FMV method.

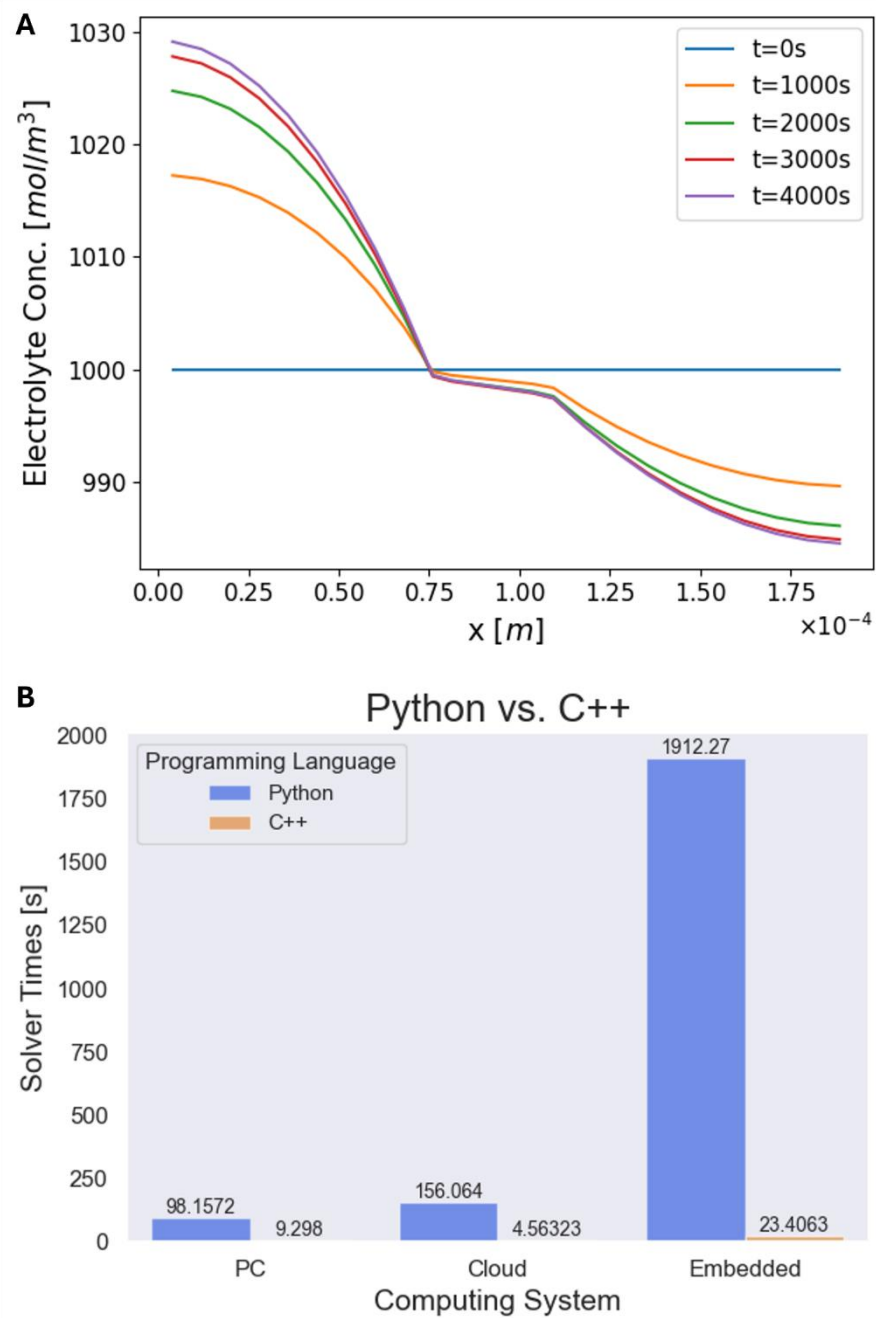


Figure 34: (A) Electrolyte concentration at different times across the battery cell during discharge. (B) The solution times of the FVM solver in Python and C++ programming languages across various computing systems.

Figure 35 shows the battery models under static and dynamic battery cycling loads. Before proceeding, it is essential to note when the solution times of the isothermal single particle model (SPM) were comparable to that of the enhanced self-correcting equivalent circuit model (ESC). These solution times were 2.256s and 3.39s for the SPM and ESC, respectively, when the simulations were performed using the Python programming language on the PC system. For the mentioned comparison, the open-circuit potential (OCV) of the battery cell was estimated from the electrode's open-circuit potential (OCP) and later fitted via polynomial approximation (Figure 44). Furthermore, the readers are referred to section S9 for further details on the ESC circuit, model equations, and model parameters ²³. Under the static cycling load, the SPM and ESPM models agree, although they deviate during dynamic HPPC cycling (Figure 35A–Figure 35C). A higher-order polynomial approximation on the volume-averaged lithium concentration quantity was used as the electrode concentration solver, while the finite-volume method was used for the electrolyte solver. Again, the HPPC cycling load required a higher computation time, possibly because of the set time for rest and discharge periods (Figure 35D). When the thermal model is added, the solution time increases by about 10 % (Figure 35E). Furthermore, a ten-fold reduction was achieved when C++ was used instead of the Python programming language (Figure 35F).

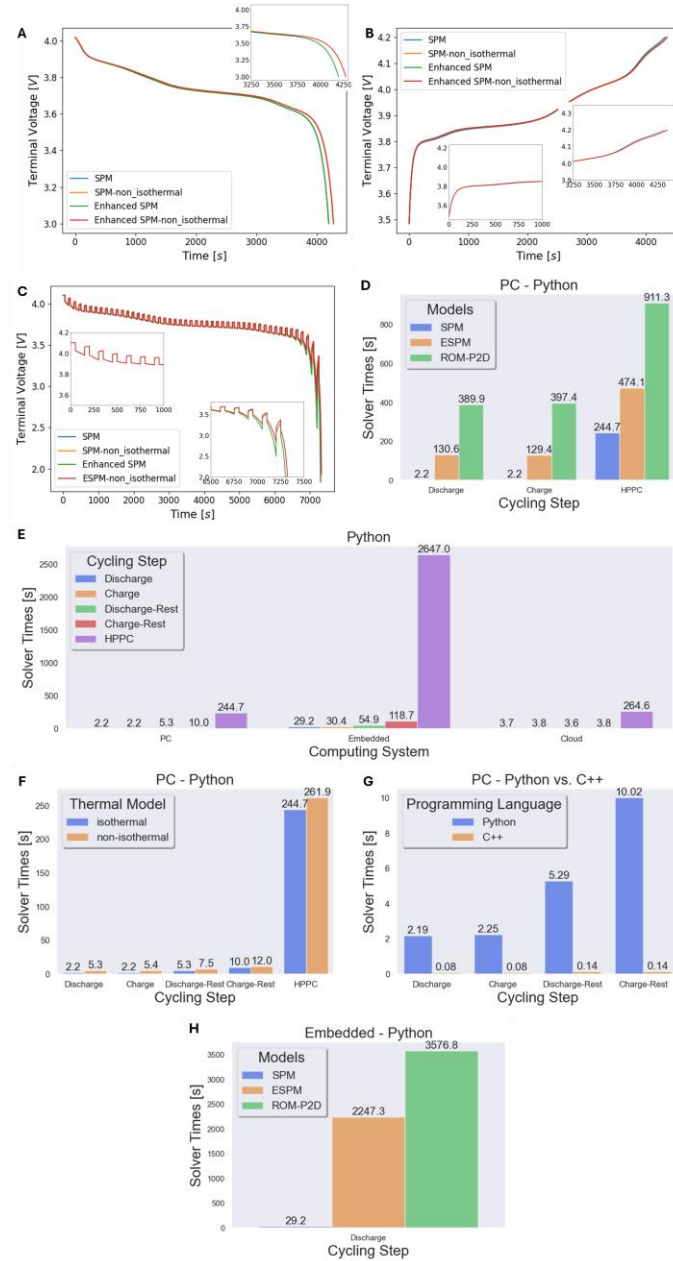


Figure 35: Time series of the battery cell terminal voltage as calculated from various battery models under (A) constant current discharge, (B) constant current charge, and (C) dynamic HPPC cycling. (D) Solution times of various charging steps as conducted using different battery models in Python in PC. (E) The solution times of the SPM under various cycling steps under a PC system. (F) The solution times of isothermal and non-isothermal SPM. (G) Comparison of the computation time between Python and C++ programming languages. (H) The solutions times of the isothermal discharge step for different battery models using Python in embedded system.

Sigma point Kalman filter (SPKF), specifically the centered difference Kalman Filter (CDKF), was applied to the single particle model under the under discharge. Random noise was added to the input discharge current and output voltage using the normal distribution (specifically, the mean of 0.0 and standard deviation of 0.01) (Figure 36A). This serves as part of the system noise and will add noise to the simulated battery cell terminal voltage. Further random noise from the normal distribution was added to the terminal voltage and this added noise acts as the sensor noise. SPKF was applied in the single particle model and its output is displayed and compared with the noisy cell terminal voltage in Figure 36B. Polynomial approximation on the averaged concentration quantities was used as the numerical scheme for the electrode SOC determination. The resultant SOC estimations are also depicted below. It should be noted that for SPKF simulations, the time difference between each iteration was set to 0.2 s. On Python programming language, it was observed that the inclusion of SPKF roughly increased the solution time twice-fold (Figure 36E). Interestingly, the simulation time for the single particle model with SPKF on embedded system with Python programming language was much lower than the actual time. It can be estimated that the current hardware is sufficient for conducting reliable battery cell real-time states estimation. However, battery pack states estimation might require multiple battery cell simulations.

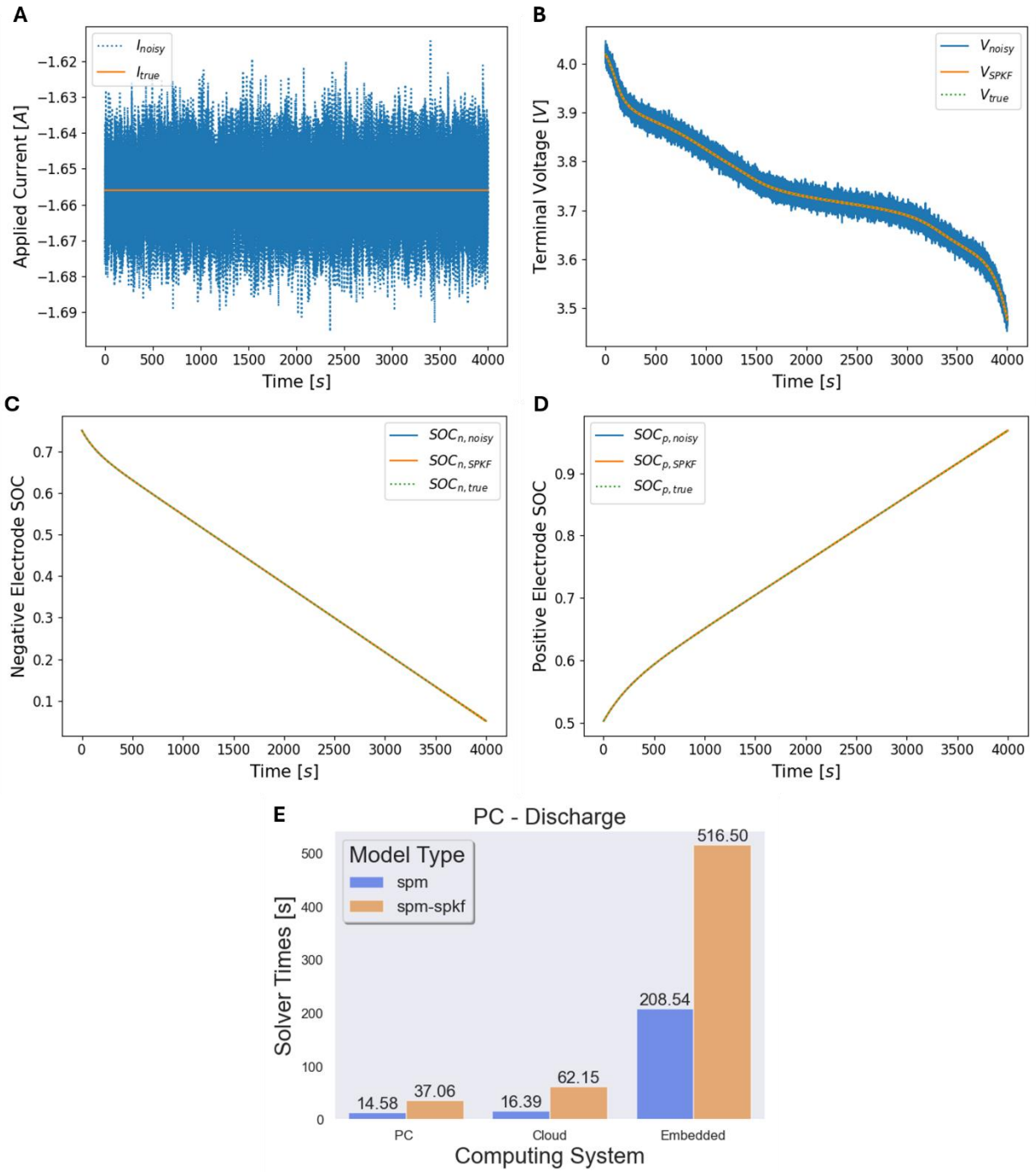


Figure 36: (A) True (or intended) vs. the actual noisy applied current. (B) The actual and SPKF estimated outputs. The estimated SOC for the (C) negative and (D) positive electrodes. (E) The computational times of the SPM vs. SPM with SPKF across multiple computing systems.

5.6 Conclusions

The demand for lithium-ion batteries (LIB) and LIB-based battery packs has steadily increased over recent years and is projected to grow in the foreseeable future. In practical applications, LIB battery packs are connected to the battery management system (BMS) that is responsible for (1) the user and LIB battery pack safety, (2) battery cell balancing, and (3) LIB pack state and health estimations. An accurate battery state estimation is challenging due to complex and non-linear phenomena that occur during its charge/discharge operations. Phenomenological cell-level models, such as the equivalent circuit model (ECM), and continuum-scale physics-based models (PBM) have been proposed in the literature for battery cell and pack state estimations. Compared to ECMs, PBM has been shown to have higher accuracy and model fidelity; however, they suffer from high computation times; however, they suffer from high computation times. The application of PBMs is critical for the next generation of BMS systems and hence a critical analysis of the computational times and model accuracy in embedded and cloud systems is required and lacking in literature.

This work compares the computational performance of PBM models, namely the single particle model (SPM) and enhanced single particle model (ESPM) under various battery cycles on embedded and cloud systems using Python and C++. In particular, representative hardware with relevant hardware specifications for embedded personal computers and cloud computing systems was used. A codebase in Python and C++ programming languages was utilized with similar object-oriented programming architecture. The PBM model parameters, particularly for the lithium cobalt oxide (LCO), were obtained from the literature. The cycling steps considered were the discharge, charge, discharge-rest, charge-rest, and hybrid pulse power characterization (HPPC).

As per the PBM model requirement, the lithium-ion concentrations in the solid positive and negative electrode particles were calculated and compared using the implicit Crank-Nicolson, eigen expansion, and polynomial approximation on the volume average concentration. Similarly, the solvers for the lithium-ion concentration in the liquid electrolyte region were solved using the finite volume method and polynomial

approximation on the volume average concentrations. The Thomas algorithm accelerated matrix operations by 46% and 21% for Crank-Nicolson and finite volume methods, respectively, and reduced memory use. Eigenexpansion and polynomial approximations displayed reasonable accuracy when the simulated electrode state-of-charge (SOC) was compared with the Crank-Nicolson scheme. Moreover, the Eigen expansion method and polynomial approximation demonstrated 5- and 735 times faster solution times for most cycling steps. Eigen expansion method and polynomial approximation proved to be, on average, 158.07 and 18.23 times faster in C++, respectively, than in Python. A similar trend across the programming languages was observed when the battery models (namely the SPM, ESPM, ROM-P2D) were analyzed. Moreover, it was interesting to note that a type of ECM, i.e., an enhanced self-correcting model, displayed a longer solution time than the single particle model. While the embedded system took longer to solve for the full electrochemical battery model due to its lower processing and memory specifications, it could still perform the simulations faster than the real-time, making the PBM suitable for real-time embedded applications. Furthermore, the inclusion of thermal models added about 20% to the solution times. When the non-linear sigma-point Kalman filter was applied to the single particle model, the simulation time increased roughly by 2-fold across all hardware. However, even on embedded systems, simulation times remained well within real-time constraints, supporting the feasibility of real-time state estimations of battery cells using continuum-scale simplified EMs, such as SPM, with reduced-order solvers.

5.7 Supporting Information

5.7.1 Section S1: Electrode Open-Circuit Potentials

The relationship between the positive and the negative electrode open-circuit potential with their respective state-of-charge is given as follows.

$$\begin{aligned}
U_p = & 0.04596 + \exp(-42.30027 * soc_p + 16.56714) - 0.04880 \\
& * \arctan(50.01833 * soc_p - 26.48897) - 0.05447 \\
& * \arctan(18.99678 * soc - 12.32362) \\
& - \exp(78.24095 * soc_p - 78.68074)
\end{aligned}
\tag{Equation 133}$$

$$\begin{aligned}
U_n = & 0.13966 + 0.68920 * \exp(-49.20361 * soc_n) + 0.41903 \\
& * \exp(-254.40067 * soc) \\
& - \exp(49.97886 * soc_n - 43.3788) - 0.028221 \\
& * \arctan(22.52300 * soc_n - 3.65328) - 0.01308 \\
& * \arctan(28.34801 * soc_n - 13.43960)
\end{aligned}
\tag{Equation 134}$$

The above relationships are plotted below along with their limits based on the lower and upper bounds of the state-of-charge.

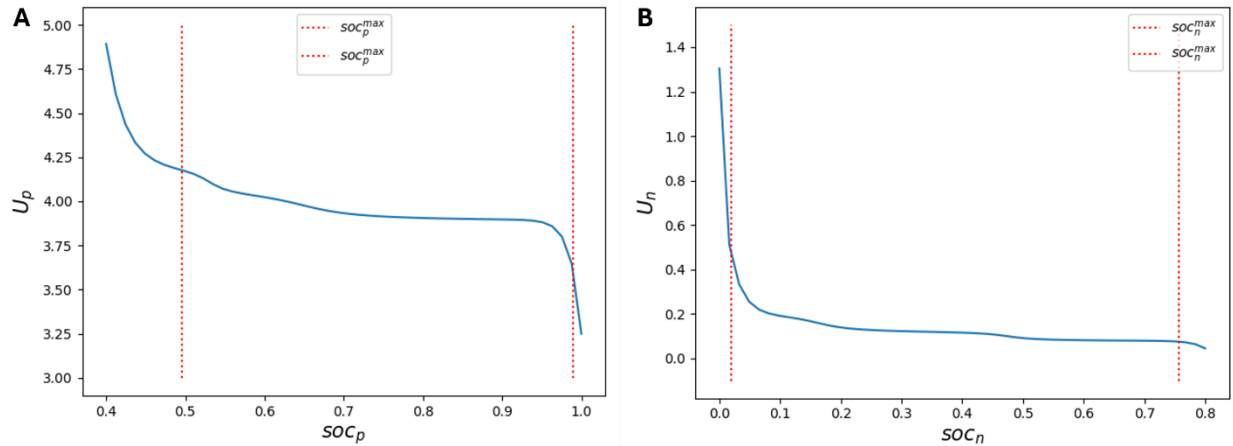


Figure 37: The (A) positive and (B) negative electrode's open-circuit potential.

5.7.2 Section S2: Crank Nickelson Scheme for Lithium Diffusivity in the Solid Electrode Region – Inverse Matrix Operation vs. Thomas Algorithm

The number of radial grid points was set to 100 when the partial differential equation (PDE) about the lithium diffusivity in the solid electrode was solved using the Crank-Nicolson Scheme. Given the battery

cell parameters in this work, the matrices in the RHS of the Equation 6 for the negative and positive electrodes are displayed below. Note that the matrices below are diagonally dominant. Further note that the elements of the mentioned matrix remain constant (except for the non-isothermal conditions), hence it can be assumed that the mentioned matrix will be diagonally dominant throughout the course of the battery cell simulations in this work.

- Negative Electrode

$$\begin{bmatrix} 1.74880000 & -0.748800000 & 0 & \dots & 0 \\ & \ddots & & & \\ & -0.123539 & 1.2496 & -0.126061 & \\ & & \ddots & & \\ 0 & & \dots & -0.2496 & 1.2496 \end{bmatrix} \quad \text{Equation 135}$$

- Positive Electrode

$$\begin{bmatrix} 1.41522491 & -0.415224913 & 0 & \dots & 0 \\ & \ddots & & & \\ & -0.000692041522 & 1.13840830 & -0.137716263 & \\ & & \ddots & & \\ 0 & & \dots & -0.138408304 & 1.13840830 \end{bmatrix} \quad \text{Equation 136}$$

Furthermore, the solution times of using the regular matrix operation or Thomas algorithm on the mentioned PDE are tabulated below.

Table 12: Solution times of the different solution methods performed in Python programming language in the PC system.

Solution Type	Electrode Type	Cycling Step	Memory Usage [MB]	Solution Time [s]
Inverse	Negative	Discharge	38.0	164.772 ± 0.551
TDMA	Negative	Discharge	3.2	96.746 ± 0.078

5.7.3 Section S3: Optimization of the Eigen Expansion Solver for the Lithium Diffusivity in the Solid Electrode Region

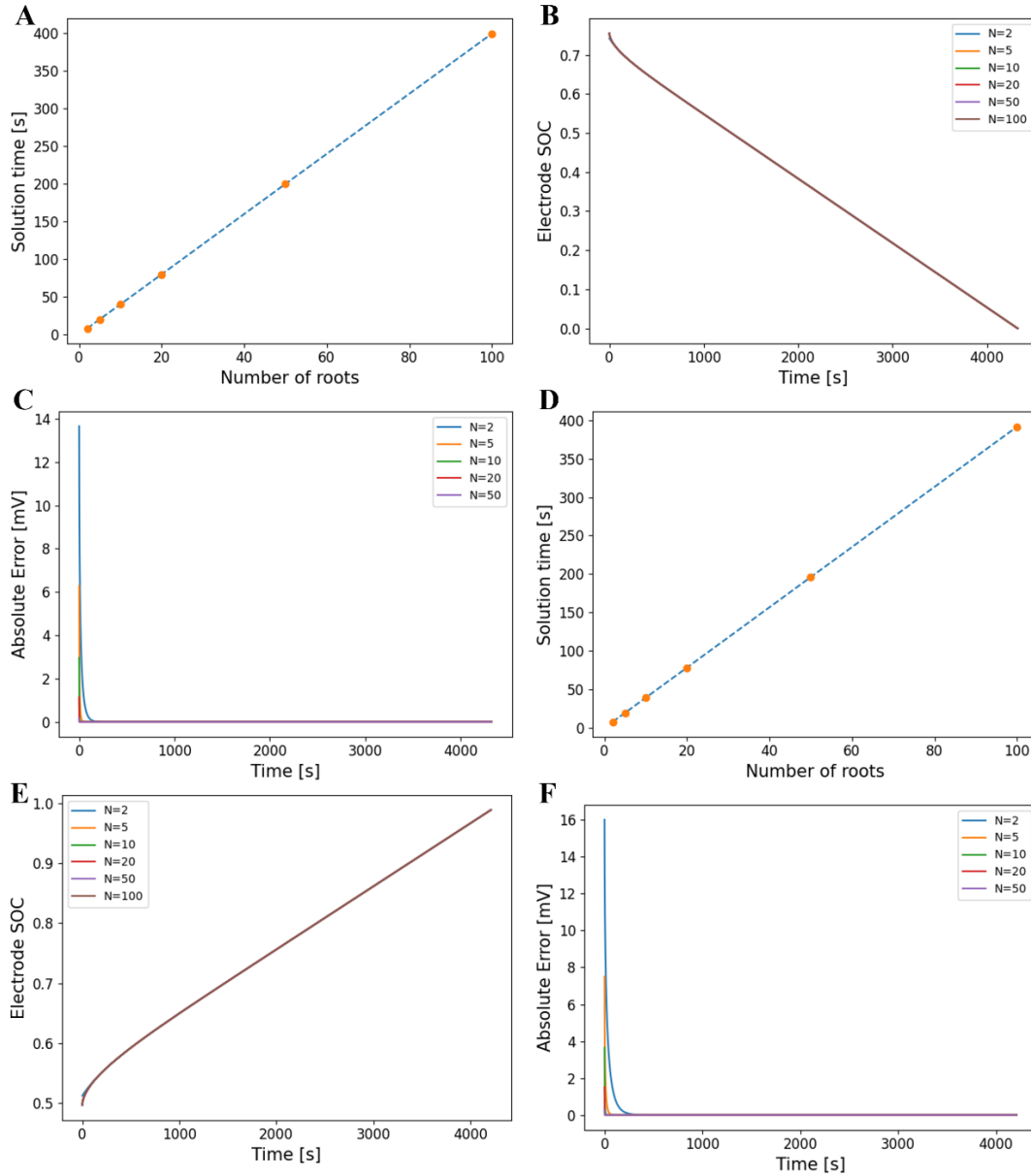


Figure 38: The (A) solution times, (B) plots, and the (C) absolute errors from the use of various number of roots for the calculation of the negative electrode's state-of-charge (SOC) during the battery cell discharge. Similarly, the (A) solution times, (B) plots, and the (C) absolute errors of the positive electrode's state-of-charge (SOC) during the battery cell. These simulation times are from the simulations performed on the personal computer in the Python programming language.

5.7.4 Section S4: Polynomial Approximation for the Lithium Diffusivity in the Solid Electrode Region

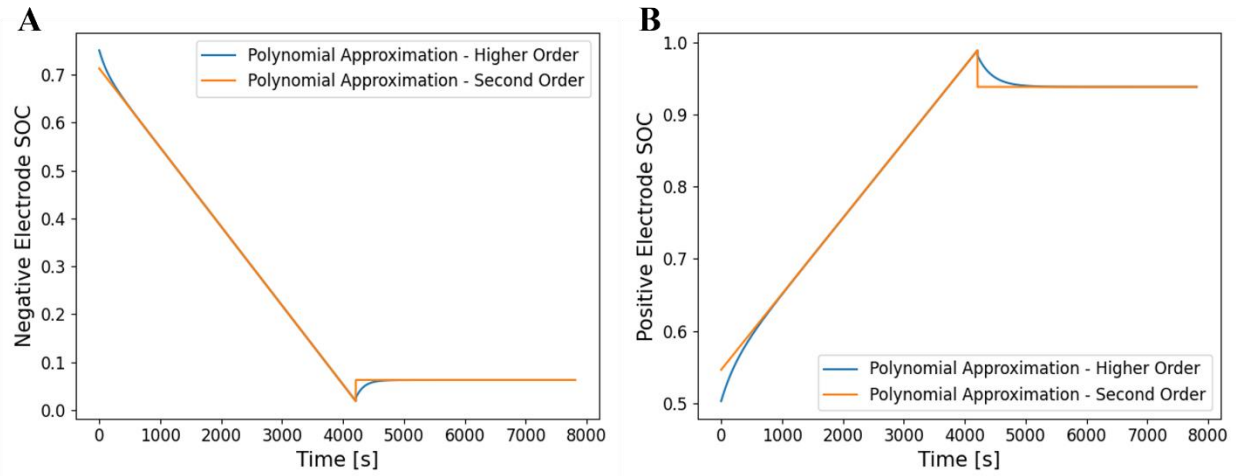


Figure 39: The polynomial approximation of various orders on the volume averaged equation of the partial differential equation pertaining to the lithium diffusivity in the solid electrode region.

Section S5: Electrode Lithium concentration solvers under static loads

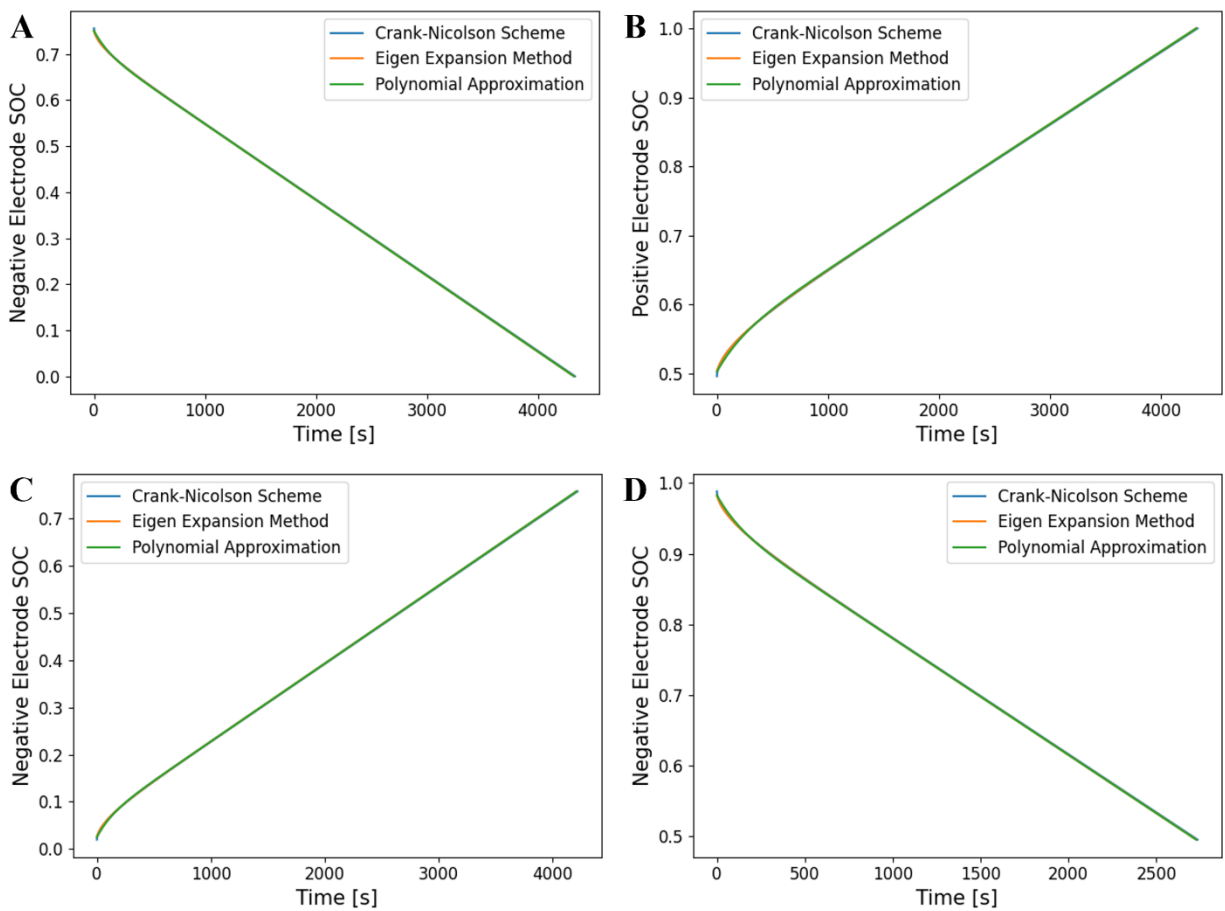


Figure 40: The solution of various solvers for the PDE pertaining to the lithium diffusivity in the solid electrode region for (A) negative electrode and (B) positive electrode under discharge, (C) negative electrode and (D) positive electrode under charge cycling steps.

5.7.5 Section S6: Absolute error of the electrode solvers compared to the Crank-Nicolson Method

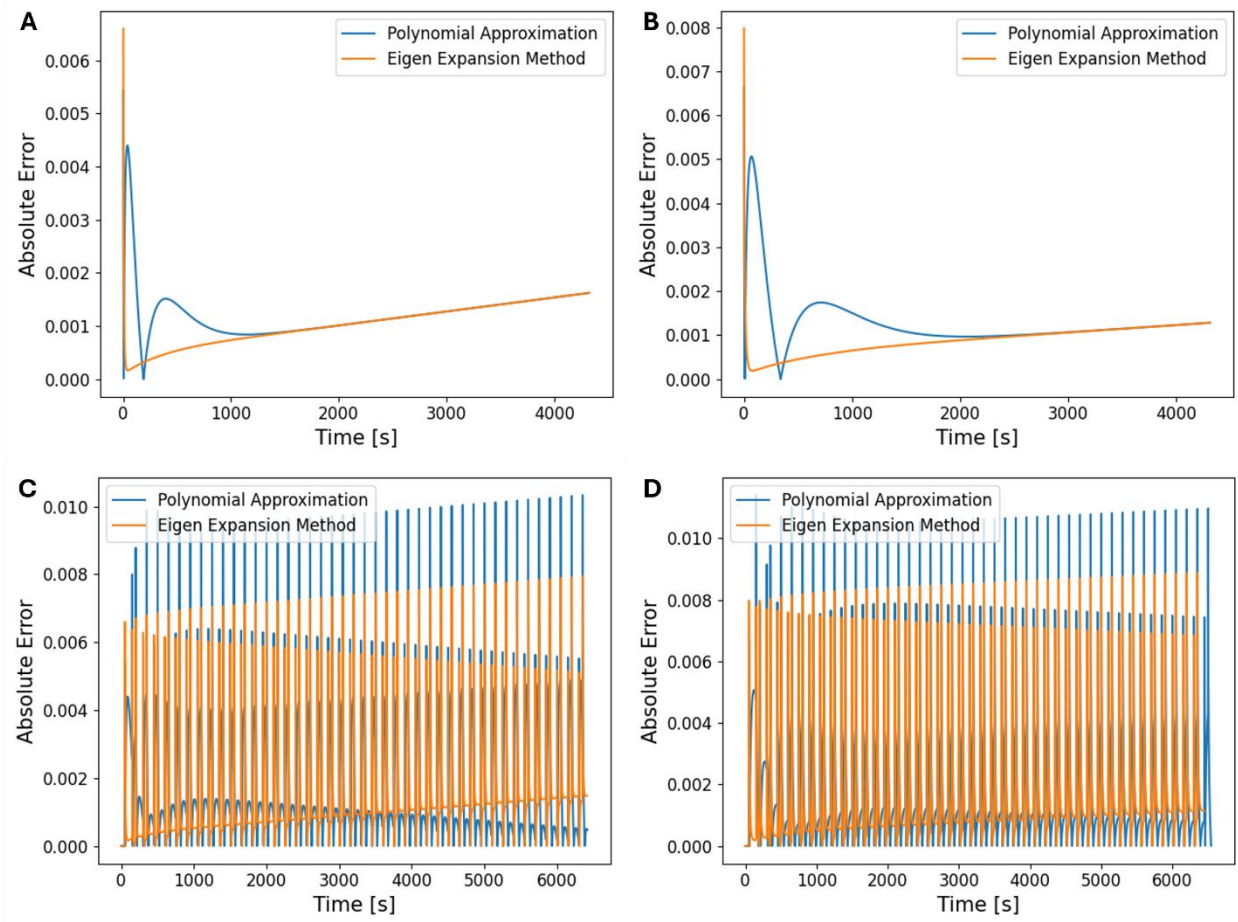


Figure 41: Absolute error between the Crank-Nicolson method and the eigen-expansion and polynomial expansion methods for the (A) negative electrode and the (B) positive electrode under the discharge cycling step. Furthermore, the absolute error for these methods for the (C) negative and the (D) positive electrodes under the HPPC cycling step.

5.7.6 Section S7: Finite Volume Method for Lithium-ion Diffusivity in the Electrolyte Region – Inverse Matrix Operation vs. Thomas Algorithm

The matrix for the electrolyte, when the nodes are set to 10 for each of the negative electrode, electrolyte, and the positive electrode regions and the $dt = 0.1$, the following matrix on the RHS of Equation 137 is obtained.

$$\begin{bmatrix} 0.3970152 & -0.0120152 & 0 & \dots & 0 \\ & \ddots & & & \\ & -0.0120152 & 0.4090304 & -0.0120152 & \\ & & \ddots & & \\ 0 & & \dots & -0.02500746 & 0.51000746 \end{bmatrix} \quad \text{Equation 137}$$

Note that the elements of the matrix above change with time and across the nodal points within a time points and the above numbers are just arbitrary values displayed to show the diagonally dominant property of the above matrix. In the Python programming language under the PC system, there was 20.5 % (126.5s vs. 100.5s) in the solution times when the Thomas algorithm was used instead of regular matrix inversion. The figure below shows the results from the inverse and the Thomas algorithm after 4000 seconds of discharge.

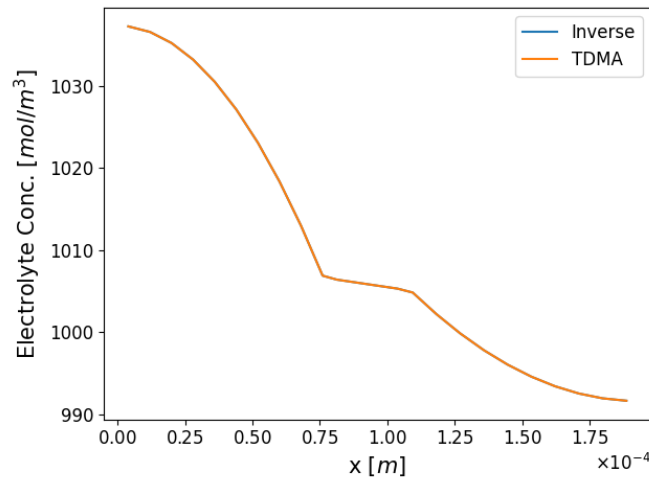


Figure 42: The results from the inverse and the Thomas algorithm for the FVM solver on the electrolyte concentration PDE after 4000s of constant discharge.

5.7.7 Section S8: Battery models under static battery cycling

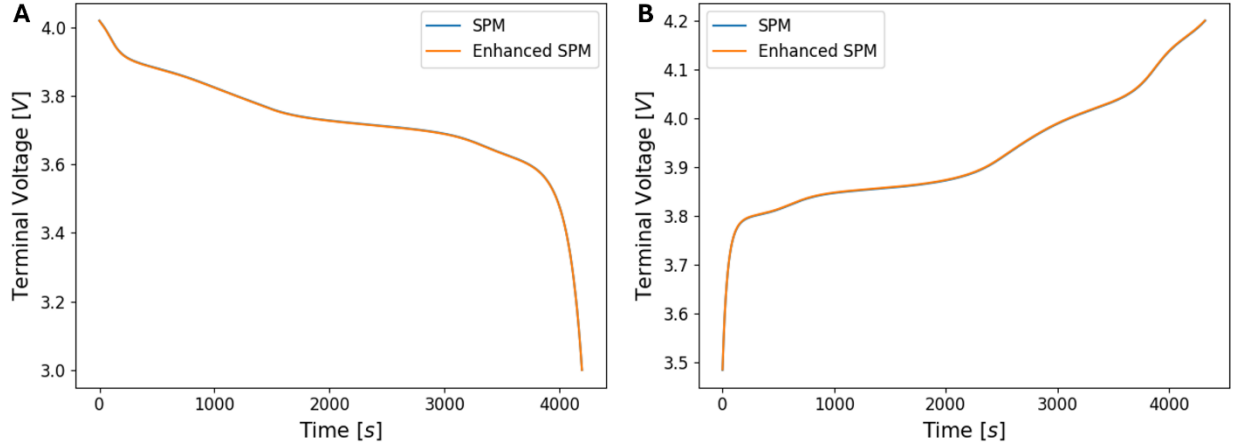


Figure 43: Battery models under constant current (A) discharge and (B) charge static battery cycle.

5.7.8 Section S9: Comparison between the isothermal single particle model and enhanced self-correcting equivalent circuit model

In this work, the enhanced self-correcting (ESC) equivalent circuit model (ECM) consists of a resistor (R_o), a resistor-capacitor pair (R_1 and C_1 electrically connected in parallel), a hysteresis element (hyst), and an open-circuit voltage (OCV) source (Figure 44A). The equations that lead to the battery cell terminal voltage for the k^{th} time step are presented below.

$$I_{R1}[k+1] = \exp\left(-\frac{\Delta t}{R_1 C_1}\right) I_{R1}[k] + \left(1 - \exp\left(-\frac{\Delta t}{R_1 C_1}\right)\right) I_{app}[k] \quad \text{Equation 138}$$

$$z[k+1] = z[k] - \frac{\eta[k] \Delta t}{Q} I_{app}[k] \quad \text{Equation 139}$$

$$h[k+1] = \exp\left(-\left|\frac{\eta[k] I_{app}[k] \gamma \Delta t}{Q}\right| h[k]\right) + \left(\exp\left(-\left|\frac{\eta[k] I_{app}[k] \gamma \Delta t}{Q}\right| h[k]\right) + 1\right) \text{sgn}(i[k]) \quad \text{Equation 140}$$

$$V_{cell}[k] = OCV(z[k]) + M_0s[k] + Mh[k] - R_1I_{R1} - R_0i[k] \quad \text{Equation 141}$$

The relationship between the battery cell OCV and the state-of-charge (SOC) was estimated using the electrode open-circuit potential (OCP) and was later fitted with a polynomial approximation of 11 degrees (Figure 44B). The ESC model parameters used for the battery cell parameters are tabulated below.

Table 13: Model parameters for enhanced self-correcting equivalent circuit model

Symbol	Value
R_0	$3.50 * 10^{-2} \Omega$
R_1	$1.00 * 10^{-1} \Omega$
C_1	$3 * 10^4 F$
M_o	$4.48 * 10^{-4}$
M_1	$1.20 * 10^{-2}$
γ	$5.24 * 10^2$

A

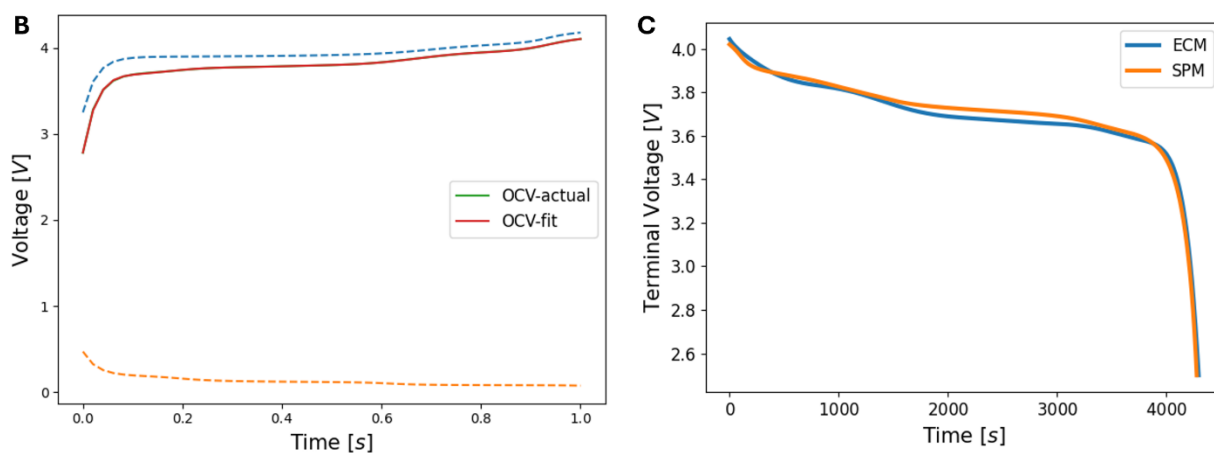
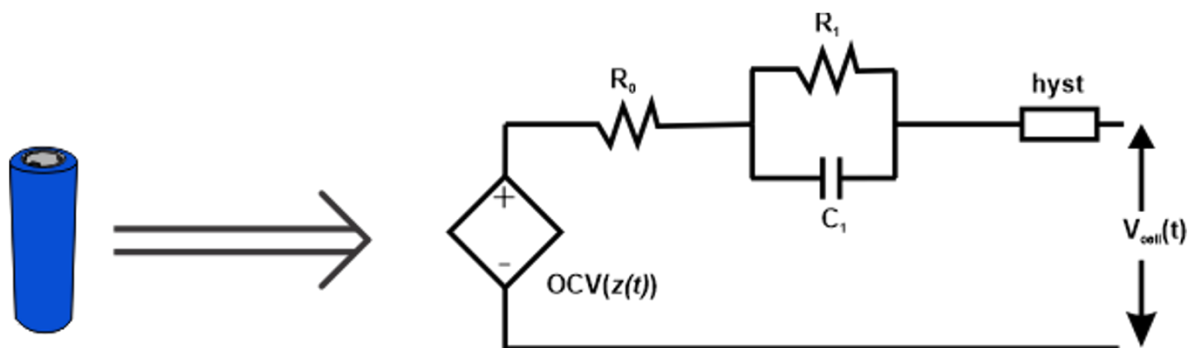


Figure 44: (A) The electrical circuit for the ESC. (B) Actual and fitted battery cell open-circuit voltage (OCV). The dashed lines depict the electrode's open-circuit potentials (OCP). (C) The obtained time-terminal voltage comparison from the SPM and ESC.

6 Thesis Conclusions and Future Works

6.1 Conclusions

During this decade, various governmental, industrial, and academic entities have shown interest and taken steps in the research, development, and commercialization of renewable energy generation and electrified transportation. The mentioned applications require energy storage technologies, and, currently, lithium-ion batteries (LIB) are used dominantly for energy storage in electric vehicles (EV), consumer electronics, and energy storage in renewable energy generation plants. Within these applications, the individual lithium-ion battery cells are connected electrically in series and parallel combinations to provide the desired output energy and power. These interconnected battery cells or modules, along with the battery management system (BMS), form the battery pack. The BMS provides various functionalities, including (but not limited to) the prevention of over-charging or over-discharging of the battery cells, managing the battery pack's thermal system, and interfacing useful information to the end-user. However, due to the non-linear nature of various physical and electrochemical phenomena that occur during battery operations, the state and health estimations of the LIB are challenging. Continuum-scale physics-based battery model uses a coupled system of partial differential equations (PDEs), about mass- and charge-transport equations, and algebraic equations (Butler-Volmer's equation) to simulate the battery cell terminal voltage and state-of-charge (SOC). Furthermore, the partial differential equations and/or ordinary equations about the energy balance and battery cell degradation mechanisms can be coupled with the above-mentioned set of PDEs to estimate the heat evolution and battery cell health during its operations. The research works in this thesis use these models, in conjunction with artificial intelligence (AI) algorithms, for enhancing the BMS functionalities.

First, BMS functionality for estimating accurate EV range was explored using continuum-scale battery cell model and application-scale vehicle dynamics model. A detailed dataset for commercial EV models manufactured from 2008 to 2021 was generated by extracting information from the internet using web-scraping techniques. This dataset contained information about the EV model, battery, dimensions, performance specifications, and range. EPA range was considered in this study, and, in case of missing

EPA range, a line-of-best fit was utilized to estimate it from NEDC and WLTP ranges. Parameters (such as battery capacity, battery voltage, top speed, and curb weight) that seem to have the most significant effect on the range were retained. Battery capacity, top speed, and curb weight showed a strong positive correlation to the range, while acceleration showed a strong negative correlation. Furthermore, other parameters were dropped based on low correlations, outliers, and poor apparent relationship with range to further analysis and calculations. Moreover, categories within parameters (body style, EV battery cooling mechanism, and LIB type) were merged to distribute the categories and prevent over-fitting on one category evenly. Next, linear regression and SVR machine-learning techniques were employed for model training and testing. Normal linear regression models seem to have the lowest RMSE of 31.428 km. Furthermore, the multi-linear regression model equation, without a standardization step, can estimate the EV's EPA range from its battery, design, and performance specifications. A newer EV model year and higher battery capacity seem to impact its EPA range positively. From this equation, EVs with water-based cooling mechanisms have a higher EPA range than those with other cooling mechanisms. Furthermore, EVs with body-style sedans and SUVs have a higher EPA range when all other designs, battery, and performance specifications are kept constant. It should be noted that most of the data points in the dataset were obtained from the EV manual and datasheets and were not validated in the laboratory setting. Furthermore, the results from the linear regression are only applicable to EVs in their pristine conditions. Hence, the effects of EV range from EV drivetrain, motor wear and tear, and battery degradation cannot be inferred.

Afterward, EV application- and battery cell continuum-scale modeling approaches were combined to estimate the range of Tesla's Model 3 (model year 2021). This EV uses lithium-nickel-cobalt-aluminum oxide (NCA) chemistry for its positive electrode and the battery cell is in the cylindrical-2170 battery form factor. The EV application model is referred to as the vehicle dynamics model and uses the forces experienced during a drive cycle to calculate the desired output power and current from its battery pack. The drive cycles considered in this study were the highway fuel economy test cycle (HWFET) and urban dynamometer drive schedule (UDDS) since these are used for the reported range by the United States

Environmental Protection Agency (EPA). The output power and current were calculated using the vehicle's drivetrain and battery pack specifications. Assuming that the battery cells have consistent capacity and internal resistance, the current required through a battery cell was calculated and then inputted into the thermal single-particle model. The thermal and material parameters required by the thermal single-particle model were obtained either from the literature or were successfully estimated using a genetic algorithm (GA). The simulation results from the final obtained parameters were tested against the ones obtained from experiments and a good agreement between the two was observed. Finally, the battery cell was subjected to the HWFET and UDDS drive cycles until its minimum cell terminal voltage was obtained, and its range was estimated using the EPA range criteria. The estimated range has a relative error of 0.081 compared with the reported EPA range. The proposed methodology in this work can be applied by an EV's BMS for estimating the range of a battery pack for a user-specific drive cycle and external condition. However, the limitations of the results should be noted.

While the parameters for the battery cell models were fitted from experimental cycling results, the power and current demand obtained from the vehicle dynamics model during its drive cycle were not experimentally validated. Based on the vehicle dynamics model assumptions, the brake force and the energy demanded by the EV's HVAC system are constant. A better model for the brake and HVAC system for the Tesla Model 3 can be implemented for a more accurate result. Moreover, the battery cell model used in this work (i.e., SPM) is only valid for low charge and discharge rates.^{43,44} Continuum-scale physics models such as the enhanced single particle model and pseudo-two-dimensional model can give more accurate results, especially in instances during the EV drive cycles where the discharge rate exceeds 0.5C.

In the second work, an AI-based algorithm was trained and used for the anomalous battery pack sensor readings for early fault detection application. Artificial neural network (ANN)-based autoencoders have been utilized in this work for early fault detection in electric vehicles (EV) battery packs. There is a user safety risk with battery packs since an occurrence of a fault and potential thermal runaway of a battery cell can propagate across the entire battery cell and can cause the battery pack to explode and/or catch fire. In

this work, a real-world dataset from the battery management systems (BMS) of a hundred EVs battery pack was used for artificial neural network (ANN) based autoencoder model training, validation, and testing. In addition, the unbalanced nature of the dataset (more normally functioning EV compared with the faulty ones) makes the use of an autoencoder as an anomaly detector a suitable candidate. Among the features in the dataset, a time series of voltage and temperature values from each cell and temperature probe, respectively, was used. It was assumed that after model training on normal functioning EV, the autoencoder would be able to detect non-conforming inputs as an anomaly. After data preprocessing, two separate autoencoders were trained on voltages and temperature measurements, respectively. Variation coefficient calculations and the model predictions from the autoencoder demonstrated the use of a thermal sensor as a better indicator of battery pack fault. Finally, a suitable threshold was obtained by plotting the 99.975th percentile of MSE for each EV from a temperature-based autoencoder. This threshold distinguished between the normal functioning EVs and the faulty ones.

The proposed use of an autoencoder for fault detection in the battery pack is able to process the sensor values from each battery cell since it takes the inputs from all the battery cells. Hence this proposed methodology is practical for applications with large LIB battery packs. Furthermore, a trained model is able to quickly perform the MSE calculations and is easier to implement compared to a physics-based battery model. Theoretically, the model can be retrained periodically with the influx of new EV data, which can make the model predictions more robust. The inclusion of more data allows the model to learn battery cell temperature sensor reading patterns across the battery cells in a pack in a variety of user drive patterns and external conditions. The proposed model is particularly suitable for cloud-based BMS systems whereby the incoming new data can be incorporated for model training.

Since the model was trained in a particular type of EV model with NMC battery chemistry, the results cannot be inferred for other EV models and/or other commercially available lithium-ion battery-positive electrode chemistries. The machine-learning-based model parameters would have to be reoptimized and retrained in case of a new EV model and/or battery chemistry. Furthermore, the model training methodology

only considered the sensor readings (either cell terminal voltage or temperature readings) as its input without considering the time series of these sensor readings. Recurrent neural network-based autoencoders can be a good candidate for further future works for model training using time-series sensor readings.

The last research work pertains to the lithium-ion battery pack state estimations, specifically state-of-charge (SOC) estimations. Complex interactions between various non-linear electrochemical phenomena, that occur during battery cell operation, have made the state estimations of the LIB battery cell and the pack an ongoing academic and commercial challenge. Physics-based electrochemical models have been shown to have accuracy and model fidelity however they suffer from high computation times. In this work, various reduced-order electrochemical solvers for the various partial differential equations (i.e., those that represent the lithium-ion concentrations in the solid electrode and liquid electrolyte regions) and battery models (specifically the single particle model (SPM) and enhanced single particle model (ESPM) have been analyzed for their solution times under different hardware systems and programming languages. In particular, the representative hardware, with relevant hardware specifications for the embedded, personal computer, and cloud computing system was used. A codebase written in Python and C++ programming languages, with similar object-oriented programming architecture, was utilized.

The implicit Crank-Nicolson, eigen expansion, and polynomial approximation on the volume average concentration were analyzed on various cycling profiles, namely constant current charge, constant current discharge, discharge-rest, charge-rest, and hybrid pulse power characterization step (HPPC). For the implicit Crank-Nicolson Scheme, Thomas algorithm was used for matrix operations since it showed 46% faster solution times than the regular matrix operations (along with significant memory usage reduction). Eigen-expansion and polynomial approximations displayed reasonable accuracy when the simulated electrode state-of-charge (SOC) was compared with the Crank-Nicolson scheme. Moreover, the eigen expansion method and polynomial approximation demonstrated 5- and 735-times faster solution times for most cycling steps. Eigen expansion method and polynomial approximation proved to be on average 158.07 and 18.23 times faster in C++, respectively, than in Python. Finally, as expected, the hardware with the

lower specifications took the longest for the same solver on the same programming language compared to the hardware with the lower specifications, with the embedded system taking over 12 times the solution time compared to the PC solution times. However, it is important to note that the embedded system took much less time to simulate the electrode SOC than real elapsed time. Similarly, the solvers written for the lithium-ion concentration in the electrolyte region in C++ performed at least 10 times faster than the ones in the Python programming language. A similar trend across the programming languages was observed when the battery models (namely the SPM, ESPM, and reduced-order P2D) were analyzed. Moreover, it was interesting to note that a type of equivalent circuit model (ECM), i.e., an enhanced self-correcting model (ESC), displayed a longer solution time than the single particle model. While the embedded system took longer to solve for the full electrochemical battery model due to its lower processing and memory specifications, it was still able to perform the simulations faster than the real-time, and this makes the electrochemical model (EM) suitable for real-time embedded applications. However, the inclusion of sequential probabilistic models, such as Kalman filters, will most probably increase the solution times of the EM models and the authors are studying this effect for future studies. Furthermore, the inclusion of thermal models added about 20% to the solution times and the models were able to be simulated below the real-time even in the embedded systems.

6.2 Future Works

Future works are proposed that can further extend the application of physics-based models for the lithium-ion-based battery management system.

Inclusion of state of battery cell health estimations in the BMS embedded systems: The works presented in this thesis for the range and state estimations utilized continuum-scale physics-based models that can be further extended to include state of health estimations. Various battery cell degradation mechanisms, such as the growth of the solid-electrolyte interface and lithium dendrite growth, lead to the increase in battery cell internal resistance and loss in battery cell capacity over usage and storage.^{210,212,214} Furthermore, the extent of these specific degradation mechanisms during the battery depends on the temperature and usage

history.²¹⁴ Various physics models have been developed in the literature to model these degradation mechanisms based on the ambient battery cell temperature and delivered current.^{215,216}

Physics-based state estimations on the microcontroller-based embedded BMS: States estimation presented in this thesis was performed on a microprocessor. Compared to microcontrollers, microprocessors possess more capable processors and higher memory. However, they also have higher power requirements.^{217,218} In practical BMS applications, the power required by the microprocessor has been supplied by the battery pack. Hence microcontroller, which can perform physics-based state estimations, can more efficiently utilize the battery pack power. Various literature works have explored the application and relevant electric circuits of microcontrollers for BMS sensor reading and battery cell balancing but have not implemented physics-based state estimations.^{191,192,217} The author is currently exploring the physics-based state estimation using the in-house developed code (used in this thesis) and BMS embedded system (Figure 46).

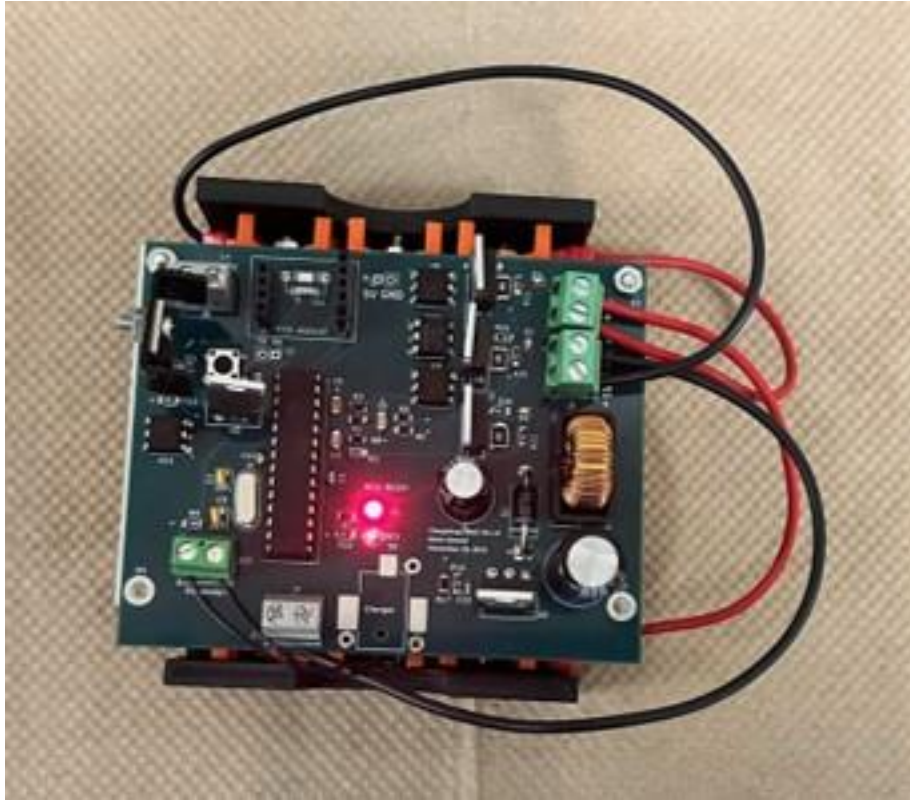


Figure 45: Inhouse BMS system, currently being designed, tested, and optimized, by the thesis author. This current version includes voltage regulation (step-down buck converter), cell balancing, cell terminal voltage readings and current readings.

References

1. United Nations Climate Change (2018). The Paris Agreement | UNFCCC. United Nations Framework Convention on Climate Change. <https://unfccc.int/process-and-meetings/the-paris-agreement/the-paris-agreement>.
2. Bilgin, B., Magne, P., Malysz, P., Yang, Y., Pantelic, V., Preindl, M., Korobkine, A., Jiang, W., Lawford, M., and Emadi, A. (2015). Making the Case for Electrified Transportation. *IEEE Transactions on Transportation Electrification* 1, 4–17. <https://doi.org/10.1109/TTE.2015.2437338>.
3. Energy Agency, I. (2020). Global EV Outlook 2020.
4. Mahmoudzadeh Andwari, A., Pesiridis, A., Rajoo, S., Martinez-Botas, R., and Esfahanian, V. (2017). A review of Battery Electric Vehicle technology and readiness levels. Preprint at Elsevier Ltd, <https://doi.org/10.1016/j.rser.2017.03.138> <https://doi.org/10.1016/j.rser.2017.03.138>.
5. Frost & Sullivan (2023). Global Lithium-ion Battery Materials Market Growth Opportunities.
6. Deng, J., Bae, C., Denlinger, A., and Miller, T. (2020). Electric Vehicles Batteries: Requirements and Challenges. *Joule* 4, 511–515. <https://doi.org/10.1016/J.JOULE.2020.01.013>.
7. Plett, G. (2015). Battery Boot Camp. In *Battery Management System* (Artech House), pp. 1–28.
8. Rahn, C. (2013). Electrochemistry. In *Battery Systems Engineering* (John Wiley & Sons Ltd), pp. 11–22. <https://doi.org/10.1002/9781118517048.CH2>.
9. Choi, J.W., and Aurbach, D. (2016). Promise and reality of post-lithium-ion batteries with high energy densities. *Nature Reviews Materials* 2016 1:4 1, 1–16. <https://doi.org/10.1038/natrevmats.2016.13>.
10. Liu, Y., Zhang, R., Wang, J., and Wang, Y. (2021). Current and future lithium-ion battery manufacturing. *iScience* 24. <https://doi.org/10.1016/J.ISCI.2021.102332>.

11. Zeng, X., Li, M., Abd El-Hady, D., Alshitari, W., Al-Bogami, A.S., Lu, J., and Amine, K. (2019). Commercialization of Lithium Battery Technologies for Electric Vehicles. *Adv Energy Mater* 9, 1900161. <https://doi.org/10.1002/AENM.201900161>.
12. Masias, A., Marcicki, J., and Paxton, W.A. (2021). Opportunities and Challenges of Lithium Ion Batteries in Automotive Applications. *ACS Energy Lett* 6, 621–630. https://doi.org/10.1021/ACSENERGYLETT.0C02584/ASSET/IMAGES/LARGE/NZ0C02584_0007.JPEG.
13. Lu, L., Han, X., Li, J., Hua, J., and Ouyang, M. (2013). A review on the key issues for lithium-ion battery management in electric vehicles. *J Power Sources* 226, 272–288. <https://doi.org/10.1016/J.JPOWSOUR.2012.10.060>.
14. Lin, Q., Wang, J., Xiong, R., Shen, W., and He, H. (2019). Towards a smarter battery management system: A critical review on optimal charging methods of lithium ion batteries. *Energy* 183, 220–234. <https://doi.org/10.1016/J.ENERGY.2019.06.128>.
15. Miao, Y., Hynan, P., Von Jouanne, A., and Yokochi, A. (2019). Current Li-Ion Battery Technologies in Electric Vehicles and Opportunities for Advancements. *Energies* 2019, Vol. 12, Page 1074 12, 1074. <https://doi.org/10.3390/EN12061074>.
16. Cheng, K.W.E., Divakar, B.P., Wu, H., Ding, K., and Ho, H.F. (2011). Battery-management system (BMS) and SOC development for electrical vehicles. *IEEE Trans Veh Technol* 60, 76–88. <https://doi.org/10.1109/TVT.2010.2089647>.
17. He, Q., Yu, B., Li, Z., and Zhao, Y. (2019). Density Functional Theory for Battery Materials. *Energy & Environmental Materials* 2, 264–279. <https://doi.org/10.1002/EEM2.12056>.

18. Gao, T., and Lu, W. (2019). Mechanism and effect of thermal degradation on electrolyte ionic diffusivity in Li-ion batteries: A molecular dynamics study. *Electrochim Acta* 323, 134791. <https://doi.org/10.1016/J.ELECTACTA.2019.134791>.
19. Plett, G.L. (2015). Microscale Cell Models. In *Battery Management Systems* (Artech House), pp. 65–135.
20. Liaw, B.Y., Nagasubramanian, G., Jungst, R.G., and Doughty, D.H. (2004). Modeling of lithium ion cells - A simple equivalent-circuit model approach. *Solid State Ion* 175, 835–839. <https://doi.org/10.1016/j.ssi.2004.09.049>.
21. Elvira, D.G., Machado, R., Plett, G.L., Trimboli, M.S., Blavi, H.V., Pastor, A.C., and Salamero, L.M. (2021). Simplified Li Ion Cell Model for BMS Coupling an Equivalent Circuit Dynamic Model with a Zero Dimensional Physics Based SEI Model. *J Electrochem Soc* 168, 110526. <https://doi.org/10.1149/1945-7111/AC3597>.
22. Wang, Q.K., He, Y.J., Shen, J.N., Ma, Z.F., and Zhong, G. Bin (2017). A unified modeling framework for lithium-ion batteries: An artificial neural network based thermal coupled equivalent circuit model approach. *Energy* 138, 118–132. <https://doi.org/10.1016/J.ENERGY.2017.07.035>.
23. Plett, G. (2015). Equivalent Circuit Models. In *Battery Management Systems Vol 1* (Artech House), pp. 30–59.
24. Ramadesigan, V., Northrop, P.W.C., De, S., Santhanagopalan, S., Braatz, R.D., and Subramanian, V.R. (2012). Modeling and Simulation of Lithium-Ion Batteries from a Systems Engineering Perspective. *J Electrochem Soc* 159, R31–R45. <https://doi.org/10.1149/2.018203jes>.
25. Tianran, Z., Daixin, L., Tao, Z., and Jun, C. (2013). Understanding electrode materials of rechargeable lithium batteries via DFT calculations. *Progress in Natural Science: Materials International* 23, 256–272. <https://doi.org/10.1016/J.PNSC.2013.04.005>.

26. Jokar, A., Rajabloo, B., Désilets, M., and Lacroix, M. (2016). Review of simplified Pseudo-two-Dimensional models of lithium-ion batteries. *J Power Sources* 327, 44–55. <https://doi.org/10.1016/J.JPOWSOUR.2016.07.036>.
27. Doyle, M., Fuller, T.F., and Newman, J. (1993). Modeling of Galvanostatic Charge and Discharge of the Lithium/Polymer/Insertion Cell. *J Electrochem Soc* 140, 1526. <https://doi.org/10.1149/1.2221597>.
28. Fuller, T.F., Doyle, M., and Newman, J. (1994). Simulation and Optimization of the Dual Lithium Ion Insertion Cell. *J Electrochem Soc* 141, 1. <https://doi.org/10.1149/1.2054684>.
29. Stetzel, K.D., Aldrich, L.L., Trimboli, M.S., and Plett, G.L. (2015). Electrochemical state and internal variables estimation using a reduced-order physics-based model of a lithium-ion cell and an extended Kalman filter. *J Power Sources* 278, 490–505. <https://doi.org/10.1016/J.JPOWSOUR.2014.11.135>.
30. Newman, J., and Balsara, N. (2021). Introduction. In *Electrochemical Systems* (Wiley), pp. 1–21.
31. Karthikeyan, D.K., Sikha, G., and White, R.E. (2008). Thermodynamic model development for lithium intercalation electrodes. *J Power Sources* 185, 1398–1407. <https://doi.org/10.1016/J.JPOWSOUR.2008.07.077>.
32. Byron Bird, R., Stewart, W.E., and Lightfoot, E.N. (2002). Diffusivity and the Mechanisms of Mass Transport. In *Transport Phenomena* (Wiley), pp. 513–542.
33. Plett, G. (2015). Continuum-scale Cell Models. In *Battery Management Systems* (Artech House), pp. 135–171.
34. Newman, J., and Thomas-Alyea, K.E. (2004). Concentrated Solutions. In *Electrochemical Systems*, pp. 297–316.

35. Doyle, M., Newman, J., Gozdz, A.S., Schmutz, C.N., and Tarascon, J. (1996). Comparison of Modeling Predictions with Experimental Data from Plastic Lithium Ion Cells. *J Electrochem Soc* *143*, 1890. <https://doi.org/10.1149/1.1836921>.
36. Jokar, A., Rajabloo, B., Désilets, M., and Lacroix, M. (2016). Review of simplified Pseudo-two-Dimensional models of lithium-ion batteries. Preprint at Elsevier B.V., <https://doi.org/10.1016/j.jpowsour.2016.07.036> <https://doi.org/10.1016/j.jpowsour.2016.07.036>.
37. Arora, P., Doyle, M., Gozdz, A.S., White, R.E., and Newman, J. (2000). Comparison between computer simulations and experimental data for high-rate discharges of plastic lithium-ion batteries. *J Power Sources* *88*, 219–231. [https://doi.org/10.1016/S0378-7753\(99\)00527-3](https://doi.org/10.1016/S0378-7753(99)00527-3).
38. Doyle, M., and Fuentes, Y. (2003). Computer Simulations of a Lithium-Ion Polymer Battery and Implications for Higher Capacity Next-Generation Battery Designs. *J Electrochem Soc* *150*, A706. <https://doi.org/10.1149/1.1569478>.
39. Srinivasan, V., and Newman, J. (2004). Discharge Model for the Lithium Iron-Phosphate Electrode. *J Electrochem Soc* *151*, A1517. <https://doi.org/10.1149/1.1785012>.
40. Christensen, J., Srinivasan, V., and Newman, J. (2006). Optimization of lithium titanate electrodes for high-power cells. *Journal of Electrochemical Society* *153*, A560–A565.
41. Plett, G. (2016). Battery State Estimations. In *Battery Management Systems*, pp. 69–165.
42. Zhang, D., Popov, B.N., and White, R.E. (2000). Modeling Lithium Intercalation of a Single Spinel Particle under Potentiodynamic Control. *J Electrochem Soc* *147*, 831. <https://doi.org/10.1149/1.1393279>.
43. Guo, M., Sikha, G., and White, R.E. (2011). Single-Particle Model for a Lithium-Ion Cell: Thermal Behavior. *J Electrochem Soc* *158*, A122. <https://doi.org/10.1149/1.3521314/XML>.

44. Santhanagopalan, S., Guo, Q., Ramadass, P., and White, R.E. (2006). Review of models for predicting the cycling performance of lithium ion batteries. *J Power Sources* 156, 620–628. <https://doi.org/10.1016/J.JPOWSOUR.2005.05.070>.
45. Moura, S.J., Argomedeo, F.B., Klein, R., Mirtabatabaei, A., and Krstic, M. (2017). Battery State Estimation for a Single Particle Model with Electrolyte Dynamics. *IEEE Transactions on Control Systems Technology* 25, 453–468. <https://doi.org/10.1109/TCST.2016.2571663>.
46. Ahmed, M., Zheng, Y., Amine, A., Fathiannasab, H., and Chen, Z. (2021). The role of artificial intelligence in the mass adoption of electric vehicles. *Joule* 5, 2296–2322. <https://doi.org/10.1016/J.JOULE.2021.07.012>.
47. Li, J., Adewuyi, K., Lotfi, N., Landers, R.G., and Park, J. (2018). A single particle model with chemical/mechanical degradation physics for lithium ion battery State of Health (SOH) estimation. *Appl Energy* 212, 1178–1190. <https://doi.org/10.1016/J.APENERGY.2018.01.011>.
48. Plett, G. (2015). State-Space Models and the Discrete-Time Realization Algorithms. In *Battery Management Systems* (Artech House), pp. 173–215.
49. Monje, C.A., Chen, Y., Vinagre, B.M., Xue, D., and Feliu, V. (2010). State-space Representation and Analysis. In *Fractional-order Systems and Controls* (Springer, London), pp. 35–74. https://doi.org/10.1007/978-1-84996-335-0_3.
50. Jacobsen, T., and West, K. (1995). Diffusion impedance in planar, cylindrical and spherical symmetry. *Electrochim Acta* 40, 255–262. [https://doi.org/10.1016/0013-4686\(94\)E0192-3](https://doi.org/10.1016/0013-4686(94)E0192-3).
51. Lee, J.L., Chemistruck, A., and Plett, G.L. (2012). One-dimensional physics-based reduced-order model of lithium-ion dynamics. *J Power Sources* 220, 430–448. <https://doi.org/10.1016/J.JPOWSOUR.2012.07.075>.

52. Smith, K.A., Rahn, C.D., and Wang, C.Y. (2007). Control oriented 1D electrochemical model of lithium ion battery. *Energy Convers Manag* 48, 2565–2578. <https://doi.org/10.1016/J.ENCONMAN.2007.03.015>.
53. DATTA, B.N. (2004). Realization and Subspace Determination. In *Numerical Methods for Linear Control Systems* (Academic Press), pp. 1–15. <https://doi.org/10.1016/B978-012203590-6/50005-7>.
54. Plett, G. (2015). Reduced-Order Models. In *Battery Management Systems* (Artech House), pp. 217–269.
55. Krishnan, H., Piyush, T., and Ramachandran, S. (2018). Theoretical Framework of the Reduced Order Models. In *Mathematical Modeling in Lithium-Ion Batteries* (Springer International Publishing), pp. 33–42.
56. Winston, P. (1992). Chapter 1: The Intelligent Computer. In *Artificial Intelligence* (Addison-Wesley Longman Publishing Co., Inc.).
57. Ng, M., Zhao, J., Yan, Q., Conduit, G.J., and Seh, Z.W. (2020). Predicting the state of charge and health of batteries using data-driven machine learning. *Nat Mach Intell* 2, 161–170. <https://doi.org/10.1038/s42256-020-0156-7>.
58. Grigorescu, S., Trasnea, B., Cocias, T., and Macesanu, G. (2019). A Survey of Deep Learning Techniques for Autonomous Driving. *J Field Robot* 37, 362–386. <https://doi.org/10.1002/rob.21918>.
59. Hu, X., Xu, L., Lin, X., and Pecht, M. (2020). Battery Lifetime Prognostics. *Joule* 4, 310–346. <https://doi.org/10.1016/j.joule.2019.11.018>.
60. Grosan, C., and Abraham, A. (2011). Rule-Based Expert Systems. In *Intelligent Systems* (Springer, Berlin, Heidelberg), pp. 149–185. https://doi.org/10.1007/978-3-642-21004-4_7.

61. Geron, A. (2019). The Machine Learning Landscape. In Hands-on Machine Learning with Scikit-Learn and TensorFlow (O'Reilly), pp. 7–17.
62. Chen, A. (2020). Machine learning : Accelerating materials development for energy storage and conversion. *InfoMat* 10, 1–24. <https://doi.org/10.1002/inf2.12094>.
63. Cunha, R.P., Lombardo, T., Primo, E.N., and Franco, A.A. (2020). Artificial Intelligence Investigation of NMC Cathode Manufacturing Parameters Interdependencies. *Batter Supercaps* 3, 60–67. <https://doi.org/10.1002/batt.201900135>.
64. Varga, B., Sagoian, A., and Mariasiu, F. (2019). Prediction of Electric Vehicle Range: A Comprehensive Review of Current Issues and Challenges. *Energies (Basel)* 12, 946. <https://doi.org/10.3390/en12050946>.
65. Abu Hanifah, R., Toha, S.F., Hassan, M.K., and Ahmad, S. (2016). Power reduction optimization with swarm based technique in electric power assist steering system. *Energy* 102, 444–452. <https://doi.org/10.1016/j.energy.2016.02.050>.
66. Li, J., Cheng, H., Guo, H., and Qiu, S. (2018). Survey on Artificial Intelligence for Vehicles. *Automotive Innovation* 1, 390. <https://doi.org/10.1007/s42154-018-0009-9>.
67. Rigas, E.S., Member, S., Ramchurn, S.D., and Bassiliades, N. (2015). Managing Electric Vehicles in the Smart Grid Using Artificial Intelligence : A Survey. 16, 1619–1635.
68. Hannan, M.A., Lipu, M.S.H., Hussain, A., Ker, P.J., Mahlia, T.M.I., Mansor, M., Ayob, A., Saad, M.H., and Dong, Z.Y. (2020). Toward Enhanced State of Charge Estimation of Lithium-ion Batteries Using Optimized Machine Learning Techniques. *Sci Rep* 10, 4687. <https://doi.org/10.1038/s41598-020-61464-7>.

69. Chen, C., Zuo, Y., Ye, W., Li, X., Deng, Z., and Ong, S.P. (2020). A Critical Review of Machine Learning of Energy Materials. *Adv Energy Mater* 10, 1903242. <https://doi.org/10.1002/aenm.201903242>.
70. Geron, A. (2019). Introduction to Artificial Neural Networks with Keras. In *Hands-on Machine Learning with Scikit-Learn and TensorFlow*, pp. 280–281.
71. Geron, A. (2019). Reinforcement Learning. In *Hands-on Machine Learning with Scikit-Learn and TensorFlow* (O'Reilly), pp. 609–664.
72. Slowik, A., and Kwasnicka, H. (2020). Evolutionary algorithms and their applications to engineering problems. *Neural Computing and Applications* 2020 32:16 32, 12363–12379. <https://doi.org/10.1007/S00521-020-04832-8>.
73. Hanifah, R.A., Toha, S.F., Ahmad, S., and Hassan, M.K. (2018). Swarm-Intelligence Tuned Current Reduction for Power-Assisted Steering Control in Electric Vehicles. *IEEE Transactions on Industrial Electronics* 65, 7202–7210. <https://doi.org/10.1109/TIE.2017.2784344>.
74. Soares, J., Vale, Z., Canizes, B., and Morais, H. (2013). Multi-objective parallel particle swarm optimization for day-ahead Vehicle-to-Grid scheduling. In *IEEE Symposium on Computational Intelligence Applications in Smart Grid, CIASG*, pp. 138–145. <https://doi.org/10.1109/CIASG.2013.6611510>.
75. Gopalakrishnan, R., Biswas, A., Lightwala, A., Vasudevan, S., Dutta, P., and Tripathi, A. (2016). Demand prediction and placement optimization for electric vehicle charging stations. In *IJCAI International Joint Conference on Artificial Intelligence*.
76. Jokar, A., Rajabloo, B., Désilets, M., and Lacroix, M. (2016). An Inverse Method for Estimating the Electrochemical Parameters of Lithium-Ion Batteries. *J Electrochem Soc* 163, A2876. <https://doi.org/10.1149/2.0191614JES>.

77. Zhang, L., Wang, L., Hinds, G., Lyu, C., Zheng, J., and Li, J. (2014). Multi-objective optimization of lithium-ion battery model using genetic algorithm approach. *J Power Sources* 270, 367–378. <https://doi.org/10.1016/j.jpowsour.2014.07.110>.
78. Rahman, M.A., Anwar, S., and Izadian, A. (2016). Electrochemical model parameter identification of a lithium-ion battery using particle swarm optimization method. *J Power Sources* 307, 86–97. <https://doi.org/10.1016/j.jpowsour.2015.12.083>.
79. Turetskyy, A., Laue, V., Lamprecht, R., Thiede, S., Krewer, U., and Herrmann, C. (2019). Artificial neural network enabled P2D model deployment for end-of-line battery cell characterization. *IEEE International Conference on Industrial Informatics (INDIN) 2019-July*, 53–58. <https://doi.org/10.1109/INDIN41052.2019.8972181>.
80. Colaço, M.J., Orlande, H.R.B., and Dulikravich, G.S. (2006). Inverse and optimization problems in heat transfer. *Journal of the Brazilian Society of Mechanical Sciences and Engineering* 28, 1–24. <https://doi.org/10.1590/S1678-58782006000100001>.
81. Santhanagopalan, S., Guo, Q., and White, R.E. (2007). Parameter Estimation and Model Discrimination for a Lithium-Ion Cell. *J Electrochem Soc* 154, A198. <https://doi.org/10.1149/1.2422896>.
82. Santhanagopalan, S., Zhang, Q., Kumaresan, K., and White, R.E. (2008). Parameter Estimation and Life Modeling of Lithium-Ion Cells. *J Electrochem Soc* 155, A345. <https://doi.org/10.1149/1.2839630>.
83. Kim, M., Chun, H., Kim, J., Kim, K., Yu, J., Kim, T., and Han, S. (2019). Data-efficient parameter identification of electrochemical lithium-ion battery model using deep Bayesian harmony search. *Appl Energy* 254, 113644. <https://doi.org/10.1016/J.APENERGY.2019.113644>.

84. Chun, H., Kim, J., Yu, J., and Han, S. (2020). Real-Time Parameter Estimation of an Electrochemical Lithium-Ion Battery Model Using a Long Short-Term Memory Network. *IEEE Access* 8, 81789–81799. <https://doi.org/10.1109/ACCESS.2020.2991124>.
85. Chun, H., Kim, J., and Han, S. (2019). Parameter identification of an electrochemical lithium-ion battery model with convolutional neural network. In *IFAC-PapersOnLine* (Elsevier B.V.), pp. 129–134. <https://doi.org/10.1016/j.ifacol.2019.08.167>.
86. Daziano, R.A. (2013). Conditional-logit Bayes estimators for consumer valuation of electric vehicle driving range. *Resour Energy Econ* 35, 429–450. <https://doi.org/10.1016/j.reseneeco.2013.05.001>.
87. Çeven, S., Albayrak, A., and Bayır, R. (2020). Real-time range estimation in electric vehicles using fuzzy logic classifier. *Computers and Electrical Engineering* 83, 106577. <https://doi.org/10.1016/j.compeleceng.2020.106577>.
88. Zhang, Y., Wang, W., Kobayashi, Y., and Shirai, K. (2012). Remaining driving range estimation of electric vehicle. In *2012 IEEE International Electric Vehicle Conference, IEVC 2012*, pp. 1–7. <https://doi.org/10.1109/IEVC.2012.6183172>.
89. Pan, C., Dai, W., Chen, L., Chen, L., and Wang, L. (2017). Driving range estimation for electric vehicles based on driving condition identification and forecast. *AIP Adv* 7, 105206. <https://doi.org/10.1063/1.4993945>.
90. Ahmed, M., Mao, Z., Zheng, Y., Chen, T., and Chen, Z. (2022). Electric Vehicle Range Estimation Using Regression Techniques. *World Electric Vehicle Journal* 2022, Vol. 13, Page 105 13, 105. <https://doi.org/10.3390/WEVJ13060105>.
91. Rauh, N., Franke, T., and Krems, J.F. (2015). Understanding the impact of electric vehicle driving experience on range anxiety. *Hum Factors* 57, 177–187. <https://doi.org/10.1177/0018720814546372>.

92. Good, D. (2017). EPA Test Procedures for Electric Vehicles and Plug-in Hybrids DRAFT Summary-Regulations take Precedence.
93. Mruzek, M., Gajdáč, I., Kučera, L., and Barta, D. (2016). Analysis of Parameters Influencing Electric Vehicle Range. In *Procedia Engineering* (Elsevier Ltd), pp. 165–174. <https://doi.org/10.1016/j.proeng.2016.01.056>.
94. Rastani, S., Yüksel, T., and Çatay, B. (2019). Effects of ambient temperature on the route planning of electric freight vehicles. *Transp Res D Transp Environ* 74, 124–141. <https://doi.org/10.1016/j.trd.2019.07.025>.
95. Mock, P., Kühlwein, J., Tietge, U., Franco, V., Bandivadekar, A., and German, J. (2014). The WLTP: How a new test procedure for cars will affect fuel consumption values in the EU. <https://doi.org/10.1016/j.enpol.2013.12.013i>.
96. How Does The EPA Calculate Electric Car Range? | CleanTechnica <https://cleantechnica.com/2020/08/18/how-does-epa-calculate-electric-car-range/>.
97. Hayes, J.G., De Oliveira, R.P.R., Vaughan, S., and Egan, M.G. (2011). Simplified electric vehicle power train models and range estimation. 2011 IEEE Vehicle Power and Propulsion Conference, VPPC 2011. <https://doi.org/10.1109/VPPC.2011.6043163>.
98. Seishiro Shibata, and Tsuguhiko Nakagawa (2015). Mathematical Model of Electric Vehicle Power Consumption for Traveling and Air-Conditioning. *Journal of Energy and Power Engineering* 9. <https://doi.org/10.17265/1934-8975/2015.03.006>.
99. Fiori, C., Ahn, K., and Rakha, H.A. (2016). Power-based electric vehicle energy consumption model: Model development and validation. *Appl Energy* 168, 257–268. <https://doi.org/10.1016/J.APENERGY.2016.01.097>.

100. Hariharan, K.S. (2013). A coupled nonlinear equivalent circuit – Thermal model for lithium ion cells. *J Power Sources* 227, 171–176. <https://doi.org/10.1016/J.JPOWSOUR.2012.11.044>.
101. Barcellona, S., De Simone, D., and Grillo, S. (2019). Real-time Electric Vehicle Range Estimation Based on a Lithium-Ion Battery Model. *ICCEP 2019 - 7th International Conference on Clean Electrical Power: Renewable Energy Resources Impact*, 351–357. <https://doi.org/10.1109/ICCEP.2019.8890095>.
102. Sun, S., Zhang, J., Bi, J., Wang, Y., and Moghaddam, M.H.Y. (2019). A Machine Learning Method for Predicting Driving Range of Battery Electric Vehicles. *J Adv Transp* 2019. <https://doi.org/10.1155/2019/4109148>.
103. Yavasoglu, H.A., Tetik, Y.E., and Gokce, K. (2019). Implementation of machine learning based real time range estimation method without destination knowledge for BEVs. *Energy* 172, 1179–1186. <https://doi.org/10.1016/j.energy.2019.02.032>.
104. Zheng, B., He, P., Zhao, L., and Li, H. (2016). A hybrid machine learning model for range estimation of electric vehicles. In *2016 IEEE Global Communications Conference, GLOBECOM 2016 - Proceedings (Institute of Electrical and Electronics Engineers Inc.)*. <https://doi.org/10.1109/GLOCOM.2016.7841506>.
105. Bolovinou, A., Bakas, I., Amditis, A., Mastrandrea, F., and Vinciotti, W. (2014). Online prediction of an electric vehicle remaining range based on regression analysis. In *2014 IEEE International Electric Vehicle Conference, IEVC 2014* <https://doi.org/10.1109/IEVC.2014.7056167>.
106. Scheubner, S., Thorgeirsson, A.T., Vaillant, M., and Gauterin, F. (2019). A Stochastic Range Estimation Algorithm for Electric Vehicles Using Traffic Phase Classification. *IEEE Trans Veh Technol* 68, 6414–6428. <https://doi.org/10.1109/TVT.2019.2918544>.

107. Rhode, S., Van Vaerenbergh, S., and Pfriem, M. (2020). Power prediction for electric vehicles using online machine learning. *Eng Appl Artif Intell* 87, 103278. <https://doi.org/10.1016/j.engappai.2019.103278>.
108. Yokoi, Y., Ichikawa, S., Doki, S., Okuma, S., Naitou, T., Shiimado, T., and Miki, N. (2004). Driving pattern prediction for an energy management system of hybrid electric vehicles in a specific driving course. In *IECON Proceedings (Industrial Electronics Conference)*, pp. 1727–1732. <https://doi.org/10.1109/IECON.2004.1431842>.
109. Lee, C.H., and Wu, C.H. (2015). A Novel Big Data Modeling Method for Improving Driving Range Estimation of EVs. *IEEE Access* 3. <https://doi.org/10.1109/ACCESS.2015.2492923>.
110. Yao, L., Xiao, Y., Gong, X., Hou, J., and Chen, X. (2020). A novel intelligent method for fault diagnosis of electric vehicle battery system based on wavelet neural network. *J Power Sources* 453, 227870. <https://doi.org/10.1016/j.jpowsour.2020.227870>.
111. Ojo, O., Lang, H., Kim, Y., Hu, X., Mu, B., and Lin, X. (2021). A Neural Network Based Method for Thermal Fault Detection in Lithium-Ion Batteries. *IEEE Transactions on Industrial Electronics* 68, 4068–4078. <https://doi.org/10.1109/TIE.2020.2984980>.
112. Bishop, W.S. (1984). Battery control and fault detection method.
113. Keates, A.W., Otani, N., Nguyen, D.J., Matsumura, N., and Li, P.T. (2010). Short circuit detection for batteries.
114. Sivertsen, D. (2009). Detecting faults in a wiring harness.
115. Judge, A., Renahan, J., and Deljevic, Z. (1999). Fault detection in a motor vehicle charging system.
116. Li, W., Zhu, J., Xia, Y., Gorji, M.B., and Wierzbicki, T. (2019). Data-Driven Safety Envelope of Lithium-Ion Batteries for Electric Vehicles. *Joule* 3, 2703–2715. <https://doi.org/10.1016/j.joule.2019.07.026>.

117. Finegan, D.P., and Cooper, S.J. (2019). Battery Safety: Data-Driven Prediction of Failure. Preprint at Cell Press, <https://doi.org/10.1016/j.joule.2019.10.013>
<https://doi.org/10.1016/j.joule.2019.10.013>.
118. Finegan, D.P., Zhu, J., Feng, X., Keyser, M., Ulmefors, M., Li, W., Bazant, M.Z., and Cooper, S.J. (2021). The Application of Data-Driven Methods and Physics-Based Learning for Improving Battery Safety. Preprint at Cell Press, <https://doi.org/10.1016/j.joule.2020.11.018>
<https://doi.org/10.1016/j.joule.2020.11.018>.
119. Ojo, O., Lang, H., Kim, Y., Hu, X., Mu, B., and Lin, X. (2020). A Neural Network-Based Method for Thermal Fault Detection in Lithium-Ion Batteries. *IEEE Transactions on Industrial Electronics*, 1–1. <https://doi.org/10.1109/tie.2020.2984980>.
120. Li, D., Zhang, Z., Liu, P., Wang, Z., and Zhang, L. (2021). Battery Fault Diagnosis for Electric Vehicles Based on Voltage Abnormality by Combining the Long Short-Term Memory Neural Network and the Equivalent Circuit Model. *IEEE Trans Power Electron* 36, 1303–1315. <https://doi.org/10.1109/TPEL.2020.3008194>.
121. Hong, J., Wang, Z., and Yao, Y. (2019). Fault prognosis of battery system based on accurate voltage abnormality prognosis using long short-term memory neural networks. *Appl Energy* 251, 113381. <https://doi.org/10.1016/J.APENERGY.2019.113381>.
122. Yu, Y., Si, X., Hu, C., and Zhang, J. (2019). A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures. *Neural Comput* 31, 1235–1270. https://doi.org/10.1162/NECO_A_01199.
123. Geron, A. (2019). Processing Sequences Using RNNs and CNNs. In *Hands-on Machine Learning with Scikit-Learn and TensorFlow2* (O'Reilly), pp. 497–515.

124. Rietmann, N., Hügler, B., and Lieven, T. (2020). Forecasting the trajectory of electric vehicle sales and the consequences for worldwide CO₂ emissions. *J Clean Prod* 261, 121038. <https://doi.org/10.1016/j.jclepro.2020.121038>.
125. Ellingsen, L.A.W., Singh, B., and Strømman, A.H. (2016). The size and range effect: Lifecycle greenhouse gas emissions of electric vehicles. *Environmental Research Letters* 11, 054010. <https://doi.org/10.1088/1748-9326/11/5/054010>.
126. Ma, H., Balthasar, F., Tait, N., Riera-Palou, X., and Harrison, A. (2012). A new comparison between the life cycle greenhouse gas emissions of battery electric vehicles and internal combustion vehicles. *Energy Policy* 44, 160–173. <https://doi.org/10.1016/j.enpol.2012.01.034>.
127. Perkins, J.H. (2017). *Changing energy: the transition to a sustainable future* (University of California Press).
128. Wu, Z., Wang, M., Zheng, J., Sun, X., Zhao, M., and Wang, X. (2018). Life cycle greenhouse gas emission reduction potential of battery electric vehicle. *J Clean Prod* 190, 462–470. <https://doi.org/10.1016/J.JCLEPRO.2018.04.036>.
129. Adnan, N., Nordin, S.M., Rahman, I., Vasant, P.M., and Noor, A. (2017). A comprehensive review on theoretical framework-based electric vehicle consumer adoption research. Preprint at John Wiley and Sons Ltd, <https://doi.org/10.1002/er.3640> <https://doi.org/10.1002/er.3640>.
130. Bolovinou, A., Bakas, I., Amditis, A., Mastrandrea, F., and Vinciotti, W. (2014). Online prediction of an electric vehicle remaining range based on regression analysis. 2014 IEEE International Electric Vehicle Conference, IEVC 2014. <https://doi.org/10.1109/IEVC.2014.7056167>.
131. Minett, C.F., Salomons, A.M., Daamen, W., Van Arem, B., and Kuijpers, S. (2011). Eco-routing: Comparing the fuel consumption of different routes between an origin and destination using field

- test speed profiles and synthetic speed profiles. 2011 IEEE Forum on Integrated and Sustainable Transportation Systems, FISTS 2011, 32–39. <https://doi.org/10.1109/FISTS.2011.5973621>.
132. Abousleiman, R., and Rawashdeh, O. (2015). Energy consumption model of an electric vehicle. 2015 IEEE Transportation Electrification Conference and Expo, ITEC 2015. <https://doi.org/10.1109/ITEC.2015.7165773>.
 133. Miri, I., Fotouhi, A., and Ewin, N. (2021). Electric vehicle energy consumption modelling and estimation—A case study. *Int J Energy Res* 45, 501–520. <https://doi.org/10.1002/ER.5700>.
 134. Used 2018 Volkswagen e-Golf Specs & Features | Edmunds <https://www.edmunds.com/volkswagen/e-golf/2018/features-specs/>.
 135. Rodgers, J.L., and Nicewander, W.A. (1988). Thirteen Ways to Look at the Correlation Coefficient. *Am Stat* 42, 59. <https://doi.org/10.2307/2685263>.
 136. Geron, A. (2019). Regularized Linear Models. In *Hands-on Machine Learning with Scikit-Learn and TensorFlow* (O Reilly), pp. 134–141.
 137. Geron, A. (2019). Support Vector Machines. In *Hands-on Machine Learning with Scikit-Learn and TensorFlow*, pp. 153–174.
 138. Gao, C., Bompard, E., Napoli, R., and Cheng, H. (2007). Price forecast in the competitive electricity market by support vector machine. *Physica A: Statistical Mechanics and its Applications* 382, 98–113. <https://doi.org/10.1016/j.physa.2007.03.050>.
 139. Gu, J., Zhu, M., and Jiang, L. (2011). Housing price forecasting based on genetic algorithm and support vector machine. *Expert Syst Appl* 38, 3383–3386. <https://doi.org/10.1016/j.eswa.2010.08.123>.
 140. Vapnik, V. (1995). SVM for Estimating Regression Function. In *The Nature of Statistical Learning Theory* (Springer), pp. 183–188.

141. scikit-learn: machine learning in Python — scikit-learn 0.24.2 documentation <https://scikit-learn.org/stable/>.
142. Ma, Y., Zhang, Z., Ihler, A., and Pan, B. (2018). Estimating warehouse rental price using machine learning techniques. *International Journal of Computers, Communications and Control* 13, 235–250. <https://doi.org/10.15837/ijccc.2018.2.3034>.
143. Plett, G. (2015). Simulating Battery Packs. In *Battery Management Systems*, vol 2 (Artech House), pp. 31–67.
144. Gillespie, T.D. (Thomas D.), Taheri, S., Sandu, C., and Duprey, B.L. *Fundamentals of vehicle dynamics* (SAE International).
145. Spanoudakis, P., Moschopoulos, G., Stefanoulis, T., Sarantinoudis, N., Papadokokolakis, E., Ioannou, I., Piperidis, S., Doitsidis, L., and Tsourveloudis, N.C. (2020). Efficient Gear Ratio Selection of a Single-Speed Drivetrain for Improved Electric Vehicle Energy Consumption. *Sustainability* 2020, Vol. 12, Page 9254 12, 9254. <https://doi.org/10.3390/SU12219254>.
146. Rosenberger, N., Rosner, P., Bilfinger, P., Schöberl, J., Teichert, O., Schneider, J., Abo Gamra, K., Allgäuer, C., Dietermann, B., Schreiber, M., et al. (2024). Quantifying the State of the Art of Electric Powertrains in Battery Electric Vehicles: Comprehensive Analysis of the Tesla Model 3 on the Vehicle Level. *World Electric Vehicle Journal* 2024, Vol. 15, Page 268 15, 268. <https://doi.org/10.3390/WEVJ15060268>.
147. Fuel Economy and EV Range Testing | US EPA <https://www.epa.gov/greenvehicles/fuel-economy-and-ev-range-testing>.
148. Kollmeyer, P.J., Naguib, M., Khanum, F., and Emadi, A. (2022). A Blind Modeling Tool for Standardized Evaluation of Battery State of Charge Estimation Algorithms. 2022 IEEE

- Transportation Electrification Conference and Expo, ITEC 2022, 243–248.
<https://doi.org/10.1109/ITEC53557.2022.9813996>.
149. CER – Market Snapshot:: Average electric vehicle range almost doubled in the last six years
<https://www.cer-rec.gc.ca/en/data-analysis/energy-markets/market-snapshots/2019/market-snapshot-average-electric-vehicle-range-almost-doubled-in-last-six-years.html>.
 150. Lu, M., Zhang, X., Ji, J., Xu, X., and Zhang, Y. (2020). Research progress on power battery cooling technology for electric vehicles. *J Energy Storage* 27, 101155.
<https://doi.org/10.1016/J.EST.2019.101155>.
 151. Chicco, D., Warrens, M.J., and Jurman, G. (2021). The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. *PeerJ Comput Sci* 7, 1–24. <https://doi.org/10.7717/PEERJ-CS.623/SUPP-1>.
 152. Saunders, L., Russell, R., and Crabb, D. (2012). The Coefficient of Determination: What Determines a Useful R2 Statistic? *Invest Ophthalmol Vis Sci* 53, 6830–6832. <https://doi.org/10.1167/IOVS.12-10598>.
 153. Kumar, R.R., and Alok, K. (2020). Adoption of electric vehicle: A literature review and prospects for sustainability. *J Clean Prod* 253, 119911. <https://doi.org/10.1016/J.JCLEPRO.2019.119911>.
 154. Álvarez Fernández, R. (2018). A more realistic approach to electric vehicle contribution to greenhouse gas emissions in the city. *J Clean Prod* 172, 949–959.
<https://doi.org/10.1016/J.JCLEPRO.2017.10.158>.
 155. Dinger, A., Martin, R., Mosquet, X., Rabl, M., Rizoulis, D., Russo, M., and Sticher, Georg. (2010). Batteries for electric cars: challenges, opportunities, and the outlook to 2020.

156. Bruen, T., and Marco, J. (2016). Modelling and experimental evaluation of parallel connected lithium ion cells for an electric vehicle battery system. *J Power Sources* 310, 91–101. <https://doi.org/10.1016/J.JPOWSOUR.2016.01.001>.
157. U.S. Department of Energy (2011). Advanced Vehicle Testing – Beginning-of-Test Battery Testing Results.
158. Davide, A. (2010). Introduction. In *Battery Management Systems* (Artech House), pp. 1–10.
159. Sun, P., Bisschop, R., Niu, H., and Huang, X. (2020). A Review of Battery Fires in Electric Vehicles. *Fire Technol.* <https://doi.org/10.1007/S10694-019-00944-3>.
160. Davide (2010). BMS Functions. In *Battery Management Systems* (Artech House), pp. 51–105.
161. Li, W., Rentemeister, M., Badedo, J., Jöst, D., Schulte, D., and Sauer, D.U. (2020). Digital twin for battery systems: Cloud battery management system with online state-of-charge and state-of-health estimation. *J Energy Storage* 30, 101557. <https://doi.org/10.1016/J.EST.2020.101557>.
162. Yao, L., Xiao, Y., Gong, X., Hou, J., and Chen, X. (2020). A novel intelligent method for fault diagnosis of electric vehicle battery system based on wavelet neural network. *J Power Sources* 453. <https://doi.org/10.1016/J.JPOWSOUR.2020.227870>.
163. Ojo, O., Lang, H., Kim, Y., Hu, X., Mu, B., and Lin, X. (2021). A Neural Network Based Method for Thermal Fault Detection in Lithium-Ion Batteries. *IEEE Transactions on Industrial Electronics* 68, 4068–4078. <https://doi.org/10.1109/TIE.2020.2984980>.
164. Wang, J., Zhang, S., and Hu, X. (2021). A Fault Diagnosis Method for Lithium-Ion Battery Packs Using Improved RBF Neural Network. *Front Energy Res* 9, 418. <https://doi.org/10.3389/FENRG.2021.702139/BIBTEX>.

165. Mohd Tohir, M.Z., and Martín-Gómez, C. (2023). Electric vehicle fire risk assessment framework using Fault Tree Analysis. *Open Research Europe* 3. <https://doi.org/10.12688/OPENRESEUROPE.16538.1>.
166. Zou, J., Zhang, J., and Jiang, P. (2019). Credit Card Fraud Detection Using Autoencoder Neural Network.
167. Zhao, Y., Deng, B., Shen, C., Liu, Y., Lu, H., and Hua, X.S. (2017). Spatio-temporal AutoEncoder for video anomaly detection. *MM 2017 - Proceedings of the 2017 ACM Multimedia Conference*, 1933–1941. <https://doi.org/10.1145/3123266.3123451>.
168. Chen, Z., Yeo, C.K., Lee, B.S., and Lau, C.T. (2018). Autoencoder-based network anomaly detection. *Wireless Telecommunications Symposium 2018-April*, 1–5. <https://doi.org/10.1109/WTS.2018.8363930>.
169. Lu, Y., Li, K., Han, X., Feng, X., Chu, Z., Lu, L., Huang, P., Zhang, Z., Zhang, Y., Yin, F., et al. (2020). A method of cell-to-cell variation evaluation for battery packs in electric vehicles with charging cloud data. *eTransportation* 6, 100077. <https://doi.org/10.1016/J.ETRAN.2020.100077>.
170. Atienza, R. (2020). Autoencoders. In *Advanced Deep Learning with Tensorflow 2 and Keras (Packt)*, pp. 78–70.
171. Geron (2019). Representation Learning and Generative Learning Using Autoencoders and GANs. In *Hands-on Machine Learning with Scikit-Learn and TensorFlow*, pp. 567–573.
172. Lopez Pinaya, W.H., Vieira, S., Garcia-Dias, R., and Mechelli, A. (2020). Autoencoders. *Machine Learning: Methods and Applications to Brain Disorders*, 193–208. <https://doi.org/10.1016/B978-0-12-815739-8.00011-0>.

173. Arora, S., and Kapoor, A. (2018). Mechanical Design and Packaging of Battery Packs for Electric Vehicles. *Green Energy and Technology* 0, 175–200. https://doi.org/10.1007/978-3-319-69950-9_8/FIGURES/9.
174. Wu, Y. (2020). Applications of Lithium-Ion Batteries. In *Lithium-Ion Batteries* (CRC Press), pp. 540–571. <https://doi.org/10.1201/B18427-21/APPLICATIONS-LITHIUM-ION-BATTERIES-CHANGHE-CAO-YUPING-WU-TEUNIS-VAN-REE>.
175. Deng, D. (2015). Li-ion batteries: basics, progress, and challenges. *Energy Sci Eng* 3, 385–418. <https://doi.org/10.1002/ESE3.95>.
176. Tarascon, J.M., and Armand, M. (2001). Issues and challenges facing rechargeable lithium batteries. *Nature* 201 414:6861 414, 359–367. <https://doi.org/10.1038/35104644>.
177. Lu, L., Han, X., Li, J., Hua, J., and Ouyang, M. (2013). A review on the key issues for lithium-ion battery management in electric vehicles. Preprint at Elsevier, <https://doi.org/10.1016/j.jpowsour.2012.10.060> <https://doi.org/10.1016/j.jpowsour.2012.10.060>.
178. Han, S., Tang, Y., and Khaleghi Rahimian, S. (2021). A numerically efficient method of solving the full-order pseudo-2-dimensional (P2D) Li-ion cell model. *J Power Sources* 490, 229571. <https://doi.org/10.1016/J.JPOWSOUR.2021.229571>.
179. Kumar, K., Sarkar, J., and Mondal, S.S. (2024). Analysis of ternary hybrid nanofluid in microchannel-cooled cylindrical Li-ion battery pack using multi-scale multi-domain framework. *Appl Energy* 355, 122241. <https://doi.org/10.1016/J.APENERGY.2023.122241>.
180. Plett, G. (2015). Thermal Modelling. In *Battery Management Systems* (Artech House), pp. 276–301.
181. Han, R., Macdonald, C., and Wetton, B. (2021). A fast solver for the pseudo-two-dimensional model of lithium-ion batteries.

182. Khaleghi Rahimian, S., Rayman, S., and White, R.E. (2013). Extension of physics-based single particle model for higher charge–discharge rates. *J Power Sources* 224, 180–194. <https://doi.org/10.1016/J.JPOWSOUR.2012.09.084>.
183. Schmidt, A., Oehler, D., Weber, A., Wetzel, T., and Ivers-Tiffée, E. (2021). A multi scale multi domain model for large format lithium-ion batteries. *Electrochim Acta* 393, 139046. <https://doi.org/10.1016/J.ELECTACTA.2021.139046>.
184. Kim, G.-H., Smith, K., Lee, K.-J., Santhanagopalan, S., and Pesaran, A. (2011). Multi-Domain Modeling of Lithium-Ion Batteries Encompassing Multi-Physics in Varied Length Scales. *J Electrochem Soc* 158, A955. <https://doi.org/10.1149/1.3597614/XML>.
185. Magri, L., Sequino, L., and Ferrari, C. (2023). Simulating the Electrochemical-Thermal Behavior of a Prismatic Lithium-Ion Battery on the Market under Various Discharge Cycles. *Batteries* 2023, Vol. 9, Page 397 9, 397. <https://doi.org/10.3390/BATTERIES9080397>.
186. Gwak, G., and Ju, H. (2019). Multi-Scale and Multi-Dimensional Thermal Modeling of Lithium-Ion Batteries. *Energies* 2019, Vol. 12, Page 374 12, 374. <https://doi.org/10.3390/EN12030374>.
187. Rizvi, S., Tahir, M.W., Ramzan, N., and Merten, C. (2024). Multiscale-multidomain model order reduction of Lithium-ion batteries for automobile application: A review. *J Energy Storage* 99, 113390. <https://doi.org/10.1016/J.EST.2024.113390>.
188. Plauska, I., Liutkevičius, A., and Janavičiūtė, A. (2022). Performance Evaluation of C/C++, MicroPython, Rust and TinyGo Programming Languages on ESP32 Microcontroller. *Electronics* 2023, Vol. 12, Page 143 12, 143. <https://doi.org/10.3390/ELECTRONICS12010143>.
189. Dokic, K., Radisic, B., and Cobovic, M. (2020). MicroPython or arduino C for ESP32 - Efficiency for neural network edge devices. *Communications in Computer and Information Science* 1187 CCIS, 33–43. https://doi.org/10.1007/978-3-030-43364-2_4/FIGURES/8.

190. Choroszucho, A., Golonko, P., Bednarek, J., Sumorek, M., and Żukowski, J. (2019). Comparison of high-level programming languages efficiency in embedded systems. <https://doi.org/10.1117/12.2537143> 11176, 1800–1808. <https://doi.org/10.1117/12.2537143>.
191. Chen, J., Qin, P., Li, Z., Che, Y., and Cai, Y. (2020). STM32-Based Platform for Testing Energy Storage Lithium Battery. Proceedings of 2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications, AEECA 2020, 192–196. <https://doi.org/10.1109/AEECA49918.2020.9213646>.
192. Li, J., Wang, S., Fernandez, C., Wang, N., and Xie, H. (2017). The battery management system construction method study for the power lithium-ion battery pack. 2017 2nd International Conference on Robotics and Automation Engineering, ICRAE 2017 2017-December, 285–289. <https://doi.org/10.1109/ICRAE.2017.8291396>.
193. STM32 Microcontrollers (MCUs) - STMicroelectronics <https://www.st.com/en/microcontrollers-microprocessors/stm32-32-bit-arm-cortex-mcus.html>.
194. Roseman, Mark. (2020). Modern Tkinter for busy Python developers : quickly learn to create great looking user interfaces for Windows, Mac and Linux using Python’s standard GUI toolkit. 257.
195. NumPy - <https://numpy.org/>.
196. Thomas, J.W. (1995). Numerical Partial Differential Equations: Finite Difference Methods (Springer New York) <https://doi.org/10.1007/978-1-4899-7278-1>.
197. Thibault, J., Bergeron, S., and Bonin, H.W. (1987). ON FINITE-DIFFERENCE SOLUTIONS OF THE HEAT EQUATION IN SPHERICAL COORDINATES. Numeri Heat Transf A Appl 12, 457–474. <https://doi.org/10.1080/10407788708913597>.
198. Lee, W. Tridiagonal Matrices: Thomas Algorithm - University of Limerick.

199. Torchio, M., Magni, L., Gopaluni, R.B., Braatz, R.D., and Raimondo, D.M. (2016). LIONSIMBA: A Matlab Framework Based on a Finite Volume Model Suitable for Li-Ion Battery Design, Simulation, and Control. *J Electrochem Soc* 163, A1192–A1205. <https://doi.org/10.1149/2.0291607JES/XML>.
200. Senthil Kumar, V. (2013). Reduced order model for a lithium ion cell with uniform reaction rate approximation. *J Power Sources* 222, 426–441. <https://doi.org/10.1016/J.JPOWSOUR.2012.09.013>.
201. Senthil Kumar, V., Gambhire, P., Hariharan, K.S., Khandelwal, A., Kolake, S.M., Oh, D., and Doo, S. (2014). An explicit algebraic reduced order algorithm for lithium ion cell voltage prediction. *J Power Sources* 248, 383–387. <https://doi.org/10.1016/J.JPOWSOUR.2013.09.089>.
202. Marelli, S., and Corno, M. (2021). Model-Based Estimation of Lithium Concentrations and Temperature in Batteries Using Soft-Constrained Dual Unscented Kalman Filtering. *IEEE Transactions on Control Systems Technology* 29, 926–933. <https://doi.org/10.1109/TCST.2020.2974176>.
203. Li, W., Fan, Y., Ringbeck, F., Jöst, D., Han, X., Ouyang, M., and Sauer, D.U. (2020). Electrochemical model-based state estimation for lithium-ion batteries with adaptive unscented Kalman filter. *J Power Sources* 476, 228534. <https://doi.org/10.1016/J.JPOWSOUR.2020.228534>.
204. Plett, G. (2016). Battery State Estimations. In *Battery Management Systems* (Artech House), pp. 69–165.
205. Welcome to Python.org <https://www.python.org/>.
206. cplusplus.com <https://cplusplus.com/>.
207. Ateeq, M., Habib, H., Umer, A., and Ul Rehman, M. (2014). C++ or python?: Which one to begin with: A learner's perspective. *Proceedings - 2014 International Conference on Teaching and*

- Learning in Computing and Engineering, LATICE 2014, 64–69.
<https://doi.org/10.1109/LATICE.2014.20>.
208. Ahmadi, M., and Qaisari Hasan Abadi, M. (2020). A review of using object-orientation properties of C++ for designing expert system in strategic planning. *Comput Sci Rev* 37, 100282.
<https://doi.org/10.1016/J.COSREV.2020.100282>.
 209. About the Unified Modeling Language Specification Version 2.5.1
<https://www.omg.org/spec/UML/2.5.1>.
 210. Farmann, A., and Sauer, D.U. (2017). A study on the dependency of the open-circuit voltage on temperature and actual aging state of lithium-ion batteries. *J Power Sources* 347, 1–13.
<https://doi.org/10.1016/J.JPOWSOUR.2017.01.098>.
 211. Whitaker, S. (1999). Overview. In *The Method of Volume Averaging* (Springer), pp. xiii–xvi.
 212. Ma, S., Jiang, M., Tao, P., Song, C., Wu, J., Wang, J., Deng, T., and Shang, W. (2018). Temperature effect and thermal impact in lithium-ion batteries: A review. *Progress in Natural Science: Materials International* 28, 653–666. <https://doi.org/10.1016/J.PNSC.2018.11.002>.
 213. Gray, W.G., and Lee, P.C.Y. (1977). On the theorems for local volume averaging of multiphase systems. *International Journal of Multiphase Flow* 3, 333–340. [https://doi.org/10.1016/0301-9322\(77\)90013-1](https://doi.org/10.1016/0301-9322(77)90013-1).
 214. Edge, J.S., O’Kane, S., Prosser, R., Kirkaldy, N.D., Patel, A.N., Hales, A., Ghosh, A., Ai, W., Chen, J., Yang, J., et al. (2021). Lithium ion battery degradation: what you need to know. *Physical Chemistry Chemical Physics* 23, 8200–8221. <https://doi.org/10.1039/D1CP00359C>.
 215. Darling, R., and Newman, J. (1998). Modeling side reactions in composite LiYMn₂O₄ electrodes. *undefined* 145, 990–998. <https://doi.org/10.1149/1.1838376>.

216. Arora, P., Doyle, M., and White, R.E. (1999). Mathematical Modeling of the Lithium Deposition Overcharge Reaction in Lithium-Ion Batteries Using Carbon-Based Negative Electrodes. *J Electrochem Soc* 146, 3543–3553. <https://doi.org/10.1149/1.1392512/XML>.
217. Nizam, M., Maghfiroh, H., Rosadi, R.A., and Kusumaputri, K.D.U. (2020). Battery management system design (BMS) for lithium ion batteries. *AIP Conf Proc* 2217. <https://doi.org/10.1063/5.0000649/1025261>.
218. Raghunathan, K.R. (2021). History of Microcontrollers: First 50 Years. *IEEE Micro* 41, 97–104. <https://doi.org/10.1109/MM.2021.3114754>.

Appendix A: Volume Averaging of Partial Differential Equations of Multiphase System

In a multiphase system, local averaging of the transport equations (within the domain containing the phase) can be applied to derive continuum equations. In battery-related microscale equations, it is important that the characteristic length scale of the volume-average be greater than the length scales of the pores in the electrode-electrolyte domain.^{33,211,213}

The phase of interest (in a two-phase system), within the volume, is referred to as α , while the other phase is referred to as β . A step function (γ_α) is used to define the presence of α -phase in the co-ordinate system and its derivative is the Dirac delta function.²¹³ A pictorial representation of the α - and β -phases, in a one-dimensional co-ordinate system, is shown in Figure D 1.

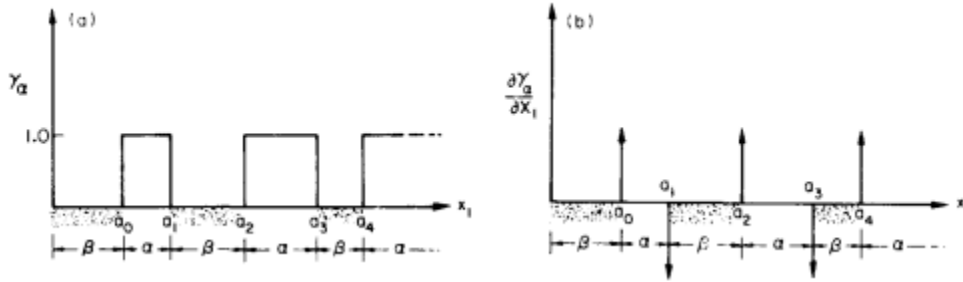


Figure 46: The step function represents the presence of α -phase in a one dimensional co ordinate system (left), while its derivative is the Dirac delta function (right).

The phase average of a property, ψ , concerns with the entire volume of the domain (V) and is defined as²¹³

$$\langle \psi_\alpha \rangle = \frac{1}{V} \int \psi \gamma_\alpha dV$$

While the intrinsic phase average concerns only over the α -phase and is defined by

$$\langle \psi_\alpha \rangle = \frac{1}{V_\alpha} \int \psi_\alpha dV$$

Physically, for a scalar and vector field, the spatial and time derivate are defined as follows. ^{33,213}

$$\nabla \psi_\alpha = \nabla \psi_\alpha + \frac{1}{V} \iint \psi_\alpha \widehat{\mathbf{n}}_\alpha dA \quad \text{Equation 142}$$

$$\nabla \cdot \psi_\alpha = \nabla \cdot \psi_\alpha + \frac{1}{V} \iint \psi_\alpha \cdot \widehat{\mathbf{n}}_\alpha dA \quad \text{Equation 143}$$

$$\frac{\partial \psi_\alpha}{\partial t} = \frac{\partial \psi_\alpha}{\partial t} - \frac{1}{V} \iint \psi_\alpha \mathbf{v}_{\alpha\beta} \cdot \widehat{\mathbf{n}}_\alpha dA \quad \text{Equation 144}$$

Where $\mathbf{n}_\alpha$ and $\mathbf{v}_{\alpha\beta}$ are the unit normal vector pointing out of the α -phase and the displacement of the α - and β -phase, respectively.

Appendix B: Non-linear Kalman Filter

The state and output equations of a state-space model of a system are represented as follows.

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1})$$

$$\mathbf{y}_k = h(\mathbf{x}_k, \mathbf{u}_k, \mathbf{v}_k)$$

Where $\mathbf{u}_k$, $\mathbf{x}_k$, $\mathbf{w}_k$, and $\mathbf{v}_k$ are the random variable vectors containing the input signal, model state, process noise, and sensor-noise signal random variables, respectively. Sequential probabilistic inference, which Kalman filters are a subset of, computes the conditional expectation of the system state using the present and past output measurements (the set containing the present and the past measurement points is denoted by $\mathbb{Y}_k$), $\mathbb{E}[\mathbf{x}_k | \mathbb{Y}_k]$.

Kalman filter has two main outputs, the state estimates and the state covariance estimate, at the current time steps.

$$\mathbf{L}_k = \boldsymbol{\Sigma}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}},k}^- \boldsymbol{\Sigma}_{\tilde{\mathbf{y}},k}^{-1}$$

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \mathbf{L}_k \tilde{\mathbf{y}}_k$$

$$\boldsymbol{\Sigma}_{\tilde{\mathbf{x}},k}^+ = \boldsymbol{\Sigma}_{\tilde{\mathbf{x}},k}^- - \mathbf{L}_k \boldsymbol{\Sigma}_{\tilde{\mathbf{y}},k} \mathbf{L}_k^T$$

The following lists the steps required for the Gaussian sequential probabilistic inference along with their equations. The first three steps pertain to the state prediction, while the last three concern with the correction.

1. The state-prediction, $\hat{\mathbf{x}}_k^-$

$$\hat{\mathbf{x}}_k^- = \mathbb{E}[\mathbf{x}_k | \mathbb{Y}_{k-1}] = \mathbb{E}[f(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}) | \mathbb{Y}_{k-1}]$$

2. The error-covariance, $\boldsymbol{\Sigma}_{\tilde{\mathbf{x}},k}^-$

$$\boldsymbol{\Sigma}_{\tilde{\mathbf{x}},k}^- = \mathbb{E}[(\tilde{\mathbf{x}}_k^-)(\tilde{\mathbf{x}}_k^-)^T] = \mathbb{E}[(\mathbf{x}_k - \hat{\mathbf{x}}_k^-)(\mathbf{x}_k - \hat{\mathbf{x}}_k^-)^T]$$

3. The system output prediction, y_k

$$\hat{y}_k = \mathbb{E}[y_k | \mathbb{Y}_{k-1}] = \mathbb{E}[h(\mathbf{x}_k, \mathbf{u}_k, \mathbf{v}_k) | \mathbb{Y}_{k-1}]$$

4. The gain factor, L_k

$$\mathbf{L} = \mathbf{\Sigma}_{\tilde{\mathbf{x}}\tilde{\mathbf{y}},k}^- \mathbf{\Sigma}_{\tilde{\mathbf{y}},k}^{-1}$$

5. The state estimate, $\hat{\mathbf{x}}_k^+$

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \mathbf{L}_k(\mathbf{y}_k - \hat{\mathbf{y}}_k)$$

6. The covariance estimate update, $\mathbf{\Sigma}_{\tilde{\mathbf{x}},k}^+$

$$\mathbf{\Sigma}_{\tilde{\mathbf{x}},k}^+ = \mathbf{\Sigma}_{\tilde{\mathbf{x}},k}^- - \mathbf{L}_k \mathbf{\Sigma}_{\tilde{\mathbf{y}},k} \mathbf{L}_k^T$$

The following expressions are specific for the Sigma-Point Kalman filter.

Step 1a

$$\hat{\mathbf{x}}_k^- = \mathbb{E}[f(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}) | \mathbb{Y}_{k-1}] \approx \sum_0^p \alpha_i^{(m)} f(\mathbf{x}_{k-1}^{x,+}, u_{k-1}, \mathbf{x}_{k-1}^{w,+}) = \sum_0^p \alpha_i^m \mathbf{x}_{k,i}^{x,-}$$

Step 1b

$$\mathbf{\Sigma}_{\tilde{\mathbf{x}},k}^- = \sum_0^p \alpha_i^{(c)} \left((\tilde{\mathbf{x}}_{k,i}^{x,-}) (\tilde{\mathbf{x}}_{k,i}^{x,-})^T \right)$$

Step 1c

$$\hat{y}_k = \mathbb{E}[h(\mathbf{x}_k, \mathbf{u}_k, \mathbf{v}_k) | \mathbb{Y}_{k-1}] \approx \sum_0^p \alpha_i^{(m)} f(\mathbf{x}_{k-1}^{x,+}, \mathbf{u}_{k-1}, \mathbf{x}_{k-1}^{v,+}) = \sum_0^p \alpha_i^m \mathbf{y}_{k,i}^{x,-}$$

Step 2a

$$\mathbf{\Sigma}_{\tilde{y},k}^- = \sum_0^p \alpha_i^{(c)} \left((\tilde{\mathbf{y}}_{k,i}) (\tilde{\mathbf{x}}_{k,i}^{x,-})^T \right)$$

Step 2b

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + L_k (\mathcal{Y}_k - \hat{\mathbf{y}}_k)$$

Step 2c

$$\mathbf{\Sigma}_{\tilde{x},k}^+ = \mathbf{\Sigma}_{\tilde{x},k}^- - \mathbf{L}_k \mathbf{\Sigma}_{\tilde{y},k}^- \mathbf{L}_k^T$$