

State Estimation in Lithium-ion Batteries Using Pulse Perturbation and Feedforward Neural Networks



Alan Gen Li

Submitted in partial fulfillment of the
requirements for the degree of
Master of Science
under the Department of Electrical Engineering
in the Fu Foundation School of Engineering and Applied Science

COLUMBIA UNIVERSITY

2020

© 2020

Alan Gen Li

All Rights Reserved

Abstract

Predicting battery stored charge, available capacity, and peak power quickly and accurately is important for understanding pack performance and stability. It is proposed that a feedforward neural network (FNN) can estimate this information using cell voltage response to an injected current. Voltage response varies with the internal chemistry, represented by charge, capacity, and impedance. These characteristics are quantified here using state of charge (SoC), state of energy (SoE), and state of power (SoP). Cell response data is collected for various states at constant temperature, resulting in 234 unique voltage responses for training and evaluating the FNN. Training is performed using 3 distinct variations on the data: (1) the full voltage response, (2) individual portions of the response, such as charging or relaxation periods, and (3) fractions of the charge and discharge periods ranging from one-half to a single open-circuit voltage measurement. Using the full response, the average mean absolute error (MAE) is 0.0057 for SoE estimation. The average MAE is below 0.0080 for SoC and SoP estimation. The results for pulse portions show that Charge-rest or Discharge-rest responses perform almost as well as the full pulse. This may inform future pulse design for further optimization. The results for pulse fractions show that error increases as the amount of input data decreases, which validates the hypothesis that pulse perturbation yields high performance in FNN. The technique can be expanded to other temperatures, with potential for estimation of other states, and even degradation mechanisms. Estimation requires 3 minutes of voltage and current data, with no charging history needed and low computational complexity. The proposed method is thus suitable for development of advanced battery management systems in electric vehicles.

Acknowledgements

Thanks are due first to Prof Matthias Preindl, whose guidance and insight have been invaluable to me. I also thank Prof Alan C. West, Weizhong Wang, and the other members of the battery management and machine learning group at Columbia for their weekly inspiration.

Table of Contents

Acknowledgments	i
List of Figures	iv
List of Tables	v
Nomenclature	vi
1 Introduction and Literature Review	1
1.1 Introduction	1
1.1.1 Definitions of battery states	3
1.1.2 Cell perturbation	5
1.2 Conventional state estimation methods	6
1.2.1 State of Charge	6
1.2.2 State of Energy	9
1.2.3 State of Power	11
1.2.4 Joint estimation	12
1.3 Data-driven state estimation methods	13
1.3.1 State of Charge	14
1.3.2 State of Energy	16
1.3.3 State of Power	17
1.3.4 Joint estimation	18
1.4 Outline	19
2 Data Collection and Processing	20
2.1 Data Collection	20
2.2 Pulse parameters and training data	22
2.3 Data processing	24

3	Neural Network Design	31
3.1	Theory	31
3.2	Implementation	33
4	Results and Discussion	37
4.1	Full pulse	37
4.2	Pulse portions and fractions	40
5	Conclusion and Future Work	43
5.1	Real-world verification	43
5.2	Outlook	44
	References	46

List of Figures

1.1	Flowchart of BMS functions	2
1.2	First-order ECM	5
1.3	Comparison of state estimation methods	6
2.1	Experimental setup	21
2.2	Testing procedure	21
2.3	Cycling voltage and current	22
2.4	Current pulse from cycler	23
2.5	Voltage responses color-coded by SoC	24
2.6	Portions of voltage responses color-coded by SoC	25
2.7	Cell SoE for three test cells	26
2.8	SoP and ECM parameter evolution with SoC	29
2.9	SoP and ECM parameter evolution with SoE	30
3.1	Diagram of general FNN structure	31
3.2	Samples of FNN training for different states	36
4.1	Box plot of MAE against nodes per hidden layer for state estimations	37
4.2	Samples of FNN predictions for different states	38
4.3	Box plot of MAE results for full pulse using chosen nodes	39
4.4	Box plots of MAE results for pulse portions	41
4.5	Box plots of MAE results for fractions of charge pulse	42
4.6	Box plots of MAE results for fractions of discharge pulse	42

List of Tables

2.1	Cell characteristics	20
2.2	Pulse portion definitions	23
3.1	Shared hyperparameters amongst FNN models	34
4.1	Comparison of parameter estimation in selected studies	40

Nomenclature

API	Application programming interface
BMS	Battery management system
CC	Constant current
CV	Constant voltage
DoD	Depth of discharge
ECM	Equivalent circuit model
EIS	Electrochemical impedance spectroscopy
ESS	Energy storage system
EV	Electric vehicle
FNN	Feedforward neural network
GPR	Gaussian process regression
GRA	Grey relational analysis
GRC	Grey relational coefficient
HPPC	Hybrid-pulse power characterization
IC	Incremental capacity
KF	Kalman filter
LIB	Lithium-ion battery
MAE	Mean absolute error
MSE	Mean squared error
NCA	Nickel cobalt aluminum
NFRA	Nonlinear frequency response analysis

NN Neural network
OCV Open circuit voltage
ODE Ordinary differential equation
P2D pseudo-2D
PDE Partial differential equation
PF Particle filter
PSO Particle swarm optimization
pu per unit
RMSE Root mean square error
RNN Recurrent neural network
RUL Remaining useful life
SEI Solid electrolyte interphase
SoC State of charge
SoE State of energy
SoH State of health
SoP State of power
SPM Single particle model
SVM Support vector machine

Chapter 1

Introduction and Literature Review

1.1 Introduction

Energy storage systems (ESS) are crucial for reducing carbon emissions and fighting climate change [1–3]. Electrochemical ESS are popular options, of which batteries and hydrogen fuel cells are competing technologies [4]. Both are used to power electric vehicles (EVs), which are a clean alternative to internal combustion engine vehicles. Increasing the share of EVs on the road is recognized as an important step to decarbonizing transport [5]. While there is ongoing research into the benefits of batteries versus fuel cells in certain applications, lithium-ion batteries (LIBs) have taken the clear lead for EVs [6]. LIBs are already used in most EVs for their high round-trip efficiency, high energy density, and relatively low cost of recharging [5, 7, 8]. For this reason, LIBs are the subject of continued research for further cost reduction, which can be achieved by improvements in capacity and energy density.

The battery management system (BMS) is a primary consideration in LIB systems. The BMS monitors and controls battery cells to increase usable capacity and power output, and ensure safety [9–12]. Generally, the BMS measures voltages, currents, and temperature to produce estimates on LIB states and safeguard the pack from overheating. It also performs cell balancing. Cell balancing circuits consist of passive or active components that act on

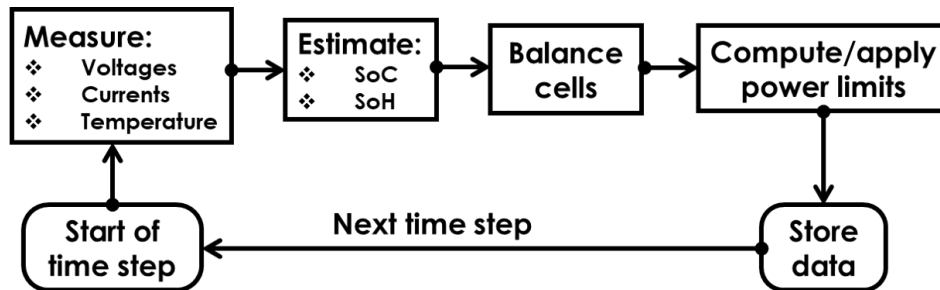


Figure 1.1: Flowchart of BMS functions

the cells within a pack to equalize SoC across the cells. As the pack is charged or discharged, stronger cells will have higher SoC than weak cells. Passive balancing aims to discharge stronger cells using components such as resistors or capacitors. Active balancing aims to redistribute charge from strong cells to weak cells. This prevents weaker cells from causing entire strings to fail and can reduce pack degradation [13–15]. Active cell balancing, like other functions of the BMS, depends on accurate state estimation. For EV applications especially, state estimation must also be cost-effective and fast, which can improve performance and stability [16–23]. A summary of BMS operation is shown in Figure 1.1.

LIB states are important metrics for LIB systems. State of charge (SoC) quantifies normalized remaining charge. SoC defines the amount of available energy in the cell and affects peak power output. State of health (SoH) quantifies LIB degradation, which occurs as cells are charged, discharged, or stored. Degradation decreases the maximum charge capacity and increases internal resistance, also lowering the peak power. In this report, SoH is treated as two quantities: state of energy (SoE), representing maximum charge capacity, and state of power (SoP), representing peak power output. SoC, SoE, and SoP are not directly measurable, so accurate estimation is important for reducing risk of failure and understanding the economic value of battery cells [24].

State estimation methods are broadly divided into two categories: conventional and data-driven. Conventional methods compute or measure parameters for electrochemical or regression models using laboratory data [23, 25–54]. Conventional approaches often have a trade-off between accuracy and computational speed, but are relatively cheap [16]. They

rely on high-fidelity battery models, which lose accuracy as the battery ages, thus providing potentially false information. Estimator parameters may require hand-tuning, which can be time consuming and inaccurate. Conventional methods also may not be applicable for cells of different chemistries. In contrast to conventional state estimation methods, data-driven state estimation may offer greater accuracy, speed, and generality [16, 21, 55–73].

This report explores a feedforward neural network (FNN) method for estimating states using cell voltage response to a current pulse perturbation. FNN are a promising data-driven method for state estimation. Battery states are complex quantities governed by many internal and external factors, and it is known from electrochemical impedance spectroscopy (EIS) techniques [74–76] and fundamental battery chemistry that the voltage response of a cell to a stimulus current varies with battery states. This makes voltage responses good candidates for FNN estimation. For training the FNN, a large dataset of voltage responses under various conditions is obtained. It is shown that FNN estimation with pulse perturbation allows for accurate, fast, and robust estimation compared to other methods.

Key theoretical concepts in this report are discussed in the following subsections. Subsection 1.1.1 defines the battery states and Subsection 1.1.2 describes LIB voltage response to a perturbation. The chapter continues to Sections 1.2 and 1.3 for the literature review. Conventional state estimation methods are reviewed in Section 1.2, and data-driven state estimation methods are reviewed in Section 1.3. The chapter is concluded with an outline of the report in Section 1.4.

1.1.1 Definitions of battery states

This subsection defines the battery states in this report. Per unit [pu] values are used when describing cell states (0.01 as opposed to 1%, for example).

SoC quantifies the remaining charge q in the cell relative to the maximum charge capacity of the cell Q_m , calculated with

$$SoC = \frac{q}{Q_m} \quad (1.1)$$

Often known as the ‘fuel gauge’ of a battery, SoC yields information on the amount of charge a cell can provide before depletion. For this is reason it is particularly important to estimate SoC quickly and accurately. Cells typically have minimum levels of SoC, defined by the depth of discharge (DoD), that reduce the usable amount of charge.

SoH is composed of two metrics: SoE, or normalized maximum capacity, and SoP. Both metrics are important for gauging LIB health. As a cell ages, both SoE and SoP decrease. This is due to degradation mechanisms such as solid electrolyte interphase (SEI) formation, lithium plating, and changes in the active material and electrode structure [77–80]. These mechanisms result in a loss of lithium inventory and active material in the LIB anode and cathode, which cause capacity fade (reduced SoE) or power fade (reduced SoP).

Real-time knowledge of SoH can be used to improve SoC estimation and BMS operation. It also determines the remaining useful life (RUL) of the battery pack. If accurate and fast SoH estimation can yield information on degradation mechanisms, the BMS could actively adjust operations to increase RUL or performance [77]. This would allow for new control methods. Thus SoH estimation should also as fast and accurate as possible.

SoE can be viewed as the total energy the cell can deliver from full charge, given by

$$SoE = \frac{Q_m}{Q_{m0}} \quad (1.2)$$

where Q_{m0} is the maximum capacity of the unaged cell. Since SoE is affected only by degradation, SoE can be used to quantify the age of the cell. It is also known as capacity fade.

SoP, also known as power fade, quantifies the peak power output of the cell. It is affected by both degradation and SoC. It is often defined with ECM parameters [44–46, 81]. First-order parameters are considered here (Figure 1.2), where V_{OC} is the open-circuit voltage (OCV) of the cell, which is a function of SoC, R_0 is series resistance, and R_1 and C_1 form the first-order RC pair. The parameters can be used to calculate peak discharge current I_{pk} .

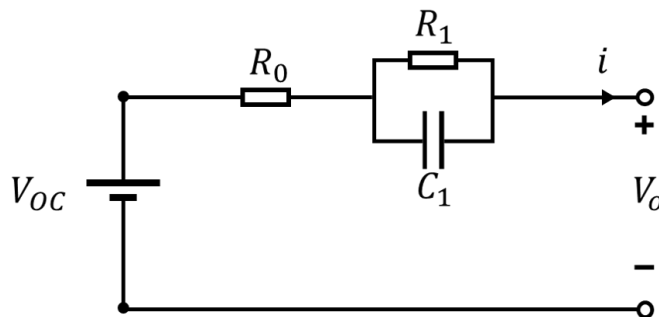


Figure 1.2: First-order ECM

SoP is then defined as

$$SoP = \frac{I_{pk}}{I_{pk0}} \quad (1.3)$$

where I_{pk0} is the peak discharge current of the unaged cell at maximum SoC.

1.1.2 Cell perturbation

It is well known from EIS that the LIB output voltage magnitude and frequency response varies with cell states and operation [33]. In EIS, sinusoidal stimuli are applied to the battery cell at a wide range of frequencies, typically from millihertz to kilohertz. The stimulus is potentiostatic if it is an applied voltage, and galvanostatic if an applied current. The response is then measured, and the ratio between the magnitude and phase of the stimuli and response then yields the complex impedance. A Nyquist diagram of the complex impedance is often used to deduce cell parameters, which can then be used in equivalent circuit models for degradation modelling [16, 82, 83]. Such models may be complex, which would hinder real-time implementation.

Pulsed-current perturbation takes advantage of the underlying principles behind EIS, but focuses on the voltage-time response. Rather than stimulate the cell at specific frequencies, as in EIS, the cell is stimulated at a spectrum frequencies simultaneously in the form of a current pulse. The response is then used in a FNN to predict LIB states. Figure 1.3 shows a comparison between conventional methods and the perturbation method. Conven-

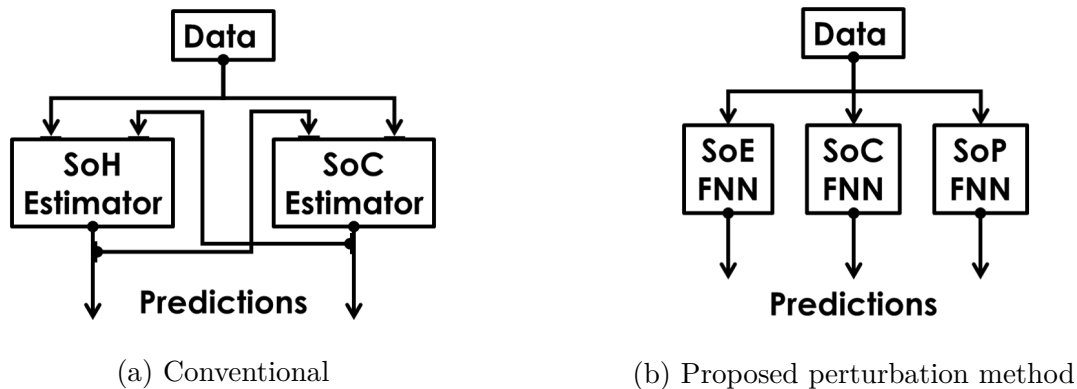


Figure 1.3: Comparison of state estimation methods

tional estimators depend on the predictions of the other estimators. In contrast, FNN can independently make predictions without knowledge of the other states.

In [61], data from pulse perturbation is used in a recurrent neural network to predict SoH metrics. The perturbation uses several high-frequency components to simulate a drive cycle. This may not be easy to replicate due to the random variation of most drive cycles, and may not be the optimal pulse shape. It is partially the aim of this report to address these issues.

1.2 Conventional state estimation methods

Conventional methods rely on battery models to faithfully reconstruct SoC or SoH based on the model parameters. Not only do these models face challenges in parametrization, but they are often used for estimating individual states for specific battery chemistries and thus lack generality [66, 70]. Still, conventional methods remain popular because they can achieve high accuracy, albeit with high computational intensity [28].

1.2.1 State of Charge

The simplest form of SoC estimation uses a look-up table to link cell OCV to SoC based on a laboratory-determined characteristic curve. Because of the overvoltages created by

polarization mechanisms within the LIB, a single SoC corresponds to a range of voltages, a phenomenon known as hysteresis. Thus a long rest period is required for the cell OCV to accurately correspond with the laboratory-determined characteristic curve, making the look-up table unusable for practical applications. Some studies have attempted to improve the basic OCV method by modelling the rest period. In [25], data collected from at least 10 minutes of cell resting is used to predict the OCV after a very long time. They demonstrate an improvement over the basic OCV method, but the time and accuracy of their proposed model are not suitable for EV applications. Contemporary methods may also use the look-up table as an input to a more advanced filtering method. It is important to consider the temperature dependency of OCV, which is shown in [26] to have a major influence on the estimation accuracy.

Coulomb-counting is another simple method of SoC estimation. In coulomb-counting, SoC is calculated using

$$SoC = SoC_0 + \frac{\eta}{Q_m} \int I dt \quad (1.4)$$

where SoC_0 is an initial value of SoC, η is the Coulombic efficiency and I is the current into the cell. Thus this method assumes that the initial SoC is known, and that Coulombic efficiency remains constant. This is not usually true because Coulombic efficiency varies with operating conditions, sensors may introduce systemic error, and thus the initial value may not be known accurately. Over time, if the SoC is not re-initialized, error accumulates and the coulomb-counting method becomes unusable.

Non-linear Kalman filters (KF) are a popular technique for SoC estimation that can yield fast and accurate results [27–30]. KF are derived from sequential probability analysis, and treat the true SoC as a ‘hidden’ state affected by process or sensor noise. The aim is to remove noise from the system to obtain an optimal estimate. To apply the algorithm, the system is discretized into a state-space representation. Using a first-order ECM, the

state-space representation is given by

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}i_k + \mathbf{w} \quad (1.5)$$

$$V_{o,k} = f(\mathbf{x}_k, i_k) = V_{OC}(SoC_k) - V_{1,k} - R_0i_k + v \quad (1.6)$$

$$\mathbf{x}_k = \begin{pmatrix} V_{1,k} \\ SoC_k \end{pmatrix}, \quad \mathbf{A} = \begin{pmatrix} e^{-\frac{\Delta t}{R_1C_1}} & 0 \\ 0 & 1 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} R_1 \left(1 - e^{-\frac{\Delta t}{R_1C_1}}\right) \\ -\eta \frac{\Delta t}{Q_m} \end{pmatrix} \quad (1.7)$$

where k is the time step, V_1 is the voltage of the RC pair, Δt is sampling period, \mathbf{w} is the process noise, v is the measurement noise, $V_{o,k}$ is a non-linear function of the input and output states. KF use several update functions to adjust the system states, covariance, and output. Various non-linear KF have been proposed. Extended KF linearize the output equation at each time step using a Taylor-series expansion, but the resultant algorithm may diverge. Unscented KF also linearize at each time step, but use ‘sigma point’ vectors to replace the mean and covariance between states. In [84] the unscented KF is applied to estimate SoC with very high accuracy. It is uncertain whether how accuracy is affected by cell degradation. Curvature KF, a more recent development explored in [28, 29], are similar to unscented KF but use multi-dimensional weighted integrals to formulate the update function. KF may be able to achieve high performance, such as root-mean square error (RMSE) below 0.01, but can be complex and require long computation times, especially when more accurate battery models are used [30].

Particle filters (PF) are based on sequential Monte Carlo methods. The aim is to use random samples, or ‘particles’, to approximate the posterior density distribution of the system states [31, 32]. In [31], a combination of KF and PF are used to estimate battery parameters. The KF are used to provide preliminary predictions, and the PF is used to make a final correction to the SoC estimation. Increasing the particle number can yield high accuracy but at the expense of significantly higher computation time.

1.2.2 State of Energy

SoE is the most common definition of SoH, though the internal resistance of the cell, which may be represented as $(R_0 + R_1)$, is also used. SoE can be obtained using coulomb-counting. The battery is fully charged, then discharged at a very low C-rate to 0 SoC. This can be very accurate, but requires several hours and thus cannot be used for real-time estimation [23].

Cell OCV is known to be affected by SoC, SoE, and temperature, and can thus be used to estimate SoH [33, 34]. In [33], the OCV of LIB cells during constant-current (CC) charging is modelled with respect to SoE. A numerical method is then used to identify the SoE based on the charging curves. The maximum reported error is 0.03, but the charging curves for various SoE all last for more than 1 hour. In [34], an OCV model of cell is developed using a first-order ECM and a superposition of Gaussian functions. The model parameters are then identified using recursive least-squares and a curve-fitting algorithm. The SoE is obtained as a function of the model parameters. The mean error is 0.03. Again, the entire OCV-SoC curve is required as the model input, making this approach unsuitable for real-time applications.

Incremental capacity (IC) analysis considers the change in SoC with voltage using the differential $\frac{dSoC}{dV_{OC}}$. Characteristic IC curves have up to five peaks, and can be linked to degradation and SoE [35, 36]. In [35], IC analysis is extended to CC charging curves to shorten estimation time. It is shown that the change in voltage and SoC around the peak displays a linear relationship with SoE. The proposed model estimates SoE with maximum RMSE of 0.0249, with charging curves lasting 45 minutes. Though this is an improvement over traditional IC analysis, it is still too long for real-time estimation. In [36], an even faster IC-based approach is presented. The differential $\frac{dSoC}{dV_{OC}}$ is calculated for 10 intervals of SoC ranging from 0 to 1. Data within each interval is modelled with respect to SoE. Mean absolute error (MAE) is 0.0108 with estimation time of 6 minutes. While this approach

may be promising, it is dependent on accurate SoC estimation and exhibits higher error at different SoC intervals.

Cell frequency response is also used to estimate SoE [37, 38]. In [37], key features of the EIS Nyquist plot are used to identify ECM parameters for a range of SoC and SoH. A belief function is then used to map ohmic resistance to SoE based on measured data. The proposed method can estimate SoE online with maximum error of 0.0373 for cells used to generate the training dataset. The model shows poor generality, however, because error increases to 0.0866 for cells outside of the training set. In [38], SoE is examined using galvanostatic non-linear frequency response analysis (NFRA). NFRA is similar to EIS, but considers the amplitudes of the voltage harmonics in addition to the voltage fundamental. The authors link SoE to the ratio between the second and third harmonics and the fundamental. The method may yield accurate results, but configuring the method to allow real-time estimation may be challenging.

Electrochemical models can accurately model chemical processes within LIB, but are often too computationally expensive to be used in real-time. To address this, simplified models are developed [39, 40]. In [39], a single-particle model (SPM) is used. The SPM is a reduced-order model described by ordinary differential equations. This simplifies the computational intensity of the pseudo-2D (P2D) model, which consists of many coupled non-linear partial differential equations (PDEs). The authors link SoE and resistance primarily to growth of the SEI layer due to the initial formation and cracking. The RMSE is shown to be 0.0103. The model does not capture other degradation mechanisms, however, such as lithium plating. Internal cell parameters are assumed. In [40], a genetic algorithm is proposed for parameter identification of a simplified P2D model. Finite analysis is used to reduce the PDE dimensions and obtain ordinary differential equations (ODE) which can be solved analytically or numerically. Maximum estimation error is 0.03. It is evident that the electrochemical models in [39] and [40] present a challenging parameter identification problem that may not be widely applicable for all battery chemistries.

PF are also used in SoH estimation [41–43]. SoH is defined using ohmic resistance in [41]. Genetic resampling is shown to improve the PF optimization process for online estimation, but there is high relative variance in the results. In [42], SoE and internal impedance are fused into one SoH metric. The SoH parameters are identified using an unscented PF algorithm to yield maximum percent error of 5%, but the input data requires a full discharge and charge cycle, meaning it cannot be used in real-time applications. A similar approach is presented in [43], who use a PF to estimate a fused SoE and resistance SoH metric, which is then used to estimate the RUL with high accuracy. A full discharge-charge cycle is required, which again restricts the approach from real-time applications.

1.2.3 State of Power

Unlike SoC and SoE, SoP is not as widely researched [17, 85]. The simplest method uses characteristic mapping, in which the SoP is defined in terms of the other states, such as SoC, SoE, and temperature. It does not consider operating history or actual operating conditions, and may require a large database to increase accuracy. More reliable estimation methods are largely based on ECM parameter identification. The hybrid-pulse power characterization (HPPC) method developed in [81], with peak current I_{HPPC} is given by

$$I_{HPPC} = \frac{V_{OC} - V_{min}}{R_0} \quad (1.8)$$

where V_{min} is the cutoff voltage. HPPC thus ignores polarization effects captured by the ECM RC pair and does not consider the change in voltage with SoC [44]. More advanced ECM definitions consider SoC and polarization voltage [45], or the variation of ECM parameters with SoC or SoE [46].

Other approaches involve multiple techniques, such as PF or KF, to estimate SoC, which is then used to predict SoP. In [47], a set of power, current, SoC, and voltage constraints are used to obtain an objective function which is minimized to predict SoP. The maximum

percent error for SoP is 6%. In [48], SoP is obtained from SoC using a look-up table. The table is generated from a fuzzy model of the polarization voltage varying with SoC and current. The proposed method is shown to have lower RMSE than the HPPC method, but it is uncertain whether the PF approach can be used for fast estimation. In [49], ECM methods are extended to a series of cells using ratio vectors of the cell parameters.

1.2.4 Joint estimation

Joint estimation is the subject of several studies, and has the potential to streamline BMS operations by eliminating interdependence between states. It may also increase accuracy.

Approaches for joint estimation are largely based on the techniques described above. In [50], the probability density function of OCV is used for parameter identification in a second-order ECM and SoE estimation. Using SoE, OCV, and temperature, the SoC is then estimated with maximum error of 0.015. At least 20 minutes of data is required, however, making this method impractical, for SoC estimation especially. In [51], the ratio between OCV decay over the discharge interval and discharge time is shown to be correlated with SoC and SoE. The method has low estimation time but very high error at low SoC. In [52], an electrochemical SPM is used to estimate SoC, SoE, and internal resistance. Like the SPM discussed above, model parameters are assumed to follow a look-up table, which lowers accuracy as the cells experience different conditions.

Kalman filters are applicable to joint estimation as well. In [53], SoC is estimated online using an extended KF, while SoE is estimated offline due to higher computational complexity. In [54], SoC is estimated at higher frequency than SoE using adaptive extended KF. Maximum error is 0.0316. An unscented KF is used in to estimate SoC and SoP.

1.3 Data-driven state estimation methods

Data-driven techniques treat the battery cell as a black-box system. No battery models are used. Rather, statistical techniques or machine learning are used to analyze data from cell processes. Typically data-driven techniques involve offline training before implementation in real applications. Whilst training and data collection may be time-consuming tasks, online estimation using data-driven methods is typically very fast, and has the potential for high accuracy [17, 20].

Neural networks (NN) and support vector machines (SVM) are popular statistical learning methods for multi-dimensional modelling [86]. Because of their ubiquity, NN and SVM fundamental concepts are discussed in more detail here. Other data-driven methods include fuzzy logic, grey relational analysis, Gaussian process regression, genetic algorithms, and particle-swarm optimization.

NN are derived from the processes between neurons in the human brain. They are widely used for classification and regression problems. Since NN are a key subject in this report, further discussion is in Chapter 3.

SVM are derived from classification algorithms, first proposed in [87] and extended to regression in [88]. SVM regression uses the concepts of an ‘epsilon tube’ defined by ε , slack variables ξ_i and $\hat{\xi}_i$, and kernel function $K()$ to generate a regression line at a test point x based on the i^{th} observed input x_i . The dual problem of SVM regression is given by

$$\max_{\beta} -\frac{1}{2} \sum_{i,j} \beta_i \beta_j K(x_i, x_j) + \sum_i \beta_i y_i - \sum_i \hat{\beta}_i \varepsilon \quad (1.9)$$

subject to

$$\sum_i \beta_i = 0, \quad \beta_i + C \geq 0, \quad \beta_i - C \leq 0, \quad (1.10)$$

for all $i = \{1 \dots N\}$, where C is the regularization parameter, y_i is the observed output, N is

total number of data points, and

$$\beta_i = C(\hat{\xi}_i - \xi_i), \quad \hat{\beta}_i = C(\hat{\xi}_i + \xi_i), \quad \xi_i \hat{\xi}_i = 0 \quad (1.11)$$

The regression line is given by

$$f(x) = \sum_i^N \beta_i K(x_i, x) + b \quad (1.12)$$

where b is the bias. In classification problems, SVM finds the classification boundary. The parameter ε controls the margin with which the boundary line separates the data, and the slack variables allow data to be within the margin, or be misclassified. The regularization parameter C controls how strict these constraints are. Small values of C , or soft-margin SVM, allow large margins and greater misclassification in the training set, which increases generalization. High values of C , or hard-margin SVM, enforce the constraints more strictly, but can lead to overfitting. Regression problems use the same concepts as classification, but the boundary function is the desired output, and not the classifications. Kernels are used for feature expansion to define the basis function of regression. The radial basis function, or Gaussian kernel, is commonly used and given by

$$K(x_1, x_2) = e^{-\frac{1}{2\sigma^2} \|x_1 - x_2\|^2} \quad (1.13)$$

where σ^2 is the variance of the curve.

1.3.1 State of Charge

NN were applied to SoC estimation in [55] and [56]. In [55], a set of drive cycles at various temperatures with known SoC are used for training and testing the the NN. To estimate SoC at a certain time, the voltage, temperature, average current, and average voltage are input to the FNN. The averages are calculated using the previous 50 to 400 time steps relative to

the present time step. MAE is 0.011 for 25°C, and 0.0217 for -20°C. While the approach may be promising, due to the inherently high variation of drive cycles it may be difficult to obtain sufficient training data to lower MAE below 0.01 for different conditions and usage patterns. In [56], a FNN is used to identify battery voltage using current, a polarization metric, temperature, and SoC measured using Coulomb-counting. The FNN output is then passed to an extended KF, which makes a final SoC prediction. Maximum error is 0.02. Their proposed method shows lower variation in temperature than direct estimation of SoC using a NN. It is uncertain, however, whether voltage, current, temperature, and polarization observations at a single time step are sufficient for NN training.

SVM for SoC estimation are used in [57] [58]. In [57], a SVM is used to estimate the battery voltage from SoC and current. The model is trained in real-time with a small number of initial values. The SVM model output is then fed to an adaptive unscented KF for SoC estimation. Absolute error is less than 2%, with computation time of several milliseconds. In [58], SVM regression is performed using several inputs measured during a drive cycle, such as current, temperature, heat dissipation, and power. The SVM is trained and tested using several sets of drive cycle data. While very high accuracy is achieved for certain datasets, accuracy is highly variable. This suggests insufficient training data or overfitting.

Fuzzy logic is used in [59, 60] for SoC estimation. Unlike Boolean logic, which has true or false, fuzzy logic can have any number of states between true and false. ‘Crisp’ outputs are obtained from the fuzzy states. In [59], fuzzy logic is applied to optimize the gain of a sliding mode observer, which is used to estimate SoC and polarization voltages in a second-order ECM. The MAE is 0.0082 for estimation at 25°C, but increases at other temperatures and usage conditions. Convergence time is also very long, at up to 1 hour. In [60], fuzzy logic is used to estimate pack SoC from cell voltages. Pack SoC is defined as the average SoC in a string of cells. The maximum error is 1%. While this may be useful for a pack, it does not yield information on the SoC variation amongst the cells.

1.3.2 State of Energy

NN have demonstrated high performance for SoE estimation [61, 62, 89]. In [61], a recurrent NN is used to estimate SoE and internal resistance using prior SoH values, SoC, current, and temperature as observed over a drive cycle. The RMSE for SoE is 0.02. Estimation accuracy is likely affected by measurement errors on the inputs, and SoC and SoE estimation remain interlinked. In [62], first-order ECM parameters and SoC are input to a NN to estimate SoE. The maximum error is 0.05. As before, the NN output is sensitive to measurement error on the input variables, which may reduce accuracy.

SVM are applied to SoE estimation in [63] and [64]. In [63], the voltage curves from 15 minutes of charging are used to train the SVM. SoE estimation error is below 3%. The 15 minute input interval can be anywhere along the curves. Still, this approach may be too slow for real-time estimation and restrictive in defining acceptable input data. In [64], voltage curves from charging are also used to train a SVM. Rather than an interval of time, however, a voltage interval is used. The curves from specific ranges of voltage are input to the SVM to yield maximum error of 0.0088. As before, the charging curve data lasts up to 20 minutes, and cannot be used for real-time estimation.

Grey system theory was first proposed in [90] and later extended to grey relational analysis (GRA) [91]. GRA uses reference sequences to assess a new input sequence. From this comparison, a grey relational coefficient (GRC) can be generated between each reference and input. This coefficient represents similarity between sequences. In [65], the GRC is calculated between training and testing sets of IC curves, as described in 1.2.2. This coefficient is then linked to SoE. Maximum error is 0.04. As noted above, IC curves depend on accurate SoC estimation and cannot be used in real-time estimation.

Gaussian process regression (GPR) is a Bayesian method for modelling complex data. The data function is assumed to follow a Gaussian process. Like SVM regression, GPR is non-parametric, and kernels are used to define the basis functions. GPR kernels represent

the covariance between each input observation, and are used for each prediction. This may be computationally expensive for large datasets. GPR with Gaussian kernels typically result in very smooth regression lines. GPR is used in [66] to perform long-term forecasting of SoE up to 200 days. The model is trained using the temperature, current, voltage, and SoE of varying load profiles. The model output is the change in SoE over a time step, rather than the actual SoE. Maximum error is 5%. This approach thus considers the usage history of the battery, which may be useful for assessing degradation. It is unlikely, however, that this model can be used in real time, due to the computational complexity of each iteration.

GPR and GRA are used in [67] to estimate SoE from features extracted from charging curves. Feature correlation with SoE is first quantified using GRA, then the 4 highest-correlated features are chosen for GPR. Estimation error is below 0.06. Not only do charging curves last up to 2 hours, but the model is likely to have high computational complexity.

1.3.3 State of Power

As noted above, methods for SoP estimation are scarce in literature compared SoC and SoE estimation. Data-driven methods are especially rare, largely because ECM-based estimation is the dominant approach. Since ECM parameter identification is highly researched, there is there appears to be less need to focus on SoP.

Data-driven methods to estimate SoP are used in [68] and [69]. In [68], a genetic algorithm is used to calculate SoP. First described in [92], genetic algorithms search for the optimal solution using principles inspired by natural selection. The genetic algorithm treats SoP estimation as an optimization problem subject to several constraints defined by current, SoC, OCV, and resistance. Convergence is achieved in less than 2 seconds, with performance shown to be more stable than conventional methods. The prediction is sensitive to measurement error in SoC and SoE, however. In [69], fuzzy logic is applied to SoP estimation for a battery pack using variations in SoE and SoC. A rough estimate of cell SoP is first obtained with model predictive control, then adjusted with fuzzy logic to produce a holistic pack SoP

estimate. This may be useful for basic BMS operations, but more advanced BMS would require cell-based estimation for real-time optimization.

1.3.4 Joint estimation

NN are used for joint estimation in [71] and [72]. In [71], a time-delay NN is used to estimate SoC and SoE. Time-delay NN are similar to FNN, but intervals of data are fed sequentially to the input layer. Within the hidden layers, each node output is linked to a certain time step. The final outputs are thus composed of several time steps of data. The time-delay NN in [71] uses voltage, current, and temperature to predict SoC and SoE at very high accuracy. The model is only validated with charging curves, so it is uncertain how effectively it can estimate states in real time. In [72], a recurrent NN is used to estimate SoC and SoE from first-order ECM parameters, voltage, and current. An adaptive extended KF is used estimate the parameters. Maximum percent error is 6%. This approach may strike a balance between computational complexity and speed, but error is too high.

A SVM is used in [70] to obtain the battery state space parameters, SoE, and a rough estimate of SoC. An unscented KF is then used to refine the SoC estimate. SoE is estimated with error of 0.02, and SoC with error of 0.004. Whilst the joint estimator may yield good results for SoC, error for SoE is one order of magnitude higher.

Particle swarm optimization (PSO), first described in [93] and [94], is used to estimate SoC and SoE in [73]. PSO was inspired by social behavior amongst living creatures such as birds or fish, whose movements as a group are hypothesized to be a result of interactions between individual group members and their neighbors. The PSO algorithm treats candidate solutions as particles with position, velocity, and inertia that move in the multi-dimensional data input space subject to individual optima and global optimum constraints. In [73], PSO is used to identify cell parameters and estimate SoE. SoC is estimated with a proportional-integral observer. SoE is estimated with maximum error of 0.02, within 12 min. SoC is estimated with maximum error below 0.01.

1.4 Outline

This report consists of 5 chapters. The introduction and literature review presented above are Chapter 1. Chapters 2-5 present the main work.

Chapter 2 details the data collection and data processing. Chapter 3 describes the design of the FNN. Chapter 4 presents and discusses results of the FNN for estimation of various states. The report is summarized and future work is described in Chapter 5.

Chapter 2

Data Collection and Processing

2.1 Data Collection

Data was collected using 3 lithium nickel cobalt aluminum (NCA) oxide cells (Samsung INR18650-30Q). NCA cells have desirable performance, but require additional safety considerations [95]. Cells are cycled simultaneously and under the same conditions to reduce the effects of individuality. Cell characteristics are summarized in Table 2.1.

Cells were cycled using the Neware BTS4000 series 5V6A cycler, held at 25°C and standard pressure. Cell voltage and cycler current are monitored at 10 Hz. Measurements are uploaded to a database for processing. Photographs of the experimental setup are shown in Figure 2.1.

Charge and discharge pulses are applied for 13 ages per cell and 6 SoC per age. The SoC

Table 2.1: Cell characteristics

Characteristic	Value
Cell chemistry	NCA
Nominal capacity [mAh]	3000
Cut-off voltage [V]	2.5
Cut-off current [mA]	150
Max charge voltage [V]	4.2
Peak charging current [A]	4
Peak discharge current [A]	15

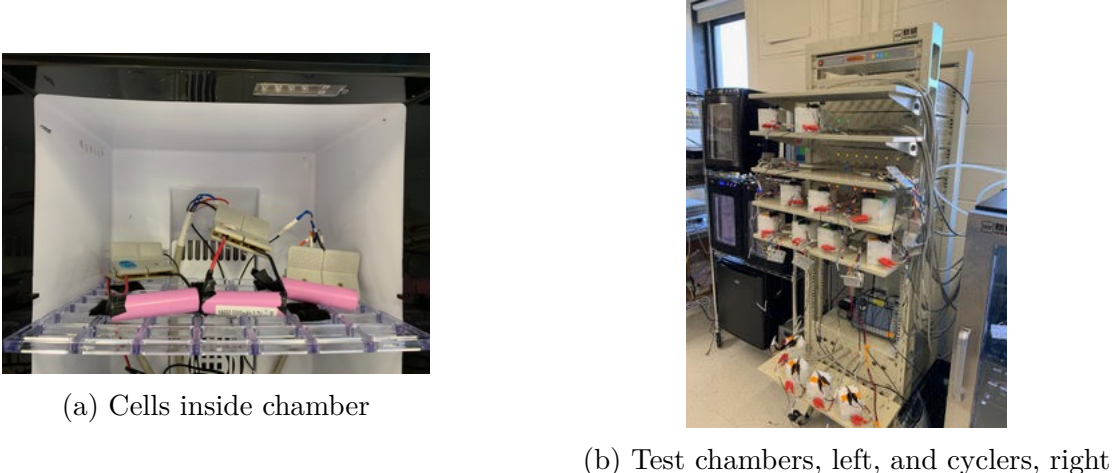


Figure 2.1: Experimental setup

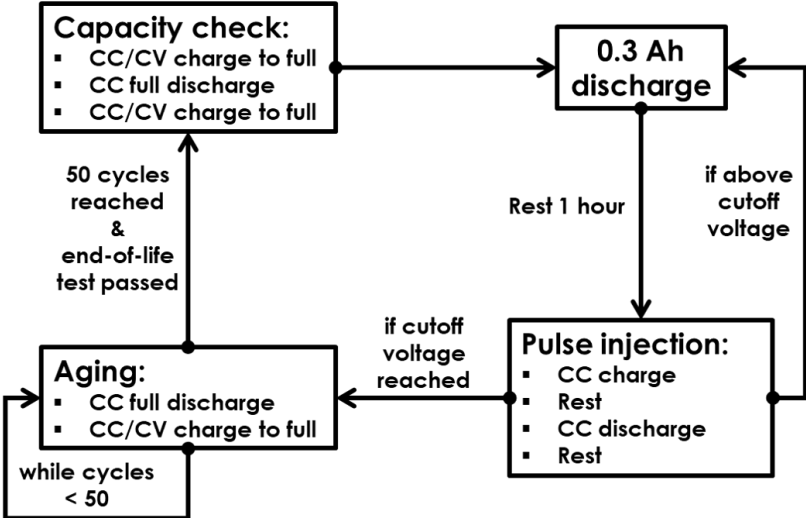


Figure 2.2: Testing procedure

range from 0.05 to 0.8 because safety functions of the battery cycler impose voltage limits that distort the pulse at higher levels of SoC. This yields 234 unique responses for training and testing the FNN.

Cycling procedure is illustrated in Figure 2.2. For each age, a capacity check was performed with a 0.1 C-rate CC discharge from full, lasting approximately 12 hours. Cells were then recharged using CC and constant voltage (CV) procedures, also lasting around 12 hours. From full charge, 0.3 Ah are discharged, then the cell is rested. After resting, pulses were applied at various SoC levels, until a 2.5V cut-off voltage was reached. Different SoC

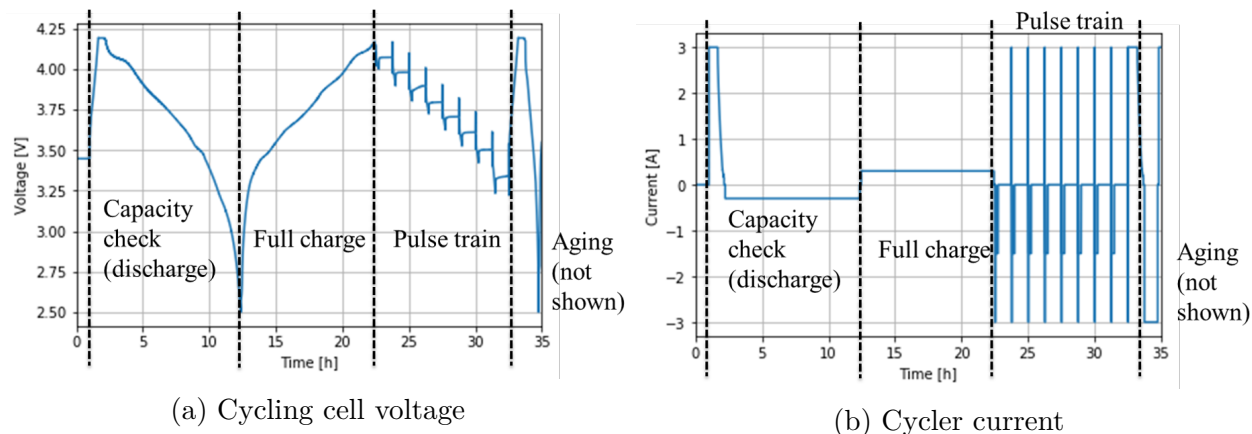


Figure 2.3: Cycling voltage and current

levels are achieved with a 0.3 Ah discharge, followed by a 1 hour rest period between each pulse. Aging was performed with 50 charge/discharge cycles at 1 C-rate, until the cell fails an ‘end-of-life’ test defined by the SoE. Voltage and current during the cycling protocols are shown in Figure 2.3.

The battery cycler assigns step numbers (step IDs) and cycle numbers to each action in the cycling protocol. Step IDs are limited by the number of unique actions within the protocol, and are used to identify the capacity check and pulse train periods. Cycle numbers accumulate throughout the entire test, and thus can be used to track aging. Both step IDs and cycle numbers are used for data extraction.

2.2 Pulse parameters and training data

A charge and discharge pulse is applied to obtain the voltage-time response of the cells, shown in Figure 2.4. Pulse amplitude is 1 C-rate, approximately 3A. There is a 30-second rest between charging and discharging. Each pulse is applied to the cell after a 1 hour rest period. Positive cycler current indicates cell charging.

This pulse shape represents the simplest possible configuration. It is used to establish a baseline for performance, and allow for further optimization of pulse shape. It also allows

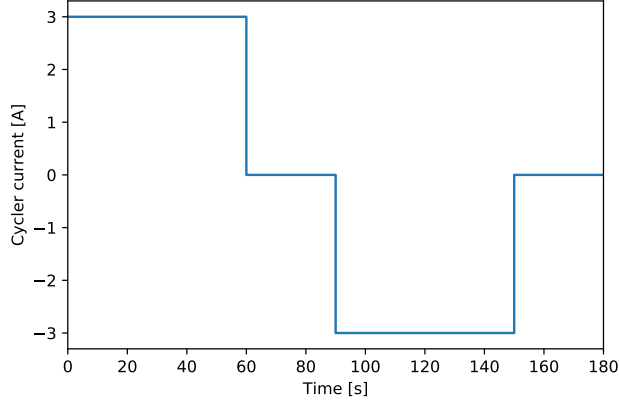


Figure 2.4: Current pulse from cyclor

for examination of individual portions. Pulse portions are defined by the names and approximate time ranges listed in Table 2.2. To extract portions from the full pulse, step IDs are used. Charge and discharge portions are isolated from each other, allowing their effects on prediction accuracy to be observed individually. The rest periods between pulses ensure that voltage responses are isolated from the effects of prior usage. Composite portions, Charge-rest and Discharge-rest, are also considered.

Table 2.2: Pulse portion definitions

Portion name	Time range [s]
Charge	[0, 60)
Rest 1	[60, 90)
Discharge	[90, 150)
Rest 2	[150, 180]
Charge-rest	[0, 90)
Discharge-rest	[90, 180]
Full pulse	[0, 180]

Fractions of the charge and discharge voltage responses, given by $\left\{\frac{1}{256}, \frac{1}{128}, \frac{1}{64}, \frac{1}{32}, \frac{1}{16}, \frac{1}{8}, \frac{1}{4}, \frac{1}{2}\right\}$, are used to further examine how prediction accuracy varies with pulse length. The fraction $\frac{1}{256}$ corresponds to a single data point, or 0.2s, which represents an OCV measurement. This can inform development of an optimal pulse length.

Voltage responses to the pulse are used to create training data. The 234 unique voltage

responses are displayed in Figure 2.5, color-coded by SoC. The portions of the voltage response are shown in Figure 2.6. The FNN requires its training data to have zero-mean, so the mean voltage curve of the entire data set is subtracted from each response before being input to the FNN for training. During testing, the same mean curve is subtracted from the test responses.

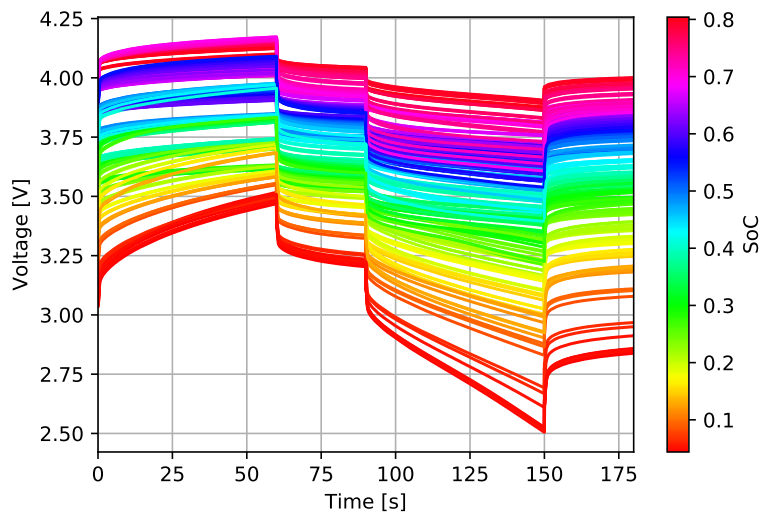


Figure 2.5: Voltage responses color-coded by SoC

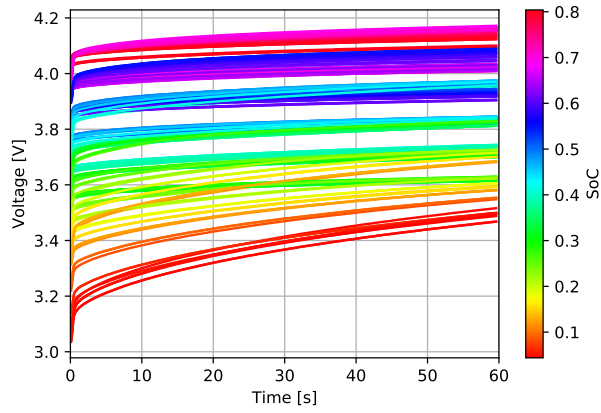
2.3 Data processing

The battery states as defined in section 1.1 are obtained by processing the voltage and current data. This yields the target data used in the FNN.

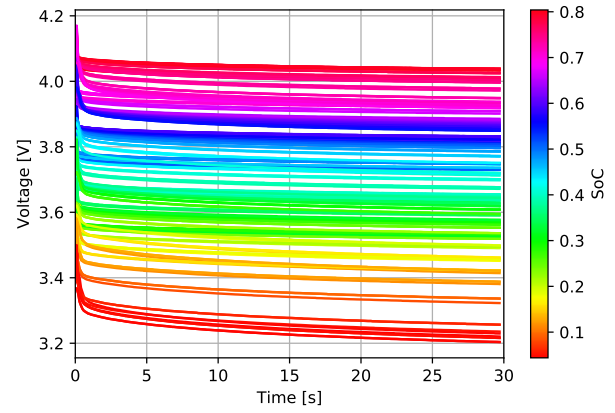
Cell maximum capacity Q_m is obtained from trapezoidal integration of the current-time curve over the capacity check period, represented by

$$Q_m(a) = \sum_{k=0}^{K_a} \left(\frac{I_k + I_{k+1}}{2} \right) \Delta t_k \quad (2.1)$$

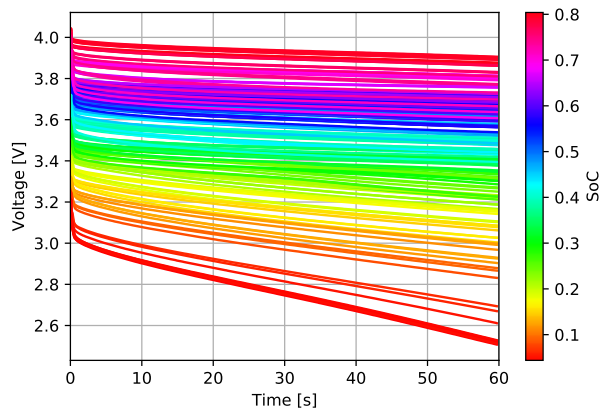
where $Q_m(a)$ is the maximum charge capacity of the cell at age a , k is the step number of the capacity check period, A is the maximum step number of the capacity check, I_k is the



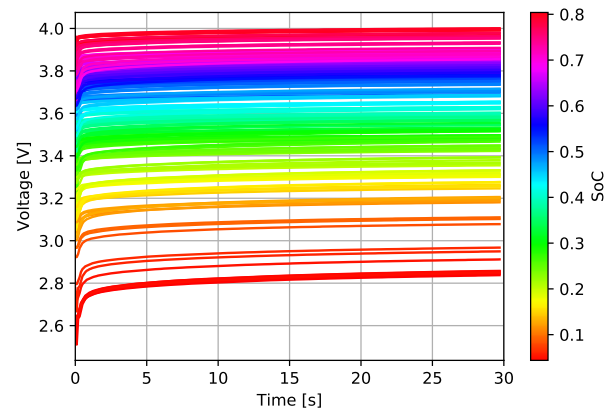
(a) Charge portion



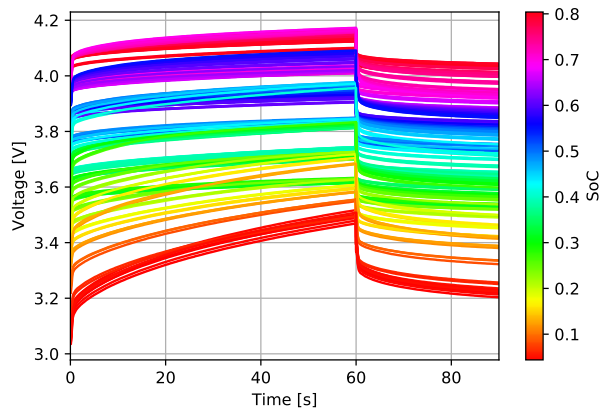
(b) Rest 1 portion



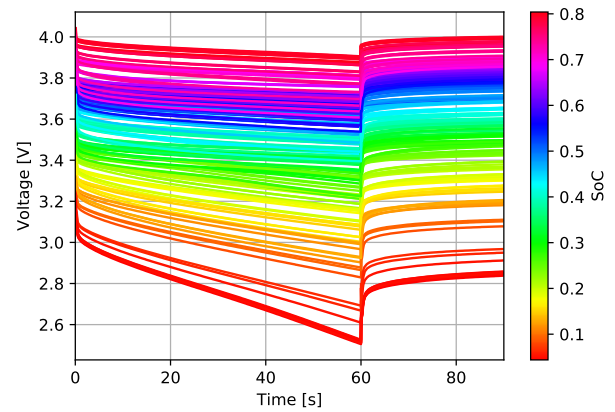
(c) Discharge portion



(d) Rest 2 portion



(e) Charge-rest portion



(f) Discharge-rest portion

Figure 2.6: Portions of voltage responses color-coded by SoC

current, and Δt_k is the time step, equal to 0.1 s.

Remaining charge capacity q is obtained similarly, but integration is performed from the end of the full-charge period to the beginning of pulse p , represented by

$$q_p(a) = Q_m(a) - \sum_{k=K_f(a)}^{K_p(a)} \left(\frac{I_k + I_{k+1}}{2} \right) \Delta t_k \quad (2.2)$$

where $K_f(a)$ is the step number of the beginning of the pulse train, and $K_p(a)$ is the step number of the beginning of pulse p . The values for $Q_m(a)$ and $q_p(a)$ are then used to obtain the target values for SoC and SoE, as defined in Subsection 1.1.1. The variation of SoE with age number for the 3 cells is shown in Figure 2.7.

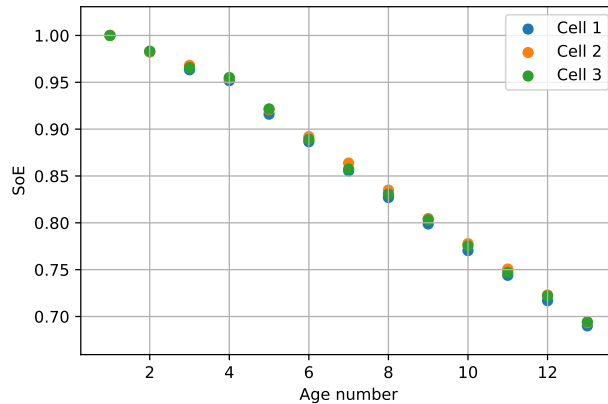


Figure 2.7: Cell SoE for three test cells

Series resistance R_0 , RC-pair resistance R_1 and capacitance C_1 are calculated from the voltage responses using the linear regression methods adapted from [96]. Open-circuit voltage V_{OC} is the voltage observed before the pulses are applied.

The ECM parameters are obtained from the Discharge and Rest 2 portions of the voltage response. The initial voltage drop ΔV at the beginning of discharge is used to calculate R_0 , given by

$$R_0 = \frac{\Delta V}{I} \quad (2.3)$$

where I is the discharge current, approximately 3A. Whilst the real-world process corre-

sponding to R_0 would occur almost instantaneously, to increase modelling accuracy ΔV is calculated over the first second.

During Rest 2, the voltage across the RC-pair V_1 is given by

$$V_1 = V_{OC} - V_o(t) = Ae^{-\frac{t}{\tau}} \quad (2.4)$$

where $V_o(t)$ is the cell voltage at the terminal, and A is some constant. To obtain $-\frac{1}{\tau}$, linear regression is performed using the logarithm of the equation, given by

$$\ln(V_{OC} - V_o) = \ln A - \frac{t}{\tau} \quad (2.5)$$

The least-squares solution is obtained with the Moore-Penrose inverse, given by

$$\theta = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y} \quad (2.6)$$

$$\theta = \begin{pmatrix} \ln A \\ -\frac{1}{\tau} \end{pmatrix}, \quad \mathbf{X} = \begin{pmatrix} 1 & t \\ \vdots & \vdots \end{pmatrix}, \quad \mathbf{y} = \begin{pmatrix} \ln(V_{OC} - V_o) \\ \vdots \end{pmatrix} \quad (2.7)$$

To decouple R_1 and C_1 , the Discharge portion is used. During discharge,

$$V_1 = V_{OC} - V_o(t) = IR_1 \left(1 - e^{-\frac{t}{R_1 C_1}}\right) \quad (2.8)$$

Thus we have

$$R_1 = \frac{V_{OC} - V_o(t = T_{dis})}{I \left(1 - e^{-\frac{T_{dis}}{R_1 C_1}}\right)} \quad (2.9)$$

where T_{dis} is the length of the discharge period. Capacitance C_1 is easily calculated once R_1 is obtained using

$$C_1 = \frac{\tau}{R_1} \quad (2.10)$$

It is noted that ECM parameters have different values during charge and discharge, but only

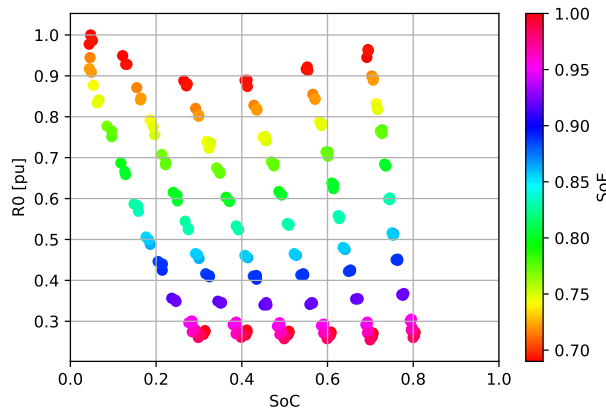
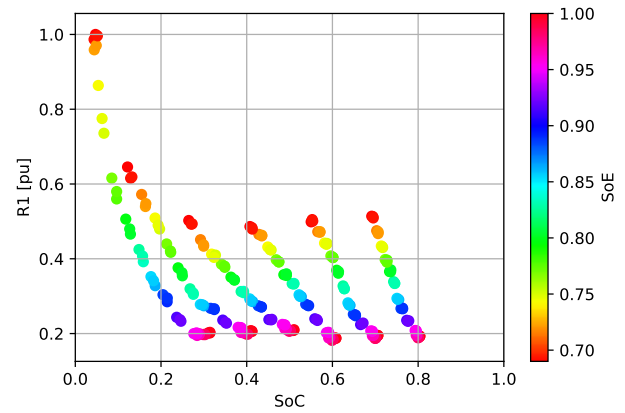
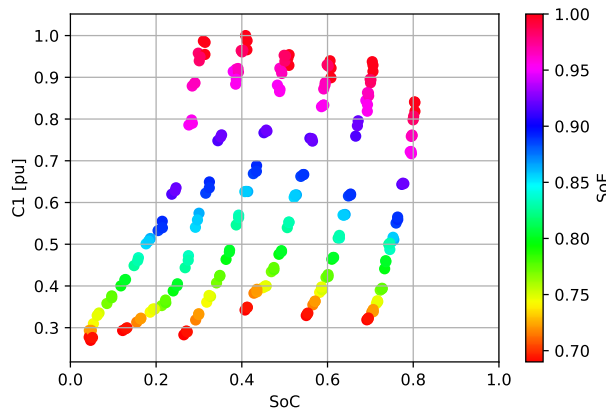
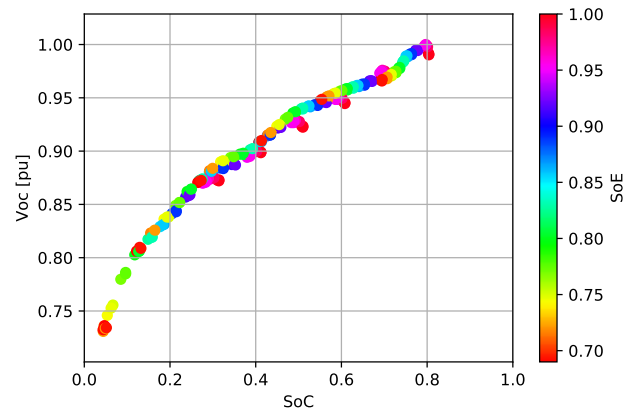
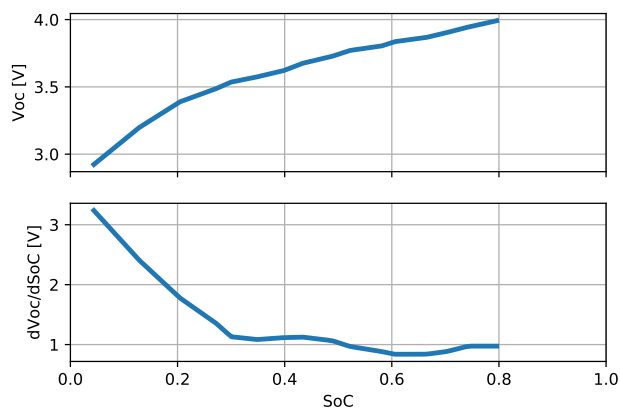
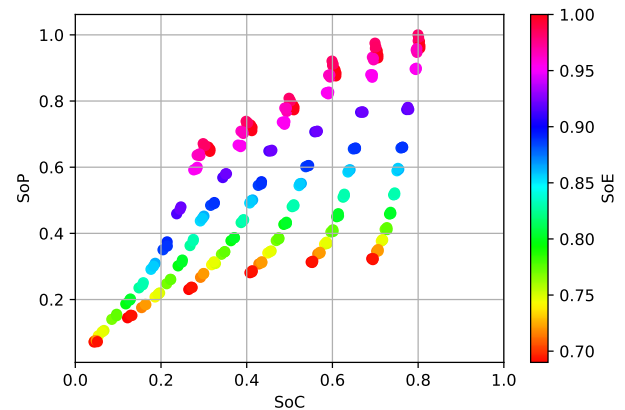
the discharge values are considered here.

There are several methods used to calculate current I_{pk} . To increase the fidelity of SoP measurement, I_{pk} is calculated instead with the model-based dynamic multi-parameter method [45], given by

$$I_{pk}(Q_m, SoC) = \frac{V_{OC} - V_{min}}{\frac{\eta \Delta t}{Q_m} \frac{\partial V_{OC}}{\partial SoC} + R_1 \left(1 - e^{-\frac{\Delta t}{R_1 C_1}}\right) + R_0} \quad (2.11)$$

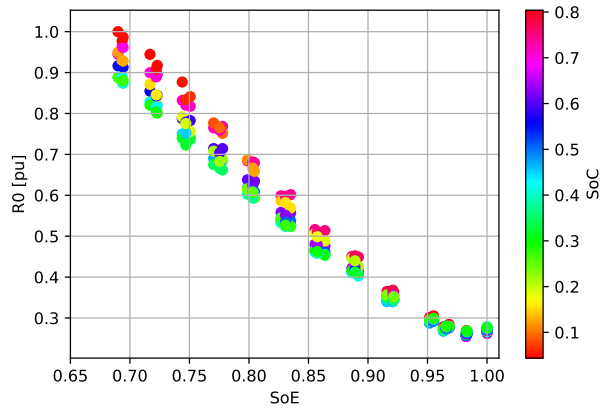
where V_{min} is 2.5 V, as shown in Table 2.1, η is coulomb efficiency, assumed constant at 99%, Δt is the time horizon, set to 60s, and $\frac{\partial V_{OC}}{\partial SoC}$ is the derivative of the V_{OC} vs SoC curve. Peak current depends on both the maximum available capacity and the SoC. With the 60s time horizon, SoP thus represents the LIB cell's peak power capabilities over the next minute. Values for $\frac{\partial V_{OC}}{\partial SoC}$ are obtained using a look-up table function created from the observed values of $V_{OC}(SoC)$.

Evolution of SoP and ECM parameters (R_0 , R_1 , C_1 , V_{OC} , and $\frac{\partial V_{OC}}{\partial SoC}$) with SoC and SoE is shown in Figures 2.8 and 2.9. To achieve uniformity in display, ECM parameters are normalized such that they range from 0 to 1. Resistances R_0 and R_1 are normalized by dividing by the maximum recorded value, representing the highest acceptable resistance before failure. Capacitance C_1 and voltage V_{OC} are normalized similarly, but the maximum value instead corresponds to the 'fresh cell'. Data smoothing of $V_{OC}(SoC)$ and $\frac{\partial V_{OC}}{\partial SoC}$ is performed with the Savitzky-Golay filter, often used when differentiating noisy data [97].

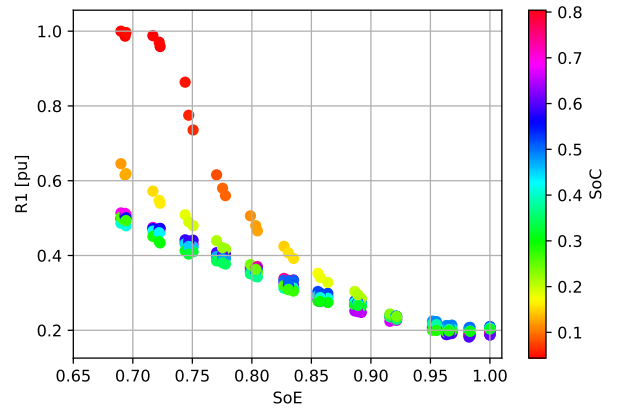
(a) Normalized resistance R_0 (b) Normalized resistance R_1 (c) Normalized capacitance C_1 (d) Normalized voltage V_{OC} (e) Smoothed V_{OC} and $\frac{\partial V_{OC}}{\partial SoC}$ curves

(f) State of Power SoP

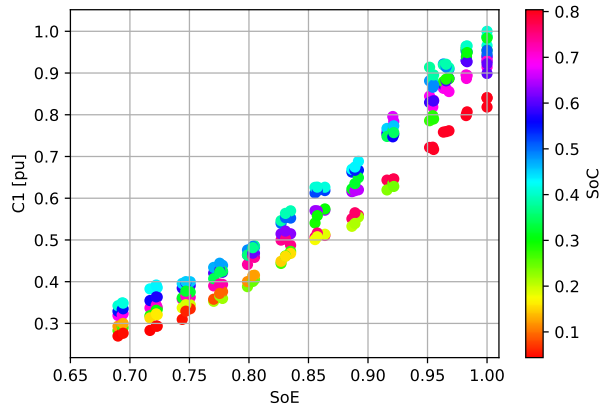
Figure 2.8: SoP and ECM parameter evolution with SoC



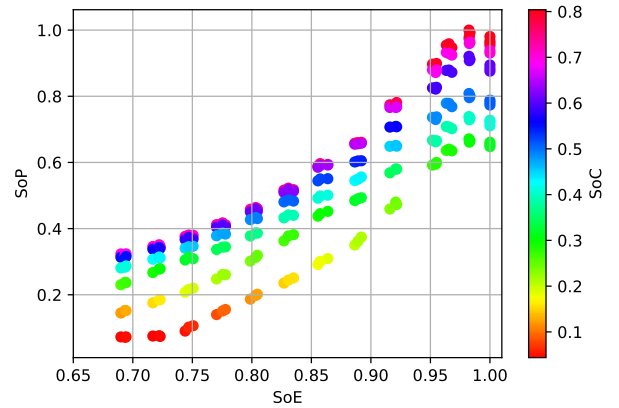
(a) Normalized resistance R_0



(b) Normalized resistance R_1



(c) Normalized capacitance C_1



(d) State of Power SoP

Figure 2.9: SoP and ECM parameter evolution with SoE

Chapter 3

Neural Network Design

3.1 Theory

NN were inspired by connections between neurons in the human brain, and NN as we know them today began to take shape in the 1960s after research into a cat's visual cortex [98]. Both FNN and recurrent NN (RNN) have demonstrated high performance in classification and regression problems.

NN are formed from several 'hidden layers' of interconnected nodes. In FNN, exemplified in Figure 3.1, node connections are only allowed to travel from input to output. In RNN, connections can be made in any direction, and loops between nodes can be formed. RNN

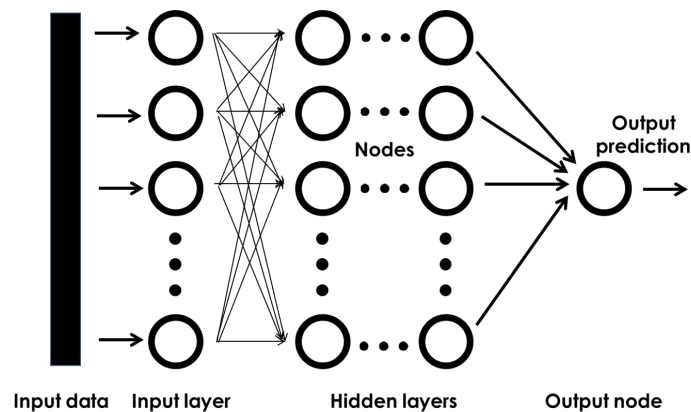


Figure 3.1: Diagram of general FNN structure

are often used for sequential data, which is useful for time-dependent predictions. Since the proposed pulse perturbation method does not require prior knowledge or vary over time, a FNN is used for state estimation.

For a FNN to learn the optimal weights between nodes, the model must be trained. Model training is supervised, meaning that each training input has a corresponding target output. After training, the FNN model is a network of matrices that accepts an input of the specified length and makes a prediction. This can be performed quickly, making FNN an attractive alternative for mapping voltage responses to the corresponding state. Since network weights have no relation to real-world parameters, FNN are considered black-box systems.

Model training is performed offline and measured in epochs (training cycles). Training data and target outputs are used. Within each epoch, there is one forward and backward pass. Nodes within the input and hidden layers are linked by network weights, and generate outputs based on the activation function. Node outputs are governed by

$$h_{n,i} = \phi\left(\sum_i^N (w_{n,i,j}h_{n-1,i} + b_{n,i})\right) \quad (3.1)$$

where $h_{n,i}$ is the node i in layer n , N is the number of layers (the input layer is not counted), $\phi()$ is the activation function, $w_{n,i,j}$ is the network weight between node i in layer n and some node j in layer $n - 1$, and $b_{n,i}$ is the bias of node i in layer n . Calculations are ‘passed forward’ through the layers and modified by the weights until the output node. For a single-output FNN, the prediction is represented by h_N . This prediction is ‘passed backward’ to be compared with the target value of the training data. The weights are then readjusted based on the optimizer function and batch size of the training data.

3.2 Implementation

The Keras module of Tensorflow software [99], was used to implement the FNN. Keras is a high-level application programming interface (API) based on Tensorflow that uses layers and models to build neural networks. A simple sequential Keras model is used in this report, meaning that the layers are linearly stacked.

FNN training is configured to minimize the mean squared error (MSE), but MAE is used here for comparison with other studies. Due to random variation of error, 20 trials were performed for each combination of state and input data. MAE varies for each trial because the training and testing data are randomly split 80/20, meaning that 80% of the entire dataset is randomly chosen for training, while 20% of the set is withheld for testing. This means that the training and testing subsets are different for each trial. The training subset is further split 80/20 for validation, meaning that FNN performance is evaluated after each training epoch using 20% of the training subset. Performance variation is also caused by the dropout layer, discussed below. Since the available dataset is relatively small, there may be more variation in MAE among each trial.

There are no universally-accepted rules for choosing FNN hyperparameters. For this reason a discussion on the chosen values is presented here. A summary of the chosen values is listed in Table 3.1. The same hyperparameters were used for estimation of each state, except the number of nodes per hidden layer. The effect of node number on estimation accuracy is explored in the following chapter.

Network structure is formed from an input layer and 2 hidden layers. To choose the node number, a sweep of nodes per hidden layer, from 16 to 256, is performed. Using two hidden layers is a reasonable choice that may reduce the total number of nodes required [100]. The optimal number of nodes is related to the complexity of the underlying data function, and thus may vary for the different states. No algorithms are used to optimize network structure, which is beyond the scope of this report.

Table 3.1: Shared hyperparameters amongst FNN models

Hyperparameter	Value
Hidden layers	2
Network weight constraint	100
Dropout rate	0.01
Learning rate	0.001
Batch size	32
Training epochs	32000
Activation function	relu
Optimizer	RMSprop

Dropout layers are sandwiched between the hidden layers. Dropouts are used to prevent convergence to local minima in the optimization process. Nodes are randomly ‘dropped out’ from the network within each epoch at the dropout rate. Larger dropout rates, together with low network weight constraints and higher learning rate may prevent overfitting [101]. Since overfitting was not observed within the training period, the dropout rate was set to a low value of 1%, network weight constraint was set high to 100, and learning rate was unchanged from default at 0.001.

Batch size refers to the number of samples evaluated by the model before network weights are updated. It shown in [102, 103] that small batch sizes can improve accuracy and convergence stability. Since the available number of samples is only 234, a batch size of 32 is adequate.

The activation function determines the node output. Non-linear functions allow FNNs to model complex data, but linearity aids in model optimization [104]. For this reason the rectified linear unit (relu) activation function is used, given simply by

$$\phi_{relu}(x) = \max(0, x) \quad (3.2)$$

Network weight optimization is performed most commonly with gradient descent algorithms. RMSprop is one such method that uses an adaptive learning rate approach. Weights

are updated using the rule

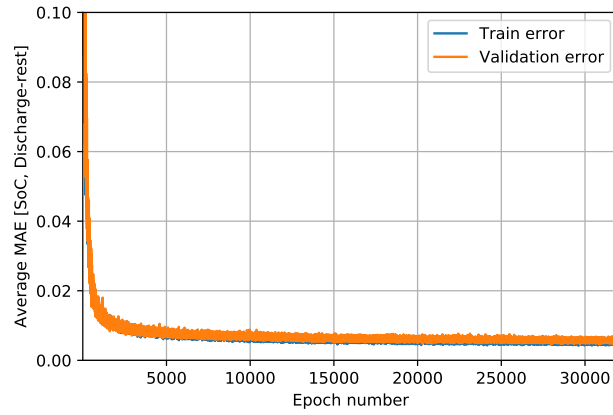
$$w_t = w_{t-1} - \frac{\eta}{\sqrt{E[g^2]_t + \epsilon}} g_t \quad (3.3)$$

where η is the learning rate, g_t are the gradients at step t , ϵ is a small value used to prevent division by 0, and $E[g^2]$ is the running average of the squared gradients, given by

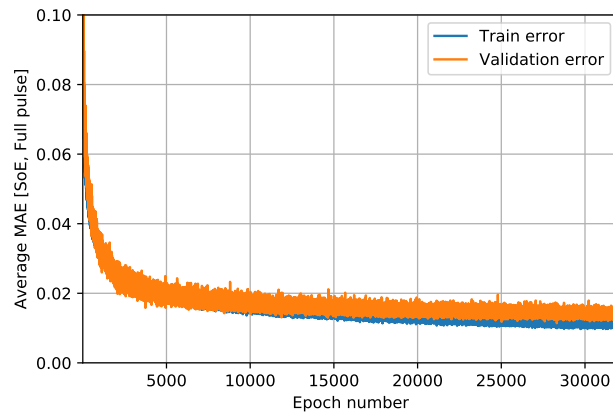
$$E[g^2]_t = 0.9E[g^2]_{t-1} + 0.1g_t^2 \quad (3.4)$$

Training was performed using 32000 epochs for all states and input data using a TITAN Xp NVIDIA graphical processing unit. Each training and evaluation cycle requires approximately 20 seconds, subject to variation with batch size and node number.

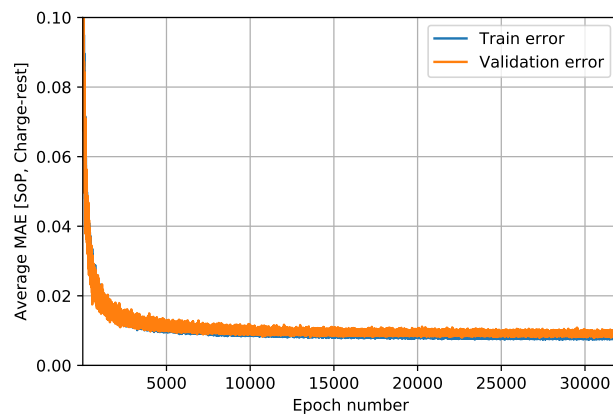
Evolution of the FNN training objective function with training epochs is shown in Figure 3.2 for the various states and different pulse portions. The curves show the average training MAE from all 20 trials. It can be seen in Figure 3.2b that validation error begins diverging from training error at around 15000 epochs. This suggests that the model error may not improve with additional training, and even increase. Since model training is a time-consuming task, excessive training should be avoided. For the other states and portions, validation error does not diverge from training error, but beyond 25000 epochs, error no longer decreases. This may suggest that larger sets of input data require additional training epochs to reach the best performance. Further verification is needed to determine the optimal number of training epochs.



(a) State of charge SoC, Discharge-rest portion



(b) State of energy SoE, Full pulse



(c) State of power SoP, Charge-rest portion

Figure 3.2: Samples of FNN training for different states

Chapter 4

Results and Discussion

Results and discussion are presented for the full pulse in Section 4.1, and pulse portions and fractions in Section 4.2.

4.1 Full pulse

The MAE achieved by the trained FNN for varying nodes are shown in Figure 4.1. Sample plots of the FNN predictions are shown in Figure 4.2. Node number should be reduced to minimize the amount of memory required when uploaded for use on a microcontroller and to minimize computation time. Thus the optimum node number must balance network size with error.

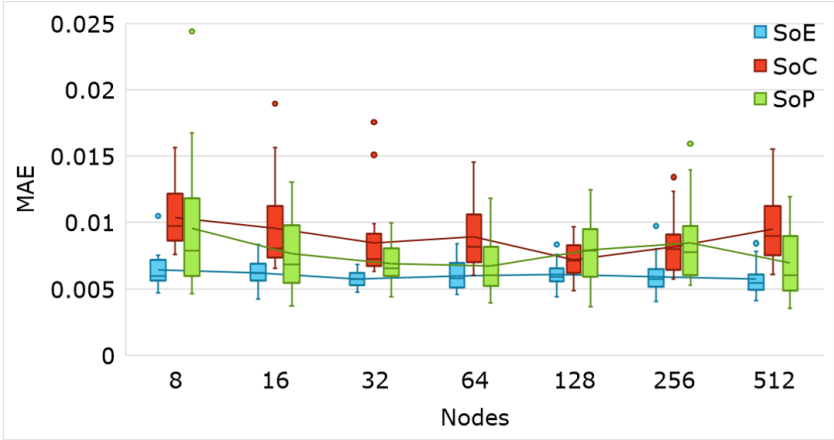
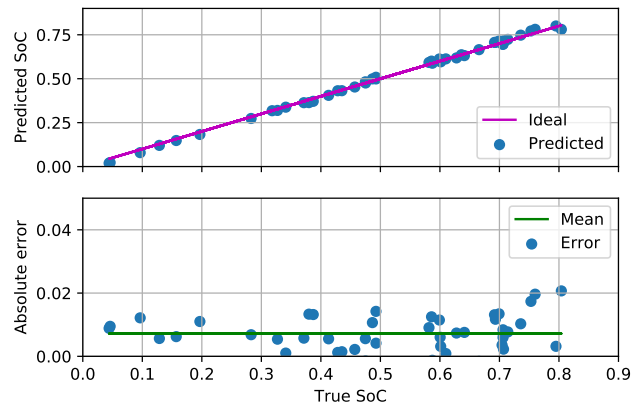
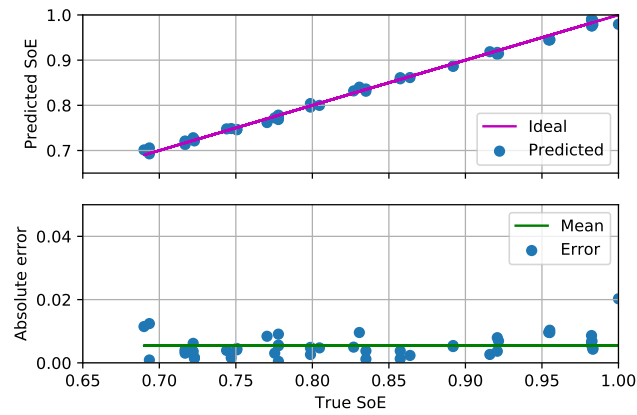


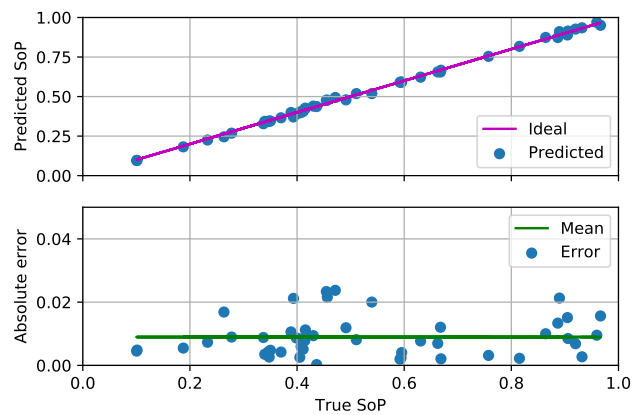
Figure 4.1: Box plot of MAE against nodes per hidden layer for state estimations



(a) State of charge SoC



(b) State of energy SoE



(c) State of power SoP

Figure 4.2: Samples of FNN predictions for different states

For SoE, MAE decreases until 32 nodes. After this, MAE increases and the error ranges increase. Thus 32 was chosen as node number for SoE. For SoC, there is a clear decrease in MAE until 128 nodes, and visible increase in MAE afterwards. Thus 128 was chosen as node number for SoC. For SoP, MAE follows a similar trend to SoE, though the increase in MAE after 32 nodes is clearer. Thus 32 was chosen as node number for SoP estimation. Above the chosen values, the MAE is seen to have a wider range, which suggests that overfitting may be occurring.

Prediction results using the chosen node numbers are presented together in Figure 4.3. It can be seen that the FNN predicts SoE and SoC with better accuracy than SoP, achieving average MAE of 0.0057 for SoE estimation, 0.0072 for SoC estimation, and 0.0069 for SoP estimation.

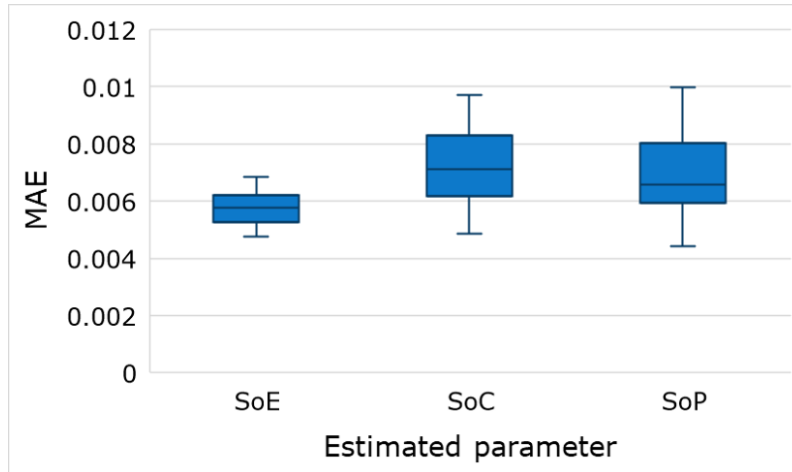


Figure 4.3: Box plot of MAE results for full pulse using chosen nodes

Variation in error ranges may be due to the ‘one-size fits all’ approach for choosing hyperparameters, except node number. Error may be reduced by assigning a unique set of hyperparameters for estimating each state. There is also uncertainty on the target data. Coulomb efficiency and OCV, for example, are approximated from the cycling data, thus affecting the target values of SoC and SoP. ECM parameter identification may also show processing error. Thus FNN estimation error may not necessarily reflect FNN accuracy, but rather data uncertainty.

Table 4.1 compares accuracy and estimation time of selected studies with the results achieved here. Parameters are defined using the conventions in this article. Mean error range refers to the minimum and maximum reported mean values, if more than one value is reported. Error metrics include MAE and RMSE. Prediction time is defined as the minimum time needed to collect input data for the estimator and generate a prediction, as reported in the study.

It is evident from Table 4.1 that the method proposed in this article is competitive against a wide range of other methods, especially for SoE estimation. FNN with pulse perturbation also strikes a balance between accuracy and speed, in addition to having high generality.

Table 4.1: Comparison of parameter estimation in selected studies

Method	Estimated parameters	Mean error range	Prediction time
FNN with pulse perturbation [<i>q.v.</i>]	SoE, SoC, SoP	[0.0057, 0.0072] MAE	3 min
Recurrent NN with pulse perturbation [61]	SoE, ($R_0 + R_1$)	[0.013, 0.021] RMSE	3 min
Regression with voltage drop analysis [51]	SoE, SoC	[0.015, 0.052] RMSE	1 min
Probability density function [50]	SoE, SoC	[0.003, 0.015] MAE	3 h
Grey relational analysis with incremental capacity curves [65]	SoE	[0.006, -] MAE	46.7 min
Incremental capacity analysis [35]	SoE	[0.0108, -] MAE	6 min
FNN with drivecycle data [55]	SoC	[0.0061, 0.016] MAE	1.67 h
Hybrid Kalman and particle filter [31]	SoC	[0.0048, 0.01] RMSE	>2 s

4.2 Pulse portions and fractions

The MAE achieved by the FNN for portions of the full pulse are shown Figure 4.4. It can be seen that Charge, Rest 1, Discharge, and Rest 2 have the highest error. Charge-rest

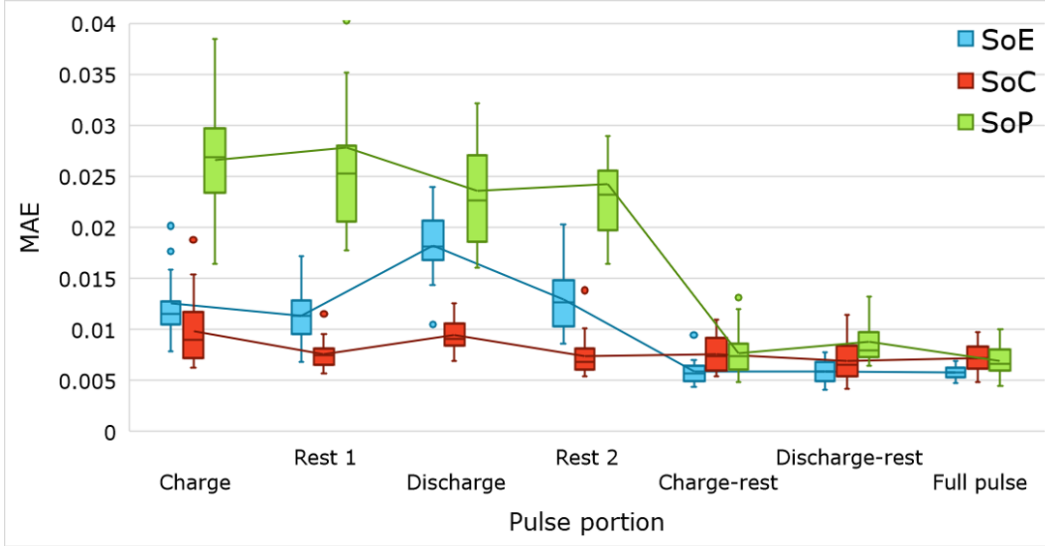


Figure 4.4: Box plots of MAE results for pulse portions

and Discharge-rest perform slightly worse than the full pulse, but no significant drop in performance is observed. For SoE, Charge and Rest 1 perform better than Discharge and Rest 2. For SoC, the rest periods perform better than the charging periods. For SoP, the individual portions yield much higher error than the composite portions or the full pulse.

Results for pulse portions suggest that a full pulse is not needed to obtain good results. Charge-rest, for example, has comparable performance to the full pulse, but is one-half the length. This is valuable information for future pulse design because a reduced pulse length is easier to implement. Results also indicate that different portions contain different state information. Rest periods, for example, appear to contain more SoC information than Charge or Discharge. A new perturbation could take advantage of this by reducing the length of the charge or discharge period, and increasing the length of the rest period.

Composite portions yield better performance than individual portions, which validates the hypothesis that the FNN pulse-perturbation method can yield accurate state estimation.

Results for estimation using fractions of the Charge and Discharge portions are presented in figures 4.5 and 4.6. Full pulse results are displayed for reference. In both figures, MAE decreases exponentially from 0.2s to approximately 0.9s. After 0.9s, MAE does not decrease, and even increases. Full pulse results still have the lowest MAE. The trend for SoC shows

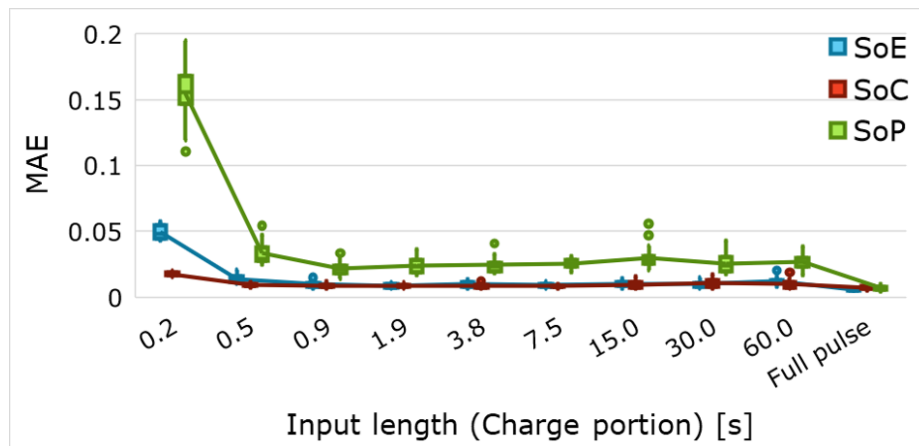


Figure 4.5: Box plots of MAE results for fractions of charge pulse

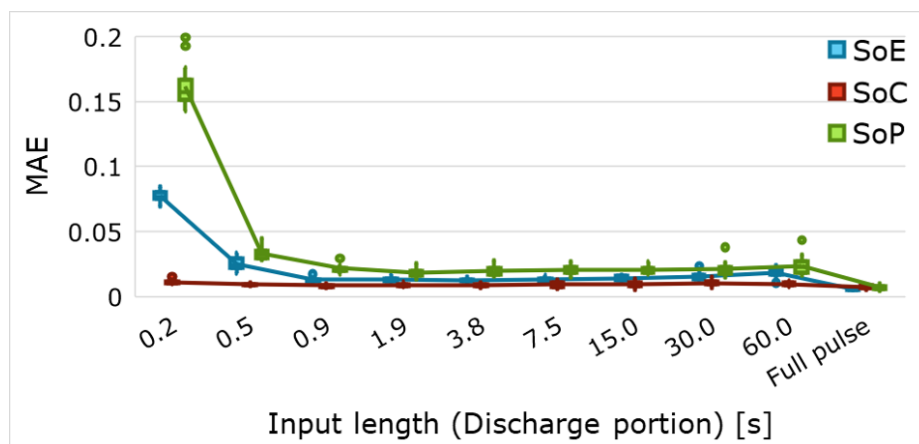


Figure 4.6: Box plots of MAE results for fractions of discharge pulse

much less reduction in error from 0.2s to 60s, especially for the Discharge portion (Figure 4.6).

The pulse fraction results confirm that using pulse perturbation is superior to using a single OC voltage measurement. The results also suggest that pulse length can and should be reduced to approximately 2s. This implies that the transient response of the pulse contains more relevant information than the steady-state response. After 2s, any additional voltage response appears to add noise to the FNN input signal. The full pulse may have lower MAE, but this is likely because of the rest portions. Further investigation is necessary to observe the effect of a full pulse with shortened Charge and Discharge portions.

Chapter 5

Conclusion and Future Work

LIB state estimation using machine learning was shown to yield fast and accurate results. Using data from a battery cycler, the FNN model uses the voltage response to a short current pulse to predict the cell states for various SoE, SoC, and SoP under constant temperature. A similar approach is used for analyzing portions of the full pulse, in addition to fractions of the Charge and Discharge portions of the pulse. It is shown that pulse perturbation is a promising technique with opportunities for further optimization.

The following sections present areas for future work. Verification of the results in the real world with respect to BMS for EVs is discussed in Section 5.1. The report is concluded in Section 5.2 with discussion of further research and outlook of the proposed method.

5.1 Real-world verification

High estimation speed and accuracy make the FNN suitable for EV applications. For real-world validation, the performance of BMS using a conventional state estimation technique and the performance of BMS using pulse perturbation and FNN must be compared. Metrics for measuring performance include prediction accuracy and time. For real-world verification, the technique can be implemented using the balancing circuit, or other features on the EV. Scheduling of pulse injection will be necessary based on the application.

An active balancing circuit could inject pulses during the load profile to avoid interfering with operation. This would require the pulse to be much lower amplitude, and perhaps shorter length. To examine this hypothesis, new pulses must be designed with different ‘base’ states. The pulse explored in this report assumes the cell is perturbed from rest, but future research must consider the effects of perturbing a cell being discharged at various C-rates.

Alternatively, if the pulse length is reduced sufficiently, the high-level BMS controller could inject the pulse at short breaks in the drive cycle – when the vehicle is stopped, for example. Pulse shape can be optimized to further improve performance. Unipolar-perturbation, such as a pure discharge-rest pulse, could decrease estimation time even further. It could also facilitate real-time implementation because it is difficult to obtain a charge-rest characteristic while the EV is at rest.

5.2 Outlook

There several directions for further research. The currently available dataset was obtained using no prior usage. A new cycling protocol can be designed to investigate the effect of prior usage on estimation accuracy. This would involve subjecting the cells to various types of driving cycles before the pulse train. Cells can also be cycled at various temperatures to ensure the proposed method is valid for different conditions.

The proposed technique can also be expanded to accommodate multiple cells within a pack. Just as a single cell’s voltage response yields information on its states, the response of a series of cells may yield information on the states of the entire series. With further research, perhaps inspired by transmission line theory, it may be possible to deduce the specific states of a single cell from the response of the series.

The pulse perturbation and FNN estimation method is widely applicable to other battery characteristics. Individual ECM parameters, for example, could be estimated instead of SoP

for comparison with the direct SoP prediction. Further study could optimize the FNN for estimation of a variety of parameters, such as degradation mechanisms, or polarization mechanisms.

Degradation mechanisms, such as SEI growth or active material loss, are likely to leave distinct ‘signatures’ in response to a perturbation. These characteristics may be difficult to identify using conventional methods, but relatively straightforward for FNN. To validate this hypothesis, new protocols for cell cycling must be designed to incite observable and distinct degradation in the cells. This may involve cycling at extreme temperatures or C-rates. The FNN can then be trained for performing classification of the mechanisms or regression of mechanism metrics.

There are many promising uses of pulse perturbation and FNN. Voltage responses from LIBs encode a significant amount of information about the internal cell parameters that FNN excel at extracting. Exploring the full potential of FNN and pulse perturbation could allow for cheaper and more reliable cell diagnostics in future battery applications.

References

- [1] IPCC, “Climate change 2014: Synthesis report,” *Intergovernmental Panel on Climate Change (IPCC)*, pp. 56–73, 2014.
- [2] IEA, “Energy, climate change and environment: 2016 insights,” *International Energy Agency (IEA)*, 2016.
- [3] —, “World energy outlook 2016,” *International Energy Agency (IEA)*, 2016.
- [4] D. W. Gao, “Energy storage for sustainable microgrid,” *Elsevier Ltd*, 2015.
- [5] Z. P. Cano, D. Banham, S. Ye, A. Hintennach, J. Lu, M. Fowler, and Z. Chen, “Batteries and fuel cells for emerging electric vehicle markets,” *Nature Energy*, vol. 3, pp. 279–289, 2018.
- [6] IEA, “Global EV outlook 2020,” *International Energy Agency (IEA)*, 2020.
- [7] N. Belmonte, V. Girgenti, P. Florian, C. Peano, C. Luetto, P. Rizzi, and M. Baricco, “A comparison of energy storage from renewable sources through batteries and fuel cells: A case study in Turin, Italy,” *International Journal of Hydrogen Energy*, vol. 41, pp. 21 427–21 438, 2016.
- [8] T. Kim, W. Song, D. Son, L. K. Ono, and Y. Qi, “Lithium-ion batteries: Outlook on present, future, and hybridized technologies,” *Journal of Materials Chemistry A*, vol. 7, pp. 2942–2964, 2019.
- [9] N. A. Chaturvedi, R. Klein, J. Christensena, J. Ahmed, and A. Kojic, “Algorithms for advanced battery-management systems,” *IEEE Control Systems Magazine*, Jun. 2019.
- [10] H. Rahimi-Eichi, U. Ojha, F. Baronti, and M. Chow, “Battery management system: An overview of its application in the smart grid and electric vehicles,” *IEEE Industrial Electronics Magazine*, June 2013.

- [11] L. Lu, X. Han, J. Li, J. Hua, and M. Ouyang, “A review on the key issues for lithium-ion battery management in electric vehicles,” *Journal of Power Sources*, vol. 226, pp. 272–288, 2013.
- [12] W. Waag, C. Fleischer, and D. U. Sauer, “Critical review of the methods for monitoring of lithium-ion batteries in electric and hybrid vehicles,” *Journal of Power Sources*, vol. 258, pp. 321–339, 2014.
- [13] W. C. Lee, D. Drury, and P. Mellor, “Comparison of passive cell balancing and active cell balancing for automotive batteries,” *2011 IEEE Vehicle Power and Propulsion Conference*, 2011.
- [14] D. F. Frost and D. A. Howey, “Completely decentralized active balancing battery management system,” *IEEE Transactions on Power Electronics*, vol. 33, no. 1, 2018.
- [15] A. Kelkar, Y. Dasari, and S. S. Williamson, “A comprehensive review of power electronics enabled active battery cell balancing for smart energy management,” *2020 IEEE International Conference on Power Electronics, Smart Grid and Renewable Energy*, 2020.
- [16] M. H. Lipu, M. A. Hannan, A. Hussain, M. M. Hoque, P. J. Ker, M. H. M. Saad, and A. Ayob, “A review of state of health and remaining useful life estimation methods for lithium-ion battery in electric vehicles: Challenges and recommendations,” *Journal of Cleaner Production*, vol. 205, pp. 115–133, 2018.
- [17] X. Hu, F. Feng, K. Liu, L. Zhang, J. Xie, and B. Liu, “State estimation for advanced battery management: Key challenges and future trends,” *Renewable and Sustainable Energy Reviews*, vol. 114, no. 109334, 2019.
- [18] M. U. Ali, A. Zafar, S. H. Nengroo, S. Hussain, M. J. Alvi, and H. J. Kim, “Towards a smarter battery management system for electric vehicle applications: A critical review of lithium-ion battery state of charge estimation,” *Energies*, vol. 12, no. 446, 2019.
- [19] M. Zhang and X. Fan, “Review on the state of charge estimation methods for electric vehicle battery,” *World Electric Vehicle Journal*, vol. 11, no. 23, 2020.
- [20] D. N. How, M. A. Hannan, M. S. Lipu, and P. J. Ker, “State of charge estimation for lithium-ion batteries using model-based and data-driven methods: A review,” *IEEE Access*, vol. 7, no. 19, pp. 136 116–13 636, 2019.
- [21] M. Bercibar, I. Gandiaga, I. Villarreal, N. Omar, J. V. Mierlo, and P. den Bossche, “Critical review of state of health estimation methods of Li-ion batteries for real applications,” *Renewable and Sustainable Energy Reviews*, vol. 56, pp. 572–587, 2016.

- [22] C. Lin, A. Tang, and W. Wang, "A review of SOH estimation methods in lithium-ion batteries for electric vehicle applications," *Energy Procedia*, vol. 75, pp. 1920–1925, 2015.
- [23] R. Xiong, L. Li, and J. Tian, "Towards a smarter battery management system: A critical review on battery state of health monitoring methods," *Journal of Power Sources*, vol. 405, pp. 18–29, 2018.
- [24] G. He, Q. Chen, P. Moutis, S. Kar, and J. F. Whitacre, "An intertemporal decision framework for electrochemical energy storage management," *Nature Energy*, vol. 3, pp. 404–412, 2018.
- [25] L. Pei, R. Lu, and C. Zhu, "Relaxation model of the open-circuit voltage for state-of-charge estimation in lithium-ion batteries," *IET Electrical Systems in Transportation*, vol. 3, no. 4, pp. 112–117, 2013.
- [26] F. Zheng, Y. Xing, J. Jiang, B. Sun, J. Kim, and M. Pecht, "Influence of different open circuit voltage tests on state of charge online estimation for lithium-ion batteries," *Applied Energy*, vol. 183, pp. 513–525, 2016.
- [27] W. Wang, D. Wang, X. Wang, T. Li, R. Ahmed, S. Habibi, and A. Emadi, "Comparison of kalman filter-based state of charge estimation strategies for Li-Ion batteries," *IEEE Transportation Electrification Conference and Expo (ITEC)*, 2016.
- [28] Y. Tian, R. Lai, X. Li, L. Xiang, and J. Tian, "A combined method for state-of-charge estimation for lithium-ion batteries using a long short-term memory network and an adaptive cubature kalman filter," *Applied Energy*, vol. 265, no. 114789, 2020.
- [29] J. Peng, J. Luo, H. He, and B. Lu, "An improved state of charge estimation method based on cubature kalman filter for lithium-ion batteries," *Applied Energy*, vol. 253, no. 113520, 2019.
- [30] P. Shrivastava, T. K. Soon, M. I. Idris, and S. Mekhilef, "Overview of model-based online state-of-charge estimation using kalman filter family for lithium-ion batteries," *Renewable and Sustainable Energy Reviews*, vol. 113, no. 109233, 2019.
- [31] M. Zhang, K. Wang, and Y. Zhou, "Online state of charge estimation of lithium-ion cells using particle filter-based hybrid filtering approach," *Hindawi Complexity*, no. 8231243, 2020.
- [32] A. Tulsyan, Y. Tsai, R. B. Gopaluni, and R. D. Braatz, "State-of-charge estimation in lithium-ion batteries: A particle filter approach," *Journal of Power Sources*, vol. 331, pp. 208–223, 2016.

- [33] Z. Guo, X. Qiu, G. Hou, B. Y. Liaw, and C. Zhang, “State of health estimation for lithium ion batteries based on charging curves,” *Journal of Power Sources*, vol. 249, pp. 457–462, 2014.
- [34] Z. Ma, R. Yang, and Z. Wang, “A novel data-model fusion state-of-health estimation approach for lithium-ion batteries,” *Applied Energy*, vol. 237, pp. 836–847, 2019.
- [35] S. Qu, Y. Kang, P. Gu, C. Zhang, and B. Duan, “A fast online state of health estimation method for lithium-ion batteries based on incremental capacity analysis,” *Energies*, vol. 12, no. 3333, 2019.
- [36] X. Tang, C. Zou, K. Yao, G. Chen, B. Liu, Z. He, and F. Gao, “A fast estimation algorithm for lithium-ion battery state of health,” *Journal of Power Sources*, vol. 296, pp. 453–458, 2018.
- [37] M. Galeotti, L. Cina, C. Giammanco, S. Cordiner, and A. D. Carlo, “Performance analysis and SOH (state of health) evaluation of lithium polymer batteries through electrochemical impedance spectroscopy,” *Energy*, vol. 89, pp. 678–686, 2015.
- [38] N. Harting, N. Wolff, F. Roder, and U. Kreuer, “State-of-health diagnosis of lithium-ion batteries using nonlinear frequency response analysis,” *Journal of The Electrochemical Society*, vol. 166, no. 2, A277–A285, 2019.
- [39] J. Li, K. Adewuyi, N. Lotfi, R. G. Landers, and J. Park, “A single particle model with chemical/mechanical degradation physics for lithium ion battery state of health (SOH) estimation,” *Applied Energy*, vol. 212, pp. 1178–1190, 2018.
- [40] R. Xiong, L. Li, Q. Yu, and H. Mu, “An electrochemical model based degradation state identification method of lithium-ion battery for all-climate electric vehicles application,” *Applied Energy*, vol. 219, pp. 264–275, 2018.
- [41] J. Bi, T. Zhang, H. Yu, and Y. Kang, “State-of-health estimation of lithium-ion battery packs in electric vehicles based on genetic resampling particle filter,” *Applied Energy*, vol. 182, pp. 558–568, 2016.
- [42] D. Liu, X. Yin, Y. Song, W. Liu, and Y. Peng, “An on-line state of health estimation of lithium-ion battery using unscented particle filter,” *IEEE Access*, vol. 6, pp. 40 990–41 001, 2018.
- [43] A. Guha and A. Patra, “State of health estimation of lithium-ion batteries using capacity fade and internal resistance growth models,” *IEEE Transactions on Transportation Electrification*, vol. 4, no. 1, pp. 135–146, 2018.

- [44] R. Xiong, H. He, F. Sun, and K. Zhao, "Online estimation of peak power capability of Li-ion batteries in electric vehicles by a hardware-in-loop approach," *Energies*, vol. 5, pp. 1455–1469, 2012.
- [45] F. Sun, R. Xiong, H. He, W. Li, and J. E. Aussems, "Model-based dynamic multi-parameter method for peak power estimation of lithium-ion batteries," *Applied Energy*, vol. 96, pp. 378–386, 2012.
- [46] P. Malysz, J. Ye, R. Gu, H. Yang, and A. Emadi, "Battery state-of-power peak current calculation and verification using an asymmetric parameter equivalent circuit model," *IEEE Transactions on Vehicular Technology*, vol. 6, no. 6, pp. 4512–4522, 2016.
- [47] S. Xiang, G. Hu, R. Huang, F. Guo, and P. Zhou, "Lithium-ion battery online rapid state-of-power estimation under multiple constraints," *Energies*, vol. 11, no. 283, 2018.
- [48] C. Burgos-Mellado, M. E. Orchard, M. Kazerani, R. Cardenas, and D. Saez, "Particle-filtering-based estimation of maximum available power state in lithium-ion batteries," *Applied Energy*, vol. 161, pp. 349–363, 2016.
- [49] B. Jiang, H. Dai, X. Wei, L. Zhu, and Z. Sun, "Online reliable peak charge/discharge power estimation of series-connected lithium-ion battery packs," *Energies*, vol. 10, no. 390, 2017.
- [50] K. Li, K. J. Tseng, F. Wei, and B. H. Soong, "A pragmatic SOH and SOC co-estimator for lithium-ion batteries in smart grid applications," *International Power Electronics Conference*, pp. 1517–1523, 2018.
- [51] S. C. Huang, K. H. Tseng, J. W. Liang, C. L. Chang, and M. G. Pecht, "An online SOC and SOH estimation model for lithium-ion batteries," *Energies*, vol. 10, no. 512, pp. 1–18, 2017.
- [52] L. Zheng, L. Zhang, J. Zhu, G. Wang, and J. Jiang, "Co-estimation of state-of-charge, capacity and resistance for lithium-ion batteries based on a high-fidelity electrochemical model," *Applied Energy*, vol. 180, pp. 424–434, 2016.
- [53] Y. Zou, X. Hu, H. Ma, and S. E. Li, "Combined state of charge and state of health estimation over lithium-ion battery cell cycle lifespan for electric vehicles," *Journal of Power Sources*, vol. 273, pp. 793–803, 2015.
- [54] B. Jiang, H. Dai, X. Wei, and T. Xu, "Joint estimation of lithium-ion battery state of charge and capacity within an adaptive variable multi-timescale framework considering current measurement offset," *Applied Energy*, vol. 253, no. 113619, 2019.

- [55] E. Chemali, P. J. Kollmeyer, M. Preindl, and A. Emadi, “State-of-charge estimation of Li-ion batteries using deep neural networks: A machine learning approach,” *Journal of Power Sources*, vol. 400, pp. 242–255, 2018.
- [56] C. Chen, R. Xiong, W. Shen, and F. Sun, “State-of-charge estimation of lithium-ion battery using an improved neural network model and extended kalman filter,” *Journal of Cleaner Production*, vol. 234, pp. 1153–1164, 2019.
- [57] J Meng, G. Luo, and F. Gao, “Lithium polymer battery state-of-charge estimation based on adaptive unscented kalman filter and support vector machine,” *IEEE Transactions on Power Electronics*, vol. 31, no. 3, pp. 2226–2238, 2016.
- [58] J. Hu, J. Hu, H. Lin, X. Li, C. Jiang, X. Qiu, and W. Li, “State-of-charge estimation for battery management system using optimized support vector machine for regression,” *Journal of Power Sources*, vol. 269, pp. 682–693, 2014.
- [59] W. Zheng, B. Xia, W. Wang, Y. Lai, M. Wang, and H. Wang, “State of charge estimation for power lithium-ion battery using a fuzzy logic sliding mode observer,” *Energies*, vol. 12, no. 2491, 2019.
- [60] L. Wang, A. Savvaris, and A. Tsourdos, “Online battery pack state of charge estimation via EKF-fuzzy logic joint method,” *2018 5th International Conference on Control, Decision and Information Technologies*, 2018.
- [61] A. Eddahech, O. Briat, N. Bertrand, J. Deletage, and J. Vinassa, “Behavior and state-of-health monitoring of Li-ion batteries using impedance spectroscopy and recurrent neural networks,” *Electrical Power and Energy Systems*, vol. 42, pp. 487–494, 2012.
- [62] D. Yang, Y. Wang, R. Pan, R. Chen, and Z. Chen, “A neural network based state-of-health estimation of lithium-ion battery in electric vehicles,” *Energy Procedia*, vol. 105, pp. 2059–2064, 2017.
- [63] X. Feng, C. Weng, X. He, X. Han, L. Lu, D. Ren, and M. Ouyang, “Online state-of-health estimation for Li-ion battery using partial charging segment based on support vector machine,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 9, pp. 8583–8592, 2019.
- [64] Z. Chen, M. Sun, X. Shu, R. Xiao, and J. Shen, “Online state of health estimation for lithium-ion batteries based on support vector machine,” *Applied Sciences*, vol. 8, no. 925, 2018.
- [65] X. Li, Z. Wang, L. Zhang, C. Zou, and D. D. Dorrell, “State-of-health estimation for Li-ion batteries by combing the incremental capacity analysis method with grey relational analysis,” *Journal of Power Sources*, vol. 410-411, pp. 106–114, 2019.

- [66] R. R. Richardson, M. A. Osborne, and D. A. Howey, “Battery health prediction under generalized conditions using a gaussian process transition model,” *Journal of Energy Storage*, vol. 23, pp. 320–328, 2019.
- [67] D. Yang, X. Zhang, R. Pan, Y. Wang, and Z. Chen, “A novel gaussian process regression model for state-of-health estimation of lithium-ion battery using charging curve,” *Journal of Power Sources*, vol. 384, pp. 387–395, 2018.
- [68] J. Lu, Z. Chen, Y. Yang, and M. Lv, “Online estimation of state of power for lithium-ion batteries in electric vehicles using genetic algorithm,” *IEEE Access*, vol. 6, pp. 20 868–20 880, 2018.
- [69] M. J. Esfandyari, M. R. Yazdi, V. Esfahanian, M. Masih-Tehrani, H. Nehzati, and O. Shekoofa, “A hybrid model predictive and fuzzy logic based control method for state of power estimation of series-connected lithium-ion batteries in HEVs,” *Journal of Energy Storage*, vol. 24, no. 100758, 2019.
- [70] Y. Song, D. Liu, H. Liao, and Y. Peng, “A hybrid statistical data-driven method for on-line joint state estimation of lithium-ion batteries,” *Applied Energy*, vol. 261, no. 114408, 2020.
- [71] H. Chaoui, C. C. Ibe-Keocha, and H. Gualous, “Aging prediction and state of charge estimation of a LiFePO₄ battery using input time-delayed neural networks,” *Electric Power Systems Research*, vol. 146, pp. 189–197, 2017.
- [72] X. Li, Z. Wang, and L. Zhang, “Co-estimation of capacity and state-of-charge for lithium-ion batteries in electric vehicles,” *Energy*, vol. 174, pp. 33–44, 2019.
- [73] X. Yang, Y. Chen, B. Li, and D. Luo, “Battery states online estimation based on exponential decay particle swarm optimization and proportional-integral observer with a hybrid battery model,” *Energy*, vol. 191, 2020.
- [74] W. Choi, H. Shin, J. M. Kim, J. Choi, and W. Yoon, “Modeling and applications of electrochemical impedance spectroscopy (EIS) for lithium-ion batteries,” *Journal of Electrochemical Science and Technology*, vol. 11, no. 1, pp. 1–13, 2020.
- [75] D. I. Stroe, M. Swierczynski, A. I. Stan, V. Knap, R. Teodorescu, and S. J. Andreasen, “Diagnosis of lithium-ion batteries state-of-health based on electrochemical impedance spectroscopy technique,” *Proceedings of the 2014 Energy Conversion Congress and Exposition*, 2020.
- [76] U. Westerhoff, K. Kurbach, F. Lienesch, and M. Kurrat, “Analysis of lithium-ion battery models based on electrochemical impedance spectroscopy,” *Energy Technology*, vol. 4, pp. 1620–1630, 2016.

- [77] M. M. Kabir and D. E. Demirocak, “Degradation mechanisms in Li-ion batteries: A state-of-the-art review,” *International Journal of Energy Research*, vol. 41, pp. 196–1986, 2017.
- [78] C. R. Birkl, M. R. Roberts, E. McTurk, P. G. Bruce, and D. A. Howey, “Degradation diagnostics for lithium ion cells,” *Journal of Power Sources*, vol. 341, pp. 373–386, 2017.
- [79] Z. Ma, Z. Wang, R. Xiong, and J. Jiang, “A mechanism identification model based state-of-health diagnosis of lithium-ion batteries for energy storage applications,” *Journal of Cleaner Production*, vol. 193, pp. 379–390, 2018.
- [80] J. Li, D. Wang, L. Deng, Z. Cui, C. Lyu, L. Wang, and M. Pecht, “Aging modes analysis and physical parameter identification based on a simplified electrochemical model for lithium-ion batteries,” *Journal of Energy Storage*, vol. 31, no. 101538, 2020.
- [81] “Battery test manual for plug-in hybrid electric vehicles,” *Idaho National Engineering & Environmental Laboratory*, 2010.
- [82] W. Wang, P. Malysz, K. Khan, L. Gauchia, and A. Emadi, “Modeling, parameterization, and benchmarking of a lithium ion electric bicycle battery,” in *2016 IEEE Energy Conversion Congress and Exposition (ECCE)*, vol. 5, 2016, pp. 1–7.
- [83] D. Andre, M. Meiler, K. Steiner, H. Walz, T. Soczka-Guth, and D. Sauer, “Characterization of high-power lithium-ion batteries by electrochemical impedance spectroscopy,” *Journal of Power Sources*, vol. 196, no. 12, pp. 5349–5356, 2011.
- [84] X. Zhang, Y. Wang, J. Wu, and Z. Chen, “A novel method for lithium-ion battery state of energy and state of power estimation based on multi-time-scale filter,” *Applied Energy*, vol. 216, pp. 442–451, 2018.
- [85] A. Farmann and D. U. Sauer, “A comprehensive review of on-board state-of-available-power prediction techniques for lithium-ion batteries in electric vehicles,” *Journal of Power Sources*, vol. 329, pp. 123–137, 2016.
- [86] V. N. Vapnik, “An overview of statistical learning theory,” *IEEE Transactions on Neural Networks*, vol. 10, no. 5, pp. 988–999, 1999.
- [87] H. Drucker, C. J. Burges, L. Kaufman, A. Smola, and V. N. Vapnik, “Support vector regression machines,” *Advances in Neural Information Processing Systems*, 1997.
- [88] B. E. Boser, I. M. Guyon, and V. N. Vapnik, “A training algorithm for optimal margin classifiers,” *Proceedings of the fifth annual workshop on Computational learning theory*, 1992.

- [89] Y. Li, K. Liu, A. M. Foley, A. Zulke, M. Berecibar, E. Nanini-Maury, J. V. Mierlo, and H. E. Hoster, “Data-driven health estimation and lifetime prediction of lithium-ion batteries: A review,” *Renewable and Sustainable Energy Reviews*, vol. 113, no. 109254, 2019.
- [90] J. Deng, “Control problems of grey systems,” *Systems & Control Letters*, vol. 1, no. 5, 1982.
- [91] Y. Kuo, T. Yang, and G. Huang, “The use of grey relational analysis in solving multiple attribute decision-making problems,” *Computers & Industrial Engineering*, vol. 55, pp. 80–93, 2008.
- [92] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Cambridge, Massachusetts: MIT Press, 1975.
- [93] J. Kennedy and R. Eberhart, “Particle swarm optimization,” *International Conference on Neural Networks*, 1995.
- [94] Y. Shi and R. Eberhart, “A modified particle swarm optimizer,” *International Conference on Evolutionary Computation Proceedings*, 1998.
- [95] Y. Miao, P. Hynan, A. Jouanne, and A. Yokochi, “Current Li-Ion battery technologies in electric vehicles and opportunities for advancements,” *Energies*, vol. 12, no. 1074, 2019.
- [96] S. Jiang, “A parameter identification method for a battery equivalent circuit model,” *SAE International*, 2011.
- [97] A. Savitzky and M. J. Golay, “Smoothing and differentiation of data by simplified least squares procedures,” *Analytical Chemistry*, vol. 36, no. 8, pp. 1627–1639, 1964.
- [98] J. Schmidhuber, “Deep learning in neural networks: An overview,” vol. 61, pp. 85–117, 2015.
- [99] M. Abadi, J. Dean, and R. Monga, “Tensorflow: Large-scale machine learning on heterogeneous systems,” *Google Research White Paper*, 2015.
- [100] D. Stathakis, “How many hidden layers and nodes?” *International Journal of Remote Sensing*, vol. 30, no. 8, pp. 2133–2147, 2009.
- [101] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting,” *Journal of Machine Learning Research*, vol. 15, pp. 1929–1958, 2014.

- [102] L. N. Smith, “A disciplined approach to neural network parameters: Part 1 - learning rate, batch size, momentum, and weight decay,” *US Naval Research Laboratory Technical Report*, no. 5510-026, 2018.
- [103] D. Masters and C. Luschi, “Revisiting small batch training for deep neural networks,” *arXiv:1804.07612 [cs.LG]*, 2018.
- [104] S. Sharma, S. Sharma, and A. Athaiya, “Activation functions in neural networks,” *International Journal of Engineering Applied Sciences and Technology*, vol. 4, no. 12, pp. 310–316, 2020.