Hybrid battery state estimation framework and advanced algorithm development

Yihuan Li

School of Electronic and Electrical Engineering

University of Leeds

Submitted in accordance with the requirements for the degree of

Doctor of Philosophy

June 2021

Declaration

The candidate confirms that the work submitted is her own, except where work which has formed part of jointly-authored publications has been included. The contribution of the candidate and the other authors to this work has been explicitly indicated below. The candidate confirms that appropriate credit has been given within the thesis where reference has been made to the work of others.

Details of the publications which has been used in this thesis are as following:

In Chapter 3:

1. Yihuan Li, Kang Li and Shawn Li. "FRA and EKF Based State of Charge Estimation of Zinc-Nickel Single Flow Batteries." Advances in Green Energy Systems and Smart Grid. Springer, Singapore, 2018. 183-191.

   As the lead author, the candidate performed all the computational as well as simulation work and wrote the paper.

   Prof. Kang Li, my supervisor, supervised the work, provided excellent technical advice, modified and proof-read the draft of the paper.

   Dr. Shawn Li discussed the concepts of this paper with me, reviewed the draft and provided some suggestions.

In Chapter 4:

2. Yihuan Li, Kang Li and Xuan Liu. "Fast battery capacity estimation using convolutional neural networks." Transactions of the Institute of Measurement and Control (2020): 0142331220966425.
   As the lead author, the candidate put forward ideas, developed methodology, performed all the computational as well as simulation work and wrote the paper.
   Prof. Kang Li, my supervisor, supervised the work, provided excellent technical advice, modified and proof-read the draft of the paper.
   Mr. Xuan Liu reviewed the draft and provided some suggestions.

3. Yihuan Li, Kang Li, Xuan Liu, Yanxia Wang and Li Zhang. "Lithium-ion battery capacity estimation—A pruned convolutional neural network approach assisted with transfer learning." Applied Energy 285: 116410.
   As the lead author, the candidate put forward ideas, developed methodology, performed all the computational as well as simulation work and wrote the paper.
   Prof. Kang Li, my supervisor, supervised the work, provided excellent technical advice, modified and proof-read the draft of the paper.
   Mr. Xuan Liu performed the experiments and discuss the data analysis.
   Dr. Yanxia Wang provided suggestions on the development of methodology and reviewed the draft.
   Mrs. Li Zhang reviewed the draft of the paper and provided

some suggestions.

Acknowledgements

First and foremost, I would like to express my sincere gratitude to my supervisor, Prof. Kang Li, for his continuous support, guidance, and encouragement during my PhD study. His extremely valuable advice and research insights and ideas have helped me build up the research expertise and guided me to find research directions. And he allocated much of his time to review and proofread my papers and provide valuable and helpful comments to improve the quality of the draft. I am extremely grateful for the support and guidance he has given me. Without his patience and encouragement, I couldn't have reached this point.

I would like to thank all my colleagues who I have worked with for a cherished time spent together. Special thanks to Dr. Yanxia Wang, Dr. Shaojun Gan, and Mr. Xuan Liu, for their treasured suggestions and experimental support which was really helpful in developing methodologies and analyzing results. I am also thankful to all my friends for their encouragement all through my studies.

My sincere gratitude extends to the China Scholarship Council, Queen's University Belfast, and University of Leeds for sponsoring my research and offering me such a great opportunity for expanding my horizon.

Last but not the least, I would like to express my deepest gratitude to my family for their love and support. Without their tremendous understanding and encouragement, it would be impossible for me to reach where I am today.

I dedicated this thesis to my parents.

Abstract

Lithium-ion batteries are playing a key role in shifting our society to a low-carbon future, they are being increasingly used in automotive and power industries to enable the intake of more renewable energy and to reduce greenhouse gas emissions. To ensure the safe, reliable, and efficient operation of batteries, a battery management system (BMS) is indispensable. Among all functionalities of the BMS, real-time battery state estimation provides important information for achieving high-fidelity and high-performance operations. This thesis focus on the development of novel techniques and frameworks to provide a coherent body of work on the estimation of battery states of interest, that is, state of charge (SOC) and state of health (SOH).

The nonlinear variants of the Kalman filter (KF) framework have proven to be powerful and elegant solutions for the real-time state estimation of battery systems. As a state-space model is fundamental for using filtering algorithms and their estimation results highly depend on the model accuracy, thus an accurate battery model that can well capture the battery dynamics is established with suitable model structure and parameters. Then the nonlinear version of the KF, namely Extended Kalman Filter (EKF) and Unscented Kalman Filter (UKF), as well as the Particle Filter (PF) are used to estimate the battery SOC.

As it is not a trivial task to obtain a precise model that can well describe the battery degradation trend, an intelligent estimation tech-

nique is proposed based on the convolutional neural network (CNN) to achieve fast and accurate online battery capacity estimation, which integrates health feature extraction, parameters identification, and capacity estimation into one framework. Taking into account the limitation of CNN on small degradation datasets, the transfer learning technique is incorporated into the CNN-based framework to improve the estimation performance on small datasets by taking advantage of the knowledge learned from large datasets. Further, in view of the limited computational capability of the current BMSs, a new network pruning technique is proposed to reduce the size and computation cost of the final model.

Considering the intrinsic coupling relationship between SOC and SOH, a co-estimation framework is proposed to estimate the SOC and capacity simultaneously. Due to the estimators can be mutually optimized in the co-estimation framework, the estimation results of SOC and capacity are both more accurate than the separated estimation methods. Further, to acquire more informative measurements, the fiber optic sensors are attached to the cell surface and their measurements are utilized for battery SOC estimation to further improve the accuracy.

Experimental data are collected from the lithium iron phosphate batteries to analyze and evaluate the efficacy of the methods and frameworks proposed in this thesis.

Abbreviations

ANN	Artificial Neural Network
$BESS$	Battery Energy Storage System
BMS	Battery Management System
CC	Constant current
CNN	Convolutional neural network
$CNN(S)$	Convolutional neural network (pre-trained on the source dataset)
$CNN(S) - TL$	Convolutional neural network (pre-trained on the source dataset) with transfer learning
$CNN(T)$	Convolutional neural network (directly trained on the target dataset)
CV	Constant voltage
DNN	Deep neural network
DP	Dual polarization model
DV	Differential voltage
ECM	Equivalent circuit model
EKF	Extended Kalman filter
EM	Electrochemical model
ESS	Energy storage system
EV	Electric vehicle
FBG	Fiber Bragg grating
FC	Fully connected

$FLOPs$	Floating point operations
FRA	Fast recursive algorithm
$G2V$	Grid-to-Vehicle
GPR	Gaussian process regression
ICA	Incremental Capacity Analysis
KF	Kalman filter
LCO	Lithium cobalt oxide
LFP	Lithium iron phosphate
LTO	Lithium-titanate oxide
MAE	Mean absolute error
$MaxE$	Max error
ML	Machine learning
NaS	Sodium sulfur
$NASA$	National Aeronautics and Space Administration
NEE	Normalized estimation error
$NiCd$	Nickel-cadmium
$NiMH$	Nickel metal-hydride
NMC	Lithium nickel manganese cobalt oxide
NN	Neural network
OCV	Open circuit voltage
$P2D$	Pseudo Two-dimensional model
$PCNN$	Pruned convolutional neural network
$PCNN(S)-TL$	Pruned convolutional neural network (pre-trained on the source dataset) with transfer learning
$PCNN(T)$	Pruned convolutional neural network (directly trained on the target dataset)
PDF	Probability density function
PF	Particle Filter

PP	Porous electrode model with the polynomial approximation
RC	Resistor-capacitor
$ReLU$	Rectified Linear Unit
RFB	Redox flow battery
RLS	Recursive least square
$RMSE$	Root mean-square error
RVM	Relevance vector machine
SOC	State of charge
SOE	State of energy
SOH	State of health
SOP	State of power
SP	Single particle model
SVM	Support vector machine
TL	Transfer learning
UKF	Unscented Kalman Filter
UPS	Uninterrupted power supply
$V2G$	Vehicle-to-Grid
$VRLA$	Valved-regulated lead acid

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The low-carbon transition is now the priority of many countries that aim at achieving carbon neutrality by the mid of this century. One of the bridges to a carbon neutral future is the battery energy storage systems, which capture the energy and store it using electrochemical solutions. It plays a vital role in the mass roll-out of electric vehicles and acceptance of significant penetration of renewable power worldwide, and requires a battery management system to monitor its internal states and maintain its functional performance, cycling life, and safe and reliable operation. This chapter starts with an introduction to the research background, it provides an overview of battery energy storage systems, battery management systems, and state estimation techniques. Then according to the current challenges facing battery management systems, the motivations and objectives of the research conducted in this thesis are presented. Finally, the outline of the thesis is provided together with a summary for each chapter.

1.1 Background

Carbon neutrality, or net-zero carbon dioxide emissions, refers to balancing the emissions and removal of carbon dioxide in the atmosphere. To slow down the global climate change and to cope with energy supply shortage, the European Union, China, and the United Kingdom, together with more than 110 other countries, have committed to achieving net-zero emissions within the next few decades [1]. Among actions in delivering a low-carbon future, the application of energy storage systems, particularly battery energy storage, is the key to enabling the integration of more renewable energy and reducing greenhouse gas emissions, mainly because it has been widely used in electric vehicles (EVs) and the electricity grid. Based on the commitment of those countries with net-zero emission target, by 2040, the total number of EVs that are projected to be on the road is expected to reach 150 to 900 millions, which is two or three orders of magnitude larger than today. In the mean time, the global installation of stationary energy storage will increase exponentially and is projected to reach up to 1300 GWh by 2040 [2].

The transportation sector and power sector are the two main sources of greenhouse gas emissions. Previous research has demonstrated that EVs are one of the most promising solutions for the transition of energy chain to a low-carbon, safe and competitive economy, they offer the potential to significantly reduce carbon emissions from the transport sector and reduce the dependence on fossil fuels. And for the power sector, battery energy storage systems (BESSs) could play vital roles in deep decarbonization as they can provide a new, carbon-free source of operating reserves and flexibility in the power system, as well as facilitate the integration of variable renewable energy sources such as solar and wind energy [3]. The growing installed capacity of renewable energy and the increasing number of EVs connected to the power grid has initiated high requirements on power bal-

ance control and power quality, and gradually changing the structure of the power network. A typical scenario of a grid system integrated with renewable energy sources, energy storage systems (ESSs) and EVs is illustrated in Figure 1.1. The renewable energy power station with solar photovoltaic panels and wind turbines produce energy to the power grid. The ESSs have potential bidirectional power flow capabilities, they can receive electrical energy from the grid during the excess of production and deliver it back when consumption is higher than generation. And similar to ESSs, the EVs with Grid-to-Vehicle (G2V) and Vehicle-to-Grid (V2G) [4] capabilities can also receive or provide electrical energy.

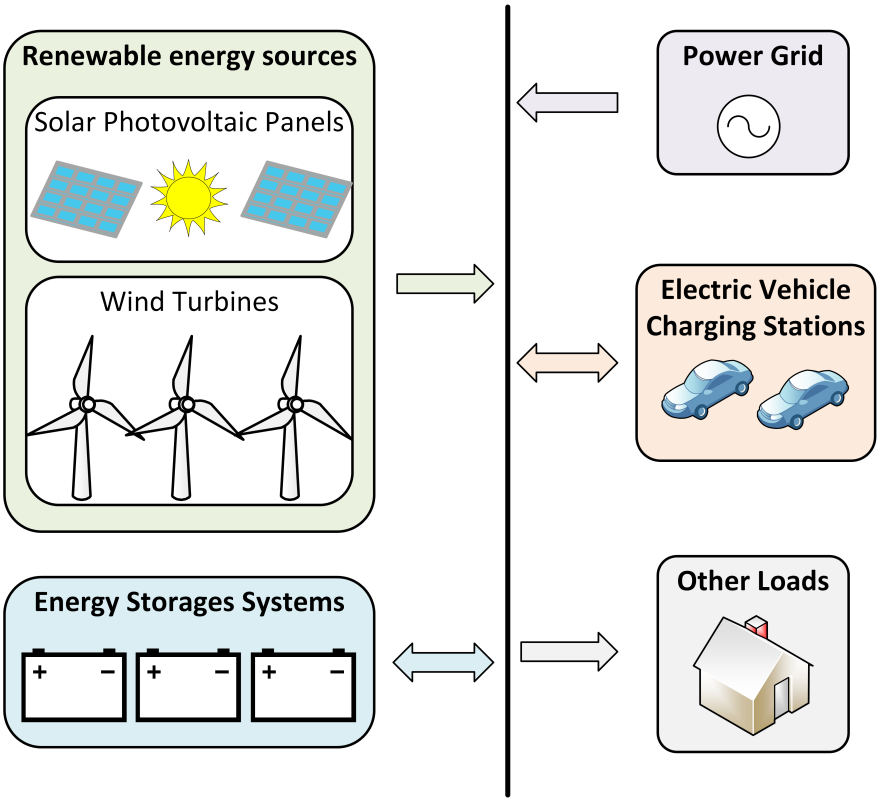


Figure 1.1: A typical grid system configuration that integrates renewable energy sources, energy storage systems and electric vehicles.

Battery energy storage systems are the key components of the future transportation sector and power grid, they are not only the power sources of EVs that de-

termine the cruising range and carrying capacity, but also electric energy buffers that can balance the energy production and consumption and improve the renewable energy integration into the power grid. In brief, BESSs can support a wide range of services needed for the low-carbon transition, from powering electric vehicles, to providing grid services including frequency regulation, reserve capacity and black-start capability [5], and upgrading mini-grids. Therefore, battery energy storage is regarded as an essential factor for countries with carbon neutrality objectives, and its deployment and commercialisation have become more widespread. More information on BESSs will be elaborated on in the following subsection.

1.1.1 Battery Energy Storage

Battery energy storage systems (BESSs), which are a sub-set of energy storage systems, are mature and popular technologies developed for storing electrical energy in electrochemical form [6]. Due to their merits of modularization, flexible installation, rapid response and decreasing prices [7], they exhibit considerable potentials for fueling the next green revolution in transportation and power sectors. By electrifying the transportation systems and balancing the intermittency issues caused by the penetration of renewable energy [8], batteries can radically reduce the carbon emissions in these two sectors. To promote the development and commercial applications of batteries, enormous amount of researches have been conducted to improve the performance of existing batteries and develop new batteries. There are many commercially available battery types, such as lead acid, nickel-based, sodium sulfur (NaS), Lithium-ion (Li-ion), and flow batteries, among which the Li-ion batteries are the current predominant BESSs for EVs. A brief introduction of these batteries and their applications is given in the following:

- The lead acid battery is the first commercially successful rechargeable battery invented by Gastom Planté in 1859 [9], it has been widely used in many applications such as EVs and grid energy storage. In this battery, the lead dioxide and metallic lead are treated as positive and negative electrodes, separately, and the sulphuric acid solution is used as the electrolyte [7]. There are two typical types of lead acid batteries, including the flooded lead acid batteries and sealed valved-regulated lead acid (VRLA) batteries, the main difference between them is that the electrolyte in VRLA batteries is immobilized [10]. The flooded lead acid battery is cheaper than the VRLA battery, but it requires regular maintenance and refilling distilled water due to the water loss, and needs to be performed in a more ventilation space due to the production of flammable gasses. Furthermore, the flooded battery must be operated in an upright posit and acid spillage is possible. As an improved type of lead acid battery, the VRLA battery can minimize water loss by employing an oxygen recombination technology, and it has a safety pressure relief valve to release the excess internal pressure. The construction of the VRLA battery means that it requires less maintenance, can be installed in any orientation and there are virtually no leakages [11]. The lead acid batteries are large and heavy, and have low energy density and short life cycles. Nonetheless, they are low-cost, have high reliability, and have a high tolerance for abuse [12]. Therefore, they are ideal candidates for those small-scale energy storage applications such as starting, lighting, and ignition for automotive and uninterrupted power supply (UPS) [13], as well as large-scale grid applications such as renewable energy balancing.

- The nickel-based battery employs a nickel hydroxide positive electrode, with either a metallic anode (e.g. nickel-cadmium (NiCd)) or a hydrogen storing negative electrode (e.g. nickel metal-hydride (NiMH)), as well as an alkaline

electrolyte. The NiCd battery has longer life cycle and can provide excellent performance under rigorous working conditions [14]. Since its invention in 1899, it has been widely applied for portable electronic devices until the mid-1990s. However, due to its low energy density and the toxicity of cadmium, the market share of NiCd battery declined rapidly. Compared to NiCd, the NiMH battery that commercially uptook in 1990 has higher energy and power density and is environmentally friendly. At the beginning of 2000, it represented the most popular technology adopted by major companies (e.g. Honda, Ford, General Motor, Toyota, etc.) for EVs and hybrid EVs [15]. However, it still suffers from several drawbacks such as high self-discharge rate, sensitive to storing and charging conditions, limited service life, and low specific energy [16]. Therefore, NiMH batteries will no longer serve in next-generation EVs and hybrid EVs and will be substituted by new technologies.

- The sodium sulfur (NaS) battery was originally developed by Ford Motor Company in the 1960s for automotive applications [16] and was commercialized in grid energy storage applications by Tokyo Electric Power Company in collaboration with NGK insulators [17]. By using molten sodium as the negative electrode and molten sulphur as the positive electrode, and separating them by a solid ceramic sodium alumina electrolyte membrane, the resultant NaS battery is characterized by high energy density, low cost of raw materials, long cycle life, and high round-trip efficiency [10; 18]. These attributes make it suitable for large-scale grid-level energy storage applications including load leveling, peak-shaving and integration of renewable energy sources. However, the major drawback of this type of battery is that it requires a high operating temperature (over 300 °C) to ensure that the active materials of the two electrodes are in a molten state and thus allow

the reactions to process. This restriction increases the application difficulty and operational cost of NaS batteries.

- Modern Redox flow battery (RFB) was invented by the National Aeronautics and Space Administration (NASA) in the 1970s [19]. Unlike traditional solid-state batteries that store energy in electrode materials, RFBs store energy in liquid electrolytes, which are generally stored in two external tanks. The electrolytes flowing through the cathode and anode often are different and are referred to as the anolyte and the catholyte, respectively. As the energy is stored in the incoming electrolytes (fuels) in the form of two dissolved redox pairs, and energy conversion between chemical and electrical energy occurs at the electrodes when the electrolytes are pumped from the tanks to cells, RFBs are sometimes referred to as regenerative fuel cells [16; 20].

  Energy capacity of the RFB is mostly scaled with the volume of the electrolytes, whilst the power is related to the electrode area or the reactor area [21]. That is, the energy and power capacity of RFB can scale over a broad range by flexible modular design, thus making RFB suitable for a variety of applications. Furthermore, the RFB has the advantages of excellent safety, fast response time and long cycle life, which makes it well suited for balancing highly variable renewable energy sources. However, one of the main drawbacks of commercial RFB is that it provides lower energy densities than other types of batteries, such as Lithium-ion batteries.

- Lithium-ion batteries, which were commercially launched by Sony in the early 1990s, are popular energy storage systems that have achieved significant penetration in various applications. For these batteries, the intercalated lithium compounds are used as the active materials of the positive electrode, and the negative electrode is usually made of carbonaceous ma-

terials such as graphite [10]. During the operation of Li-ion batteries, the Li-ions migrate across the electrolyte located between the two electrodes. In particular, the Li-ions are deintercalated from the positive electrode and then inserted into the negative electrode when the battery is charged, while the process is reversed during discharge [16]. The Li-ion batteries outperform other types of batteries (e.g. lead acid, NiCd and NiMh), the properties of low molecular weight, high energy density, long cycle life, low self-discharge rate and environmental friendliness make them overwhelmingly attractive and thus ubiquitous in portable electronics. Furthermore, Li-ion batteries are regarded as the systemic enabler of a major shift for the next generation of EVs and grid energy storage applications [7; 22].

The huge application prospects of Li-ion batteries have attracted great research attention, and significant research efforts have been devoted to the development of advanced Li-ion battery technologies in recent years. Based on the development of basic solid-state chemistry of materials, the positive and negative electrodes and electrolytes have been improved and optimized [23]. Several types of commercialized Li-ion batteries have been developed, such as lithium cobalt oxide (LCO), lithium nickel manganese cobalt oxide (NMC), lithium iron phosphate (LFP) and lithium-titanate oxide (LTO). As their electrodes and electrolytes materials are different, they have different chemistry and characteristics. For example, LCO battery has high specific energy and dominate the portable electronics market, NMC and LFP batteries are becoming increasingly popular in EVs applications, while LTO battery tends to have a growing market share in large-scale grid applications due to its long cycle life, high power density, and well low/high-temperature tolerance. It is noteworthy that LFP battery has the characteristics of long cycle life, high current rating, good thermal stability, high safety level and environmental resilience, which make it one of the most promising Li-ion

batteries for EVs and grid energy storage applications [16; 24]. Therefore, LFP batteries are chosen as the subjects of this work, and the later developed online state estimation frameworks are all based on the experimental test of these batteries.

A comparison of different types of batteries is summarized in Table 1.1 [10; 16; 25; 26; 27; 28; 29; 30], the main characteristics in terms of cycle life, nominal voltage, specific energy, specific power, round trip efficiency, daily self-discharge rate, charging and discharging temperature of the aforementioned batteries are listed.

As it is shown, the lithium-ion batteries are superior to other types of batteries in many terms. Notwithstanding their desirable performance, the operation reliability and safety necessitate an effective and dependable battery management system for state estimation, charge/discharge control, and planned maintenance, etc.

1.1.2 Battery Management System

The battery management system (BMS) is a key element to ensure the safe, reliable, and efficient operation of the battery as well as to maximize its energy/power delivery capabilities and prolong its lifetime. As illustrated in Figure 1.2, the core functions of BMS generally include the following:

- Data acquisition:

  This module measures and collects the battery data at proper sampling frequency and precision, which includes current, voltage, and surface temperature.

- State estimation:

  Battery internal states, such as state of charge (SOC), state of power (SOP), state of health (SOH) and internal temperature, are critical indicators of the

Table 1.1: A comparison of different types of batteries.

Type	Cycle Life	Nominal Voltage	Specific Energy	Specific Power	Round-trip Efficiency	Self-discharge per day	Charging Temperature	Discharging Temperature
Lead acid	200-1000	2.1 V	30-50	75-300	75-85%	0.1-0.3%	-40-50 °C	-40-60 °C
NiCd	2500	1.2 V	50-75	150-300	∼70%	0.2-0.6%	0-40 °C	-20-70 °C
NiMH	180-2000	1.2 V	60-120	250-1000	∼65%	0.6-1%	0-40 °C	-20-65 °C
NaS	2500-4000	2.1 V	150-240	150-230	75-90%	–	300-350 °C	300-350 °C
RFB	>12000	1.15-1.55 V	10-30	80-150	75-90%	Negligible	0-50 °C	0-50 °C
LCO	<1000	3.6-3.7 V	90-200	1500	97%	0.1-0.3%	0-45 °C	-20-60 °C
NMC	>1000	3.65 V	100-240	300-1500	92%	<0.2%	0-45 °C	-20-60 °C
LTO	>10000	2.3 V	70	3000-5100	96%	0.1-0.4%	-20-45 °C	-30-60 °C
LFP	2500-12000	3.2 V	100-140	150-500	89%	<0.1%	0-45 °C	-30-60 °C

battery performance and reveal the operating status of the battery. These variables can not be measured directly, the only information available to infer these states is generated by the measurements. However, erroneous internal state values can greatly affect the operation of the battery, result in abuse or waste of battery available capability, which may cause premature failure [31]. Therefore, real-time high-accuracy state estimation is essential for the safe, efficient and reliable operation of the battery systems.

- Charge/discharge control:
  This module regulates the charge/discharge current and voltage to avoid inappropriate operations, thereby preventing damage to the battery and prolonging its service life.

- Cell balancing:
  A battery pack usually consists of a number of cells configured in series/parallel formations to provide sufficient power and voltage for specific applications. Small differences between cells due to production tolerances or operating conditions tend to be magnified with battery usage, which can significantly affect the overall pack performance [16]. Meanwhile, weaker cells could become overstressed resulting in premature failure of the battery. To avoid this problem, cell balancing is proposed to compensate for weaker cells by equalising the charge on all the cells in the pack and thus extending battery life.

- Thermal management:
  As the temperature can significantly affect battery characteristics, an efficient thermal management module is essential to control the temperature and confine the battery pack to operate within a proper temperature range.

- Safety protection:

The protection module is responsible for ensuring the safety of the battery under almost all circumstances and protecting the user from the consequences of battery failures. Special circuits are designed to prevent the battery from operating beyond the tolerance conditions, i.e., to operate the battery within its predefined limits thereby avoiding overcharge/overdischarge, overheating, overpressure, etc. Moreover, this module also offers proactive fault detection and alarming functions.

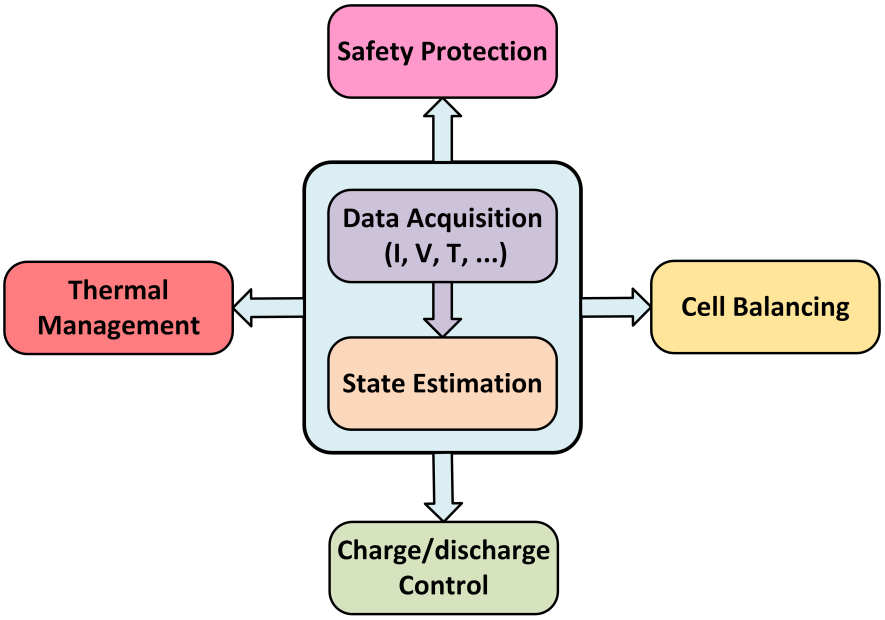


Figure 1.2: The core functions of a typical BMS

The BMS is indispensable for integrating the battery energy storage into the EVs and electric grid. Without efficient BMSs, the battery may be operated and maintained inappropriately, leading to catastrophic hazards and failures.

1.1.2.1 Battery State Estimation

Although the BMS offers various sophisticated functions, the internal state estimation is still one of the most critical tasks in BMS due to it is usually the basis

for battery charge/discharge control, protection, optimization, and maintenance. The battery states of interest are:

- State of charge (SOC) which is an indicator that reflects the real-time remaining capacity of the battery. The SOC is influenced by current, temperature and battery aging level, and has fast time-varying dynamics.
- State of health (SOH) which is an indicator that reflects the aging or degradation level of the battery. The battery aging is mostly influenced by operating conditions (i.e. load, temperature, humidity) and SOC, and has slow time-varying dynamics.
- State of energy (SOE) which is an indicator that reflects the residual energy of the battery.
- State of power (SOP) which reflects the amount of electrical power that a battery can deliver for a given time interval under specified operating conditions. The SOP is influenced by operating temperature, SOC, SOH and time.
- Internal temperature which is different from the battery surface temperature, can directly affect the battery performance leading to degradation and failure.

As the safety standards and regulations of batteries in automotive and grid applications become more stringent, battery state estimation is definitely a fast-growing area of research. The two major states, SOC and SOH, have been extensively studied over the years to achieve fast and accurate online estimation. A large number of approaches have been proposed, which can be roughly divided into two categories: model-based approaches combined with filtering algorithms, and model-free approaches such as machine learning-based techniques. The SOC

and SOH estimation techniques reported in the literature will be systematically elaborated in Chapter 2.

1.1.2.2 Fiber Optic Sensors in BMS

Most of the existing research works estimate the battery states using the external measured parameters such as current, voltage, and surface temperature. These external parameters are weakly informative to represent the holistic mapping of the battery states thus limit the estimation accuracy, and lead to over-conservative usage of the battery [32]. Therefore, an intuitive way to improve the estimation performance is to gain additional information of the battery dynamic behavior, such as internal temperature and electrode volume change. Fiber optic sensors, which are immune to electromagnetic interference, robust to corrosive environments, have multiplexing and small dimensions, are attractive candidates for battery sensing applications [33]. They are capable of sensitively measuring multiple parameters simultaneously, including strain, temperature and pressure. During the charge and discharge processes, the cell electrode volume changes due to Li-ions intercalation/ deintercalation processes in the electrode materials, which manifest as changes in the strain on the cell surface. Besides, the cell temperature also changes during the charge and discharge processes due to the electrochemical reactions, resistive heating, and enthalpy changes. Both parameters can help gain deeper insight into the internal dynamics of the battery, and are regarded as a complementary to the current and voltage measurements. Fiber Bragg grating (FBG) sensors are sensitive to strain and temperature and thus their applications in BMS have been increasingly investigated in recent years. To demonstrate the potential of FBG sensors to aid state estimation in BMS, Sommer et al. attached a pair of FBG sensors externally to lithium-ion pouch cells to monitor intercalation state transition points across various charge/discharge rates [34]. Sharp and repeatable features correlated with intercalation state tran-

sitions can be observed in the strain curves across different charge/discharge rates, and these novel features can enable better state estimation. Nascimento et al. presented a comparative study of surface temperature monitoring performance between the thermocouples and fiber sensors, and demonstrated that the FBG sensors were better choices for surface temperature monitoring under normal and abuse operating conditions and failure detection [35]. Fortier et al. embedded the FBG sensors inside the coin cell and monitored strain, internal and external temperatures simultaneously, the challenges and obstacles encountered during the integration of FBG sensors inside coin cells were discussed [36]. Similarly, after exploring the relationship between the external FBG sensing signals and diffusion processes [32], the team of Raghavan et al. embedded the FBG sensors inside pouch cells to directly monitor the internal temperature and electrode strain [33], the results have revealed that batteries with embedded FBG sensors are highly comparable to those without FBG sensors in terms of seal integrity, capacity retention and projected cycle life. Besides, this work demonstrated that the FBG sensors were possible for embedding into large-format cells as a low-cost, field-deployable option for direct internal state monitoring for BMSs. Peng et al. [37; 38] designed sensitivity-enhanced FBG sensors and mounted them onto the cell surface, the relationship between the strain and SOC/depth of discharge was investigated.

Since increasingly encouraging results indicate that the FBG signals can be used for battery state estimation, there is a growing interest in developing new state estimation algorithms to incorporate the conventional measurements (i.e. current, voltage and surface temperature) and the FBG sensor signals (i.e. strain and internal temperature) to improve the performance. [39] is the second part of the twofold paper [33; 39], it estimated the SOC and SOH using dynamic time warping and Kalman filtering algorithms based on the internal strain signals obtained from these FBG sensors at different temperature conditions. Similarly,

Rente et al. [40] achieved accurate SOC estimation results using dynamic time warping algorithm by correlating the cell surface strain data obtained from FBG sensors with the SOC, the results indicated that installing FBG sensors on the cell surface is a feasible, cost-effective and non-invasive approach for assisting SOC estimation.

1.1.3 State estimation

The hidden variables, which can characterise the internal condition or status of a system, are often extracted from indirect, inaccurate and uncertain measurements. This process is called state estimation, and the hidden variables are often named as system states. The main objective of state estimation is to minimize the estimation error between the actual and estimated values. Due to the importance of state estimation in monitoring, control, and fault diagnosis of engineering processes and systems, it has been extensively researched and widely applied across many engineering and scientific disciplines since the concept was developed in the 1960s [41].

The earliest interest in the state estimation problems can even be traced back to 1795 in the astronomical field, where the motion of planets and comets was studied using telescopic observations. Karl Friedrich Gauss investigated the problem of how to infer the parameters from the measurements in an optimal way to characterise the orbits of the revolution of celestial bodies, and developed a mathematical approach in 1809 based on the famous least square estimation method that he invented in 1795. The maximum likelihood estimation, which lays the foundation of the modern state estimation theory, was introduced by Fisher in 1912. And then in the 1940s, Kolmogorov and Wiener developed estimation methods for discrete-time and continuous-time stochastic systems respectively. The developed methods minimize the least mean-square errors, and form the basis for the recursive estimation and stochastic filtering theory. Wiener filter designed by

Wiener is mainly used for signal processing and communication theory, but not for state estimation, as it only deals with stationary processes. Kalman extended this theory for more generic non-stationary processes and developed a new theory which was later named as the Kalman filter (KF) [42]. Then, this algorithm was successfully employed in the Apollo's guidance and navigation system by NASA and was soon applied to state estimation problems as one of the most pragmatic approaches. Sequential measurements and a linear state space model are used in KF to provide an optimal state estimation for the system with white Gaussian noise. Kalman and Bucy further developed a continuous version of the KF called the Kalman-Bucy filter [43; 44; 45].

State estimation problems can be classified based on whether the system is linear or nonlinear. The aforementioned Wiener filter and Kalman filter are optimal linear filters. Linear state estimation deals with the estimation problems for linear systems or linear models. But in reality, almost all the practical systems are nonlinear, and linear estimators may fail to produce satisfactory results for nonlinear systems. Though a number of methods have been proposed to solve the problems of state estimation in nonlinear systems and processes, no single solution clearly outperforms other strategies. Two main streams of nonlinear system state estimation methods include model-based and data-driven methods. If the system model can be well developed based on the first principle laws, the model-based method would be a cost effective approach for real time state estimation. A number of model-based approaches have emerged over the past several decades for state estimation of nonlinear systems, such as the variants of KF and sliding mode observers. The extensions of KF formulation, such as the extended Kalman filter (EKF) and the unscented Kalman filter (UKF), enable the framework of KF to be applied to state estimation in nonlinear systems with Gaussian noise hypotheses. To deal with state estimation problems for nonlinear system without Gaussian noise hypotheses, the particle filter (PF) was proposed based on Monte

Carlo simulation technology to estimate the states by calculating the conditional probability of the state. Based on these basic filters, many other algorithms are proposed according to the demand in real applications. Adaptive EKF and Adaptive UKF can achieve higher accuracy by adaptively updating the process and measurement noise covariance. Unscented PF was proposed to generate a proposed distribution to obtain the posterior probability using the information of the latest observations. And the Adaptive PF is capable of adjusting the number of particles at each iteration. This adaptive adjustment is expected to produce better estimation performance and reduce the computational cost when the initial number of particles is over-estimated. The Regularized PF was proposed to overcome the sample impoverishment problem by changing the discrete approximation of the posterior probability density function (PDF) to a continuous one at the resampling stage with a re-scaled kernel procedure [46].

With the development of sensor technology and storage technology, data-driven methods have been increasingly investigated for state estimation, which could circumvent the need to model large or complicated systems. In particular, machine learning (ML) techniques such as support vector machine, Gaussian process regression, and deep neural network, are popular data-based state estimation techniques, and have become a common tool for modelling complex or even unknown systems due to their simplicity in handling data and powerful approximation capability. These methods do not require prior knowledge of the systems, and are not restricted by the system assumptions. The applications of both model-based and data-based techniques in the fields of SOC and SOH estimation have been widely studied, and a review will be given in Chapter 2.

1.2 Research motivations and objectives

Lithium-ion battery energy storage systems are complex dynamic systems with high nonlinearity and environmental sensitivity. As aforementioned, accurate state estimation plays a vital role in BMS for ensuring safe, efficient and reliable battery operation. Estimation methods are consistently developed to achieve further improvement in terms of accuracy, reliability, and practicability. And the newly developed methods are expected to be more intelligent since a large volume of historical experimental data is available now and various operating conditions are taking into account. This thesis aims to expand upon the current state-of-the-art advanced battery state estimation algorithms, by developing accurate and reliable frameworks for SOC and SOH estimation.

For model-based SOC estimation methods, an accurate model that can well describe the system dynamics is essential. Considering the complex operating conditions and the complicated electrochemical reactions inside the battery, it is a challenging task to obtain a perfect battery model. For battery capacity estimation, using data-driven methods has several advantages over other conventional methods. Particularly, the convolutional neural network (CNN) is powerful to automatically capture hidden information from a huge amount of historical data, flexible to express any complex systems by adjusting the weights and layers, and capable of retaining the same expressibility as the deep neural networks (DNN) with fewer parameters. Thus a CNN-based framework is proposed for capacity estimation in this work. However, the primary use of the CNN mainly lies in high-dimensional data, like images, videos and speech recognition. To apply it in such a temporal dimension, the structure of the CNN should be carefully defined, and the input signal should be reconstructed. Besides, the performance of CNN-based capacity estimation framework on small dataset is unsatisfactory as a proper CNN model requires a large amount of training data. Advanced tech-

niques should be investigated to improve its performance on small target dataset. Moreover, in view of the limited computational capability of the current BMSs, the CNN model size need to be further compressed. And last but not least, the SOC estimation is affected by battery aging, while the SOH estimation can also be affected by inaccurate SOC estimates. Therefore, the intrinsic coupling relationship between SOC and SOH should be considered during their estimation.
To tackle these challenges, the main objectives of this research work can be summarized as follows:

- To comprehensively review the state-of-the-art techniques on battery state estimation, the main emphasis is given to SOC and SOH.
- To develop an effective battery model for SOC estimation, as an accurate model is fundamental to ensure reliable state estimation.
- To develop a CNN-based SOH estimation framework, where the SOH is represented by the capacity. This method can perform the modelling and estimation in a single unified step as the neural network can directly map the measurements (e.g., current, voltage and surface temperature) to the battery capacity.
- To investigate advanced techniques that can improve the CNN performance on small degradation dataset of batteries, e.g.,via the transfer learning technique.
- To investigate network pruning techniques to reduce the size and computational complexity of the CNN model while maintaining its performance.
- To develop a SOC and SOH joint estimation method that can synergistically optimize their respective estimation results, in order to achieve higher estimation accuracy for both states.

- To investigate the application of FBG measured signals in assisting battery state estimation.

1.3 Thesis Structure

Chapter 1 introduces the research background including battery energy storage, battery management system and state estimation techniques, and presents the research motivations and objectives.

Chapter 2 reviews the battery modelling and state estimation techniques. Various battery modelling and identification methods are discussed, the advantages and drawbacks of several different estimation techniques of battery SOC and SOH are analyzed, with a view to later develop qualified state estimation frameworks.

Chapter 3 describes the model-based battery SOC estimation method proposed in this study. The detailed algorithm used in the model construction is firstly introduced, then the proposed model construction processes are presented. Next, a preliminary introduction to the filtering algorithms is presented, the concepts and formulas of the KF, EKF, UKF, and PF are introduced in detail. Finally, different filtering algorithms are applied to estimate the SOC based on the resultant model, and the results are analysed comparatively.

Chapter 4 details the CNN-based framework developed in this study to simulate battery degradation trend. CNNs, transfer learning, and network pruning techniques, which are then integrated into one framework to produce the final model, are introduced at the beginning of this chapter. Then the detailed process for model construction are elaborated. Finally, estimation results of the resultant model are analyzed to verify the efficacy of the proposed framework.

Chapter 5 extends the research work presented in Chapter 3 and Chapter 4 and a battery SOC and SOH coestimation framework based on Gaussian process regression and CNN is proposed and evaluated. New signals obtained by fiber optic

sensors are introduced and applied for SOC estimation. Experimental results are analyzed to verify the performance of the proposed framework and the utility of the newly involved fiber sensor data.

Finally, **Chapter 6** summarizes the work that has been conducted so far, and the scope for the future research which can be expanded on this thesis is set out.

Chapter 2

Literature Review

As introduced in Chapter 1, lithium-ion batteries have shown superior performance in energy density, power density, cycle life, and round-trip efficiency in comparison with other types of batteries. Thus they are versatile and attractive battery energy storage systems, and represent the most convincing choice for both electric vehicles and power grids in the upcoming years. In order to ensure the operation safety, reliability and longevity of the battery, an efficient battery management system (BMS) with advanced techniques is necessary. Among all functionalities of the BMS, battery internal state estimation is essential for controlling, optimizing, and managing the operation of the battery. Generally speaking, battery state estimation methods can be classified into two categories, i.e., model-based and model-free methods. Since an accurate battery model is an important reference for model-based estimation methods, a comprehensive review of battery modelling approaches is provided at the beginning of this chapter. Then a systematic literature review over state estimation techniques is performed, the main emphasis is given to state of charge and state of health.

2.1 Battery models

The battery model is a prerequisite for model-based battery internal state estimation. The battery models presented in literature mainly fall into the following three categories: physics-based electrochemical models (EM), electrical equivalent circuit models (ECMs), and black-box models [47], as shown in Figure 2.1.

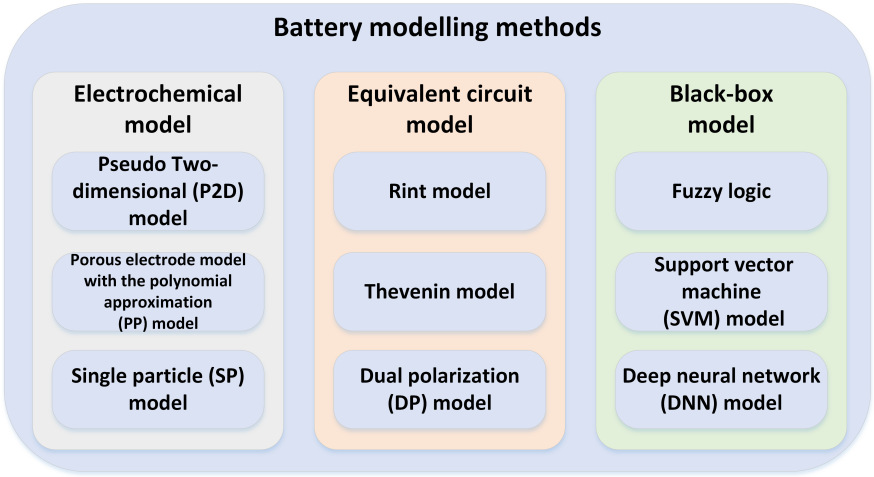


Figure 2.1: The classification of battery modelling methods.

Batteries are electrochemical systems, the electrochemical models can provide full information on the battery internal electrochemical dynamics. Electrochemical models can explain how the potential is produced and affected by the internal electrochemical reactions of the cell using a set of coupled partial differential equations. The electrochemical models are regarded as the most accurate among all battery models, as they interpret key behaviours of battery cells at the microscopic scale based on the chemical reaction taking place inside the battery. However, the electrochemical models are difficult to model because they require detailed first principle knowledge and the computational complexity restricts their real-time applications. For real-time calculation, it is important to strike a balance between the model accuracy and the model complexity. ECMs are one of the most commonly used battery models because they are accurate and not as com-

plicated as electrochemical models. The ECMs are expressed by a combination of voltage and current sources, capacitance, and resistance, where the parameters like capacitance and resistance are prescribed offline and unchanged without adaption during operation. However, these parameters vary with the battery working conditions and battery aging, the non-adaption of these varying parameters will degrade the model accuracy and may introduce errors in battery state estimation [48]. Black-box models are empirically established from experimental data fittings to infer relationships between various battery parameters such as the terminal voltage, throughput current, temperature and SOC. Their equations rarely describe battery properties. Compared to ECMs and electrochemical models, though they are not physically interpretable, they are simple and easy to implement. In this study, a black-box mathematical model is established to describe the relationship between the terminal voltage and SOC, which is used in combination with the coulomb counting method to form a state space model for filter-based SOC estimation.

2.1.1 Electrochemical Model

Newman's group has pioneered the work of the electrochemical model. In 1993, the Pseudo Two-dimensional (P2D) model was developed for lithium-ion batteries based on the porous electrode theory and the concentrated solution theory [49; 50]. The P2D model uses a set of nonlinear and coupled partial differential equations and algebraic equations, which including the mass and charge conservation in electrodes and electrolyte, the corresponding boundary conditions, and the electrochemical kinetics equations, to describe the solid and electrolyte dynamics of lithium-ion batteries. To this day, this model still remains the most popular Li-ion battery model and has been thoroughly tested and validated. However, the number of parameters and equations involved in the P2D model makes the model highly computationally complex, limiting the model real-time application. To fa-

cilitate computation simplicity, scholars have made continuous effort to develop simplified P2D models by using various mathematical (e.g., finding the analytical or approximate solution of each partial differential equation by mathematical method) and physical (e.g., proposing new hypotheses to eliminate part of the electrochemical process) simplification methods.

The PP model is a porous electrode model that incorporates the parabolic approximation into the P2D model, the resultant PP model retains the complexity of the previous P2D model, but is mathematically simpler [51; 52]. To further reduce the computational time without compromising accuracy, the single particle (SP) model was proposed, which treats each electrode as a single spherical particle whose surface area is equivalent to the active area of the solid porous electrode and neglects the presence of the liquid electrolyte [53; 54; 55]. That is, the SP model treats the transport phenomena in a simple manner and remove the partial differential equation of the liquid electrolyte phase [52]. It has been confirmed that the calculation speed of this simplified model is thousands of times faster than that of the conventional P2D model [56]. Though the SP model adequately describes the general charge-discharge behavior of the battery, they are not suitable for batteries with high charge/discharge current rates, thick electrodes, and lowly conductive electrodes, which is mainly because of the absence of the electrolyte physics [57; 58].

There is little doubt that good electrochemical models are the most accurate among all battery models, as they are capable of comprehensively describing the battery spatiotemporal dynamics. However, the complexity and high computational requirements for solving the electrochemical model equations in comparison to other battery models limit their real-time application. Thereby, the equivalent circuit models, which have flexible structures, fast computational speed, and simple control design, appear to be promising for the model-based real-time battery state estimation and have gained a lot of interest in automotive applications.

2.1.2 Equivalent Circuit Model

To build models that are accurate enough but not as complicated as the electrochemical models, the equivalent circuit modelling approaches are investigated. The electrical equivalent circuit models provide a great trade-off between the model accuracy and the computational complexity, which utilize the electrical circuit elements to mimic battery dynamics without considering the physical basis of the batteries. Compared with the electrochemical models, the ECMs are easy to understand, offer fast simulation of the system, require few computational resources, and can be embedded in very basic (and cheap) microcontrollers [59]. As presented in Figure 2.1, the ECMs mainly include the Rint model, Thevenin model, dual polarization (DP) model and their revisions. The circuit diagrams of different ECMs are illustrated in Figure 2.2 [60].

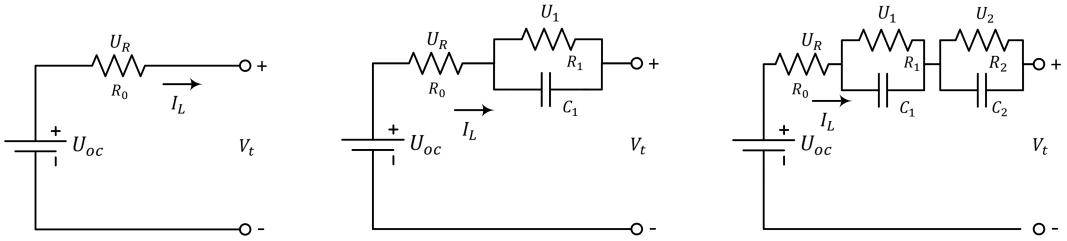


Figure 2.2: Schematic diagram of ECMs:(a) The Rint model; (b) The Thevenin model; and (c) The DP model [60].

Where I_L is the load current, V_t is the terminal voltage, R_0 refers to the battery internal resistance, U_{oc} is the open circuit voltage which is a function of SOC, R_1, R_2 are the equivalent polarization resistances and C_1, C_2 are the equivalent polarization capacitances to model the polarization characteristic, U_R, U_1, U_2 are the over-potential voltages across R_0, R_1 and R_2 respectively. The discretised voltage equations and the parameters θ for these three models are listed below [60]:

Rint Model

$$V_{t,k} = U_{oc}(SOC_k) - I_{L,k}R_0 \tag{2.1}$$

$$\theta = R_0 \tag{2.2}$$

Thevenin Model

$$U_{1,k+1} = U_{1,k}e^{\frac{-\Delta t}{\tau_1}} + I_{L,k}R_1\left(1 - e^{\frac{-\Delta t}{\tau_1}}\right) \tag{2.3}$$

$$V_{t,k} = U_{oc}(SOC_k) - I_{L,k}R_0 - U_{1,k} \tag{2.4}$$

$$\theta = [R_0, R_1, \tau_1]^T \tag{2.5}$$

DP Model

$$U_{1,k+1} = U_{1,k}e^{\frac{-\Delta t}{\tau_1}} + I_{L,k}R_1\left(1 - e^{\frac{-\Delta t}{\tau_1}}\right) \tag{2.6}$$

$$U_{2,k+1} = U_{2,k}e^{\frac{-\Delta t}{\tau_2}} + I_{L,k}R_2\left(1 - e^{\frac{-\Delta t}{\tau_2}}\right) \tag{2.7}$$

$$V_{t,k} = U_{oc}(SOC_k) - I_{L,k}R_0 - U_{1,k} - U_{2,k} \tag{2.8}$$

$$\theta = [R_0, R_1, R_2, \tau_1, \tau_2]^T \tag{2.9}$$

Where k refers to the kth sample time, Δt is the sampling period, the time constants of the two RC networks are $\tau_1 = C_1R_1$ and $\tau_2 = C_2R_2$.

As shown in Figure 2.2(a), The Rint model is the simplest ECM, which consists of an ideal voltage source in series with a resistor. To simulate the diffusion process and polarization effects, the parallel-connected resistor-capacitor (RC) networks are utilized to form more complicated ECMs [61], and the number of RC networks varies from 1 to n according to the dynamics of the load profile and the required modelling accuracy. The model order refers to the number of RC networks, and the first and second-order models are commonly used RC models for SOC estimation and power predictions [60]. The first-order model, also

called the Thevenin model, connects a parallel RC network in series with the Rint model (Figure 2.2(b)). The single RC network describes the transient response during battery charging and discharging process [62]. The second-order model, also known as the DP model, connects two parallel RC circuits in series with the Rint model (Figure 2.2(c)), which can refine the description of the transient response of the battery caused by transfer, diffusion and other factors [63]. The polarization resistance R_1 and the polarization capacitance C_1 characterize the electrochemical polarization, while R_2 and C_2 are used to simulate the concentration polarization [64; 65]. Higher-order models have also been investigated, as adding more RC networks may improve the model accuracy. However, more RC networks involve more parameters, and thus increase the model complexity and require more computational effort. Therefore, it is important to get a compromise in the balance of computational effort and model accuracy [66]. The effect of the number of RC networks on the model accuracy was systematically evaluated in [63], which concluded that the DP model has optimal simulation performance among the investigated battery models.

Accurate model parameters identification methods are necessary for the efficient utilization of ECMs. They can be classified into two categories, traditional offline identification methods and online methods. The traditional offline methods, such as curve fitting and recursive least squares (RLS) method, have been extensively utilized in online SOC estimation. However, the parameters of ECMs would change according to the operating conditions, such as temperature, charge/discharge rate, SOC and aging level, it is injudicious to utilize the invariant parameters [60; 67]. In this regard, ECM parameters need to be recalibrated regularly to ensure their extensibility for different operating conditions. Thus, online recursive identification algorithms are more appropriate. Plett introduced dual estimation methods to concurrently estimate the battery state and time-varying model parameters using extended Kalman filtering algorithm [68] and

unscented Kalman filtering algorithm [69]. In [48], the model parameters were adapted online using the RLS method, and the battery SOC and capacity were estimated simultaneously with the joint EKF, which fully decoupled the model identification and state estimation processes to eliminate the cross interference. Considering that the battery behaviors consist of both fast dynamics (FD) and slow dynamics (SD) and estimating all the parameters together will suffer from numerical problems and poor accuracy, Zhang et al. [70] proposed a novel online parameter identification method based on weighted RLS to decouple these two dynamics, and estimate the FD and SD parameters separately. The DP model was used in this paper and it is assumed that the time constant τ_1 is less than τ_2. Herein, the parameters R_0, R_1, C_1 are regarded as the FD part, which describe the battery internal resistance, charge transfer, and double layer effect respectively. While R_2 and C_2 are considered as the SD parameters. In [71], nine different parameters identification methods were used and compared for nine ECMs, the results suggested that PSO is an ideal choice for online parameters identification of second-order RC models. And higher-order RC models are not suitable for online parameter identification.

2.1.3 Black-box Model

It is not trivial to establish the precise battery model using the electrochemical model and ECM approaches due to the complicated electrochemical reactions inside the cells and the uncertain external operating conditions. The data-driven methods appear to be powerful tools for accurate battery model establishment and have been extensively researched in recent years. They are easy to use and are free of a priori knowledge of battery dynamics, and are suitable for different battery systems and operating conditions as long as sufficient, diverse, and comprehensive training data are available. The machine learning techniques have merits of flexibility, reliability, strong adaptability and generalizability, and can

approximate highly nonlinear dynamics with desired accuracy.

Diverse ML techniques have been exploited in battery modelling and state estimation. In [72], the practicable capacity was used as the input to a radial basis function neural networks (RBFNN) along with terminal voltage and current, which considering the effect of battery aging on SOC estimation. The model can offer good robustness under varying temperatures, different loading profiles, and different degradation levels. Hannan et al. [73] used the terminal voltage, current and temperature as inputs to a back-propagation neural network to estimate SOC. The number of neurons in the hidden layer and the learning rate of this model were optimized by backtracking search algorithm to improve the model accuracy and robustness. Chemali et al. [74] used a 4-layer deep neural network to directly map the battery measurements to SOC. By using the temperature as one of the inputs, the intrinsic behavior of the battery under different ambient temperatures can be encoded into the weights of the network such that the trained model is capable of providing accurate and robust estimation results at different operating temperatures. Furthermore, by forming a state space model using the coulomb counting method and ML techniques, the closed-loop SOC estimation can be realised using filtering algorithms or observers. Similar work has been reported in several papers, which use machine learning techniques such as feedforward neural network [75], RBFNN [76], least squares support vector machine [77], and long short-term memory (LSTM) [78] to capture the nonlinear relationship between the terminal voltage and battery SOC and estimate the SOC using filters or observers. Since the filtering algorithms are implemented for the state-space model, which is a discrete-time, stochastic model that consists of a state function that describes how a state transitions in time, and a measurement function relating the measured data to a state, the battery model is represented in the form of state space model. Therefore, the resulting ML model was normally treated as the measurement function of a state space model for battery

SOC estimation, while the formula of the coulomb counting method was treated as the state function.

Though ML-based methods have benefits such as high modeling accuracy, strong adaptability, and generalizability, the model performance can be severely affected by the optimization methods and the quality of the data adopted. Generally, an immense number of data are required to train a model suitable for various operating conditions.

2.2 Review of battery SOC estimation approaches

SOC is the ratio of remaining capacity ($Q_{remaining,k}$) to maximum available capacity (Q_{max}) of the battery, as expressed in Equation (2.10).

$$SOC_k = \frac{Q_{remaining,k}}{Q_{max}} \times 100\% \tag{2.10}$$

Where SOC_k is the SOC at time k, $Q_{remaining,k}$ is the releasable capacity at time k. Q_{max} is the maximum possible charge that can be extracted from a fully charged battery in the current cycle, which decreases as the battery degrades. For ease of implementation, battery nominal capacity is usually used as reference Q_{max} value, however, it is not applicable for aged cells and needs to be updated using efficient SOH estimation methods. Figure 2.3 illustrates the stored energy status at different SOCs, where 100% and 0% represent the fully charged and fully discharged conditions of the battery, respectively. The empty portion can be recharged and the inactive portion reflects the permanent loss due to aging.

The approaches for battery SOC estimation can be classified into two main categories, namely direct and indirect methods. As shown in Figure 2.4, for the direct estimation approaches, SOC is mainly calculated by open circuit voltage (OCV) based method, which is a look-up table based method, and coulomb counting

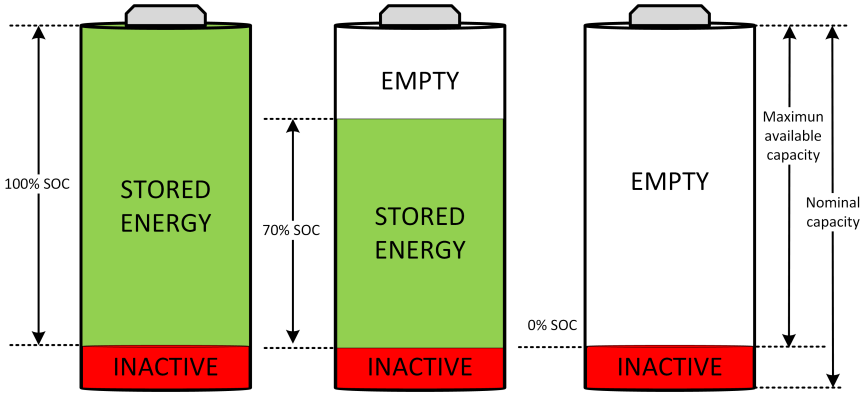


Figure 2.3: Battery stored energy status at different SOCs (100%, 70%, and 0%).

method. Whereas the OCV method requires a very long resting time with no load attached to measure the OCV, the coulomb counting method suffers from cumulative errors caused by sensor noise and offsets induced by initialization [79]. Therefore, the model-based method has been developed for online battery SOC estimation. In the model-based estimation method, an effective battery model is crucial for estimation accuracy. Battery models, such as equivalent circuit model (ECM) and electrochemical model, in the form of standard state space, are usually selected to estimate battery states. However, as introduced in Section 2.1, the complexity of batteries internal physicochemical reactions and the uncertainty of external operation makes it difficult to build accurate battery models. The machine learning-based estimation methods can be employed to model the nonlinear relationship between the measurements and SOC without requiring any prior knowledge of the system.

2.2.1 Coulomb counting method

The coulomb counting method, also known as Ampere-hour (Ah) integral method, is to calculate the SOC by integrating the measured current with respect to time

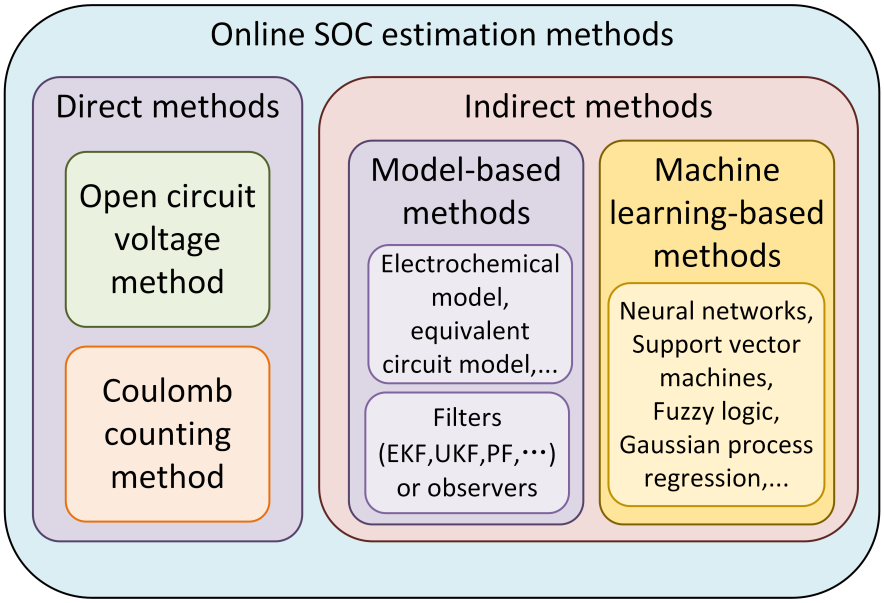


Figure 2.4: Categories of online SOC estimation methods.

while the battery is charging or discharging [80], it can be expressed as:

$$SOC_k = SOC_0 - \frac{\int_{t_0}^{t_k} \eta I_{k-1} dt}{Q_{max}} \tag{2.11}$$

where SOC_k and SOC_0 denote the SOC at time t_k and t_0 respectively. t_0 is the initial time, $t_k = t_0 + k \times \Delta t$, Δt denotes the sampling interval. I_{k-1} is the current at time $k-1$, η is coulombic efficiency. The maximum available capacity Q_{max} usually uses the value of the nominal capacity. Conventionally, the current direction is negative ($\text{sign}(I) < 0$) during charge and positive during discharge ($\text{sign}(I) > 0$).

Although it is a simple and straightforward SOC estimation method, the shortcomings are apparent due to its open-loop nature [81]. The error in current measurements will be inevitably accumulated over time due to the time integration of current, which can significantly affect the estimation accuracy. Besides, the initial SOC is difficult to be accurately determined in real-time applications especially when the battery is only charged/discharged within a limited range.

Moreover, operating conditions such as aging and temperature can make the actual maximum available capacity deviate from the nominal capacity and affects the coulombic efficiency η, leading to inaccurate SOC estimation. Therefore, the coulomb counting method is more suitable to work in combination with other supporting techniques, such as the model-based methods, to enhance the robustness [82].

2.2.2 Open-circuit voltage method

OCV is the thermodynamic equilibrium potential of battery under no load condition [83] and its relationship with SOC can be established by step-wise measuring OCV for different values of SOC [60]. That is, the OCV method infers the SOC using an offline predefined OCV-SOC relationship.

This method has inherent difficulties in practical applications. The relationship differs among batteries, even though they are fabricated from the same materials and structures [84]. However, it is a very time-consuming process to measure the OCV of each battery at each SOC, as reaching a satisfactory equilibrium condition to measure an OCV requires a long resting time. Besides, the OCV method is an open-loop estimator sensitive to measurement errors and uncertain disturbances. Some kinds of batteries, in particular lithium iron phosphate batteries, have a relatively flat OCV-SOC curve, which means that a small OCV measurement error may lead to a large SOC deviation [85]. Therefore, a voltage sensor with extremely high accuracy is needed. Furthermore, taking into account the influence of operating conditions, such as temperature and aging, on the OCV-SOC relationship, various modified OCV-SOC tables have been proposed. For example, by adding the temperatures to the OCV-SOC table, an offline OCV-SOC-temperature table was established in [86]. All these difficulties hinder the practical application of the OCV method, make it more suitable for laboratory application [82].

2.2.3 Model-based Methods

In the model-based method, different types of battery models, which are introduced in Section 2.1, are usually combined with the coulomb counting method to form a state space model, where the SOC is a state variable that acts as a bridge between the state equation and the measurement equation. Then based on the state space model, various filtering algorithms or observers are employed to estimate or infer the battery SOC. For example, in [87; 88], sliding mode observer is used to estimate the SOC. In [68; 89; 90; 91; 92], the battery SOC is estimated by the extended Kalman Filter (EKF) algorithm based on different battery models. And the Unscented Kalman Filter (UKF) algorithm has been investigated to estimate the SOC in [69; 93; 94; 95]. The results show that the UKF method is superior to the EKF method in SOC estimation. The adaptive EKF method improves the accuracy of the EKF-based SOC estimation by adaptively updating the noise covariance, for example, the covariance matrix of the process noise is estimated using Maybeck's estimator in [76], Sage-Husa adaptive filter algorithm is used to estimate the noise covariance in [96], and in [97], the terminal voltage and model parameters are estimated online by an adaptive EKF algorithm, where the filter's innovation sequence is utilized. An adaptive UKF method is presented for lithium-ion battery SOC estimation in [98], the adaptive adjustment of the noise covariances in the SOC estimation process is implemented by the idea of covariance matching in the unscented Kalman filter context. Compared with EKF, UKF and AEKF, the AUKF can estimate the SOC more accurately. Gao et al. [99] employed particle filter (PF) for SOC estimation and the computation time of their algorithm is six times faster than EKF. An unscented PF (UPF) algorithm is introduced by He et al. [100] for the estimation of SOC of high-power lithium-ion batteries. The numerical calculations show an improvement in UPF over UKF in minimizing Root Mean Squared Error and Maximum absolute error. H_∞ filter is used to estimate the SOC of lithium-ion batteries in [101; 102].

Adaptive H_∞ filter is presented to estimate the SOC of a lithium-ion battery in [103], and in comparison with AEKF and square-root UKF, it offers better overall performance regarding the estimation accuracy, computational cost and time efficiency.

Figure 2.5 is a schematic diagram of model-based SOC estimation methods. The input for both real battery system and battery model can be the loading current and ambient temperature. In most cases, the terminal voltage is used as the actual measurement. The principle of the model-based SOC estimation method is to calculate the difference between the actual measurement and the value estimated by the measurement equation of the state space model for generating a measurement residual, and then feed it back to the model through a filter or observer for updating the model parameters and states.

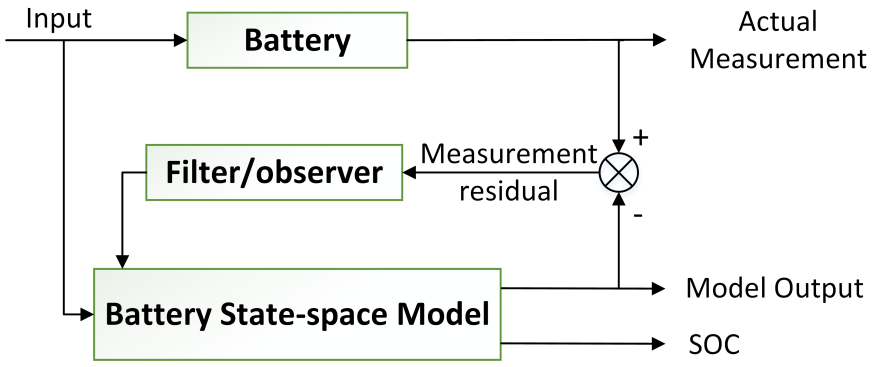


Figure 2.5: The schematic of model-based SOC estimation methods.

As a closed-loop estimation method, the model-based SOC estimation method exhibits desirable advantages and is widely used. It is insensitive to the initial SOC value, and can achieve accurate online estimation results as long as the model is precise enough. However, to build an accurate battery model is always time-consuming and laborious, and requires a priori knowledge of the battery system. Therefore, the model-based method is not applicable to all types of battery [104].

2.2.4 Machine Learning-based Methods

With the advancement of computer technologies, machine learning techniques have been increasingly employed to estimate the battery SOC. Techniques such as artificial neural networks (ANN) [73; 105; 106], deep neural networks [74; 107], recurrent neural networks [108; 109], fuzzy logic [110], and Gaussian process regression (GPR) [111; 112] can directly map the measured signals (e.g. current, terminal voltage, and surface temperature) to the SOC. Conventionally, the direct measurements are usually considered as model inputs to calculate the SOC, which is the model output. The main benefit of the ML-based methods is that a priori knowledge of the battery dynamics is no longer required. Besides, various real-world operating conditions can be considered during the model training process by adding additional input to the model, and this method is suitable for all types of batteries [60]. The main problem of the ML-based SOC estimation methods is the high demand for the huge amount of training data. Besides, due to the implementation of low-cost microcontrollers in BMSs, the model size and computational complexity need to be taken into account.

2.3 Review of battery SOH estimation approaches

The SOH can be used to recognize the dynamic health status and evaluate the aging or degradation level of the battery. It is a parameter that reflects the present condition of the battery described in percentage, with 100% being a fresh cell. Due to the battery degradation caused by irreversible internal chemical reactions and physical processes, the SOH decrease with time and usage of the battery. Battery capacity and internal resistance are the two most used SOH indicators, which reflect the energy capability and power capability respectively [113; 114]. According to the study of internal aging mechanisms, the capacity fading is mainly caused by the loss of lithium inventory and loss of active material

of the electrodes [115]. And the increase of resistance is mainly caused by solid electrolyte interface (SEI) layer formation and growth [114]. Most studies use the actual capacity to evaluate the SOH, and the SOH is defined as:

$$SOH = \frac{Q_{max}}{Q_N} \times 100\% \tag{2.12}$$

where Q_{max} denotes the actual maximum available capacity that a battery can supply from a fully charged state, and Q_N is the battery nominal capacity. Generally, a battery cell is regarded as being at the end of life and need to be replaced when the actual capacity decreases to 80% of the nominal capacity [114].
Numerous SOH estimation approaches have been reported in the literature, which can be roughly categorized into three groups: model-based, Incremental Capacity Analysis (ICA)-based, and machine learning-based approaches. A schematic of the available battery SOH estimation methods is illustrated in Figure 2.6.

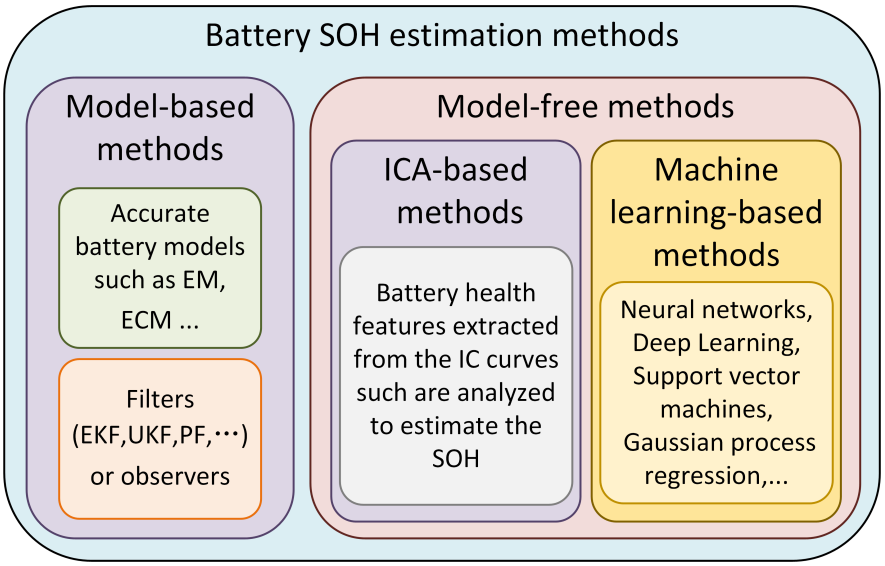


Figure 2.6: Categories of online SOH estimation methods.

2.3.1 Model-based Methods

In model-based battery SOH estimation approaches, physical-based electrochemical models [116], empirical models [117], thermal models [118] and fusion models [119] are often used in conjunction with observers or adaptive filtering algorithms to achieve online capacity estimation. For example, Yu et al. [120] used the adaptive H_∞ filter to estimate the battery capacity based on the Thevenin model with online identified parameters. Zheng et al. [121] estimated the capacity using proportional-integral observers based on an electrochemical model. Xiong et al. [122] extracted five parameters that are strongly correlated with battery aging from the electrochemical model to estimate the capacity. Zheng et al. [123] applied sequential extended Kalman filters to estimate capacity based on the empirical model. Different types of battery models were compared and analysed in [124]. And three widely used filtering algorithms for estimating the capacity, i.e. extended Kalman filter, particle filter and recursive least squares, were investigated in [125], and their performances are compared and analysed in terms of accuracy and convergence speed. Furthermore, considering that the battery SOH is a slow time-varying state, while SOC and other model parameters have fast time-varying dynamics, multi-scale estimation frameworks have been investigated to estimate the SOH and fast-varying parameters/states separately, such as multi-scale EKF [126], multi-scale dual H_∞ filter [127], and multi-scale dual adaptive PF [128]. These frameworks usually use a micro-scale filter/observer/least square method to estimate the SOC or identify the parameters, and use a macro-scale filter/observer to estimate the battery SOH. The model-based approaches have the merits of achieving fairly accurate SOH estimation and sometimes offering clear physical insights into the battery dynamics. Notwithstanding, their performance highly depends on the model quality, whereas building an accurate battery model is not an easy task as the requirement of a large amount of prior knowledge is not always available.

2.3.2 Incremental Capacity Analysis-based Methods

The differential analysis-based methods correlate the features extracted from the differentiated curves of some electrical, thermal, or mechanical parameters with battery capacity fade. For example, incremental capacity (IC) analysis and differential voltage (DV) analysis have been frequently used in recent years as effective tools to analyze the battery behavior and fading mechanism [114; 129]. IC is calculated as the derivative of the capacity with respect to its terminal voltage (dQ/dV) while the battery is charged/discharged under a small and constant current rate. The DV curves (dV/dQ) is defined as the inverse of IC. A schematic diagram of ICA-based battery SOH estimation methods is shown in Figure 2.7. The voltage plateaus shown in the V-Q curve can be easily identified from the IC curves after the differential operation. The features extracted from the curves such as IC peak position, peak magnitude, peak shape, corresponding peak voltage/SOC, and peak area, are analyzed to estimate the battery capacity. For example, [130] have extracted five different features from the IC curves, the first two are peaks and the last two are valleys, the rest is the shoulder of the IC curves. The capacity is estimated by analyzing the position, value and associated area changes of these features. As described in [131], the IC peak values are tracked to estimate the capacity for single cells as well as battery packs. In [132], three corresponding SOC positions are extracted from the SOC based IC and DV curves for battery capacity estimation. While [133] use a regional voltage, which is calculated by the terminal voltage corresponding to the IC peak, for fast capacity estimation. However, the IC/DV analysis is sensitive to measurement noise and subject to operation temperature, further, it requires very low current rate, therefore their applications are severely constrained [134].

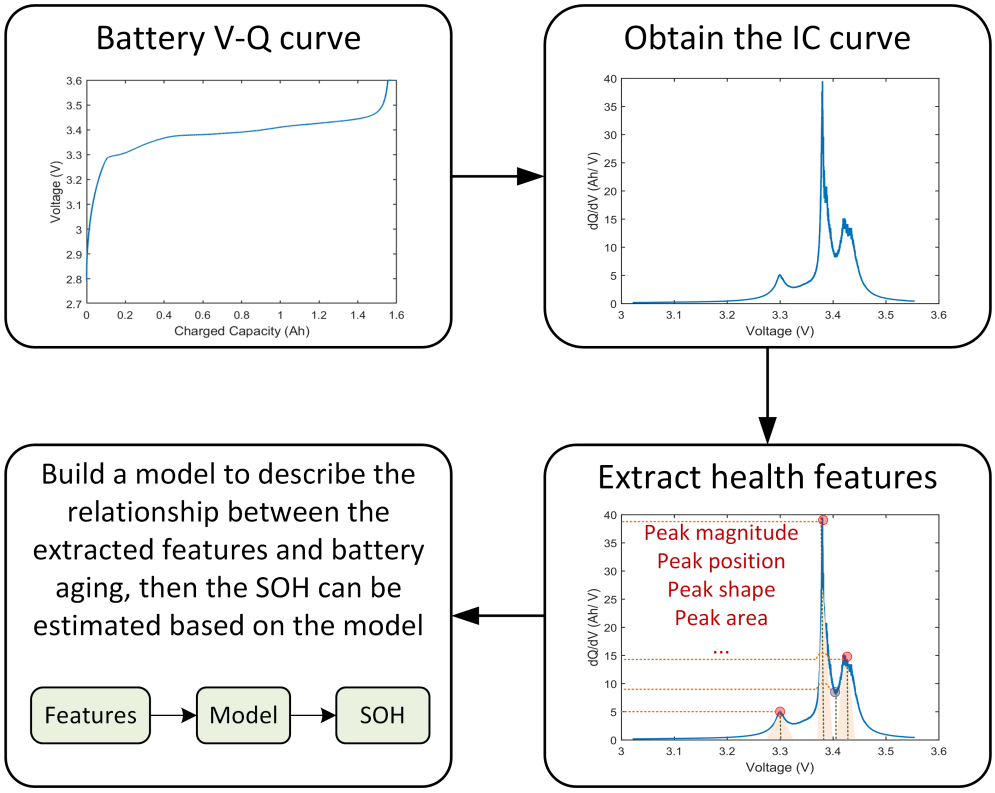


Figure 2.7: The schematic diagram of ICA-based SOH estimation methods.

2.3.3 Machine Learning-based Methods

Taking into account the limitations of model-based and ICA-based methods caused by complex internal principles and uncertain working conditions, machine learning-based battery capacity estimation methods have been extensively studied in recent years. These methods can handle data with nonlinear relationship without requiring prior knowledge on the complex physical principles of the battery and are easy to implement. Methods such as GPR [135], kernel ridge regression [136], support vector machine (SVM) [137], and support vector regression [138], just to name a few, have been successfully applied in battery capacity estimation. Among these methods, Liu et al. [139] employed a GPR-based framework to simultaneously predict capacity and quantify the uncertainty of predicted values. Li et al. [140] proposed a GPR-based method to estimate

the battery SOH using the health features extracted from partial incremental capacity curves. The accuracy, robustness and effectiveness of this approach are verified on four batteries from the NASA battery degradation dataset. Feng et al. [141] developed a SVM-based online SOH estimation method. The SVM model is first established offline using features extracted from the battery charging curves of cells at different SOHs. Then the model is employed for online SOH estimation by comparing the features of the measured charging voltage segment and the stored models. Guo et al. [142] extracted health features from charging measurements, and eight features which are most relevant to the capacity degradation are selected as the inputs to a relevance vector machine (RVM) model. It is worth noting that the performance of these methods heavily relies on the manually extracted features, which requires significant manual and computational efforts, and the resultant model is often application specific and may not be generalized. To address this bottleneck, some other machine learning-based techniques such as Elman neural network [143], LSTM [144], random forest [145] and Convolutional Neural Network (CNN) [146] have been applied to automatically extract features from the measurements. The Elman neural network was applied in [143] to estimate the battery capacities of the following cycles until its end-of-life in real-time, only using the charged capacities in the past cycles as inputs of the network. A LSTM network was designed in [144] for capacity estimation using direct measurable data, i.e. current and voltage, and the model's robustness and flexibility in dynamic environments has been extensively verified with data collected from more than seventy Lithium-ion batteries cycled with more than ten driving profiles. In [145], the charge capacity recorded in a specific voltage region was used as the input to a random forest regression model, the trained model can directly use online recorded data for capacity estimation. And [146] directly used the voltage, current and charge capacity of each cycle as inputs to train a CNN model for capacity estimation. The performance of three different neural

network models for capacity estimation, i.e. feed-forward neural network, LSTM and CNN, were compared in [147], and the test results revealed the difficulty of the resultant models in dealing well with limited available battery data. It is clear that these machine learning-based methods have shown great potentials in battery capacity estimation, yet their performance is heavily dependent on the size of the training dataset. Only models trained with sufficient data are capable of achieving satisfactory accuracy. However, to collect a large battery degradation dataset requires a substantial number of cycling tests, which is extremely time-consuming and costly. To this end, transfer learning techniques can be incorporated into these methods to improve the estimation performance on small dataset. In [148], the battery health was estimated by combining the kernel ridge regression and transfer learning to improve the prediction accuracy. In [149], transfer learning was applied to achieve accurate, quick and steady prediction based on a LSTM model. A CNN model combining the concepts of transfer learning and ensemble learning was used for capacity estimation in [150] with voltage, current and charge capacity as the inputs of the network. Similar work has been reported in [151].

The disadvantages of the ML-based methods lie in the big computation cost incurred during the training process [152], the requirement of large training datasets which is time-consuming and laborious, and relatively high model complexity.

2.4 Chapter Summary

This chapter provides a systematic literature review on battery modelling and state estimations. The states of interest in this thesis are SOC and SOH. As model-based methods are one of the most popular state estimation methods and accurate battery models are indispensable for these methods, this chapter starts with an introduction to three commonly used battery models, namely electro-

chemical models, equivalent circuit models and black-box models. Then a technical review on SOC and SOH estimation methods is presented. The model-based methods are featured by closed-loop nature and are insensitive to initialization error and uncertain disturbances. The machine learning-based methods have shown great potential for battery state estimation since a huge amount of batter test data will be available with the advent of the big data platform of batteries. However, to implement the estimation method on BMSs with limited computational capability, the issues existing in different estimation methods still need to be further addressed.

Chapter 3

Battery State of Charge Estimation

Battery state of charge (SOC) is an important parameter that provides information about the battery real-time remaining capacity. As an unobservable quantity, SOC estimation is vital for the battery management system to guarantee the operation safety and reliability of batteries. However, the battery is often exposed to highly dynamic load demands, it is not easy to estimate the SOC accurately. To infer SOC from the measured signals, such as current, terminal voltage and temperature, is still a challenging task due to the highly nonlinear and non-stationary battery dynamics and the sophisticated battery internal chemical reactions. During the last decades, a large number of state estimation methods have been developed for battery SOC estimation [153], as summarized in Chapter 2, these methods can be roughly classified into three groups: conventional methods including Coulomb counting and open-circuit voltage based techniques, model-based methods, and machine learning methods. In this chapter, an integrated approach is proposed for battery SOC estimation, which construct a state-space model using fast recursive algorithm to correlate the SOC with terminal voltages and estimate the SOC using filtering algorithms.

3.1 Model Construction

To estimate the SOC using model-based filtering method, a state space model is first established, while the SOC is defined as an independent state variable in the state space model. As summarized in Chapter 2, electrochemical models (EMs), equivalent circuit models (ECMs), and black-box models are the commonly used battery models. The EM is capable of reflecting the electrochemical reactions inside the battery, however, it requires in-depth knowledge of the battery and to identify all parameters is a very complicated process. The ECMs are proposed to strike a balance between model complexity and accuracy, and they describe the charging and discharging processes using a combination of voltage source, capacitances, and resistances. The parameters of the ECM change under different operating conditions and need to be updated either by look-up tables or by polynomial functions. Compared to the ECMs and EMs, black-box models do not require detailed first principle knowledge of the battery and and can describe the nonlinear relationship between the battery SOC and terminal voltage with desired accuracy. In this chapter, the fast recursive algorithm is used to determine the model structure and identify the model parameters.

3.1.1 Fast recursive algorithm

In the field of system identification, the linear-in-the-parameter model is a popular model structure for approximating a large class of nonlinear systems. These models linearly combine a set of model terms that are nonlinear functions of the system variables. Such models have an excessive number of candidate terms which may cause overfitting and high computational complexity, therefore, model selection algorithms have been proposed to generate parsimonious models with a much smaller number of terms. In particular, a fast recursive algorithm (FRA) [154] was proposed to simultaneously select most significant model terms and

estimate model parameters.

Consider a nonlinear discrete-time dynamic system represented by a linear-in-the-parameters model, which is identified by N data samples $\{x(i), y(i)\}_{i=1}^{N}$

$$\mathbf{y} = \mathbf{\Psi\Theta} + \mathbf{\Xi} \tag{3.1}$$

Where $\mathbf{y} = [y(1), ..., y(N)]^T \in \Re^N$ denotes the system output, $\mathbf{\Psi} = [\varphi_1, ..., \varphi_j, ..., \varphi_S] \in \Re^{N\times S}$ is the regression matrix that contains all candidate model terms, each term $\varphi_j \in \Re^{N\times 1}, \varphi_j = [\varphi_j(x(1)), ..., \varphi_j(x(N))]^T (j = 1, ..., S)$ represents a nonlinear function of N input samples, $\mathbf{\Theta} = [\theta_1, ..., \theta_S]^T$ are the unknown parameters to be identified, and $\mathbf{\Xi} = [\varepsilon_1, ..., \varepsilon_N]^T$ is the model residual vector.

Two recursive matrices, namely information matrix M_k and residual matrix R_k, are predefined in FRA to fulfill the forward model selection procedure as:

$$\mathbf{M}_k = \mathbf{\Psi}_k^T \mathbf{\Psi}_k \tag{3.2}$$

$$\mathbf{R}_k = \mathbf{I} - \mathbf{\Psi}_k \mathbf{M}_k^{-1} \mathbf{\Psi}_k^T \tag{3.3}$$

where $\mathbf{\Psi_k} \in \Re^{N\times k}$ contains the first k columns of the full regression matrix $\mathbf{\Psi}$, additionally, $k = 1, ..., S$, and $\mathbf{R}_0 = \mathbf{I}$.

Thus, when the first k columns in $\mathbf{\Psi}$ are selected, the estimation of parameters that minimizes the cost function and the associated minimal cost function can be formulated as

$$\hat{\mathbf{\Theta}}_k = \mathbf{M}_k^{-1} \mathbf{\Psi}_k^T \mathbf{y} \tag{3.4}$$

$$E_k = \mathbf{y}^T \mathbf{y} - \hat{\mathbf{\Theta}}_k^T \mathbf{\Psi}_k^T \mathbf{y} \tag{3.5}$$

When $\{\varphi_j, j = 1, ..., S\}$ in $\mathbf{\Psi}$ are mutually linearly independent, the residual matrix $\mathbf{R}_k$ will have the following distinguished properties:

$$\mathbf{R}_{k+1} = \mathbf{R}_k - \frac{\mathbf{R}_k \varphi_{k+1} \varphi_{k+1}^T \mathbf{R}_k^T}{\varphi_{k+1}^T \mathbf{R}_k \varphi_{k+1}}, \ k = 0, 1, ..., (S-1) \tag{3.6}$$

$$\mathbf{R}_k^T = \mathbf{R}_k, \quad \mathbf{R}_k \mathbf{R}_k = \mathbf{R}_k, \quad k = 0, 1, ..., S \tag{3.7}$$

$$\mathbf{R}_k \mathbf{R}_j = \mathbf{R}_j \mathbf{R}_k = \mathbf{R}_k, \quad k \geq j; \ k, j = 0, 1, ..., S \tag{3.8}$$

$$\mathbf{R}_k \varphi_j = 0, \quad \forall j \in \{1, ..., k\} \tag{3.9}$$

Now, Equation (3.5) can be expressed as

$$E_k = \mathbf{y}^T \mathbf{R}_k \mathbf{y} \tag{3.10}$$

To simplify the formulas and decrease the computational complexity, three quantities are consequently defined as

$$\begin{cases} \varphi_j^{(k)} \triangleq \mathbf{R}_k \varphi_j, \quad \varphi_j^{(0)} \triangleq \mathbf{R}_0 \varphi_j = \varphi_j \\ a_{k,j} \triangleq \left(\varphi_k^{(k-1)}\right)^T \varphi_j^{(k-1)}, \quad a_{1,j} \triangleq \varphi_1^T \varphi_j \\ b_k \triangleq \left(\varphi_k^{(k-1)}\right)^T \mathbf{y}, \quad b_1 \triangleq \left(\varphi_1^{(0)}\right)^T \mathbf{y} = \varphi_1^T \mathbf{y} \end{cases} \tag{3.11}$$

where $j = 1, ..., S$, and $k = 0, 1, ..., S$. According to the properties of $\mathbf{R}_k$ and the new quantities definition, the net contribution of a new model term φ_{k+1} to the

cost function can be explicitly calculated as

$$\Delta E_{k+1} = \frac{\left(\mathbf{y}^T \varphi_{k+1}^{(k)}\right)^2}{\left(\left(\varphi_{k+1}^{(k)}\right)^T \varphi_{k+1}^{(k)}\right)} = \frac{\left(b_{k+1}^T\right)^2}{a_{k+1,k+1}}, \quad k = 0, 1, ..., S-1 \tag{3.12}$$

By calculating the net contribution of each term, the model terms with maximum contributions will be selected one by one. Finally, after all important model terms have been selected, the parameter for each selected term is calculated as

$$\hat{\theta}_j = \frac{b_j - \sum_{i=j+1}^{k} \hat{\theta}_i a_{j,i}}{a_{j,j}}, \quad j = k, k-1, ..., 1 \tag{3.13}$$

Note: the summation term in Equation (3.13) is zero for $j = k$, i.e. $\sum_{i=k+1}^{k} \hat{\theta}_i a_{k,i} = 0$. Equations (3.12) and (3.13) constitute the main steps of the FRA, which selects model terms one by one based on (3.12) and calculates the model parameters for the resultant model based on (3.13).

3.1.2 The proposed model

In this chapter, the FRA is used to capture the nonlinear relationship among terminal voltage, battery SOC and current. Figure 3.1 shows the process of building the measurement function. Symbols I and SOC refer to discharge current and state of charge, respectively. And the subscript k means for the $k-th$ time step.

The procedure is summarized below:

Step 1 - Initialization. Identify model term pool, such as $1/SOC$, $\sqrt{SOC}$, $ln(SOC)$, e^{SOC}, $sin(SOC)$ and $cos(SOC)$, collect data and build the data matrix (information matrix).

Step 2 - Fot the $(k+1)$-th time step, the terminal voltage V_{k+1} is regarded as the model output, and the current I_{k+1} and SOC_{k+1} are used to generate the candidate model terms based on the identified model term pool.

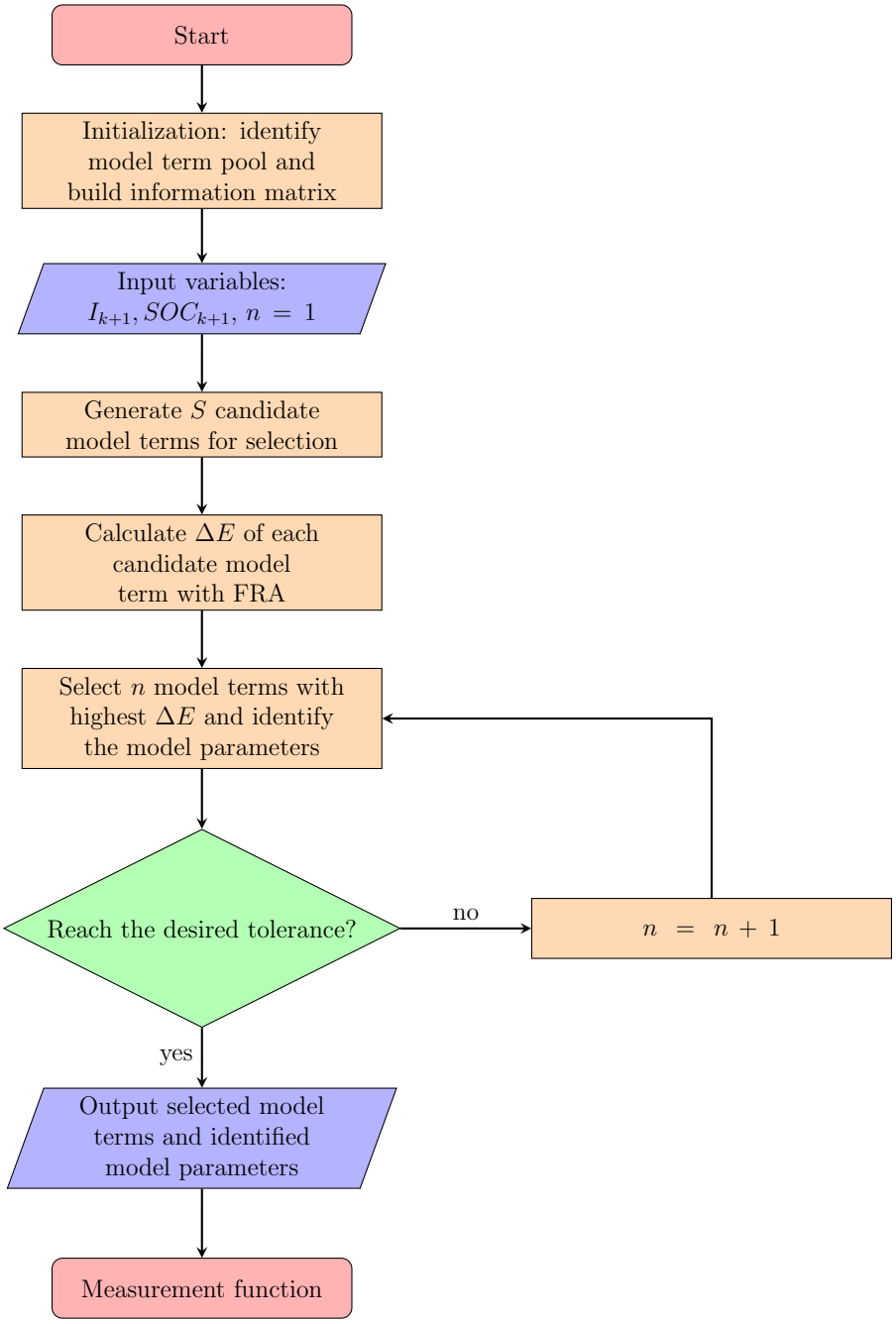


Figure 3.1: Flowchart of building measurement function based on FRA

Step 3 - Then the contribution of each term in the candidate pool is calculated using Equation (3.12).

Step 4 - Select the model term with highest contribution, and calculated its pa-

rameter using Equation (3.13). With the selected model terms and the identified parameters, the model output is obtained, denoted by $\hat{V}_{k+1}$. The root mean square error between the measured terminal voltage V_{k+1} and model output $\hat{V}_{k+1}$ over all the training samples is calculated by:

$$RMSE = \sqrt{\frac{1}{N}\sum_{k=0}^{N-1}\left(V_{k+1} - \hat{V}_{k+1}\right)^2} \tag{3.14}$$

Step 5 - Repeat Step 3 until the model accuracy is satisfactory. As a result, the model structure has been determined and the parameters of the selected model terms have been identified.

3.1.3 Battery state space model

As introduced in Chapter 2, according to Equation 2.11, the SOC at time instant $k+1$ can be calculated by:

$$SOC_{k+1} = SOC_k + \frac{\eta \Delta t}{Q} I_k \tag{3.15}$$

The coulombic efficiency η is set to 1. The SOC is considered as a hidden state variable in the state-space model. This is the state function of the state space model.

The measurement function is established according to the process described in section 3.1.2 using the fast recursive algorithm, which is expressed as follows:

$$V_{k+1} = \mathbf{\Psi_n}(I_{k+1}, SOC_{k+1})\mathbf{\Theta_n} \tag{3.16}$$

where $\mathbf{\Psi_n}(I_{k+1}, SOC_{k+1})$ refers to n model terms and consists of current, SOC, and their nonlinear forms that are selected by FRA from the candidate feature pool, and $\mathbf{\Theta_n} = [\theta_1, ..., \theta_n]^T$ is the coefficient vector for the selected nonlinear

terms.

3.2 Filtering Algorithms

3.2.1 Kalman Filter

Kalman filter (KF) [42] is a computationally efficient framework that provides optimal state estimation for linear systems or systems with linear state space model by minimizing the mean square error between the actual and estimated data. It has been extensively studied and applied in virtually every technical or quantitative field. Consider a dynamic discrete process represented by a linear stochastic equation using state vector x_k:

$$x_k = Ax_{k-1} + Bu_{k-1} + \omega_{k-1} \tag{3.17}$$

where A is the transition matrix that associates the state of the previous moment with the current state. And matrix B correlates the control input u_{k-1} with the current state of the system. The process noise ω_{k-1} is assumed to be white Gaussian noise, with zero mean and and covariance Q_{k-1}.
The measurements z_k are linearly related to the state x_k:

$$z_k = Cx_k + v_k \tag{3.18}$$

where C is the measurement matrix, and measurement noise v_k is assumed to be white Gaussian noise with zero mean and covariance R_k. The process noise ω_k is uncorrelated with the measurement noise v_k, and both of them are assumed to

be independent of the state of the system.

$$\begin{aligned} &E[\omega_k v_j^T] = 0 \\ &E[\omega_k x_j^T] = 0, \quad E[\omega_k \omega_j^T] = Q_k \delta_{k-j}, \quad \forall k, j \\ &E[v_k x_j^T] = 0, \quad E[v_k v_j^T] = R_k \delta_{k-j}, \quad \forall k, j \end{aligned} \tag{3.19}$$

where $E[\cdot]$ is the expectation operator, and δ_{k-j} is the Kronecker delta function, that is $\delta_{k-j} = 0$ if $k \neq j$, and $\delta_{k-j} = 1$ if $k = j$.

The KF estimates the state x_k based on the system dynamic model and the noisy measurements. The measurements sequence $Z_j = [z_1, z_2, z_3, ..., z_j]$, contains all of the measurements that up to and including j^{th} instant. The *a priori* $\hat{x}_k^-$ and *a posteriori* $\hat{x}_k$ estimates of system state at time instant k are defined based on the available measurements. If the available measurements are prior to time k, z_k is unavailable for state estimation at k^{th} instant, it is *a priori* estimate, we denote as $\hat{x}_k^-$. If the measurements z_k at time-step k are available for use in the estimation of state x_k, the estimate is *a posteriori*, which we denote as $\hat{x}_k$. $\hat{x}_k^-$ and $\hat{x}_k$ are results at the prediction (also called time update) stage and update (also called measurement update) stage, respectively.

The *a priori* state $\hat{x}_k^-$ can be predicted by taking the expectation of state x_k conditioned on all measurements before k^{th} instant. The predicted state is given by:

$$\begin{aligned} \hat{x}_k^- &= E[x_k | Z_{k-1}] \\ &= E[(Ax_{k-1} + Bu_{k-1} + \omega_{k-1}) | Z_{k-1}] \\ &= AE[x_{k-1} | Z_{k-1}] + Bu_{k-1} + E[\omega_{k-1} | Z_{k-1}] \\ &= A\hat{x}_{k-1} + Bu_{k-1} \end{aligned} \tag{3.20}$$

Similarly, the predicted measurement can be obtained as:

$$\begin{aligned}\hat{z}_k^- &= E[z_k|Z_{k-1}] \\ &= E[(Cx_k + v_k)|Z_{k-1}] \\ &= CE[x_k|Z_{k-1}] + E[v_k|Z_{k-1}] \\ &= C\hat{x}_k^-\end{aligned} \tag{3.21}$$

The *a priori* error and *a posteriori* error between the actual state and the estimated state are

$$e_k^- = x_k - \hat{x}_k^- \tag{3.22}$$

$$e_k = x_k - \hat{x}_k \tag{3.23}$$

Then the *a priori* covariance P_k^- of the state estimation error can be calculated as

$$P_k^- = E[e_k^- e_k^{-T}] = AP_{k-1}A^T + Q_{k-1} \tag{3.24}$$

After predicting the state and error covariance, the next step is to update the *a priori* state $\hat{x}_k^-$ and covariance P_k^-. The state vector is updated using the predicted state and the new measurement z_k, and the discrepancy between the actual measurement and the predicted measurement is weighted by Kalman Gain to minimize the *a posteriori* error covariance [155].

$$K_k = P_k^- C^T (CP_k^- C^T + R)^{-1} \tag{3.25}$$

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - C\hat{x}_k^-) \tag{3.26}$$

$$P_k = E[e_k e_k^T] = (I - K_k C)P_k^- \tag{3.27}$$

The Kalman gain K_k is calculated in equation (3.25) to evaluate the updated state and covariance. The factor $(z_k - C\hat{x}_k^-)$ in equation (3.26) is called the

residual, which denotes the discrepancy between measurements and predicted measurements.

The simplified diagram of Kalman filter is illustrated in Figure 3.2. The estimation process of KF consists of two steps, i.e. the prediction (time update) step and the update (measurement update) step. The prediction step in Kalman filter obtains the priori estimates of the current states and error covariance based on the past states of the system. The update step correct the *priori* estimates of the states and obtain the updated *posteriori* estimates by using the new measurement value, and then provides a feedback to the prediction step to implement iterative estimation.

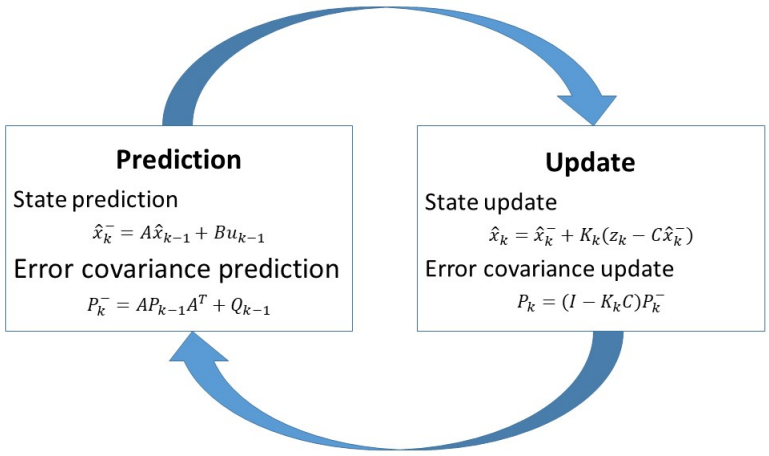


Figure 3.2: Simplified diagram of the KF

3.2.2 Extended Kalman Filter

The KF is an optimal linear filter, but in reality, most real-world systems are nonlinear. Applying KF in nonlinear systems will introduce large errors and sometimes may cause filter divergence. Considering the significance of nonlinear filtering for the diverse fields in science and engineering, researchers started

working on nonlinear filtering problems in the early 1960's [156], and a number of methods have been developed.

The state of a nonlinear dynamic system is governed by a nonlinear stochastic difference equation which is denoted by a differentiable function $f_k(.)$, and the measurements are nonlinear related to the state vector by a nonlinear function $h_k(.)$.

$$x_{k+1} = f_k(x_k, u_k, w_k) \tag{3.28}$$

$$z_k = h_k(x_k, v_k) \tag{3.29}$$

where k is the time index, x_k refers to the state, u_k is the input, w_k is the process noise, z_k represents the measurement, and v_k is the measurement noise. The functions $f_k(.)$ and $h_k(.)$ are time-varying nonlinear process and measurement equations.

In KF and its variants, vector w_k and v_k in equations (3.28) and (3.29) are assumed to be uncorrelated zero-mean white Gaussian noise with covariance matrixes Q_k and R_k respectively.

The EKF was developed to extend the application of the KF to nonlinear systems by replacing the nonlinear function with its linear approximation. It linearizes the estimation by calculating the Jacobian matrices of the nonlinear functions $f_k(.)$ and $h_k(.)$. The EKF is also called the First-Order filter, because it expands nonlinear functions using the Taylor Series and only uses the first order term. The linearized transformation of the nonlinear functions in the EKF is the approximation of the actual functions, therefore, the estimation results calculated by EKF is not guaranteed to be optimal estimates of the actual states of a nonlinear system.

The function $f(.)$ and $h(.)$ are linearized using Taylor-series expansion at each time step, supposing that $f(.)$ and $h(.)$ are differentiable at all time, the elements

of the state transition matrix are defined as:

$$\mathbf{A}_k = \frac{\partial f(x_k)}{\partial x_k}|_{x_k=\hat{x}_k^-} \tag{3.30}$$

$$\mathbf{C}_k = \frac{\partial h(x_k)}{\partial x_k}|_{x_k=\hat{x}_k^-} \tag{3.31}$$

Similar to the Kalman filter, the EKF algorithm also has two parts: prediction and measurement update stages. As shown in equation (3.32) and (3.33), at time instant k, the predicted state $\hat{\mathbf{x}}_k^-$ and error covariance $\mathbf{P}_k^-$ are first calculated without the measurement y_k. Then based on the predicted $\mathbf{P}_k^-$ and the measurement y_k, the state $\hat{\mathbf{x}}_k$ and error covariance $\mathbf{P}_k$ are updated using the Kalman gain $\mathbf{K}_k$ computed by (3.34).

$$\hat{\mathbf{x}}_k^- = f(\hat{\mathbf{x}}_{k-1}, u_{k-1}) \tag{3.32}$$

$$\mathbf{P}_k^- = \mathbf{A}_k\mathbf{P}_{k-1}\mathbf{A}_k^T + \mathbf{Q_{k-1}} \tag{3.33}$$

$$\mathbf{K}_k = \mathbf{P}_k^-\mathbf{C}_k^T\left(\mathbf{C}_k\mathbf{P}_k^-\mathbf{C}_k^T + \mathbf{R_k}\right)^{-1} \tag{3.34}$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k\left(\mathbf{y}_k - h(\hat{\mathbf{x}}_k^-)\right) \tag{3.35}$$

$$\mathbf{P}_k = \left(\mathbf{I} - \mathbf{K}_k\mathbf{C}_k\right)\mathbf{P}_k^- \tag{3.36}$$

3.2.3 Unscented Kalman Filter

The Unscented Kalman Filter (UKF) was introduced by Julier and Uhlmann [157; 158], which is a recursive Minimum Mean Square Error (MMSE) estimator that addresses the limitations of the EKF caused by linear approximation and provides more accurate, robust and efficient prediction performance and easier implementation, while the computational cost of the algorithm is the same order of magnitude as the EKF. The UKF is also called sigma-point Kalman filter (SPKF) due to the use of sigma-point approach, which is a deterministic sampling framework [159].

The UKF is still restricted to Gaussian distributions. The core and foundation of the UKF algorithm is unscented transformation (UT), a method for calculating the statistics of a random variable which undergoes a nonlinear transformation and builds on the principle that it is easier to approximate a probability distribution than an arbitrary nonlinear function or transformation[160]. The schematic representation of the unscented transformation is shown in Figure 3.3. Firstly, a set of sigma points $\{\chi_i\}_{i=0}^{2n}$ are deterministically sampled from the the mean $\bar{\mathbf{x}}$ and the covariance $\mathbf{P}_x$ of the state variables at the previous moment, n is the number of state variables, and the weights for the mean and covariance are $w_x^{[i]}$ and $w_p^{[i]}$, respectively. Then each sigma point is propagated through the nonlinear function $y = g(\chi_i)$. Finally, the approximated mean $\bar{y}$ and covariance P_y are computed to capture the prior distribution of the state variables at current moment.

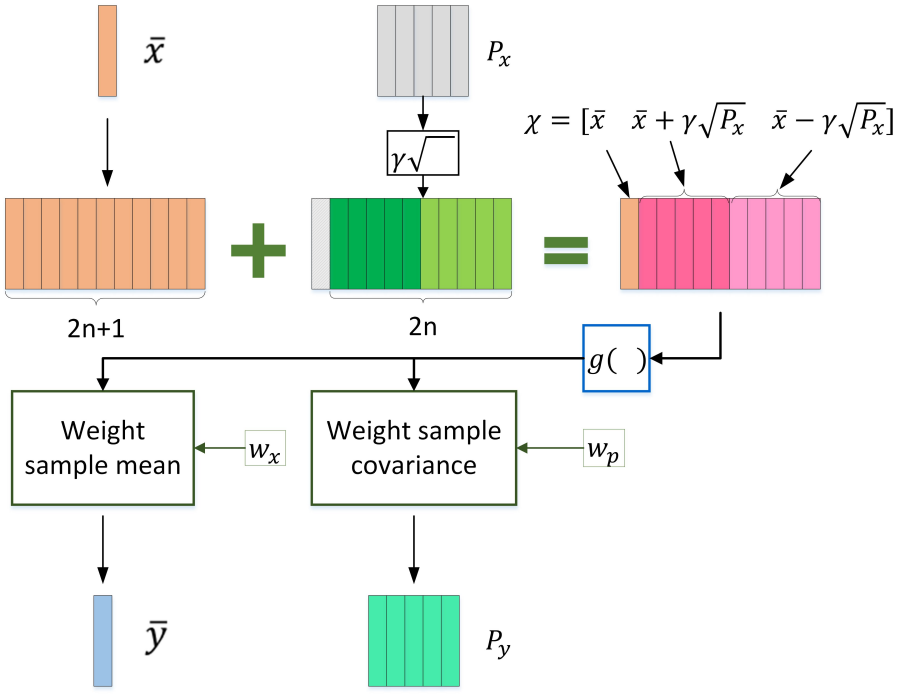


Figure 3.3: Schematic diagram of the UT

The UKF is a straightforward application of the UT in the recursive Kalman filter framework [159]. Through unscented transformation, the mean and covariance of

$\mathbf{y}$, who has a nonlinear relationship with $\mathbf{x}$ given as $\mathbf{y} = g(\mathbf{x})$, can be obtained. The UT algorithm is summarized as follows:

- Choosing the sigma points

$$\begin{aligned} \chi^{[0]} &= \bar{\mathbf{x}} \\ \chi^{[i]} &= \bar{\mathbf{x}} + \left[\gamma\sqrt{\mathbf{P}_x}\right]_i, \quad i = 1, ..., n \\ \chi^{[i]} &= \bar{\mathbf{x}} - \left[\gamma\sqrt{\mathbf{P}_x}\right]_i, \quad i = n+1, ..., 2n \end{aligned} \tag{3.37}$$

  where $\left[\gamma\sqrt{\mathbf{P}_x}\right]_i$ denotes the i^{th} column of $\left[\gamma\sqrt{\mathbf{P}_x}\right]$, $\gamma = \sqrt{n+\lambda}$ is a composite scaling parameter calculated by the dimension n of the state and the scaling parameter λ.

$$\lambda = \alpha^2(n + \varepsilon) - n \tag{3.38}$$

  where α $(0 \leq \alpha \leq 1)$ represents the scaling parameter that determines the distribution of the sigma points, in general, α should be a small number. ε is the other parameter that can adjust the spread of the points. The bigger the α and ε are, the further the sigma points are away from the mean.

- Set the corresponding sigma point weights

$$\begin{aligned} w_x^{[0]} &= \frac{\lambda}{n+\lambda} \\ w_p^{[0]} &= w_x^{[0]} + (1 - \alpha^2 + \beta) \\ w_p^{[i]} &= w_x^{[i]} = \frac{1}{2(n+\lambda)}, \quad i = 1, ..., 2n \end{aligned} \tag{3.39}$$

  The suggested values for α, β and ε are 0.01 or 0.001, 2 and $3 - n$, respectively [159].

- Calculate the mean an covariance of $\mathbf{y}$ from the weighted and transformed

points

$$\bar{\mathbf{y}} = \sum_{i=0}^{2L} w_x^{[i]} g(\chi^{[i]}) \tag{3.40}$$

$$P_y^- = \sum_{i=0}^{2L} w_p^{[i]} \left(g(\chi^{i]}) - \bar{\mathbf{y}}\right) \left(g(\chi^{[i]}) - \bar{\mathbf{y}}\right)^T \tag{3.41}$$

Considering the nonlinear system model given in equation (3.28) and (3.29), the UKF algorithm is briefed as:

- Initialization

$$\hat{x}_0 = E[x_0], \quad P_{x_0} = E[(x_0 - \hat{x}_0)(x_0 - \hat{x}_0)^T] \tag{3.42}$$

- Prediction

  Calculate sigma-points at time step $k-1$ ($k = 1, ..., \infty$)

$$\chi_{k-1} = \left[\hat{x}_{k-1} \quad \hat{x}_{k-1} + \gamma\sqrt{P_{k-1}} \quad \hat{x}_{k-1} - \gamma\sqrt{P_{k-1}}\right] \tag{3.43}$$

  Propagate the sigma points through the process model from time $k-1$ to k, and predict the state and error covariance using these weight points

$$\chi_{k|k-1}^{[i]} = f(\chi_{k-1}^{[i]}, u_{k-1}) \tag{3.44}$$

$$\hat{x}_k^- = \sum_{i=0}^{2n} w_x^{[i]} \chi_{k|k-1}^{[i]} \tag{3.45}$$

$$P_{x_k}^- = \sum_{i=0}^{2n} w_p^{[i]} \left(\chi_{k|k-1}^{[i]} - \hat{x}_k^-\right) \left(\chi_{k|k-1}^{[i]} - \hat{x}_k^-\right)^T \tag{3.46}$$

- Measurement-update

  Propagate the sigma points through the measurement model and predict the measurement, measurement error covariance and cross-covariance at

time instant k

$$Z_{k|k-1}^{[i]} = h(\chi_{k|k-1}^{[i]}, u_k) \tag{3.47}$$

$$\hat{z}_k^- = \sum_{i=0}^{2n} w_x^{[i]} Z_{k|k-1}^{[i]} \tag{3.48}$$

$$P_{z_k} = \sum_{i=0}^{2n} w_p^{[i]} \left(Z_{k|k-1}^{[i]} - \hat{z}_k^- \right) \left(Z_{k|k-1}^{[i]} - \hat{z}_k^- \right)^T \tag{3.49}$$

$$P_{x_k z_k} = \sum_{i=0}^{2n} w_p^{[i]} \left(\chi_{k|k-1}^{[i]} - \hat{x}_k^- \right) \left(Z_{k|k-1}^{[i]} - \hat{z}_k^- \right)^T \tag{3.50}$$

Calculate the Kalman gain and update the mean and covariance of the state

$$K_k = P_{x_k z_k} P_{z_k}^{-1} \tag{3.51}$$

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - \hat{z}_k^-) \tag{3.52}$$

$$P_{x_k} = P_{x_k}^- - K_k P_{z_k} K_k^T \tag{3.53}$$

3.2.4 Particle Filter

The above-mentioned Kalman filter and its variants have a fundamental limitation that the noise in the process and measurements should be Gaussian, and the processes with non-Gaussian noise cannot be estimated using the variants of Kalman filters. Whereas the Particle filter(PF) does not require the Gaussian hypotheses [161]. The particle filter, also including variants known as the bootstrap filter, Sequential Importance Sampling (SIS), condensation algorithm, interacting particle approximations, Monte Carlo filter, Sequential Monte Carlo (SMC) filter [41; 162], has become a very popular class of numerical methods for the optimal estimation problems in non-linear non-Gaussian scenarios since their introduction in 1993 [163]. In comparison with standard approximation methods, such as the popular EKF, the main advantage of PFs is that they do not rely on

any local linearisation technique or any crude functional approximation.
Particle filtering is based on Monte Carlo simulation with sequential importance sampling (SIS), it approximates the complete posterior state density recursively [162; 164]. The key idea of PF is to represent the required posterior density function by a set of independent random samples (particles) with associated weights, and to update the posterior by involving the new measurements.
The particle set is recursively located, weighted and propagated according to the Bayesian theory, the PF is a probability-based estimator that is derived from the Bayesian approach. For the record, the particle filter needs a Bayesian formulation of the problem.

$$x_k = f(x_{k-1}, w_k) \leftrightarrow p(x_k|x_{k-1}) \tag{3.54}$$

$$y_k = h(x_k, v_k) \leftrightarrow p(y_k|x_k) \tag{3.55}$$

where k is the time index, w_k and v_k are white noises, and are not necessarily Gaussian.
The SIS algorithm is the basis for all particle filter algorithms. The importance function $q(x_{0:k}|y_{0:k})$ and importance weights of SIS algorithm can be evaluated recursively:

$$q(x_{0:k}|y_{0:k}) = q(x_0)\prod_{i=1}^{k} q(x_i|x_{0:i-1}, y_{0:i}) \tag{3.56}$$

$$\tilde{w}_k^{(i)} \propto \tilde{w}_{k-1}^{(i)} \frac{p(y_k|x_k^{(i)})p(x_k^{(i)}|x_{k-1}^{(i)})}{q(x_k^{(i)}|x_{0:k-1}^{(i)}, y_{1:k})} \tag{3.57}$$

Also the weights should be normalized

$$\tilde{w}_k^{(i)} = \frac{\tilde{w}_k^{(i)}}{\sum_{i=1}^{N} \tilde{w}_k^{(i)}} \tag{3.58}$$

But the SIS method suffers from a serious drawback: the weight will tilt after several updating iterations which causes substantial computational efforts being wasted on the particles which have scarcely any contribution to the final result. Resampling method is proposed to remove the particle that has low weight and retains the rest. Currently, wide attentions are paid to multinomial resampling, stratified resampling, systematic resampling and residual resampling [46].

Sampling Importance Resampling (SIR) Filter is the most classical form of particle filter and is easy to implement [165]. It can be easily derived from the SIS algorithm by choosing appropriate importance density and the resampling step. The advantage of SIR method is that it is easy to evaluate the importance weights and sample the importance density.

The choice on importance function determines the performance of the method. A simple, though not optimal, choice for the importance function is to use the prior distribution, that is

$$q(x_k|x_{k-1}, y_k) = p(x_k|x_{k-1}) \tag{3.59}$$

The weight update equation can be simplified

$$w_k^{(i)} = \frac{p(y_k|x_k^{(i)})p(x_k^{(i)}|x_{k-1}^{(i)})}{q(x_k^{(i)}|x_{k-1}^{(i)}, y_{0:k})} w_{k-1}^{(i)} = p(y_k|x_k^{(i)})w_{k-1}^{(i)} \tag{3.60}$$

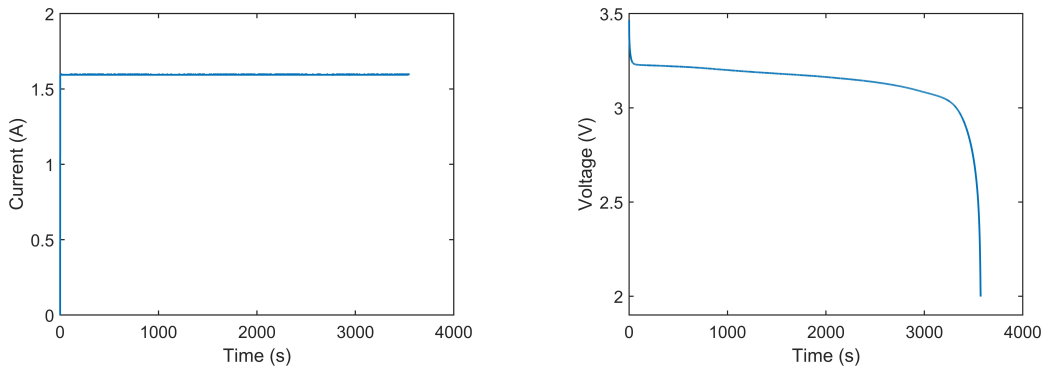
3.3 Experimental Data

A cylindrical lithium iron phosphate battery cell was tested under the room temperature (25°C) using a BTS 4000 battery test system made by NEWARE, and its parameters are summarized in Table.3.1. Thermocouples with measurement error less than 0.1°C are attached to measure the cell surface temperature. The sampling frequency for all the equipment used in the test was set as 1 Hz.

Table 3.1: Specifications of the battery.

Battery type	$LiFePO_4$
Nominal capacity	1.6 Ah
Nominal voltage	3.2 V
Min /Max Voltage	2.0V/3.6V
Initial State-of-Charge	100%

The battery was charged with a constant current (CC) of 1.6 A (1C) until the voltage reaches the upper cutoff voltage 3.6 V, and then followed by a constant voltage (CV) charge at 3.6 V which was terminated when current fell below the cutoff value of 75 mA. The cell was then discharged with a 1.6 A constant current to the lower cutoff voltage of 2.0 V. The current and voltage of this CC discharge process measured at an ambient temperature of 25°C are shown in Figure 3.4.



(a) (b)

Figure 3.4: The current and voltage measured during the CC discharge cycle.

3.4 Results and Discussion

To produce an accurate estimation of the SOC based on the real-time measurements of terminal voltages and currents, a battery model in the form of state space should be established first. The overall process of the proposed model-based SOC estimation is illustrated in Figure 3.5, which can be divided into two steps: battery modelling and SOC estimation. The modelling step consists of testing data collection, candidate model terms generation, optimal model terms selection and model performance validation, which has been given in detail in Section. 3.1.2. Then based on the built model, the SOC is estimated using suitable algorithm. The algorithm, such as Kalman filter family and sliding mode observer, is selected according to the application and required accuracy. With the battery testing experiment introduced in Section. 3.3, the data collected from the first 5 discharging cycles are used to build the measurement function to correlate the terminal voltage with SOC and current, and the data of the 7-th discharging cycle are used to validate the performance of the model as well as to compare the performance of different filtering algorithms for SOC estimation.

3.4.1 Evaluation of the model

As described in Section.3.1.2, the model terms include current, SOC, and their nonlinear forms, which are selected by fast recursive algorithm from the candidate model terms. The statistical validation results of the trained model are shown in Figure 3.6, as can be observed, the RMSE of the model decreases as the number of selected model terms increases. When 9 model terms are selected (red cycle in Figure 3.6), the RMSE is 0.0055 V, after which the RMSE does not decrease significantly even if more terms are selected. Considering the balance between accuracy and model complexity, the number of selected model terms are determined to be 9.

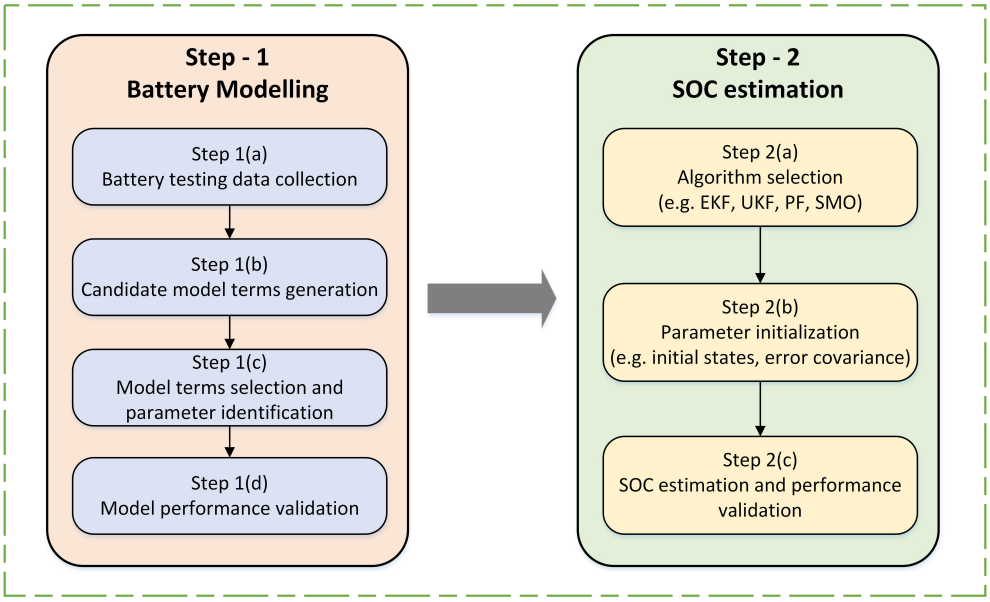


Figure 3.5: Procedures of the proposed model-based SOC estimation

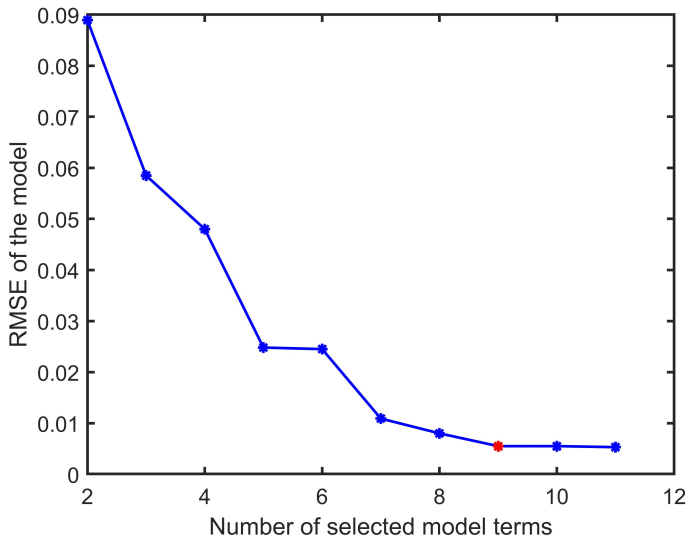


Figure 3.6: The RMSE of model with different number of model terms

The modelling results on the validation data are shown in Figure 3.7. As can be observed, the model outputs are close to the measured data, that is, the model built by FRA can well describe the nonlinear relationship between the SOC and voltage. The RMSE is 5.9 mV, which is 0.18% of the battery nominal voltage.

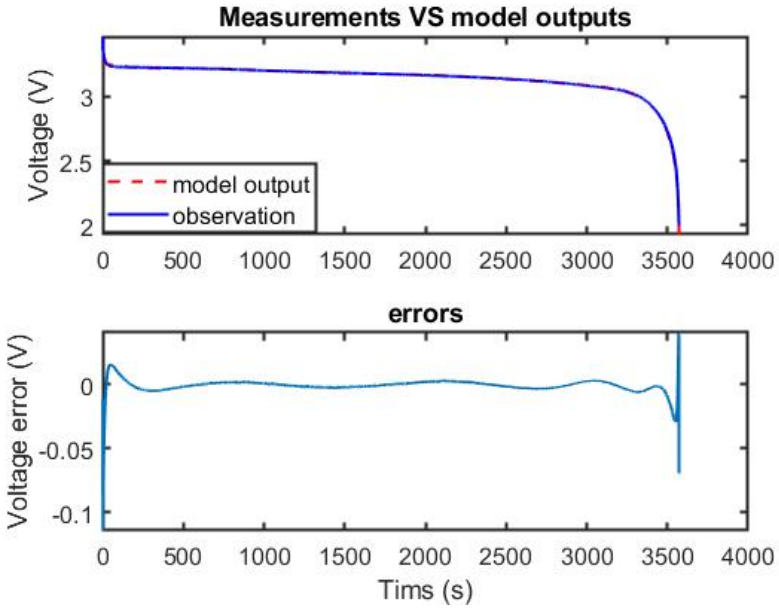


Figure 3.7: Modelling results

Table 3.2 summarizes the RMSE of the model on subsequent cycles, which increases as the cycle increases. It can be observed that the RMSE is larger than 10 mV after 10 cycles, and the model performance further degrades after 80 cycles. Therefore, to obtain acceptable SOC estimation results, the model parameters should be updated regularly using newly collected cycling data. The update frequency depends on the requirements for SOC estimation accuracy, for the best results, the model should be updated every 10 cycles.

Table 3.2: Modelling results on validation cycles

cycle number	8	9	10	11	20	40	60	80	100
RMSE (mV)	7.1	7.3	7.4	10.3	11.8	12.2	12.1	15.7	16.5

3.4.2 SOC Estimation based on the proposed model

The battery state space model is expressed as follows:

$$\begin{cases} SOC_k = SOC_{k-1} + \frac{\Delta t}{Q} I_{k-1} \\ \\ V_k = h(I_k, SOC_k) = \mathbf{\Psi_9}(I_k, SOC_k)\mathbf{\Theta_9} \end{cases} \tag{3.61}$$

The selected models terms $\mathbf{\Psi_9} = [\varphi_1, \varphi_2, ..., \varphi_9]$ and the corresponding parameters $\mathbf{\Theta_9} = [\theta_1, \theta_2, ..., \theta_9]^T$ of the measurement function are listed in Table. 3.3. Then the SOC is estimated based on this model using different filtering algorithms. In this section, EKF, UKF and PF are used to estimate the SOC, and their results are compared. The initial value of SOC is set to 0.8 in all tests, while the correct initial value is 1, thus 20% initial error is introduced to validate the performance of the filtering algorithms with erroneous initial state value.

Table 3.3: Model terms and parameters of the measurement function.

	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8	i=9
θ_i	0.1176	7.7379	273.9758	4.91722	21.3713	10.3254	-294.8140	-41.5292	17.3574
φ_i	I	$\sqrt{SOC}$	$sic(SOC)$	e^{SOC}	$cos(SOC^3)$	SOC^2	SOC	$cos(SOC^2)$	e^{SOC^2}

According to the detailed algorithm described in Section. 3.2.2, the flowchart of the EKF-based battery SOC estimation is illustrated in Figure 3.8.

Based on Equation.(3.61), the matrix A_k and C_k in Equation. (3.30) and (3.31)

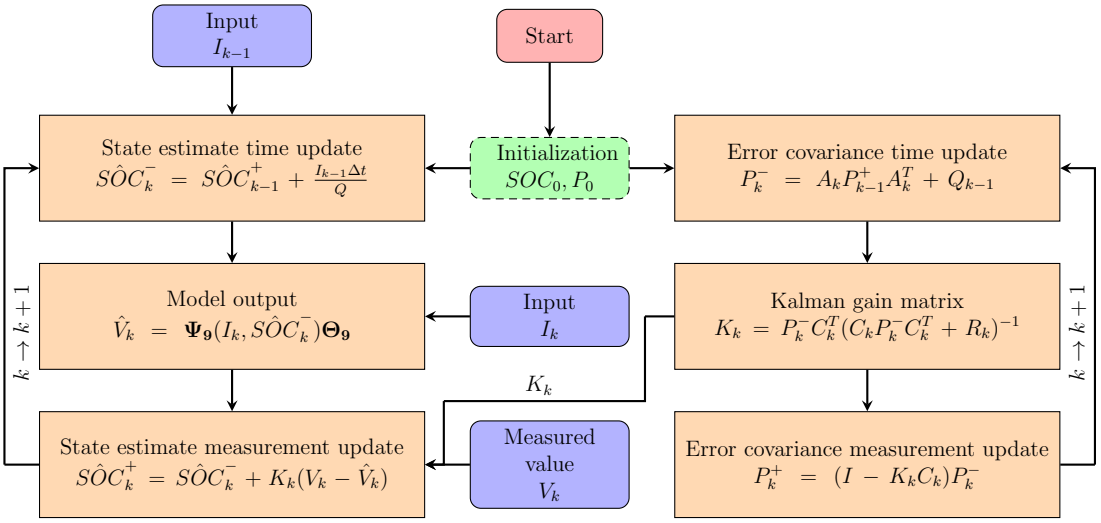


Figure 3.8: Flowchart of EKF-based SOC estimation

can be deduced as:

$$\begin{cases} A_k = 1 \\ C_k = \frac{\partial h(I_k, SOC_k)}{\partial SOC_k} = \frac{\partial \mathbf{\Psi_9}(I_k, SOC_k)}{\partial SOC_k}\mathbf{\Theta_9} \end{cases} \tag{3.62}$$

The EKF-based battery SOC estimation results are shown in Figure 3.9, where the blue line represent the reference SOC and the red line is the SOC estimation results using EKF. As shown in this figure, the SOC estimation results can quickly converge to the reference value and the estimation error is close to zero after convergence.

The flowchart of UKF-based battery SOC estimation is illustrated in Figure 3.10. As the number of state is 1 in this test, the sigma point χ_{k-1} generated at $(k-1)$-th time step is a 1×5 vector. Figure 3.11 shows the SOC estimation results using UKF, similar to that of the EKF-based method, a faster convergence speed can be observed and the estimates can track the reference SOC value well.

Figure 3.12 shows the flowchart of PF-based SOC estimation. In the PF-

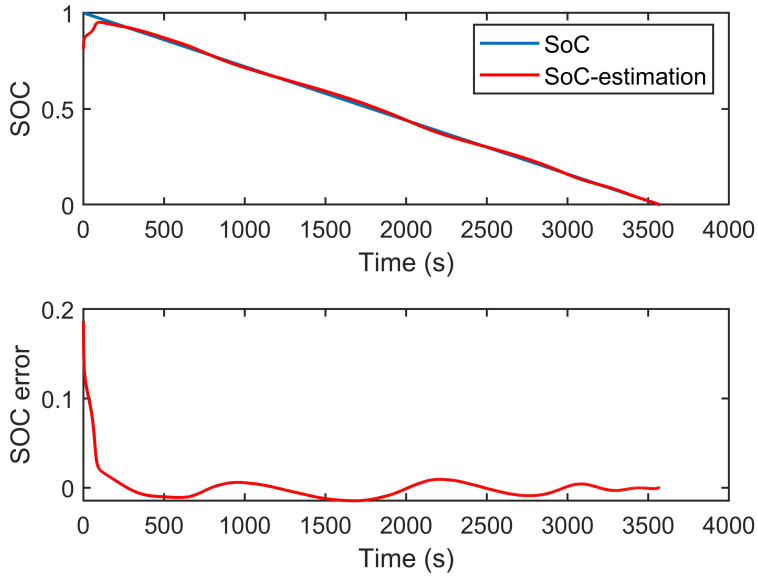


Figure 3.9: EKF-based battery SOC estimation results

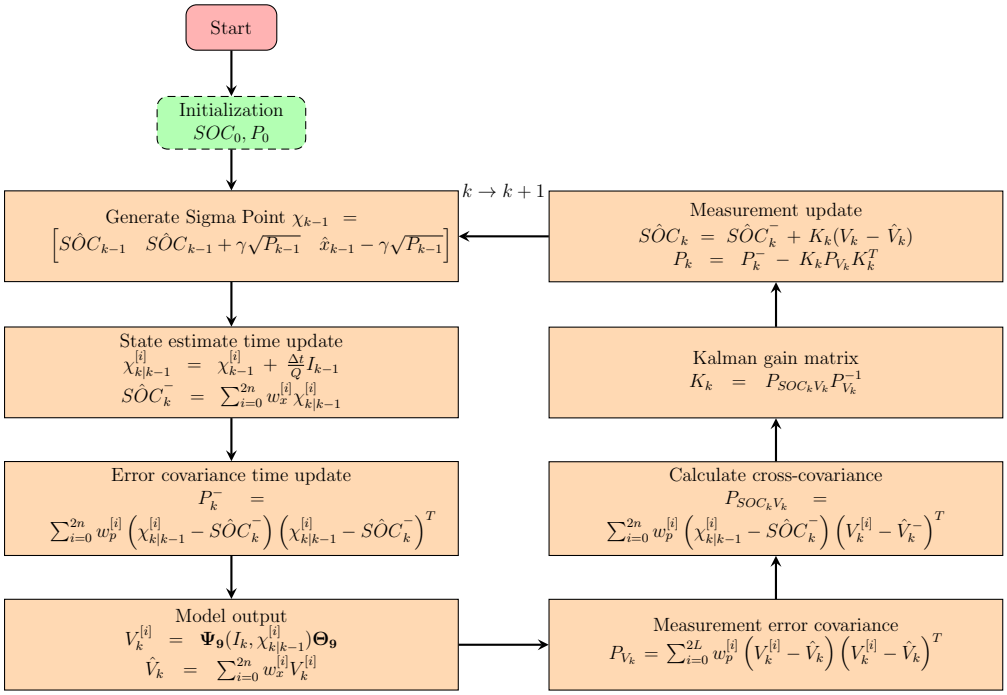


Figure 3.10: Flowchart of UKF-based SOC estimation

based SOC estimation, the number of particles is set to 140 and the systematic

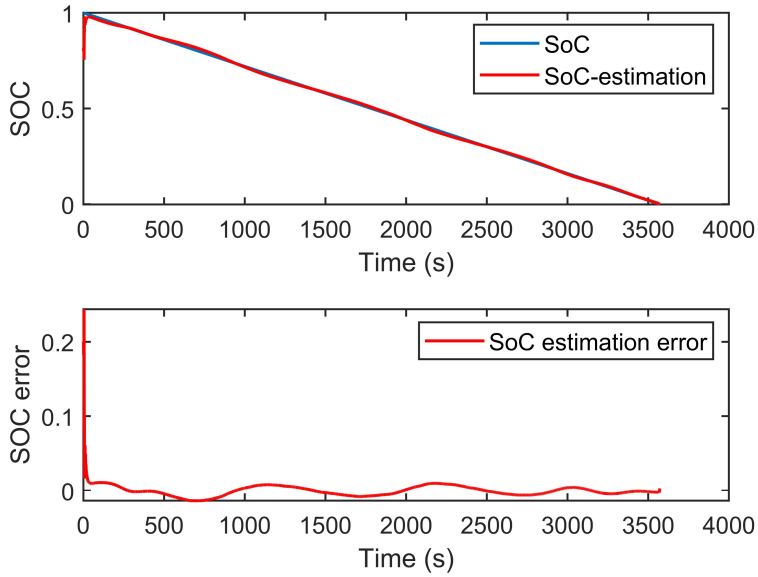


Figure 3.11: UKF-based battery SOC estimation results

resampling strategy is applied. The estimation results are shown is Figure 3.13. The estimated SOC values are quite close to the reference values, also a quick convergence speed can be observed.

3.4.3 Summary of the results

Based on the model identified using FRA, three filtering algorithms are applied to estimate the battery SOC, namely EKF, UKF and PF. Figure 3.9, 3.11, and 3.13 show the estimation results of these algorithms, and all of these algorithms can generate accurate estimation results when the initial state value is inaccurate. To clearly compare the performance of these algorithms, the estimation results in terms of RMSE and computation time of each algorithm are summarized in Table.3.4. The computation time is recorded by the 'tictoc' function in matlab. It is clear that among these three algorithms, the PF provide the most accurate estimation results, however, its computation time is more than 100 times that of UKF. The computation time of EKF is even longer than that of PF due to

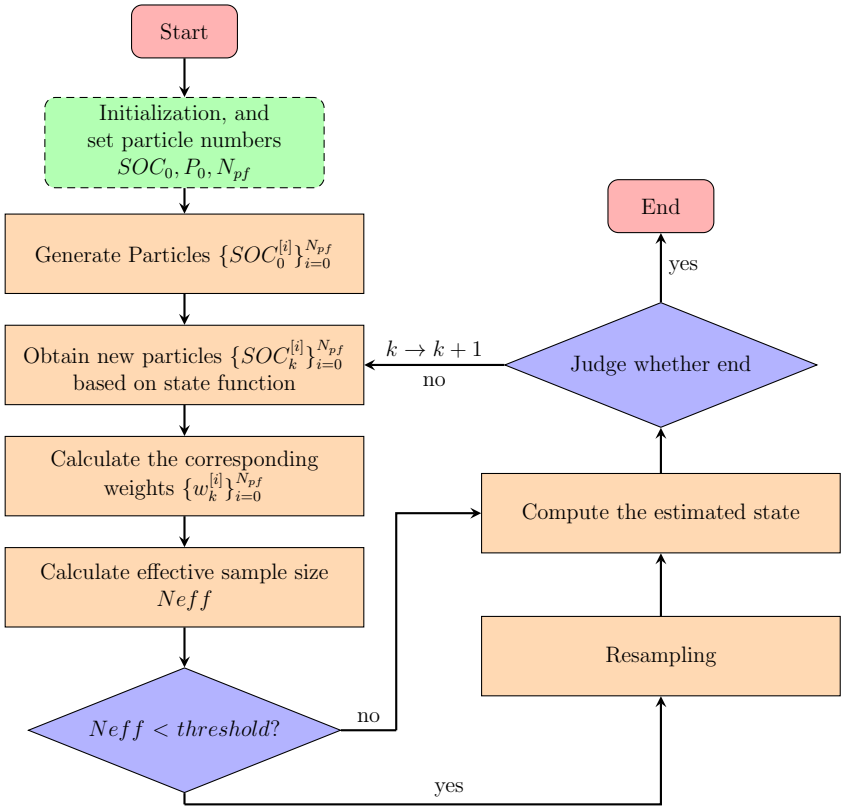


Figure 3.12: Flowchart of PF-based SOC estimation

the calculation of Jacobian matrix increases the computation complexity. While the UKF neither require the linearization of the nonlinear system nor require hundreds or thousands of particles to represent the states, thus the computation time of SOC estimation using UKF is the shortest.

Table 3.4: Comparison of filters' performance

	RMSE (%)	Computation time (s)
EKF	1.64	25.7319
UKF	1.05	0.1699
PF	0.75	21.6863

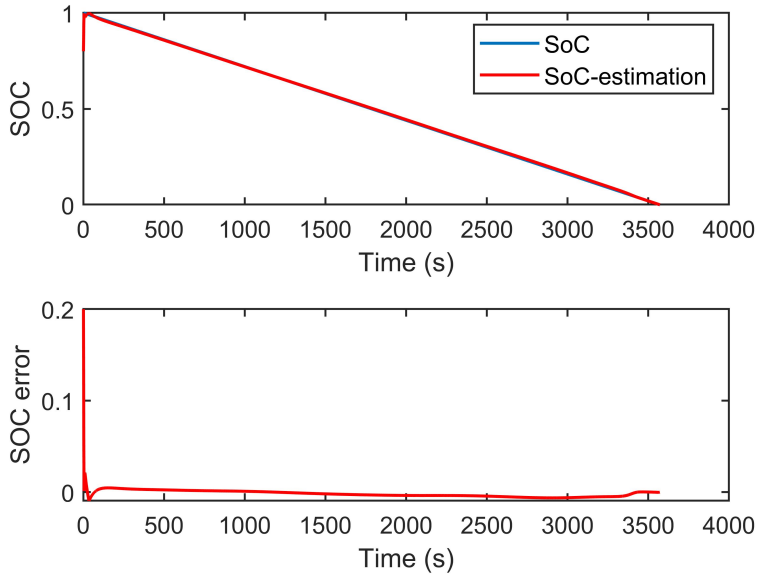


Figure 3.13: PF-based battery SOC estimation results

Further, the comparison of their estimation errors is shown in Figure 3.14, from which the convergence speed of PF is proven to be the fastest among the applied filters, and the UKF converges faster than EKF. Besides, the PF-based estimation results seem to be the most stable and closest to zero. In summary, at the cost of computational complexity, the PF-based SOC estimation method provides the highest estimation accuracy. And the performance of UKF-based SOC estimation method is better than EKF-based method in terms of accuracy and computational efficiency.

The proposed method is also validated on the charging cycles. The resultant model has 9 model terms and the RMSE of the model is 9.5mV. As shown in Figure 3.15, the SOC estimation results of the charging process produced by different filtering algorithms are similar to those of the discharging process. Among the EKF, UKF and PF, the PF produces the most accurate SOC estimation results (RMSE is 0.69%) and its calculation time (17.3639 s) is shorter than that of the EKF-based estimation method (30.8647 s). The UKF is the fastest that

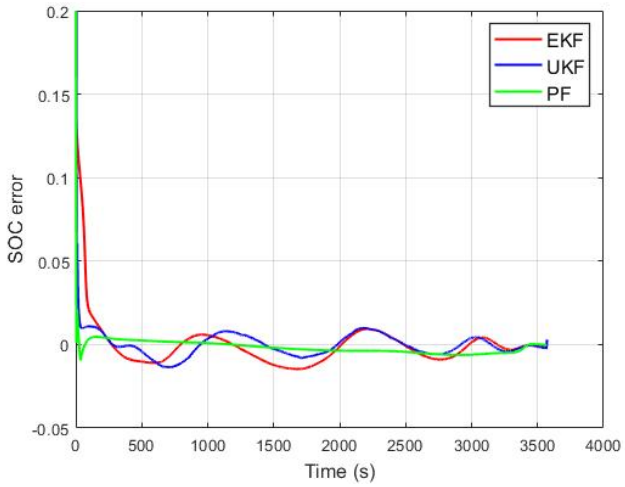


Figure 3.14: Error of SOC estimation under different filters

only takes 0.3466 s and the RMSE is 1.04%, which is less than that of the EKF (1.58%).

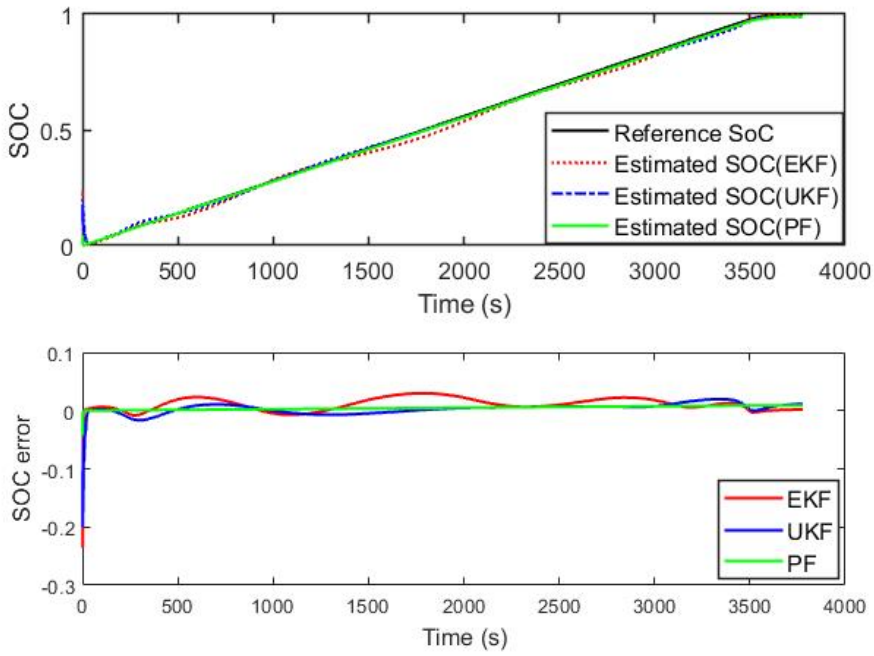


Figure 3.15: SOC estimation results on charging data using different filtering algorithms

3.5 Chapter Summary

Battery SOC is an important characteristic parameter for BMS, and it should be monitored in real-time to ensure the battery operation safety. Since SOC cannot be measured directly, many state estimation methods have been proposed for SOC estimation based on the direct measurable variables. In this chapter, a model-based SOC estimation method has been proposed. Due to the estimation performance of the model-based methods highly depend on the model accuracy, the first aim of this chapter is to establish a measurement function with satisfactory accuracy. Therefore, a computationally efficient and stable model identification method, i.e. fast recursive algorithm, was presented first and was then applied to capture the nonlinear relationship between the terminal voltage and SOC, and its performance was verified on test data. Based on the well established battery model, three filtering algorithms, namely EKF, UKF and PF, were successively applied to estimate the SOC. The estimation results of these algorithms were summarized and analyzed, and the results indicate that all of these three algorithms are capable of correcting initial SOC error and providing accurate estimation results. Further, the comparison results suggest that the PF-based method is the best choice among them when the estimation accuracy is of most concern, while the UKF-based method is the best choice if the computation time is of most concern.

Chapter 4

Battery Capacity Estimation Using Convolutional Neural Networks

Accurate estimation of the battery capacity, which is one of the most popular indicators for assessing the battery aging, is crucial to maintain the battery performance and guarantee the safe and reliable operation of the battery. It is of great importance to trace the capacity in real-time, as the capacity information is one of the key factors affecting SOC estimation results [166] and enables detection of premature faults [167]. Due to a range of time-varying situation-dependent internal and external factors, it is still a challenging task to estimate capacity accurately. As a class of deep neural networks, the Convolutional Neural Network (CNN) is powerful to capture hidden information from a huge amount of input data, making it an ideal tool for battery capacity estimation. Therefore, in this chapter a CNN based battery capacity estimation method is proposed, which can accurately estimate the battery capacity using limited available measurements, without resorting to other online information. And the proposed method only requires partial charging segment of voltage, current and charge capacity curves,

making it possible to achieve fast online health monitoring. To employ CNN for capacity estimation using partial charging curves is however not trivial, a comprehensive approach covering time series-to-image transformation, data segmentation, and CNN configuration is presented at the beginning of this chapter. However, the CNN proposed for capacity estimation has thousands of parameters to be optimized and demand a substantial number of battery aging data for training. These parameters require massive memory storage while collecting a large volume of aging data is time-consuming and costly in real-world applications. To tackle these challenges, a novel framework incorporating the concepts of transfer learning and network pruning is then proposed in this chapter to build compact Convolutional Neural Network models on a relatively small dataset with improved estimation performance. The resultant CNN model is capable of achieving fast online capacity estimation at any time, and its effectiveness is verified on a target dataset collected from four Lithium iron phosphate battery cells, and the performance is compared with other Convolutional Neural Network models.

4.1 Deep Learning Algorithms

4.1.1 Deep Neural Networks

The recent surge of interest in deep learning algorithms is primarily due to the fact that they enable researchers to make significant improvements in many applications, as well as the advent of faster computational power and the abundance of available training data. In this section, the Deep Neural Networks (DNNs) refers to the fully-connected neural networks that employ deep architectures. There are more than one hidden layers between the input and the output layer, and each neuron in one layer is connected to all neurons in the adjacent layer, which makes the network fully connected. Each layer consists of a linear operation followed by

a point-wise nonlinearity (activation function) [168]. The architecture of DNN is shown in Figure 4.1. These fully-connected architectures do not require special assumptions about the input data, and are 'universal approximators' capable of representing any functions [169].

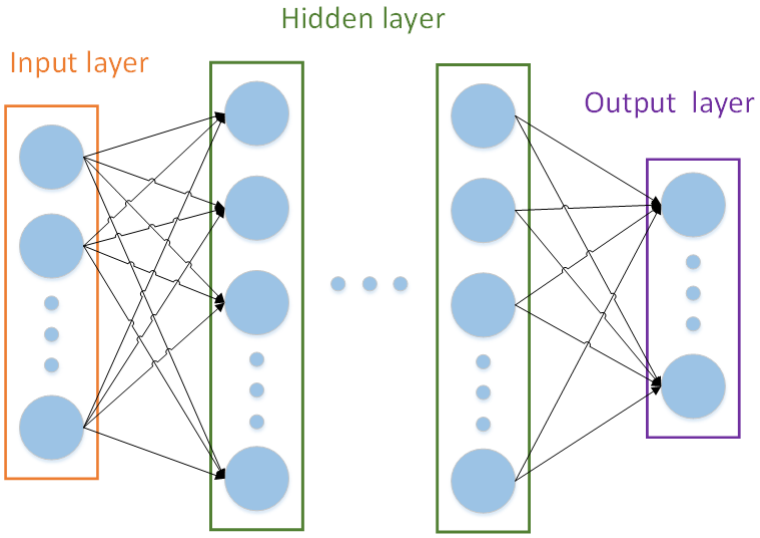


Figure 4.1: Architecture of DNN

Let $\mathbf{x} \in \mathbb{R}^{m\times 1}$ represent the input to a fully connected layer, $y_i \in \mathbb{R}$ represent the i^{th} output of this layer. Each output activation is composed of a weighted sum of all input activations. The y_i can be calculated as

$$y_i = f(w_{1i}x_1 + w_{2i}x_2 + ... + w_{mi}x_m) \tag{4.1}$$

where the $f(\cdot)$ is a nonlinear activation function, the weights w_{ji} are learnable parameters in the network that connect the j^{th} input neuron to the i^{th} output neuron. For a layer of n neurons, a n-by-m matrix is defined as $\mathbf{w} = [w_1, w_2, ..., w_m]$, where $w_j = [w_{j1}, w_{j2}, ..., w_{jn}]^T$, $\mathbf{y} \in \mathbb{R}^{n\times 1}$ can be conveniently computed as a matrix multiplication

$$\mathbf{y} = f(\mathbf{wx}) \tag{4.2}$$

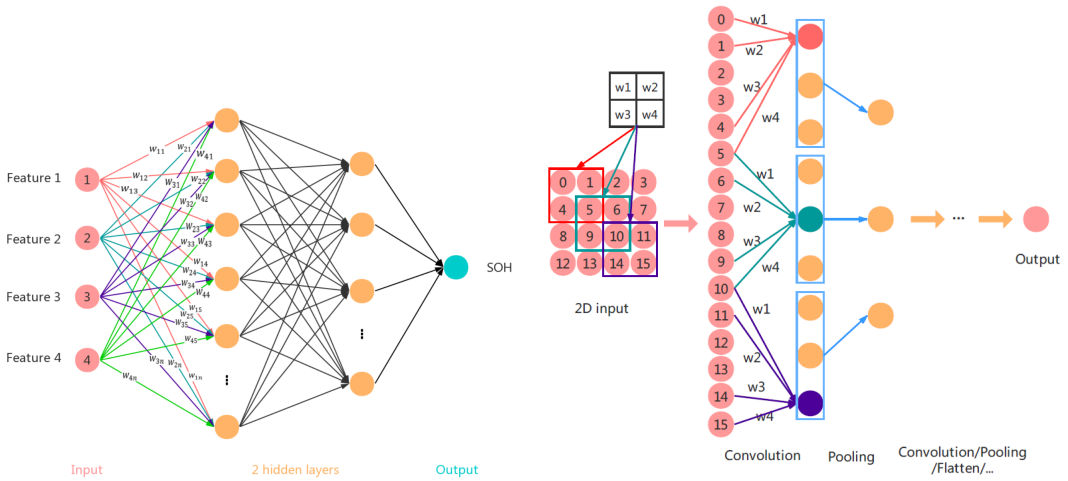
Due to the fully connected structure of a DNN, a large amount of parameters need to be determined, which requires a significant amount of storage and entails large computational complexity. Sparse connection and limited weights number between the input and output neurons, which can be realized if each output is a function of a fixed-size window of inputs, is used to make the computation more efficient. Even further efficiency can be achieved by repeated using the same weight values to calculate each output. Convolution neural network is then introduced as a good example.

4.1.2 Convolutional Neural Networks

The Convolutional Neural Network (CNN) is probably one of the most popular class of Artificial Neural Networks (ANNs) in recent years. Compared with traditional DNNs with the same number of layers, the number of parameters (weights) of a CNN that are required to maintain the accuracy is significantly reduced, due to the sparse connectivity, shared weights, and pooling architectures. As illustrated in Figure 4.2, these three architectures can be described as follows:

- Sparse Connectivity

  In traditional DNNs, a matrix multiplication is applied to describe the interaction among each input neuron and each output neuron. While in CNNs, the sparse connectivity architecture is applied, which is achieved by making the size of filter smaller than the input, and enforcing a local connectivity pattern among neurons of adjacent layers. In other words, the inputs of a neuron in layer l are from a subset of neurons in layer $l-1$. Figure 4.2(b) shows how the nodes numbered 0,1,4,5 surrounded by the red line are connected to the next node 0. It is obvious that the direct connections in a CNN are sparse. This architecture can describe the



(a) Simple DNN (b) Simple CNN

Figure 4.2: (a) A fully connected 3-layer feedforward neural network. (b) A convolutional neural network, with convolutional layer as the first layer and pooling layer as the second layer. Here, the filter size is 2× 2 with stride (1,1), and the pooling size is 1× 3 with stride (1,1).

complicated interactions among neurons more efficiently and also reduce the overfitting risk due to the number of parameters are dramatically reduced.

- Shared Weights

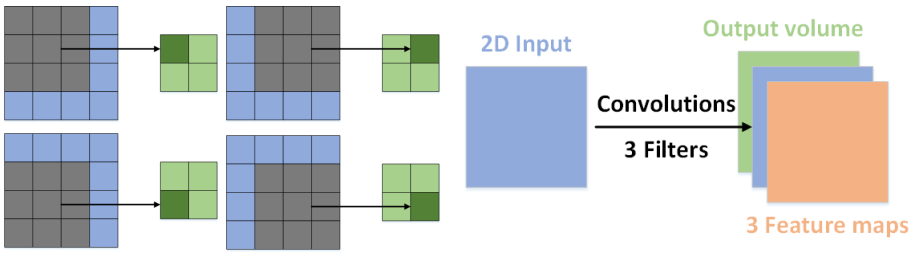
  In conventional DNNs, each weight matrix is multiplied by all the neurons of the input layer and used only once. While in CNNs, the same weights are used for more than one activation function, which is called shared weights. As shown in Figure 4.2(b), a 2×2 weight matrix with stride being set to $(1, 1)$ is used across the whole visual field, every four neurons of the input generate one output node. The shared weights architecture has endowed the CNN with a property called equivariance, meaning that the output will change in the same way as the input changes [170].

- Pooling

  Then, the use of pooling architecture replaces the outputs of the convolu-

tional layer with summary statistic, and this subsampling operation makes the output insensitive to small translation of the input. In Figure 4.2(b), the pooling size is 1×3 with stride being set to $(1, 1)$, thus a 3-by-3 matrix becomes a 3-by-1 matrix after the pooling stage.

A CNN is composed of multiple layers, such as convolution layers, pooling layers, and fully connected layers, and a series of convolution, pooling and matrix multiplication operations are performed during the calculation process. In the convolution operation, the filter (also called weight matrix) slides over the input space and generates a set of output nodes, and each output node is calculated by convolving the input with the filter. The number of involved inputs for one output node (or called the size of the receptive field) is decided by the filter size and how far away the receptive fields are for neighboring output neurons are decided by the filter stride. Figure 4.3(a) is an example of a 3×3 filter (gray area) with stride 1 acting on a 4×4 input, a 2×2 feature map (green square) is produced, and each cell represents a neuron. Note that in this example, we do not consider the depth of the input and the filter. All the output nodes produced by the same filter form a feature map, which is a matrix, while the number of feature maps is decided by the number of filters, as shown in Figure 4.3(b).



(a) (b)

Figure 4.3: Example of Convolutional layer: (a) Example of a 3×3 filter (gray matrix) with stride 1 acting on a 4×4 input. (b) The input is convolved with 3 filters, thus produces 3 two-dimensional feature maps.

In other words, all the nodes in one output feature map share the same weights.

For the rth feature map in layer l, the node $\mathbf{C}_{r,l}^{a,t}$ at ath row and tth column can be calculated by

$$\mathbf{C}_{r,l}^{a,t} = f(\sum_{v} \sum_{i=0}^{m_l-1} \sum_{j=0}^{n_l-1} \boldsymbol{x}_{v,l}^{a \cdot s_l+i, t \cdot d_l+j} \boldsymbol{\omega}_{v,r,l}^{i,j} + \boldsymbol{b}_{r,l}) \tag{4.3}$$

where $\boldsymbol{x}_{v,l} \in \mathbf{R}^{p \times q \times v}$ denotes the input of the lth layer, of which the number of channels (also called depth) is v, and each channel has a size of $p \times q$. The inputs can be the initial input signal or the output of the preceding layer. For the output of the preceding layer, v refers to the number of feature maps in the $(l-1)$th layer. $\boldsymbol{\omega}_{v,r,l} \in \mathbf{R}^{m_l \times n_l}$ refers to the vth channel of the rth filter in layer l, with size $m_l \times n_l$ and stride set to (s_l, d_l). $\boldsymbol{b}_{r,l}$ is the bias for the rth feature map. $f(.)$ is the activation function that endows the network with the ability to learn complex nonlinear relationships in the data. The activation function used in this work is Rectified Linear Unit (ReLU), which is given by

$$f(x) = \begin{cases} 0 \quad for \quad x < 0 \\ \\ x \quad for \quad x \geq 0 \end{cases} \tag{4.4}$$

Pooling is a down-sampling operation which reduces the size of the feature maps extracted in a convolution layer as well as the number of parameters introduced to the following layers by either max pooling strategy

$$\mathbf{P}_{r,l+1}^{a,t} = \underset{0 \leq i \leq m_{(l+1)}-1, 0 \leq j \leq n_{(l+1)}-1}{max} \{\mathbf{C}_{r,l}^{a \cdot s_{(l+1)}+i, t \cdot d_{(l+1)}+j}\} \tag{4.5}$$

or average pooling strategy

$$\mathbf{P}_{r,l+1}^{a,t} = \frac{\sum_{i=0}^{m_{(l+1)}-1} \sum_{j=0}^{n_{(l+1)}-1} (\mathbf{C}_{r,l}^{a \cdot s_{(l+1)}+i,,t \cdot d_{(l+1)}+j})}{m_{(l+1)} \times n_{(l+1)}} \tag{4.6}$$

where $(m_{(l+1)}, n_{(l+1)})$ is the size of the pooling region, while $(s_{(l+1)}, d_{(l+1)})$ is the strides of the pooling filter in layer $l+1$.

The final output of the CNN is calculated as:

$$\mathbf{O} = f(\sum_{j=1}^{z} \boldsymbol{x}(j)\boldsymbol{\omega}(j) + \boldsymbol{b}) \tag{4.7}$$

where $\boldsymbol{x}$ denotes the input of the output layer, $\boldsymbol{\omega}$ and $\boldsymbol{b}$ are the weights and bias that connect the $\boldsymbol{x}$ and final output, respectively.

All the weights and bias are tunable parameters (θ) which are updated by minimizing the loss function $J(\theta)$ through an optimization algorithm. For prediction problems, it is common to use MSE loss function [171], that is $J(\theta) = E_{mse}$. To update the parameters, the Adam algorithm [172] is used in this work, which has been suggested as the default optimization method for deep learning applications [173].

$$g_t = \nabla_\theta J_t(\theta_{t-1}) \tag{4.8}$$

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t \tag{4.9}$$

$$v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2 \tag{4.10}$$

$$\tilde{m_t} = \frac{m_t}{1-\beta_1^t} \tag{4.11}$$

$$\tilde{v_t} = \frac{v_t}{1-\beta_2^t} \tag{4.12}$$

$$\theta_t = \theta_{t-1} - \frac{\alpha \cdot \tilde{m_t}}{\sqrt{\tilde{v_t}} + \epsilon} \tag{4.13}$$

where g_t is set to be the gradient of the loss function $J(\theta)$ at tth training iteration.

m_t and v_t are the estimated first moment (the mean) and second moment (the uncentered variance) of the gradient respectively, and $\tilde{m}_t$ and $\tilde{v}_t$ are their bias-corrected values. β_1 and β_2 are exponential decay rates, while β_1^t and β_2^t are β_1 and β_2 to the power t. θ_t is the updated parameters.

CNNs are effective tools for extracting features from a high-dimensional data, and have been widely used in a range of fields, such as image processing, text classification, and speech recognition. These high-dimensional signals usually have high spatial or temporal correlations in adjacent variables, which can be effectively extracted through the convolution operations. Due to the fact that time series data is ubiquitous and is constantly generated in many engineering processes and in our daily life, there are imperative needs to develop efficient techniques to extract useful information from time series data. Considering the merits of CNNs in terms of automatic feature extraction and low overfitting risk, their applications in dealing with large amount of time series signals have also been investigated. For example, some reports have confirmed the potential of CNNs in extracting the representative features from time series data. In [174], a CNN is used for solving a human activity recognition problem where the inputs of the network are multichannel time series signals collected from inertial sensors, and the outputs are related human activities. In this application, the filters in the CNN move along the temporal dimension for each sensor (each sensor corresponds to a row in the 2-dimensional input). In [175], a multi-scale CNN is used for time series data classification problems. The CNN architecture has multiple branches in its first layer that can extract features of different frequency and time scales. Further, CNNs have also been used for time series forecasting and estimation, and fault diagnosis.

4.2 Related Works

This section presents the concepts of transfer learning and network pruning, which are the key elements of the proposed battery capacity estimation method.

4.2.1 Transfer Learning

Most machine learning technologies are developed under the assumption that the data used for training and testing should be drawn from the same feature space and have the same distribution [176]. In other words, the statistical model trained on a dataset can not be directly applied to another dataset with different distribution, and the model has to be reconstructed and retrained from scratch on the new dataset. However, collecting enough data to retrain a new model is often time-consuming and costly in real-world applications, and on the other hand, the training process often takes a long time. Transfer learning was proposed to handle such cases, aiming to reduce the need and effort for data recollection and model re-training while leveraging the knowledge learned from a source task to a different but related task though the data of these two tasks may have different distributions [177]. Based on the transfer learning, the latter task requires much less data to retrain the model.

Generally speaking, the transfer learning can be achieved in two ways for neural network models: one is to utilize the original pre-trained network except for its last fully-connected (FC) layer as a fixed feature extractor for the new task, and the other approach is to fine-tune the parameters of the pre-trained network using data of the new task. Choosing the suitable transfer learning technique mainly depends on the size and similarity of the target dataset to the original one used in pre-training [177]. It is widely accepted that the first several layers of a network are used to extract low-level features while the following few layers are used to extract high-level features [178], the FC layers are used to learn

non-linear combinations of these features. Therefore, the first transfer learning technique is suitable for cases when the target dataset is quite similar to the source dataset. On the other hand, fine-tuning the pre-trained network is more suitable for a target dataset which has different distribution statistics from the original dataset.

When the CNN is applied for battery capacity estimation, it is obvious that the model trained on one battery degradation dataset can not be directly applied to another type of batteries with different specifications and charging/discharging policies. When the battery specification changes or the operation condition varies significantly, the CNN model needs to be retrained or even rebuilt from scratch on the new battery degradation data, which is quite time-consuming and requires a huge amount of training data. Considering that different batteries have different local degradation patterns under different operating conditions but have similar overall degradation trends [179], and the transfer learning technique does not require the test data to have the same statistical distribution as the source data [180], an effective solution to overcome the limitations of CNN-based capacity estimation method is to introduce the concept of transfer learning. To this end, this chapter combines the CNN with transfer learning, aiming to eliminate the need to recollect a complete battery cycling data and build a completely new CNN model.

4.2.2 Network Pruning

When CNNs are used for battery capacity estimation, the deep and complex network structure and a large number of parameters to tune demand huge memory storage and high computational cost, which hinder its implementation in embedded devices [181]. To acquire a smaller and much more compact model, network pruning, which can remove redundant network connections based on a predefined criteria, has proven to be an effective approach. After pruning, the model will

have much fewer parameters with little or no impact on performance, while the computational complexity and storage space can be significantly reduced. According to the granularity level, the pruning can be applied at four levels, namely weight-level, filter-level, channel-level and layer-level [182]. Though a number of methods have been proposed so far for network pruning at different granularity level, they are all based on the similar framework, that is, to evaluate the importance of each weight/ kernel/ channel/ layer and remove those insignificant ones. The core of these methods is to choose a suitable importance indicator that determines which element should be removed. Different criteria have been proposed in the literature, for example, the absolute values of weights were directly used as the importance indicator in [183], the second order derivatives of a layer-wise loss with respect to the corresponding weights were used to identify the importance of each weight in [184], while [185] calculated the percentage of zero activation of a neuron after the ReLU mapping to determine which neuron should be removed, based on the assumptions that neurons with higher percentage values provide less power to the results.

Considering that the CNN structure used in capacity estimation often has less number of filters and layers, weight-level and channel-level pruning are more suitable. Further, neuron pruning simplifies the network structure and also reduces the number of weights, therefore it is often more effective than the weight pruning approach only [186]. As a consequence, a channel-level pruning method is proposed in this chapter to remove unimportant neurons, where each neuron is viewed as one channel. Considering that the FC layers are less sensitive to pruning than the convolutional layers [183], and the FC layers have the most number of parameters, we choose the FC layers as the pruning object. In this way, more parameters can be pruned with minimal impact on the output. The final compressed model will use much less memory and require less calculations, which makes it possible to implement the resultant model in the on-board BMS. The

detailed pruning method will be elaborated in the following sections.

4.3 The Proposed Methodology

In traditional capacity estimation methods, the estimated performance relies on the number of health features extracted from the measurement data and the way they are combined [187]. However, it is not easy to manually extract features from the raw data effectively and efficiently. Therefore, a CNN-based capacity estimation framework is proposed in this chapter due to the CNNs are capable of making full use of the large volume of historical measurements and extracting features automatically. In this section, the proposed framework is described in detail. First, the input generation method including data normalization, segmentation and time series-to-image conversion is introduced, and the target output of the model is clarified. Then the construction of the proposed CNN model is described step by step, which incorporates both the concepts of transfer learning and network pruning.

4.3.1 Input Generation and Model Output

The input variables used for battery capacity estimation in this chapter are current, terminal voltage and charge capacity, where the charge capacity is calculated by integrating the current with respect to time for a partial charging segment. These variables are highly correlated, embedding lots of information. To make the most of the embedded information in these measurements and to automatically extract features, a CNN model is built and a time series-to-image conversion method is proposed to convert the measured time series signals to 3-dimensional (3-D) images, as the CNN usually takes images as the input and the resultant 3-D images embed the spatially and temporally correlated information among these variables. Generally, with the same number of weights (e.g. a $2 \times 2 \times 3$

filter for 3-D image input and a 3×4 filter for 2-D temporal input both have 12 parameters), more information can be extracted from the converted 3-D images than from the temporal signals, since the converted 3-D input can link both adjacent signals and signals after a period of time together in single visual field of the filter. Given that there is little chance to fully charge/discharge a battery from a fixed initial state in practical applications [141], partial charging curves based capacity estimation methods are more practical. Therefore, data segmentation is performed before the conversion stage to generate partial charging curves with flexible start/end point.

Figure 4.4 illustrates the steps of the proposed input generation method. Note that the raw data, i.e. current, voltage and charge capacity, have different scales, therefore directly converting these variables to 3D inputs will slow the training process and significantly degrade the model performance. Thus, data normalization is applied first. In this work, the min-max normalization strategy is adopted, which retains the original distribution of data and all transformed data fall into the range of [-1,1], reflecting both the charging and discharging phases. Then M data chunks are segmented from one charge cycle, as illustrated in the sub-picture in the lower right corner of Figure 4.4, each data chunk is a $n \times 3$ matrix that represents a partial charging segment, where 3 is the number of involved variables, and n denotes n continuous data points for each variable. Two adjacent data chunks have c overlapping data points. That is, n continuous data points of the three variables with the same starting point are extracted for M times in one cycle, and each time the starting point moves down $n - c$ points. Finally, each partial charging segment is converted into a 3 dimensional image with the size of $\sqrt{n} \times \sqrt{n} \times 3$. In this transformation stage, the number of data chunks M generated in a cycle contains L data points in total, is determined by the segmentation length n and the overlap size c between two adjacent data chunks, $M = floor(\frac{L-n}{n-c}) + 1$. The function floor(.) gives the greatest integer less than or

equal to the input parameter.

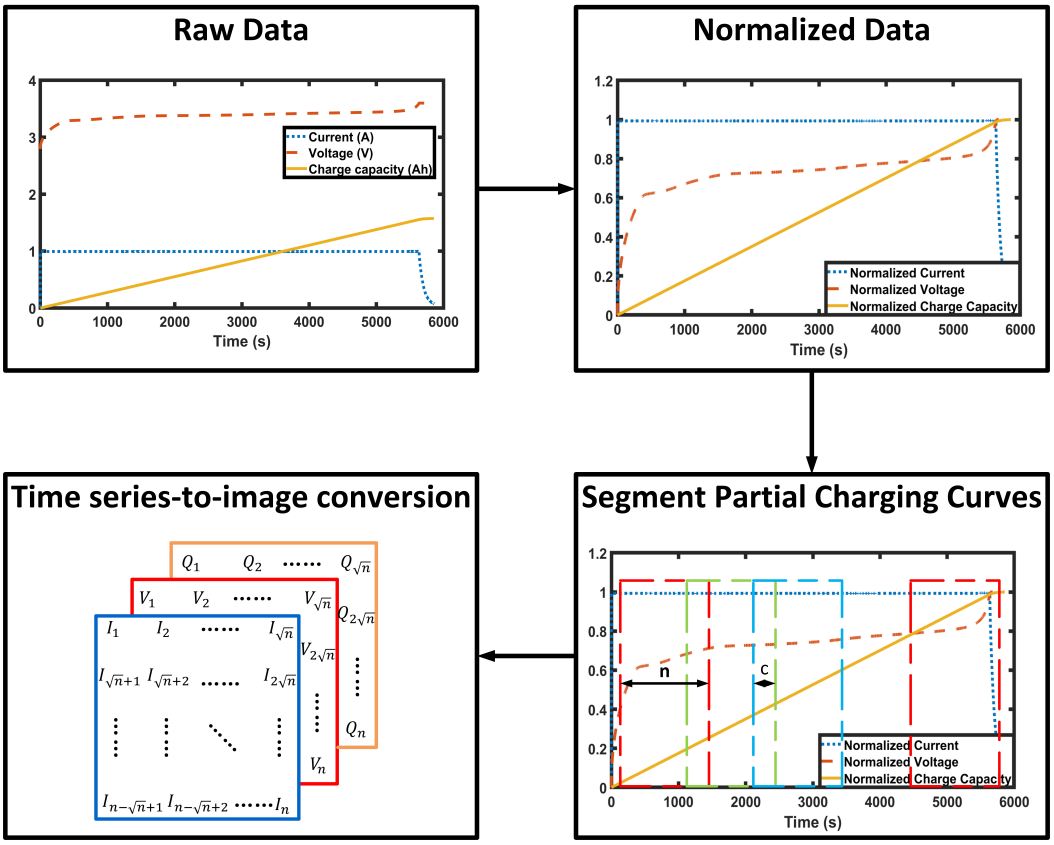


Figure 4.4: Input generation steps: data normalization, charging curves segmentation, and time series-to-image conversion

This input generation method can increase the number of samples used for CNN training, and allow fast online capacity estimation only using flexible partial charging curves, moreover, it is an enabling block to apply the CNNs for time series signals. Each input sample is associated with an output sample, which is the full discharge capacity that is calculated by integrating the discharge current over time for the entire full discharge cycle which immediately follows the charge cycle that generated the input sample.

4.3.2 Network Construction

As shown in Figure 4.5, the CNN model proposed in this chapter is constructed in three consecutive steps. Firstly, a CNN architecture is designed based on a classical and effective CNN structure named LeNet-5 [188], and the CNN model is pre-trained using the source dataset. Secondly, the knowledge learned from the source dataset is transferred to a new task, and the trainable parameters are fine-tuned using the target dataset to produce the target model. Finally, a pruned CNN model is constructed by pruning insignificant neurons of the target model using the proposed pruning algorithm. The detailed process of each stage will be further elaborated below.

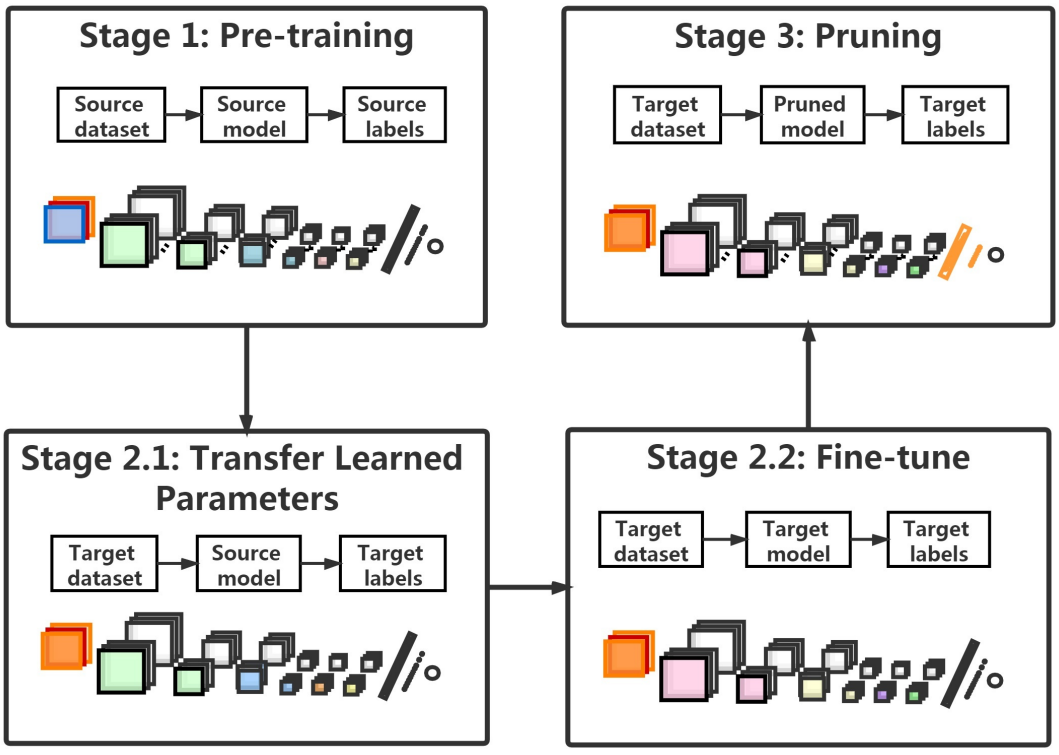


Figure 4.5: Model construction stages: (1) pre-training, (2) transfer learned parameters and fine-tuning, (3) pruning

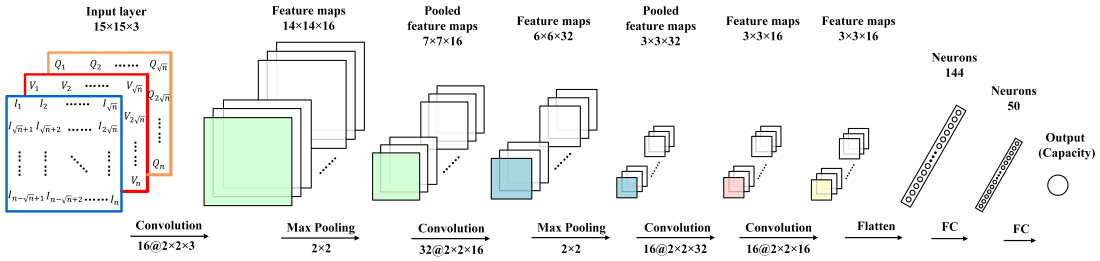


Figure 4.6: Model structure of the pre-trained CNN model

4.3.2.1 Pre-training

The designed CNN architecture is graphically illustrated in Figure 4.6 and the output size and number of parameters for each layer are summarized in Table 4.1. It has 4 convolutional layers, 2 max pooling layers, a flatten layer and 2 fully-connected layers. The zero-padding method [189] is used in the last two convolutional layers to control the size of the feature maps. As aforementioned, the inputs are $\sqrt{n} \times \sqrt{n} \times 3$ matrices, their weight and height dimensions both are $\sqrt{n}$, which are determined by the length n of the data chunk, and 3 is the depth of the input, which is determined by the number of variables. In this work, n is set to be 225, thus the size of the input is $15 \times 15 \times 3$. The output size $w \times h \times c$ refers to the size of the feature maps, where w, h, c denote the weight, height and channel, respectively. For each convolutional layer, $N@k \times m \times d$ denotes the filter design, which implies that N filters with the size of $k \times m \times d$ are convolved with the input of this layer, where the depth of the input is d, and the depth/channel of the output feature maps must be N. The model pre-trained on the source dataset is denoted as CNN(S), and there are totally 12693 parameters in this model, which are trained using the source dataset.

The source dataset was split into three sets: training, validation and testing. The CNN model is trained from scratch using the training and validation sets, and then tested on the testing set. All the 12693 parameters are trained and then used in the following steps.

Table 4.1: Layer configurations and number of parameters of the pre-trained CNN model

Layer	Layer name	Stride	Output size	Parameters
L1	Conv_1(16@2 × 2 × 3)	(1,1)	(14 × 14 × 16)	208
L2	Maxpool_1(2 × 2)	(2,2)	(7 × 7 × 16)	0
L3	Conv_2(32@2 × 2 × 16)	(1,1)	(6 × 6 × 32)	2080
L4	Maxpool_2(2 × 2)	(2,2)	(3 × 3 × 32)	0
L5	Conv_3(16@2 × 2 × 32)	(1,1)	(3 × 3 × 16)	2064
L6	Conv_4(16@2 × 2 × 16)	(1,1)	(3 × 3 × 16)	1040
L7	Flatten	-	144	0
L8	FC1	-	50	7250
L9	FC2	-	1	51

4.3.2.2 Transfer Learning and Fine-tuning

In real-life applications, it is not a trivial task to collect and annotate battery long-term cycling data. While the CNN architectures used for battery capacity estimation usually contain tens of thousands of free parameters to train, requiring a large number of labeled training data and long training time. To circumvent this problem for practical applications, transfer learning is adopted to transfer the knowledge learned from the source dataset to a new task with a much smaller dataset. In this work, transfer learning is achieved through the following two steps (Stage 2.1 and Stage 2.2 in Fig. 4.5): first, the structure and parameters of the CNN model pre-trained on the source dataset are transferred to the target dataset, then specific layers are fine-tuned using the target dataset. The resultant

CNN model after the transfer learning is denoted as CNN(S)-TL in the remainder of this chapter.

4.3.2.3 Pruning

As aforementioned, the CNN(S)-TL model used for capacity estimation fine-tuned using the target dataset has 4 convolutional layers, 2 max pooling layers, a flatten layer and 2 fully-connected layers with a total of 12693 parameters. In real-world applications, such a CNN model will require large computation and memory resources, which limits its implementation in on-board BMS. An intuitive way to solve this problem is to reduce the model complexity and the number of parameters. Inspired by the work presented in [190] that selects the best hidden neurons and avoids redundant structure, a FRA-based network pruning method is proposed in this work to remove all redundant neurons based on measuring the contributions of all neurons to the outputs. Compared with other pruning methods, the proposed method identifies and preserves the most important neurons of networks and re-assign weights to the retained neurons. In other words, the important features are retained, which may help to improve the accuracy of capacity estimation for the CNN model refined on a rather small task dataset. Further, the ability of the proposed method to simultaneously update weights and select neurons implies that there is no need to fine-tune the network after pruning. As shown in Table 4.1, more than half of the total parameters of the CNN model are from the last two fully-connected layers, the proposed method is therefore deployed to these two layers. The redundant neurons and all the incoming and outgoing connections associated with these neurons are removed, leading to significantly reduced memory usage and computational complexity for online capacity estimation.

Supposing $X \in \Re^{N\times(L+1)}$ is the input matrix to the first fully-connected (FC1) layer, where N is the number of samples and L is the length of the vector formed

in the flatten layer and inputted to the FC1 layer, and the input matrix X includes a bias vector with all elements being set to 1. $Y \in \Re^{N\times 1}$ is the output of the second fully-connected (FC2) layer, i.e. the output of the CNN(S)-TL model, which can be written as

$$\mathbf{Y} = f(\mathbf{X\Theta})\mathbf{W} + \mathbf{\Xi} \tag{4.14}$$

where $\Theta \in \Re^{(L+1)\times S_1}$ represents the parameter matrix of the FC1 layer, which consists of the $L \times S_1$ weight matrix and $1 \times S_1$ bias vector. S_1 is the number of neurons inputted to the FC2 layer, namely, the number of latent features obtained by the CNN-TL model to estimate capacity. W denotes the parameter matrix of weights and bias of the FC2 layer, and Ξ is the residual vector.

To remove redundant neurons in FC1 and FC2 layers without sacrificing the overall performance of the network, the detailed procedure is illustrated in Figure 4.7, and the neuron selection process is elaborated as follows.

Take the FC1 as an example, the pruning procedure consists of 3 steps:

Step 1 - Calculate the contribution of each neuron in the FC1 layer. Let $\mathbf{Y}^f = f(\mathbf{X\Theta})$, $\mathbf{Y}^f = [\mathbf{y}_1^f, \mathbf{y}_2^f, ..., \mathbf{y}_N^f]^T \in \Re^{N\times S_1}$ is the output of the FC1 layer. According to Equation (3.12), the net contribution of each neuron in this layer can be calculated as

$$\Delta E_{k+1} = \frac{\left(\left(\mathbf{Y}^f\right)^T x_{k+1}^{(k)}\right)^2}{\left(\left(x_{k+1}^{(k)}\right)^T x_{k+1}^{(k)}\right)}, \quad k = 0, 1, ..., L-1 \tag{4.15}$$

Step 2 - Rank the neurons based on their contributions, and then select the highest ranked neuron. With the selected neurons, a new output of the FC1 layer is obtained, denoted by $\hat{\mathbf{Y}}^f$. The root mean square error between $\mathbf{Y}^f$ and $\hat{\mathbf{Y}}^f$

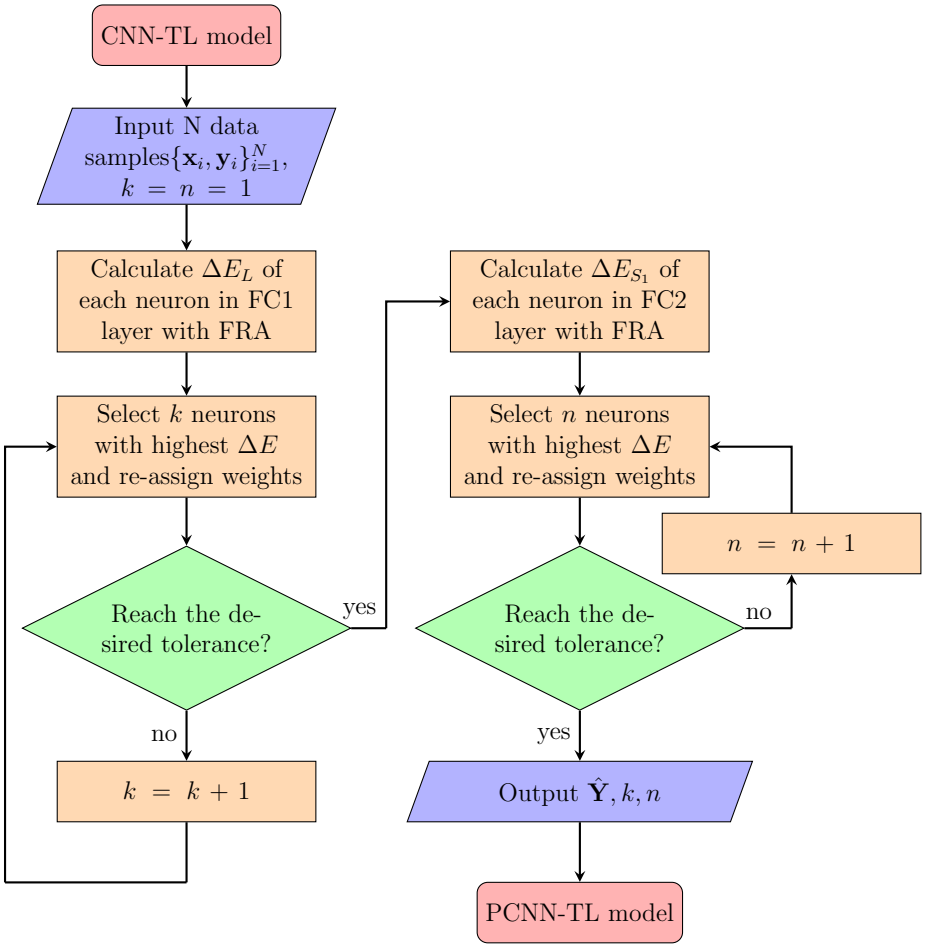


Figure 4.7: Flowchart of the FRA-based neuron pruning process

over all the training samples is given as

$$\begin{aligned} rmse^f &= \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left((\hat{\mathbf{y}}_i^f - \mathbf{y}_i^f)(\hat{\mathbf{y}}_i^f - \mathbf{y}_i^f)^T\right)} \\ &= \frac{1}{\sqrt{N}}\left\|\hat{\mathbf{Y}}^f - \mathbf{Y}^f\right\|_F \end{aligned} \tag{4.16}$$

where $\|\cdot\|_F$ is the Euclidean norm.

Step 3 - Repeat Step 2 until the maximum number of neurons in the FC1 layer is reached or the $rmse^f$ is smaller than the predefined error bound. The parameters

for the selected terms are calculated by

$$\hat{\theta}_j = \frac{\left(x_j^{(j-1)}\right)^T \mathbf{Y}^f - \sum_{i=j+1}^{k} \hat{\theta}_i \left(x_j^{(j-1)}\right)^T x_i^{(j-1)}}{\left(x_j^{(j-1)}\right)^T x_j^{(j-1)}} \tag{4.17}$$

where $j = k, k-1, ..., 1$, and k is the total number of the selected terms.
The neuron pruning process for FC2 is the same as for the FC1 layer. Repeat step 1-3, and until the equivalent or even better result with fewer neurons is achieved. The proposed method using FRA for neuron pruning can simultaneously reduce the model size and re-tune the weights, which eliminates the need to fine-tune the network after neuron pruning, and reduces the number of computing operations without sacrificing the accuracy. After pruning, the new network is more compact than the original CNN(S)-TL model in terms of the model size, and hence the computational complexity is significantly reduced. The resultant model is denoted as pruned CNN(S)-TL (PCNN(S)-TL) in the rest of this chapter.

4.4 Experimental data

The detailed information of the source battery degradation dataset used to pre-train the CNN model and the target dataset used to validate the performance of the proposed PCNN(S)-TL model are summarized in Table 4.2. These include one large-scale test dataset for 16 cells and one small-scale dataset for 4 cells, and each cell in the first dataset has roughly gone through 1000 charging/discharging cycles, while cells in the small dataset were only tested for 30 reference cycles. Therefore, the large-scale dataset is employed as the source dataset to pre-train the CNN model, and then the learned knowledge will be transferred to the small-scale dataset. Note that these two sets of cells are all the same type, but have different specifications and are tested under different cycling scenarios.

Table 4.2: Summary of the two battery degradation datasets

	Source dataset	Target dataset
Available cycles for each cell	roughly 1000 charge /discharge cycles	30 reference charge /discharge cycles
Number of cells	16	4
Nominal capacity	1.1 Ah	1.6 Ah
Nominal voltage	3.3 V	3.2 V
Battery type	$LiFePO_4$	$LiFePO_4$

4.4.1 Source dataset

The source dataset is collected from 124 commercial lithium iron phosphate (LFP)/graphite batteries that are manufactured by A123 System (APR18650M1A), with a nominal voltage of 3.3 V and nominal capacity of 1.1 Ah [191]. They are cycled to failure under fast-charging policy at a constant temperature of 30°C. Under the fast-charging experimental procedure, denoted as "C1(Q1)-C2", the cell is first charged from 0% to Q1 state of charge (SOC) at a constant current (CC) C1, then charged from Q1 to 80% SOC at constant current C2. After reaching 80% SOC, all cells are charged at 1C until the voltage reaches its upper cutoff potential 3.6 V. A 1C rate represents that the charge/discharge current will charge/discharge the entire battery in 1 hour [192]. For these batteries with a nominal capacity of 1.1 Ah, 1C equals to a charge/discharge current of 1.1 A. Finally a constant voltage (CV) mode continues until the charge current falls to 22 mA. The whole charging policy is illustrated in Figure 4.8. All cells are discharged under a CC-CV protocol, discharging at CC of 4C until the cell voltage falls to 2.0 V with a current cutoff of 22 mA. And the sampling frequency was 0.5Hz.

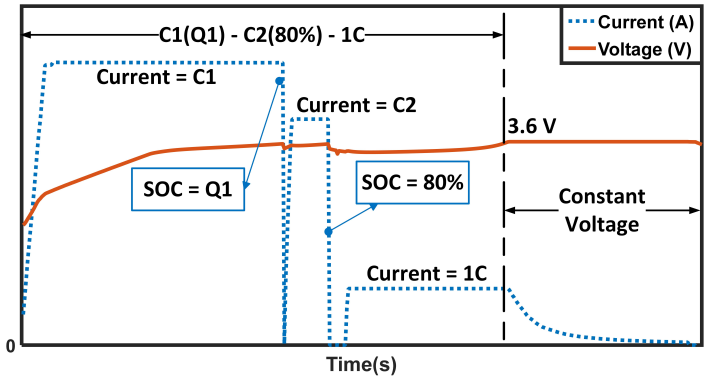


Figure 4.8: Charging policy of the source dataset

The first 16 cells in dataset 'batch3' are used in this work and they are charged with different policies. These 16 cells are divided into 2 groups, where the first group consists of 12 cells and is used to train the network, while the second group consists of the remaining 4 cells and is used as the testing set. The detailed policies applied to charge these 16 cells from 0% to 80% SOC are summarized in Table 4.3, and the groups used for training and testing are also given in the table. During the pre-training process, samples generated from the Group 1 are first shuffled and randomly split into a training set and a validation set with the ratio of 7:3, which are then used to pre-train the CNN model. The Group 2 is then used to test the performance of the trained CNN model.

4.4.2 Target dataset

To validate the proposed PCNN(S)-TL based capacity estimation algorithm, data of 4 commercial cylindrical LFP cells with nominal voltage of 3.2 V and nominal capacity of 1.6 Ah are used as target dataset in this work. These cells used in the target dataset are different from the cells used in the source dataset as shown in Table 2, and they are tested in parallel using a BTS 4000 battery test system

Table 4.3: Summary of the policies for charging the cells from 0% to 80% SOC

	Battery	Charging policies	Cycles
	1	'5C(67%)-4C'	1008
	2	'5.3C(54%)-4C'	1062
	3	'5.6C(19%)-4.6C'	1266
	4	'5.6C(36%)-4.3C'	1114
	5	'5.6C(19%)-4.6C'	1047
Group 1	6	'5.6C(36%)-4.3C'	827
(Training set)	7	'3.7C(31%)-5.9C'	666
	8	'4.8C(80%)-4.8C'	1835
	9	'5C(67%)-4C'	827
	10	'5.3C(54%)-4C'	1038
	11	'4.8C(80%)-4.8C'	1077
	12	'5.6C(19%)-4.6C'	816
	13	'5.6C(36%)-4.3C'	931
Group 2	14	'5.6C(19%)-4.6C'	815
(Testing Set)	15	'5.6C(36%)-4.3C'	857
	16	'5.9C(15%)-4.6C'	875

made by NEWARE. Thermocouples with measurement error less than 0.1 °C are attached to measure the cell surface temperature.

High charging current rates are used to accelerate the aging speed of these cells,

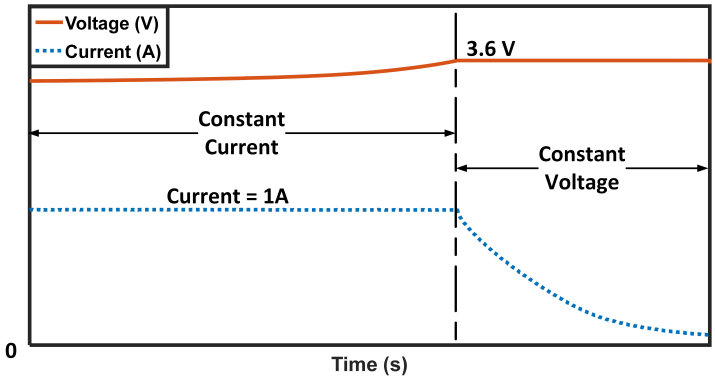
and a reference cycle is tested under the temperature of 25 °C every 30 cycles. The protocol of the reference cycle recommended by the manufacturer includes a CC-CV charging and a CC discharging process. As shown in Figure 4.9(a), under the CC-CV charging policy, the cells were charged at a uniform CC of 1 A until the voltage reaches the upper cutoff voltage 3.6 V, and then charged with 3.6V CV to a current cutoff of 75mA. All cells were subsequently discharged with a 1 A CC to the lower cutoff voltage of 2.0 V. The sampling frequency for all the equipment used in this experiment was set as 1Hz. The resting time between the charge and discharge process was set to 1 hour.

The CNN(S)-TL model is fine-tuned using the data collected from the reference cycles of the 3 cells, while the data from the remaining cell is used for testing. The capacity degradation trends for these 4 cells are shown in Figure 4.9(b).

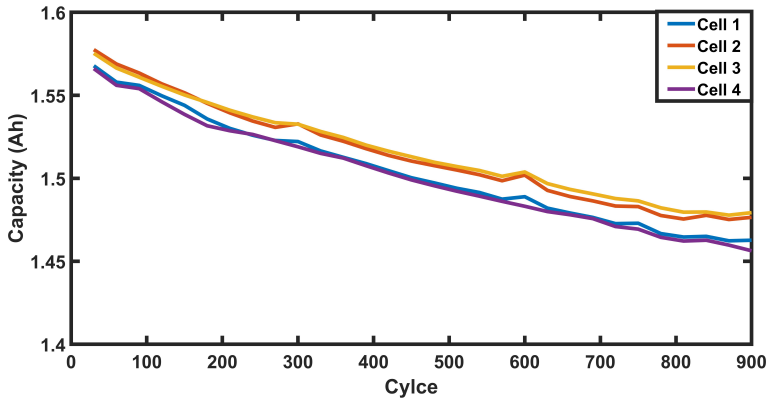
4.5 Implementation Details

4.5.1 Input structure

Training samples generation process is illustrated in Figure 4.10, and M_i partial charging segments with the length of 225 consecutive data points were extracted from the $i\ th$ charging cycle, $i = 1, 2, ..., N_t$, where N_t is the total number of charging cycles involved in the training dataset. After a series of trial-and-error tests, it is found that the length of 225 data points is the best trade-off between the number of generated training samples and the information embedded in each sample. Each segment was formulated to a $15 \times 15 \times 3$ matrix. This leads to $\sum_{i=1}^{N_t} M_i$ data segments fed into the network to train the model. For a given overlap size c, the number of samples generated from the $i\ th$ charge cycle containing



(a) CC-CV charging strategy



(b) Capacity degradation trends for cell 1 - cell 4

Figure 4.9: Charging strategy (a) and actual capacity degradation (b) of the four cells in target dataset

L_i data points in total is calculated as follows

$$M_i = floor(\frac{L_i - 225}{225 - c}) + 1 \tag{4.18}$$

where the function $floor(\cdot)$ gives the largest integer less than or equal to the input value. The overlap size was set as $c = floor(225 \times 0.8)$ in this work.

For the test dataset, only one partial charging segment was extracted per cycle, and this segment has a fixed length of 225 data points and a random starting point. The training data of the source and target datasets are used to pre-train

the CNN(S) and fine-tune the CNN(S)-TL model, respectively, and they are also used to prune the input neurons of the fully-connected layers. The testing data from the source dataset is used to verify the effectiveness of the CNN-based capacity estimation method on cells with different charging policies, while the testing data of the target dataset is used to verify the effectiveness of the transfer learning strategy. Further, the testing data of the target dataset are also used to validate the performance of the pruned model.

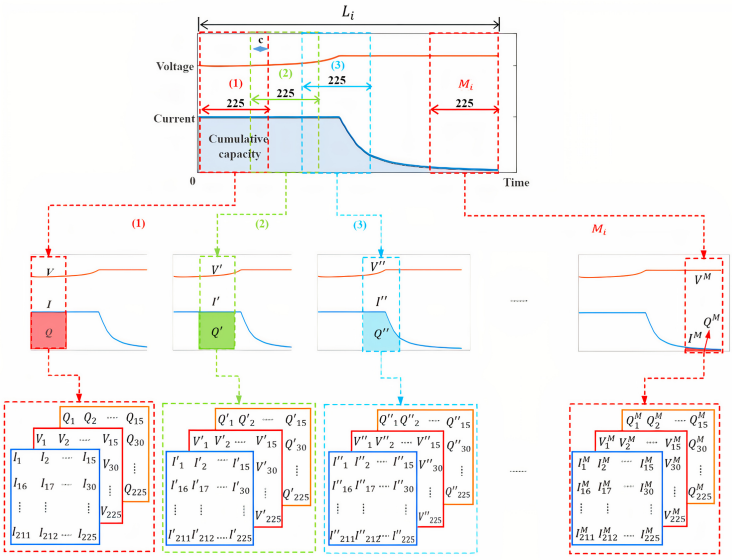


Figure 4.10: Illustration of training samples generation. M_i samples are segmented from one charge cycle, each sample refers to a partial charge cycle with a length of 225 data points. Three variables, i.e. voltage, current and cumulative charge capacity, of each partial charge segment are then together form a $15\times15\times3$ input matrix. Note that the charge capacity is calculated by integrating the current with respect to time for a partial charging segment.

4.5.2 Training Protocols

Important hyperparameters are pre-defined before training and fine-tuning the model to maximize the performance of the model. When the CNN model is trained from scratch, all the kernels are initialized based on Xavier uniform ini-

tializer and the bias are initialized to zero [193]. The maximum training epochs are set to 80 with the mini-batch size of 128 samples. To avoid overfitting problem, early stopping method with patience set to 5 is applied to stop training once the model performance has not improved for 5 consecutive epochs on the validation dataset. The Adam algorithm with learning rate of 0.001 is used to update the parameters.

The CNN(S)-TL model is fine-tuned from the pre-trained CNN(S) model on the target dataset. First, all layers of the pre-trained CNN(S) are copied to the CNN(S)-TL model. Considering that the target dataset is much smaller than the source dataset, and the features extracted from the first few layers are universal, the parameters of layers prior to the third convolutional layers are retained unchanged. Then the subsequent layers are fine-tuned to learn the specific features on target dataset with a learning rate of 0.0001, which is ten times smaller than the one used in the scratch training.

In the pruning stage, all the layers copied from the CNN(S)-TL model are fixed except for the last two fully-connected layers. When the desired performance of the model is achieved with the minimum number of neurons using the proposed FRA-based network pruning method, it eliminates the necessity of fine-tuning the network after removing redundant neurons, due to that new weights have already been assigned to remaining neurons using the proposed algorithm.

4.5.3 Evaluation Criteria

In the experiment, the total mean absolute error (MAE), root mean-square error (RMSE) and normalized estimation error (NEE) are used to assess the model performance. They are effective measures to assess the deviations in distances between the estimated capacities $\hat{\mathbf{Y}}$ and the reference values $\mathbf{Y}$. The formulas are defined as follows:

$$E_{mae} = \frac{1}{N}\left\|\hat{\mathbf{Y}} - \mathbf{Y}\right\|_1 \tag{4.19}$$

$$E_{rmse} = \frac{1}{\sqrt{N}}\left\|\hat{\mathbf{Y}} - \mathbf{Y}\right\|_F \tag{4.20}$$

$$E_{nee} = \frac{1}{\sqrt{N}\mathbf{Q}}\left\|\hat{\mathbf{Y}} - \mathbf{Y}\right\|_F \times 100\% \tag{4.21}$$

where N is the sample size of test data, Q is the nominal capacity. $\|\cdot\|_1$ and $\|\cdot\|_F$ are the L1 norm and Euclidean norm, respectively. $\mathbf{Y} = \left[\{y_i\}_{i=1}^N\right]^T$ and $\hat{\mathbf{Y}} = \left[\{\hat{y}_i\}_{i=1}^N\right]^T \in \Re^{N\times 1}$, y_i is the reference capacity for the $i\ th$ sample, while $\hat{y}_i$ is the corresponding estimated value.

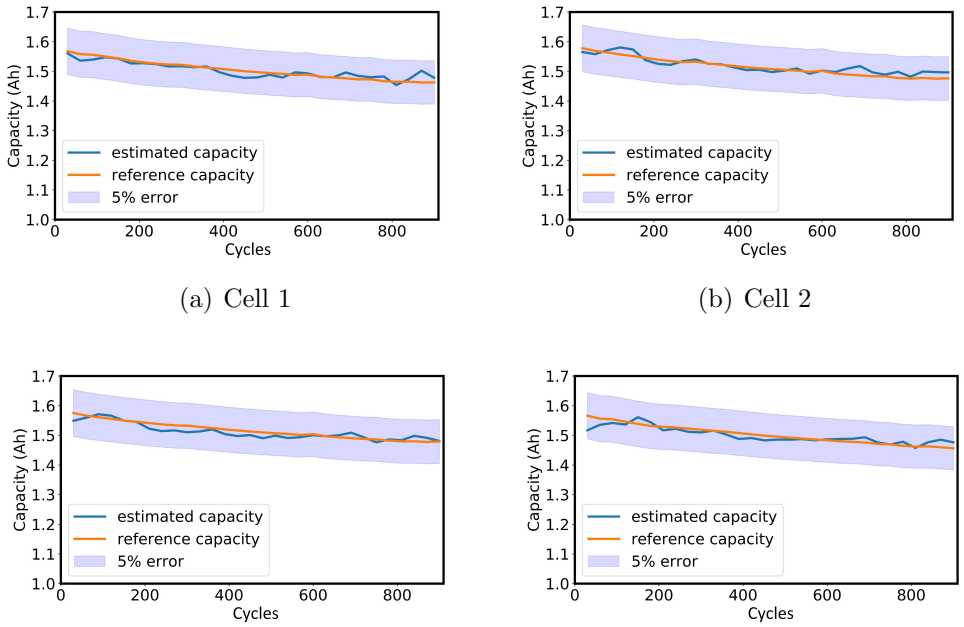
In addition, the floating point operations (FLOPs) are considered to measure the computational cost in this study. As the pruning algorithm is applied to the fully-connected layers in this study, only the number of FLOPs for a fully-connected layer is considered, which is calculated as $FLOPs_{fc} = (2 \times I) \times O$, where I and O are the number of input and output neurons, respectively.

4.6 Results and Discussion

In this section, the datasets introduced in section 4.4 are used to validate the performance of the proposed method. The capacity estimation results of the proposed method on different cells are given. The performance of the proposed model is compared with the CNN model, CNN with network pruning, and CNN with transfer learning. The performance improvements of the proposed framework in terms of accuracy and model size are analysed. Moreover, the effect of the number of layers fine-tuned during the transfer learning stage is investigated to explain the reasons why we choose to fine-tune 4 layers.

4.6.1 Capacity estimation results

The capacity estimation results of the proposed PCNN(S)-TL model on target dataset are shown in Figure 4.11 and Figure 4.12. As shown in Table 4.2 and presented in Section 4.4.2, each cell in the target dataset has 30 reference cycles, and a reference cycle is conducted for every 30 cycles. The red lines in Figure 4.11 and 4.12 show the degradation trend of each cell under 900 cycles. It should be noted that only the reference cycles (the 30-th, 60-th,..., 900-th cycle) are used for capacity estimation and have actual reference/estimated values. It can be observed from Figure 4.11 that the proposed model can produce accurate estimation on the test cells, and the estimated capacity can well track the degradation trend. Here the four-fold cross validation approach is used, when one cell is used for testing, data of the other three cells are used to generate the training samples according to the input generation method introduced in Section 4.3.1 to fine-tune and prune the pre-trained CNN model. The average NEE for the 4 cells is 0.84%. Similar estimation results are obtained when using discharging segments to estimate the capacity, and the average NEE for the 4 cells is 0.83%. However, in practical electric vehicle applications, CC charging process is possible while the discharging process is usually highly dynamic, hence limiting the application of using discharging data to estimate the capacity. Therefore, this work mainly considers using charging data to estimate capacity to provide stable and reliable estimation results. The experimental results have revealed that the parameters of the developed model can be updated using the data from one cell, and then the resultant model can be directly applied to other cells who have the same specifications and similar charging policies. Figure 4.12 illustrates that updating the proposed model parameters using the data of the first several cycles, the degradation trend for the following cycles can be well captured. Here data of the first 2/3 cycles of all cells in the target dataset are used to fine-tune and prune the model pre-trained on the source dataset, and the remaining 1/3 cycles are



(a) Cell 1

(b) Cell 2

(c) Cell 3

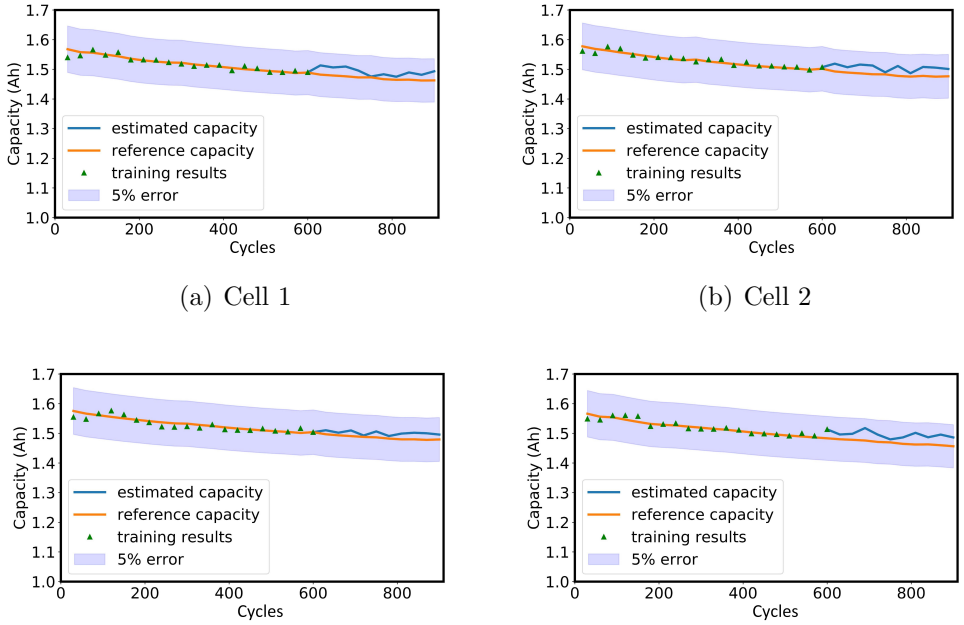
(d) Cell 4

Figure 4.11: Capacity estimation results produced by the PCNN(S)-TL model for the four cells in target dataset. For testing the performance on each cell, data of other three cells are used to generate training samples and fine-tune and prune the model.

used for testing. Both figures demonstrate the performance of the PCNN(S)-TL model for battery capacity estimation.

4.6.2 Algorithm verification

The proposed battery capacity estimation method has been further verified in this section. The CNN model directly trained using the target dataset is denoted as CNN(T), while CNN(S) refers to the model trained using the source dataset. The models acquired using network pruning, transfer learning and both of them are denoted as PCNN(T), CNN(S)-TL and PCNN(S)-TL, respectively. During the knowledge transfer stage, the first 4 layers (2 convolutional layers and 2 max pooling layers) are fixed, and the remainder layers are fine-tuned. The values



(a) Cell 1 (b) Cell 2

(c) Cell 3 (d) Cell 4

Figure 4.12: Capacity estimation results produced by the PCNN(S)-TL model for the four cells in target dataset. For testing the performance on each cell, the first 2/3 cycles of all cells in the target dataset are used to generate training samples and fine-tune and prune the model, the remaining 1/3 cycles of each cell are used for testing.

of the aforementioned evaluation criteria for these models are averaged based on 100 repetitions of the experiments, and the test results of 4 cells are summarized in Table 4.4 in terms of MAE, RMSE and NEE. Further, the values of the model size, total number of parameters (of the entire model), and pruned FLOPs (of the last two fully-connected layers) of the CNN(T), PCNN(T), CNN(S)-TL and PCNN(S)-TL models are averaged for the 4 cells, which are listed in Table 4.5. The results presented in these tables confirm the effectiveness of the proposed capacity estimation methods.

Table 4.4 reveals that, firstly, the proposed network pruning method does not degrade the estimation accuracy. The average NEE of PCNN(T) is 1.00%, which is 9.91% smaller than that of the CNN(T) model. The NEE of PCNN(S)-TL is

Table 4.4: Capacity estimation results produced by CNN(T), PCNN(T), CNN(S)-TL, and PCNN(S)-TL on target dataset

Cell No.	Assess	CNN(T)	PCNN(T)	CNN(S)-TL	PCNN(S)-TL
Cell1	MAE (Ah)	0.0137	0.0122	0.0112	0.0109
	RMSE (Ah)	0.0170	0.0152	0.0135	0.0132
	NEE (%)	1.06	0.95	0.84	0.83
Cell2	MAE (Ah)	0.0127	0.0111	0.0114	0.0104
	RMSE (Ah)	0.0164	0.0145	0.0141	0.0130
	NEE (%)	1.03	0.91	0.88	0.81
Cell3	MAE(Ah)	0.0129	0.0130	0.0114	0.0112
	RMSE (Ah)	0.0175	0.0174	0.0139	0.0135
	NEE (%)	1.09	1.09	0.87	0.84
Cell4	MAE (Ah)	0.0170	0.0133	0.0113	0.0111
	RMSE (Ah)	0.0199	0.0167	0.0138	0.0137
	NEE (%)	1.24	1.04	0.86	0.86
Average	MAE (Ah)	0.0137	0.0120	0.0113	0.0109
	RMSE (Ah)	0.0177	0.0160	0.0138	0.0134
	NEE (%)	1.11	1.00	0.86	0.84

Table 4.5: The model size, total parameters and FLOPs of the CNN(T), CNN(S)-TL, PCNN(T) and PCNN(S)-TL models.

Model	Model size (KB)	Reduced size (%)	Parameter numbers	Parameters pruned (%)	Pruned FLOPs (%)
CNN(T)/CNN(S)-TL	199	-	12693	-	-
PCNN(T)	55	72.36	5711	55.01	95.70
PCNN(S)-TL	63	68.34	6793	46.48	80.97

2.33% smaller than that of the non-pruned CNN(S)-TL model. Secondly, due to the transfer learning technique, the estimation performance of CNN(S)-TL is better than CNN(T), and the NEE is decreased by 22.52% (from 1.11% to 0.86%). Thirdly, the PCNN(S)-TL model achieves the best result, of which the NEE is

24.32% smaller than the CNN(T) model directly trained on the target dataset. More statistics are listed in Table 4.5. Compared with the non-pruned network, the model size of the PCNN(T) and PCNN(S)-TL are dramatically reduced, being only 27.64% and 31.66% of the original model size, leading to the size reduction of 72.36% and 68.34% respectively. By removing the redundant neurons in the fully-connected layers, almost 50% of the parameters of the entire network are removed and more than 80% of the FLOPs in the last two fully-connected layers are reduced, which makes the network more compact and reduces the computational cost. Comparing the PCNN(T) with the PCNN(S)-TL model, it can be found that the model size and the number of parameters of the PCNN(S)-TL model are slightly higher than that of the PCNN(T) model. Since the CNN(S)-TL model was trained on a larger dataset, the specific features extracted in the fully-connected layers for the capacity estimation are expected to be much richer than that of the CNN(T) model. That is to say, the CNN(S)-TL model requires to retain more neurons to capture these features during the network pruning process. Consequently, the PCNN(S)-TL model will be slightly bigger than the PCNN(T) model.

In summary, the CNN model with transfer learning and network pruning techniques can achieve more accurate estimation results with much less neurons than a model trained on the target dataset from scratch.

4.6.3 Factors affecting the model performance

To investigate the influence of the number of layers fine-tuned during the transfer learning stage, the identical training and testing datasets were utilized for all tests. In each test, the entire structure and parameters of the CNN(S) model were copied first, and the last n layers were fine-tuned on the target sets while the first $6 - n$ layers were fixed. It should be noted that, the whole CNN has 9 layers in total, the two max pooling layers added after the first two convolu-

tional layers and the flatten layer after the last convolutional layer do not have parameters, therefore, maximally six layers can be fine-tuned. Thus, n refers to the number of tunable layers, and was selected from 1 to 6, where 1 implies that all layers except for the last FC layer are fixed, and 6 implies that the entire network needs to be fine-tuned. A complete fine-tuning and testing procedure was executed 100 times for each selected n, and the averaged max error (MaxE), MAE, RMSE, NEE of 100 runs are summarized in Table 4.6, where FC stands for the fully-connected layer, 2FC refers to the last two fully-connected layers, Conv represents the convolutional layer, and the number prefixing to Conv stands for the number of convolutional layers involved in fine-tuning from back to front.
As shown in Table 4.6, the estimation error decreases as the number of fine-tuned layers increases. When only fully-connected layers are fine-tuned, the NEEs are all greater than 1.3%, while if one or more convolutional layers are fine-tuned, the NEE are decreased by at least 30.88% (decreasing from1.36% to 0.94%). This improvement implies that for batteries with different specifications and subject to different charge/discharge policies, they have different high-level specific features projected at the last few convolution layers. The specific high-level features learned from the source dataset can not precisely describe the target dataset, thus they need to be learned using the target dataset.

Further, Figure 4.13 shows the number of neurons selected in FC1 and FC2 layer of the PCNN(S)-TL model during the network pruning process for different number of fine-tuned layers n. Two observations can be concluded from this bar chart. Firstly, the number of neurons selected in FC2 layer almost does not change with n. Secondly, for the FC1 layer, as the number of fine-tuned layers increases, the number of selected neurons first increases until it reaches the maximum value at $n = 3$, and then gradually converges. Only relatively fewer neurons are selected in FC1 layer when only fine-tune the last one or two layers ($n = 1, 2$), while the maximum number of neurons is selected when the last convolutional

Table 4.6: Comparison of the model performance on battery capacity estimation with different number of fine-tuned layers.

Assess	Fine-tuned layers (from back to front)	FC n=1	2FC n=2	2FC + Conv n=3	2FC + 2Conv n=4	2FC + 3Conv n=5	2FC + 4Conv n=6
MaxE (Ah)	fine-tune	0.0644	0.0475	0.0377	0.0292	0.0277	0.0275
	fine-tune & pruning	0.0578	0.0478	0.0395	0.0294	0.0273	0.0265
MAE (Ah)	fine-tune	0.0271	0.0177	0.0120	0.0112	0.0112	0.0113
	fine-tune & pruning	0.0262	0.0182	0.0121	0.0109	0.0113	0.0109
RMSE (Ah)	fine-tune	0.0321	0.0218	0.0151	0.0135	0.0132	0.0135
	fine-tune & pruning	0.0307	0.0223	0.0154	0.0132	0.0132	0.0129
NEE (%)	fine-tune	2.01	1.36	0.94	0.84	0.83	0.84
	fine-tune & pruning	1.92	1.39	0.96	0.83	0.83	0.81

layer is fine-tuned together with the fully-connected layers ($n = 3$). Then the number of selected neurons decreases as the number of fine-tuned convolutional layers increases. This again implies that the last few convolutional layers can extract more specific features. When $n = 1, 2$, the number of useful specific features for interpreting the target data is insufficient, as a consequence, only a limited neurons in FC1 are needed which can contribute to the final specific features captured in FC2, and the estimation performance can not be maintained (Table 4.6). When n increases from 1 to 3, the increase of the number of selected neurons and the decrease of the estimation error reveals that as the number of fine-tuned layers increases, the number of useful extracted features increases and the number of redundant neurons decreases. However, when n is further increased from 3 to 6, the features extracted from the data have already been fully exploited through the fine-tuning of the convolutional layers proceeding to FC1 and FC2, therefore the number of selected neurons in FC1 and FC2 can be reduced. This is observable from Fig. 6 that as n increases from 3 to 6, the number of selected neurons in FC1 and FC2 is slightly decreased. The aforementioned analysis reveals that there is a trade-off in the transfer learning between the number of layers required to be fine-tuned, the model complexity, and associated computational effort.

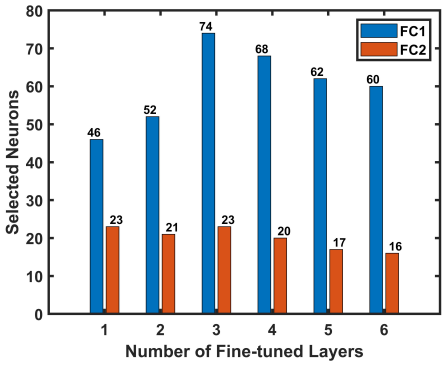


Figure 4.13: The number of neurons selected in the last two fully-connected layers of the PCNN(S)-TL model versus the number of fine-tuned layers

In summary, according to Table 4.6 and Figure 4.13, fine-tuning 4 layers (2 fully-connected layers and 2 convolutional layers) are the best trade-off among estimation performance, model complexity, fine-tuning effort and computational cost.

4.6.4 Discussions

By incorporating the concepts of transfer learning and network pruning, the final model updated offline using the small target dataset can produce more accurate online capacity estimation results with a more compact structure and lower computation cost. With real-time recorded signals (i.e. current, voltage and charge capacity) of a partial charging curve, the proposed model can quickly estimate the capacity online. These distinctive features open up a great potential for implementing the model in embedded systems for industrial applications. Note that the datasets used in this study only include one testing temperature, while in practice, the battery operating temperature varies. This can be tackled by adding the temperature as an additional input variable to the model. This study has also demonstrated that our proposed method is sufficiently robust to varying operating temperatures and different charging policies, as the batteries used to generate the source dataset and the target dataset were tested at different

temperatures and charging/discharging policies. Further, we have demonstrated the effectiveness of our proposed method on battery capacity estimation on the similar type of batteries, though their specifications and testing profiles are different. However, whether the transfer learning can be applicable to different battery types is still an open question. Moreover, the input data used in this experiment has a high time resolution, and it can yield a model with good estimation results, however the redundant information may lead to a high computational cost. While data with a low time resolution may contain insufficient information or even lose important features, the resultant model may have poor estimation performance. Therefore, it is of great significance to investigate the impact of the time resolution of input data on the estimation accuracy, which will provide guidelines for selecting a proper time resolution.

Finally, in this work, we first build the CNN model offline using both source data and target data, and then use the developed model for online capacity estimation. A potential limitation for this method is that the parameters may drift as the battery cycling life increases, and the estimation error may increase and eventually exceed tolerable limit. This problem however can be overcome by updating the model parameters regularly using cycling test data which could be collected when the battery systems are under maintenance, or in other occasions where managed charging and discharging tests can be conducted. Given the operation safety is a paramount requirement for battery powered systems, regular maintenance is a necessity, and we strongly recommend including cycling test as one of the key battery maintenance procedures.

4.7 Chapter Summary

In this chapter, a CNN-based battery capacity estimation framework is proposed to achieve fast online capacity estimation only using partial charging segment

with flexible starting point and a fixed length of 225 consecutive points, and the feature extraction and capacity estimation are automatically executed in one framework. The input generation method is introduced at the beginning of the framework, which can increase the number of the samples used for CNN training and it is an enabling block to apply the CNNs for capacity estimation using partial charging segment. Then the designed CNN architecture that incorporate the transfer learning and network pruning techniques is described in detail. In the experiment, a large battery degradation dataset is used as the source dataset to build the CNN model which is then transferred to a small degradation dataset of target batteries with different specifications tested under different cycling profiles. Then the resultant model is obtained by pruning the neurons of the transferred model using the proposed contribution-based method. By comparing the CNN model directly trained on the target dataset and separately pruned/transferred models with the resultant model, it can be revealed that both the model size and computational cost of the proposed model are significantly reduced, that is the resultant CNN model is more compact and can achieve higher estimation accuracy on small degradation datasets with lower computational cost.

Chapter 5

Battery SOC and capacity Co-estimation

This chapter is an extension of the research work presented in Chapter 3 and Chapter 4 and a battery SOC and capacity co-estimation framework is proposed based on the fusion of a Convolutional Neural Network model and a Gaussian Process Regression model. As aforementioned, battery SOC and SOH are important indicators that can quantitatively evaluate the remaining capacity and degradation degree of the battery, respectively. Accurate SOC and SOH estimation is critical to ensure safe and reliable operation of battery systems. In Chapter 3, the SOC is estimated using the nominal capacity as the maximum available capacity, however, the capacity fades as the battery degrades, inaccurate maximum available capacity information will influence the accuracy of the SOC estimation. Therefore, battery aging should be taken into account and the capacity should be updated regularly to improve the performance of the SOC estimation. Considering the intrinsic coupling relationship between SOC and SOH, a joint estimation method of SOC and SOH is more favorable, as it can synergistically optimize their respective estimation results. In this chapter, the CNN-based battery capacity estimation method presented in Chapter 4 along with the Gaus-

sian Process Regression method are leveraged to achieve more accurate capacity and SOC estimation.

Moreover, a new set of signals measured by the Fiber Bragg gratings (FBG) sensors is introduced in this chapter and utilized for SOC estimation. The newly measured signals contain both the strain and temperature information, since the measured Bragg wavelength variations are subject to both strain and temperature changes. The lattice expands and contracts during the lithium intercalation/deintercalation process, leading to the changes in strains and volume. The stability and safety of the battery can be affected by these induced strains, which are among the main reasons leading to potential material failure and other forms of performance degradation if the strains exceed certain level of thresholds [194]. Besides, temperature is often monitored to provide early warning of potential thermal hazards, which generally manifest as the thermal runaway and may cause irreversible damage to battery cells. Therefore, strain and temperature are crucial parameters to monitor during the electrochemical processes of battery charging/discharging to ensure safe operation of batteries. The FBG sensors introduced in this chapter have been proven to be a suitable solution to perform strain and temperature measurements for batteries.

5.1 Experimental Setup

Cells used in this section are those used in the experiments presented in Section 4.4.2, where 4 commercial cylindrical LFP cells with a nominal voltage of 3.2 V and a nominal capacity of 1.6 Ah are used in the experiment. All cells are tested under a constant temperature of 25 °C, and the battery current, voltage and surface temperature are recorded during the CC-CV charging and CC discharging process. Besides, fiber-optic sensors, which are a promising new sensing option for battery cell monitoring, are used in this experiment to provide more

informative measurements to improve the SOC estimation accuracy. As shown in Figure 5.1, three Fiber Bragg gratings (FBG)-based fiber-optic sensors are directly attached to the surface of each cell without affecting its packaging and integrity. This non-invasive installation approach does not cause potential safety issues and the FBG sensors can be easily mounted on the battery cells.

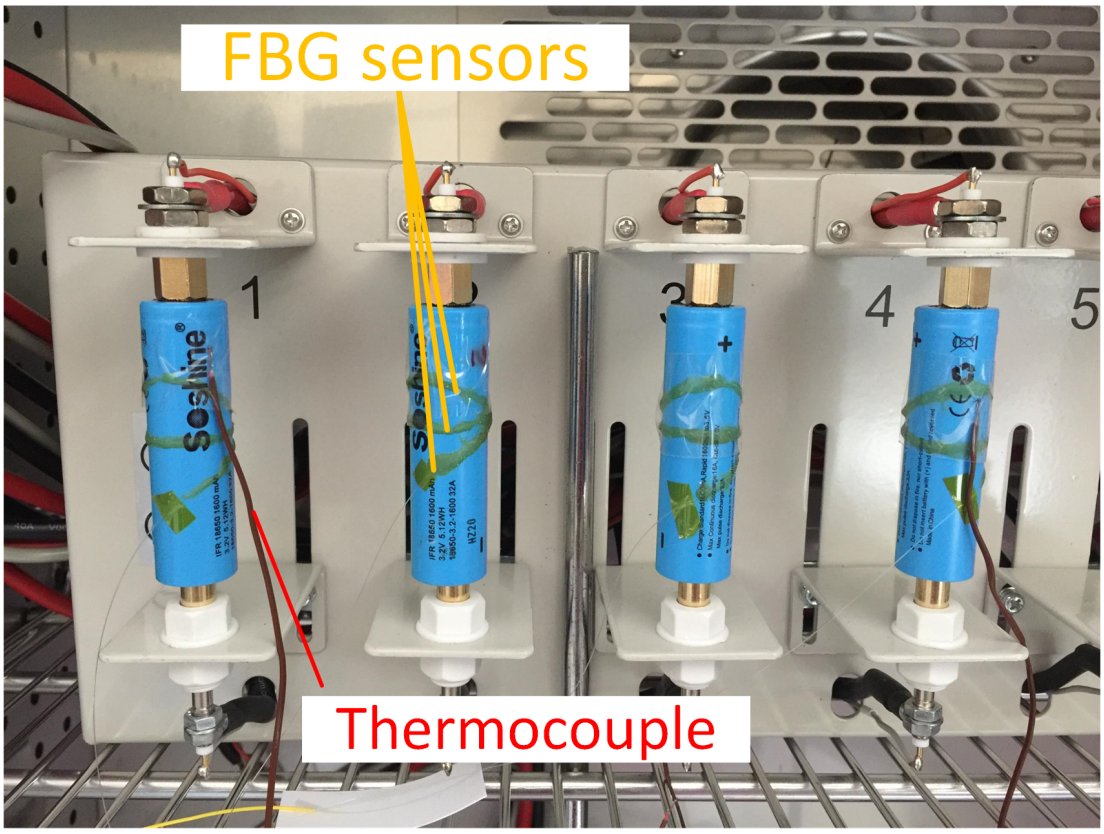


Figure 5.1: Cells with FBG sensor integrated.

The FBG sensors are sensitive to strain and temperature variations. These two significant parameters are directly related to the complex processes inside the cells, and the temperature increase and mechanical stress will cause capacity loss and potential risk of batteries. When the battery surface temperature or strain changes, the reflected wavelength changes from the base wavelength λ to λ_s, thus the wavelength shift $\Delta\lambda$ is related to both strain and temperature variations

[33; 194]. As the three FBG sensors are co-located within a small footprint but have a slightly different orientation, the radial strain signal can be decoupled from the temperature measurement [40]. In this section, the wavelengths that contain both strain and temperature information are directly used for SOC estimation. The average wavelength shift of the three FBG sensors and the voltage of a cell subject to the CC-CV charging and CC discharging mode is presented in Figure 5.2. In Figure 5.2, step (1) corresponds to the charging phase with a constant current of 1 A, and step (2) is the 3.6 V constant voltage charging phase, while the final step (3) corresponds to the discharging phase with a constant current of 1 A.

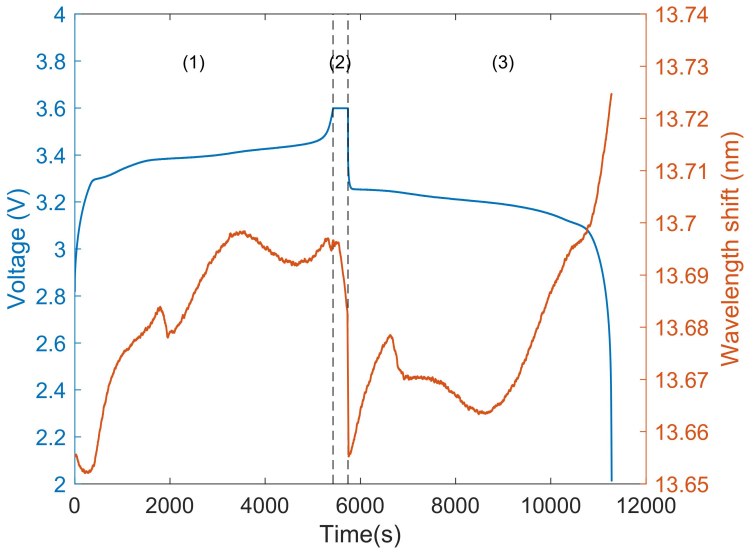


Figure 5.2: Wavelength shift of a CC-CV charging and CC discharging cycle

Since simple functions cannot describe the relationship between SOC and wavelength shift appropriately, therefore, the Gaussian Process Regression algorithm is used to estimate the SOC using this new set of signals.

5.2 Methodology

In this section, the detailed battery SOC and capacity co-estimation framework is presented. Firstly, the proposed framework of the co-estimation method is introduced. Secondly, a brief overview of the Gaussian Process Regression (GPR) theory is introduced and the implementation procedure of GPR-based battery SOC estimation method is presented. Further, the CNN-based battery capacity estimation method, which is described in detail in Chapter 4, is briefly introduced.

5.2.1 The Co-estimation Framework

Figure 5.3 shows the flowchart of the proposed battery capacity and SOC co-estimation framework. The main steps of this framework are introduced as follows: at each time instant, firstly, battery current, voltage, and FBG signals are sampled in real-time, then the current, voltage and the charge capacity are normalized and transformed to a three-dimensional (3-D) image with the size of $15 \times 15 \times 3$ and inputted to the PCNN-TL model trained in Chapter 4 for online estimation of the battery capacity. Subsequently, the estimated capacity is used in the GPR model to correct the imprecise capacity estimation value, and the updated capacity along with other normalized measurements are used to estimate the SOC online. Finally, the charge capacity calculated by integrating the current with respect to time is replaced by the charge capacity calculated using the estimated SOC, and the new 3-D input generated with the updated charge capacity is fed into the PCNN-TL model to estimate the capacity.

The co-estimation framework is more practical and provide more accurate estimates than the traditional state estimation methods since it utilizes the coupling relationships between the capacity and SOC. By online updating the capacity value in SOC estimation, the impact of battery degradation is taken into account. Therefore the performance of SOC estimation for aged batteries can be

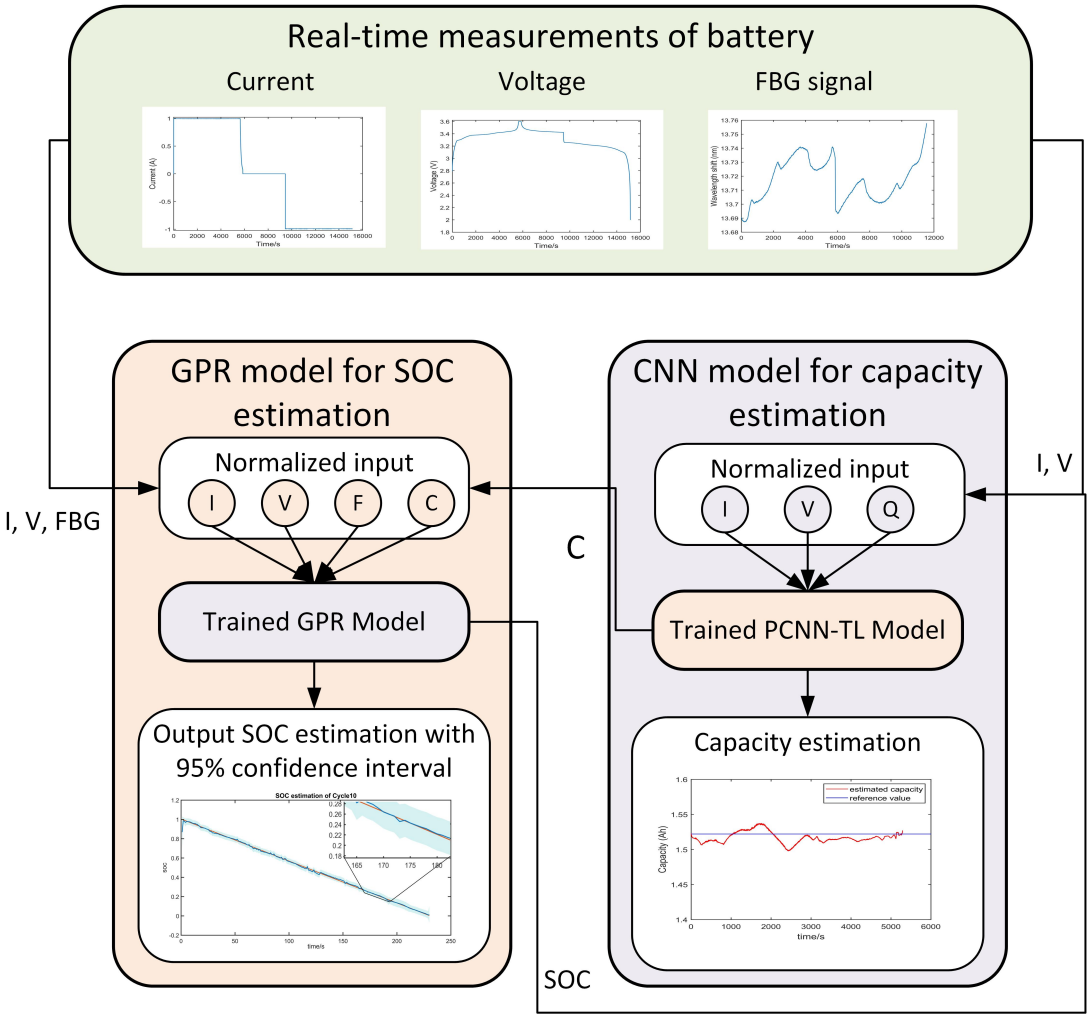


Figure 5.3: The flowchart of the proposed battery capacity and SOC co-estimation framework

improved.

5.2.2 SOC estimation

5.2.2.1 Gaussian Process Regression

Different from Chapter 3, GPR, which is a probabilistic and non-parametric machine learning method, is used for battery SOC estimation in this chapter. The GPR method is capable of quantifying the uncertainty of the estimation rather than just provide a point estimate of the SOC, and hence providing more informative outputs than the KF algorithm. In essence, based on the GPR method, the estimation result of SOC is given in the form of probability distribution, which consists of the mean of the estimation value and confidence intervals.

Let $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{N}$ denote a labelled training dataset with N samples, where $\mathbf{x}_i \in \Re^D$ is a D - dimensional input vector, and $\mathbf{y}_i \in \Re$ is the corresponding output. Suppose that there exists a latent function $f(.)$, to map inputs $\mathbf{x}_i$ to outputs $\mathbf{y}_i$:

$$\mathbf{y}_i = f(\mathbf{x}_i) + \varepsilon_i \tag{5.1}$$

where $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$ is an independent and identically distributed noise contribution.

In the GPR, the function $f(\mathbf{x})$ is assumed to follow a multivariate Gaussian distribution, and can be described as:

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), K(\mathbf{x}, \mathbf{x})) \tag{5.2}$$

where $\mathcal{GP}$ denotes a Gaussian process. The mean function $m(\mathbf{x})$ and covariance function $K(\mathbf{x}, \mathbf{x})$, which can fully describe the function $f(\mathbf{x})$, are denoted by:

$$m(\mathbf{x}) = E(f(\mathbf{x})) \tag{5.3}$$

$$
\begin{aligned}
K(\mathbf{x}, \mathbf{x}) &= E[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}'))] \\
&= \begin{bmatrix}
\kappa(\mathbf{x}_1, \mathbf{x}_1) & \kappa(\mathbf{x}_1, \mathbf{x}_2) & ... & \kappa(\mathbf{x}_1, \mathbf{x}_N) \\
\kappa(\mathbf{x}_2, \mathbf{x}_1) & \kappa(\mathbf{x}_2, \mathbf{x}_2) & ... & \kappa(\mathbf{x}_2, \mathbf{x}_N) \\
... & ... & ... & ... \\
\kappa(\mathbf{x}_N, \mathbf{x}_1) & \kappa(\mathbf{x}_N, \mathbf{x}_2) & ... & \kappa(\mathbf{x}_N, \mathbf{x}_N)
\end{bmatrix}
\end{aligned} \tag{5.4}
$$

The mean function reflects the expected function value at input $\mathbf{x}$, and the prior mean function is often set to zero in order to avoid expensive posterior computations and hence only the covariance function is inferred [195]. The covariance function $K(\mathbf{x}, \mathbf{x})$, also called the kernel of the Gaussian process, reflects the dependence between the function values at different input points $\mathbf{x}_i$ and $\mathbf{x}_j$. All the assumptions on the properties of the function to be modelled, such as smoothness and periodicity, are reflected in the covariance function. The squared exponential (SE) kernel is commonly used and has become the de-facto default kernel for Gaussian processes, it is defined as:

$$
\kappa(\mathbf{x}_i, \mathbf{x}_i) == \sigma_f^2 exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_i\|^2}{2\lambda^2}\right) \tag{5.5}
$$

where σ_f^2 denotes the signal variance that quantifies the variation of the latent function from its mean, and λ is the characteristic length scale that determines the relative importance of the input variables in estimating the target output.
Based on Equation (5.1) and (5.2), the joint distribution of the training output $\mathbf{y}$ can be expressed as:

$$
\mathbf{y} \sim \mathcal{N}(0, K(\mathbf{x}, \mathbf{x}) + \sigma^2 \mathbf{I}) \tag{5.6}
$$

where $\mathbf{I}$ is a $N \times N$ unit matrix. Generally, the unknown hyperparameters $\Theta = (\sigma^2, \sigma_f^2, \lambda)$ of the covariance function need to be optimized in the training process by maximizing the log of the marginal likelihood function of output $\mathbf{y}$. The log marginal likelihood is given by:

$$\log p(\mathbf{y}|\mathbf{x}, \Theta) = -\frac{1}{2}\mathbf{y}^T[K(\mathbf{x}, \mathbf{x}) + \sigma^2\mathbf{I}]^{-1}\mathbf{y} - \frac{1}{2}\log\left|K(\mathbf{x}, \mathbf{x}) + \sigma^2\mathbf{I}\right| - \frac{N}{2}\log 2\pi \tag{5.7}$$

After obtaining the optimal hyperparameters using gradient-based method, and giving a testing dataset $\mathcal{D}_* = \{(\mathbf{x}_{*i}, \mathbf{y}_{*i})\}_{i=1}^{N_*}$, the joint multivariate Gaussian distribution of the training output $\mathbf{y}$ and the testing output $\mathbf{y}_*$ can be written as:

$$\begin{bmatrix} \mathbf{y} \\ \\ \mathbf{y}_* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} K(\mathbf{x}, \mathbf{x}) + \sigma^2\mathbf{I} & K(\mathbf{x}, \mathbf{x}_*) \\ \\ K(\mathbf{x}_*, \mathbf{x}) & K(\mathbf{x}_*, \mathbf{x}_*) \end{bmatrix}\right) \tag{5.8}$$

where $K(\mathbf{x}, \mathbf{x}_*)$ is the covariance matrix between the testing inputs and the training inputs and $K(\mathbf{x}, \mathbf{x}_*)^T = K(\mathbf{x}_*, \mathbf{x})$, and $K(\mathbf{x}_*, \mathbf{x}_*)$ is the covariance matrix of testing inputs $\mathbf{x}_*$. Then the predictive posterior distribution is derived for the estimation on the new/testing inputs $\mathbf{x}_*$, which can be completely specified by the mean and covariance:

$$\mathbf{y}_*|\mathbf{x}_*, \mathbf{x}, \mathbf{y} \sim \mathcal{N}(\bar{\mathbf{y}}_*, K_*) \tag{5.9}$$

where the mean $\bar{\mathbf{y}}_*$ of the predictive distribution, which gives the point estimate of the testing output, is given by:

$$\bar{\mathbf{y}}_* = K(\mathbf{x}_*, \mathbf{x})\left[K(\mathbf{x}, \mathbf{x}) + \sigma^2\mathbf{I}\right]^{-1}\mathbf{y} \tag{5.10}$$

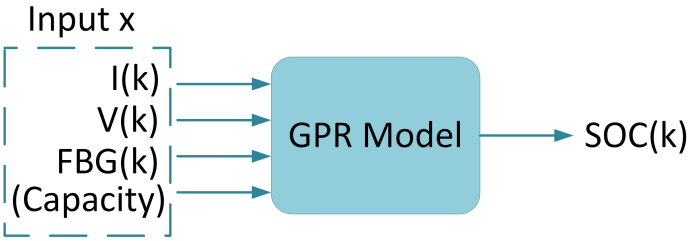


Figure 5.4: GPR-based battery SOC estimation

and the covariance matrix K_* provides a measure of the uncertainty in the estimate of the test output [196], and it is given by:

$$K_* = K(\mathbf{x}_*, \mathbf{x}_*) - K(\mathbf{x}_*, \mathbf{x}) \left[K(\mathbf{x}, \mathbf{x}) + \sigma^2 \mathbf{I}\right]^{-1} K(\mathbf{x}, \mathbf{x}_*) \tag{5.11}$$

5.2.2.2 GPR-based SOC estimation

In this chapter, the GPR is used to estimate the battery SOC for given measurement inputs. As shown in Figure 5.4, the input variables to the GPR model are current $I(k)$, voltage $V(k)$ and wavelength data $FBG(k)$ at time k, and the capacity information of the corresponding cycle is updated by the estimation output of the PCNN-TL model (as trained in Chapter 4) to improve the SOC estimation results. Here the wavelength data is obtained from the fiber optic sensors, and two important parameters (i.e. battery surface strain and temperature) can be decoded and extracted from the wavelength for the characterization of the lithiation/delithiation process. The model output is the estimated SOC at time k, denoted by $SOC(k)$.

The GPR-based SOC estimation method mainly consists of two parts, offline training of the model and online estimation of SOC using the trained model. The steps for training a GPR model and then performing SOC estimation are illustrated in Figure 5.5, where the blue part on the left represents the offline training process, and the orange part on the right refers to the testing process.

The detailed steps can be described as follows:

Training process:

Step 1 - Determine and normalize the training dataset, $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{N}$, where $\mathbf{x}$ contains current, voltage and wavelength measurements as well as the estimated capacity of the corresponding cycle, and $\mathbf{y}$ is the reference value of SOC.

Step 2 - Select a kernel function that can well represent the underlying target function.

Step 3 - Set the initial values for the hyperparameters in the specified kernel function as well as the noise variance.

Step 4 - Optimize the hyperparameters with training data by maximizing Equation (5.7), the log marginal likelihood function.

Testing process:

Step 5 - With the optimal hyperparameters, the GPR model is referred to as the 'trained' model. Then in the testing process, the normalized testing inputs are fed into the trained GPR model, and the target SOC is outputted in form of mean and covariance, which provides both the SOC estimation results and the uncertainty measurements.

5.2.3 CNN-based capacity estimation

As mentioned in Chapter 4, the battery aging during its utilization will lead to capacity degradation, which will impact the accuracy of SOC estimation. Considering the importance of timely maintenance and replacement of aged batteries, and the requirement for improved SOC estimation performance, it is meaningful to update the actual capacity in real-time.

In this section, the same CNN model described in Chapter 4 is used to perform the battery capacity estimation. The estimated capacity is then used as an input to the GPR model introduced in Section 5.2.2.2, along with battery current, voltage and wavelength data to update the SOC estimation.

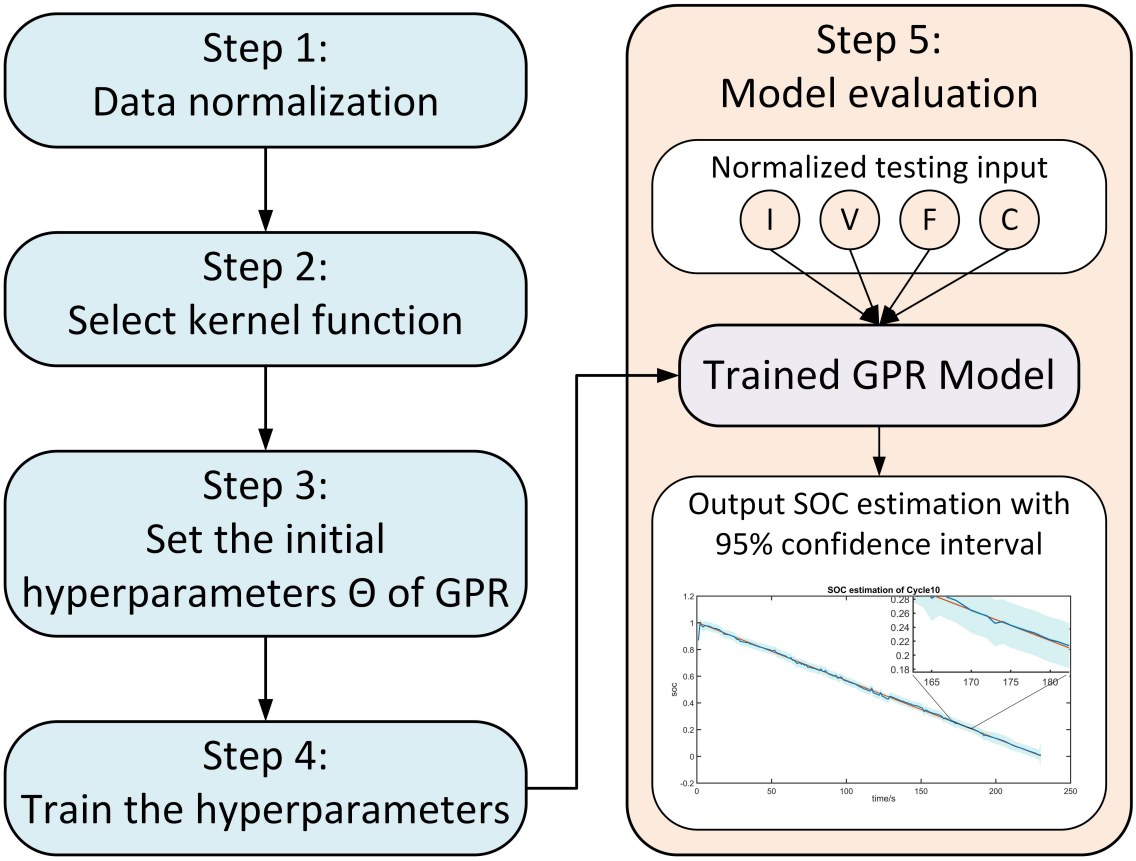


Figure 5.5: Flowchart of the GPR-based battery SOC estimation

5.3 Estimation Results and Discussions

The performance of the proposed SOC and capacity co-estimation framework is verified on the aforementioned dataset. The RMSE is used to evaluate the estimation accuracy and the standard deviation is used to characterize the estimation uncertainty.

The SOC estimation results of cell 1 over discharge profiles of reference cycle 10 are shown in Figure 5.6. It is evident that the estimated SOC with updated capacity information are much closer to the reference SOC. As illustrated in Figure 5.6(c), though the errors shown in both figures converge to zero, the error of SOC estimation with updated capacity is within 2%, while the error of SOC estimation without updating the capacity is within 6%. Further, the RMSE of the SOC estimation with and without the updated capacity value are 0.62% and 3.59%, respectively. This has clearly shown that accurate capacity estimation is important for SOC estimation. While estimating the SOC, battery capacity is simultaneously estimated using the well-trained CNN model, and the estimation results are shown in Figure 5.7. The blue and red solid lines represent reference and estimated capacity value of cycle 10, respectively. As shown in Figure 5.7, the estimated capacity is close to the reference value, while its fluctuation is similar to that of the SOC estimation error (red line in Figure 5.6(c)), when the error of SOC estimation converges to zero, the estimated capacity also converges to the reference value.

To investigate the effect of using FBG wavelength signals as model input for the SOC estimation, the data collected from cell 1 was utilized for testing. Two different GPR models were built for SOC estimation, one took current, voltage and capacity as the model input, and the other used current, voltage, FBG wavelength signal, and capacity as the model input. The estimation results on cycle 10 of cell 1 are shown in Figure 5.8, and from the enlarged view of Figure 5.8(a) and 5.8(c),

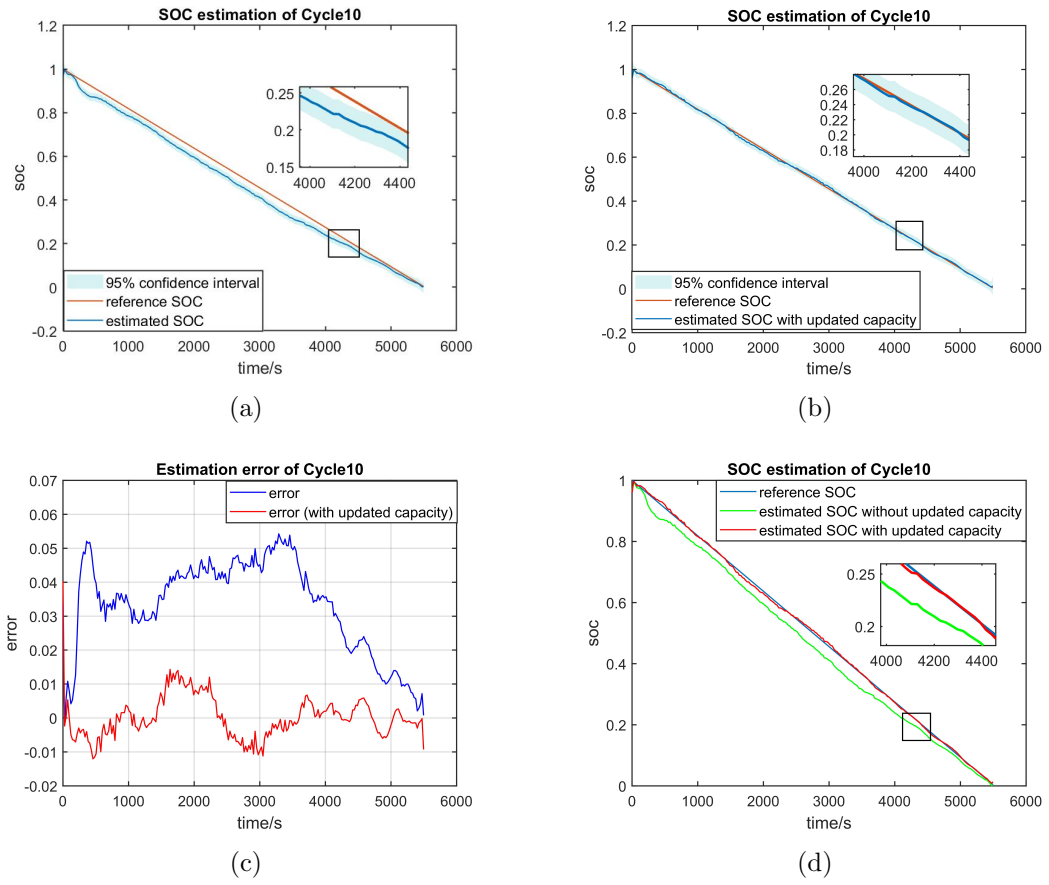


Figure 5.6: SOC estimation results of cell 1: (a) SOC estimation results without updated capacity information. (b) SOC estimation results with updated capacity information. (c) SOC estimation error. (d) SOC estimation results with/without updated capacity.

it is evident that the the shaded blue area is wider when the FBG signal is not fully utilized, which means the estimation uncertainty is higher. Further, Figures 5.8(b) and 5.8(d) confirm that the SOC estimation is more accurate when the FBG signal is utilized as an input to the estimation model. Quantitatively, when using the updated capacity information, the RMSE of the SOC estimation with and without using the FBG signals as input are 0.62% and 1.48%, respectively, and the mean standard deviation of the estimation with and without FBG signals are 1.02% and 1.78%, respectively.

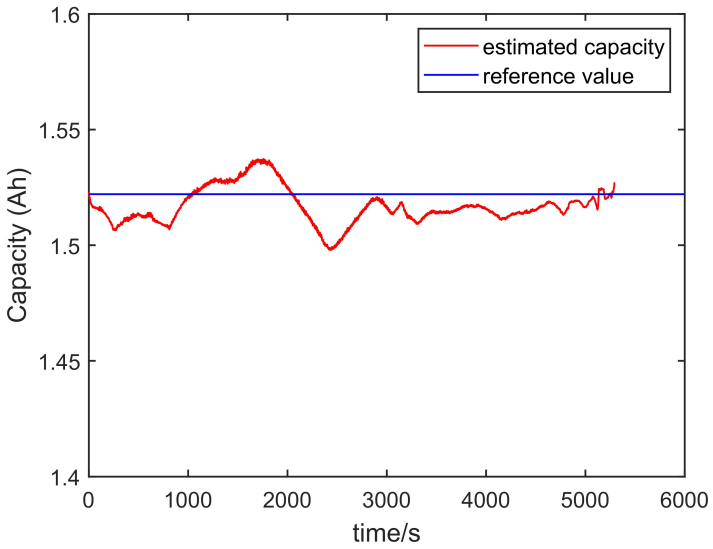


Figure 5.7: The estimation results of battery capacity for reference cycle 10 of cell 1

The RMSE and standard deviation of SOC estimation under different input conditions are summarized in Table 5.1. Two observations can be concluded from Table 5.1. Firstly, the use of updated capacity can greatly improve the SOC estimation accuracy (reduces the RMSE from 4.3% to 1.48%, or from 3.59% to 0.62%), but do not affect the estimation uncertainty. Secondly, using FBG signals as input to estimate the SOC can not only reduce the estimation RMSE (the RMSE decrease from 4.3% to 3.59% and from 1.48% to 0.62% achieved 16.51% and 58.11% reductions, respectively), but also reduce the estimation uncertainty (achieving 42.7% reduction on the estimation standard deviation from 1.78% to 1.02%).

The SOC estimation results of the GPR-based method are then compared with those of model-based methods proposed in Chapter 3. As summarized in Chapter 3, among EKF, UKF and PF algorithms, UKF is the fastest yet gives medium SOC estimation performance in this case. Considering that the GPR method can also quickly produce estimation results, the UKF-based method is used for

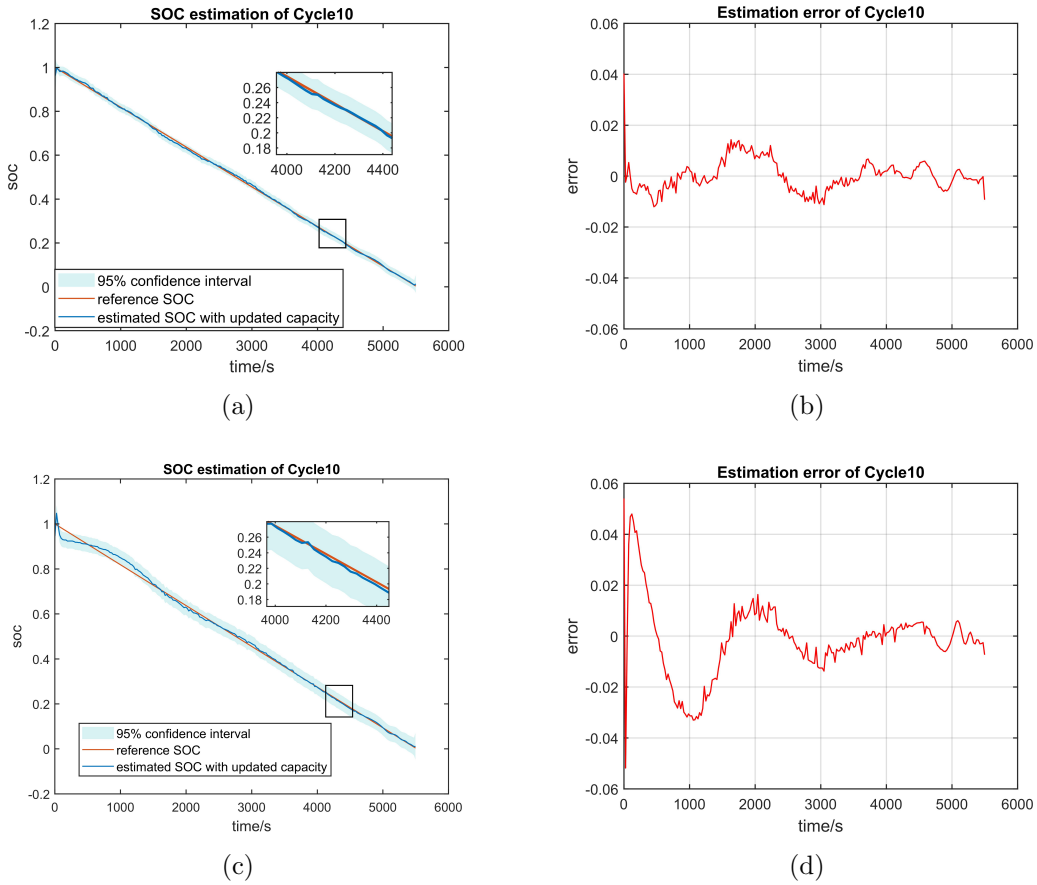


Figure 5.8: SOC estimation results of cell 1: (a) SOC estimation results using FBG signals. (b) Error of SOC estimation using FBG signals. (c) SOC estimation results without FBG signals. (d) Error os SOC estimation without FBG signals

Table 5.1: SOC estimation results with/without updated capacity and with/without FBG measurements.

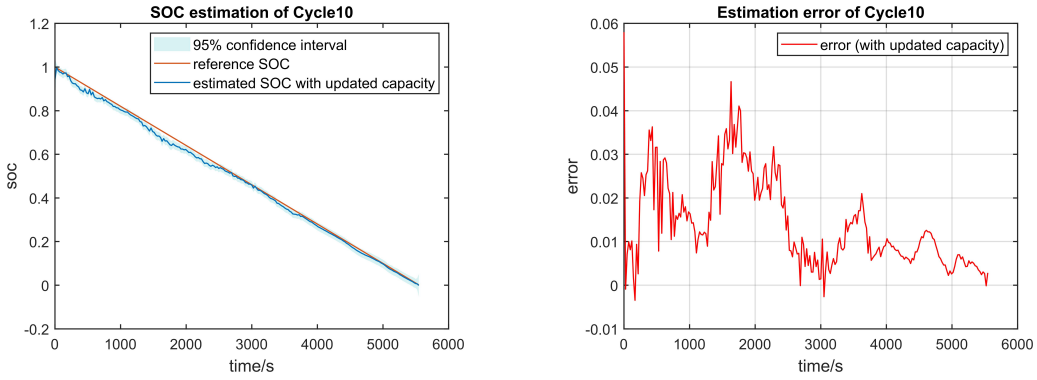
Assess	Without FBG signals		With FBG signals	
	Without updated Q	With updated Q	Without updated Q	With updated Q
RMSE	4.3%	1.48%	3.59%	**0.62%**
Mean Standard Deviation	1.77%	1.78%	1.02%	**1.02%**

comparison. In this test, when considering the FBG measurements, the correlations among the terminal voltage and current, SOC, maximum available capacity,

and FBG measurements is identified first by the FRA method. Then the SOC is estimated by the UKF method, and the RMSE of the SOC estimation with and without the updated capacity value are 3.29% and 4.25%, respectively. When the FBG measurements are not used, the terminal voltage is correlated only with the current, SOC, and maximum available capacity. And based on this model, the RMSE of the UKF-based SOC estimation with and without the updated capacity value are 4.62% and 5%, respectively. The results again validate that with the updated capacity, the SOC can be estimated more accurately, and the use of FBG measurements can further improve the estimation accuracy. However, these results are not as accurate as the GPR method. One reason might be that the non-linear forms of the inputs (current, SOC, capacity, FBG signals) in the manually constructed feature pool are not sufficiently rich enough to capture the effects of aging and fully utilize the useful information embedded in the FBG measurements. Therefore, our co-estimation framework uses GPR method rather than model-based method to estimate the SOC.

To validate the generalization ability of the proposed method, the model trained on cell 1 is directly applied to estimate the SOC of cell 2, and satisfactory SOC estimation results are also achieved. Taking reference cycle 10 of cell 2 as an example, as shown in Figure 5.9, the estimation results can still well track the reference values, though the max error is around 5%, it converges to zero at the end of the cycle. Similar trend can be observed from the capacity estimation results shown in Figure 5.10, when the SOC estimation error is large, the capacity estimation error is large, while the estimated SOC converged to the reference SOC, the estimated capacity also converged to the reference value. Further, the RMSEs for the discharge profiles of the first ten reference cycles of cell 2 are summarized in Figure 5.11. As shown in Figure 5.11, the RMSE of SOC estimation is always lower when the capacity is estimated and updated in SOC estimation, which is less than than 2% for these ten reference cycles, while the RMSE of the

model prediction without using the updated capacity is within 5%, the results again confirm that the proposed co-estimation framework can estimate the SOC more accurately by updating the capacity value.



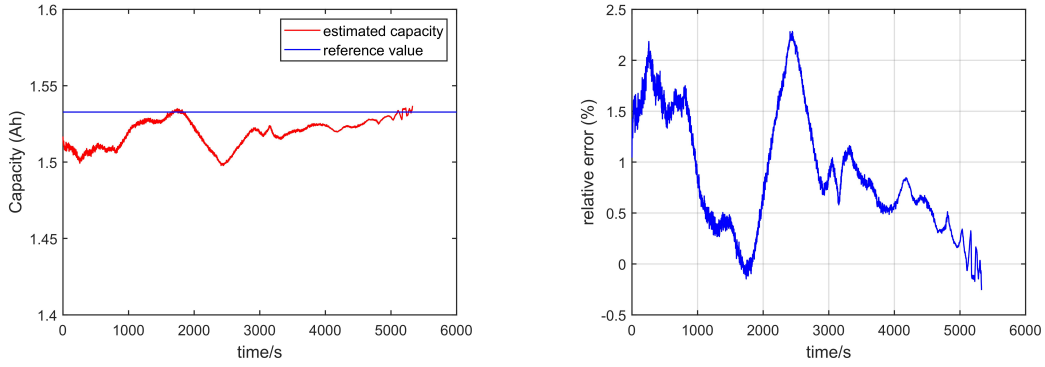
(a) (b)

Figure 5.9: SOC estimation results over discharge profile of reference cycle 10 of cell 2 using model trained on cell 1: (a) SOC estimation results with updated capacity information. (b) SOC estimation error.

In summary, the proposed co-estimation framework has shown to significantly improve the SOC estimation accuracy by updating the imprecise battery capacity in time, and accurate SOC estimation can in turn improve the accuracy of capacity estimation, while traditional SOC estimation methods without capacity calibration cannot eliminate the influence of the erroneous capacity value. Furthermore, the FBG measurements can provide more information on the battery dynamics, therefore, using FBG measurements to assist SOC estimation, the estimation uncertainty can be decreased and estimation accuracy can be improved.

5.4 Chapter Summary

The battery capacity and SOC are two important parameters for BMS to ensure safe and reliable operations. To improve the estimation performance of these two parameters, this chapter has proposed a CNN-GPR based capacity and SOC co-



(a) (b)

Figure 5.10: Capacity estimation results for reference cycle 10 of cell 2 (a)Capacity estimation results using online estimated SOC. (b) The relative error between estimated and reference capacity.

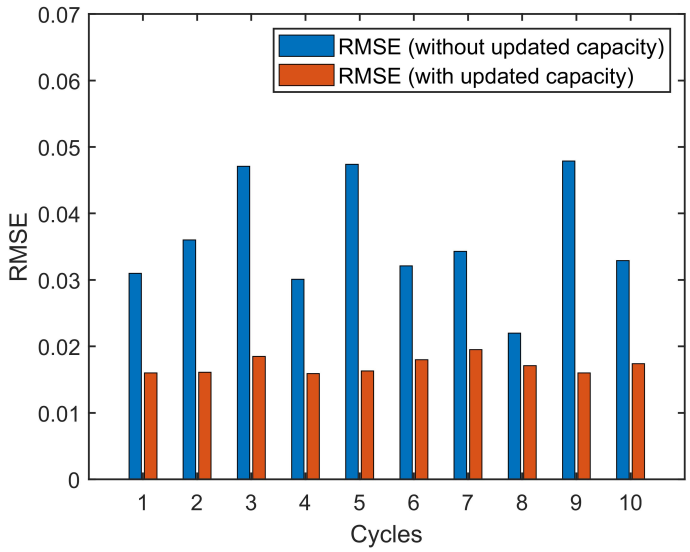


Figure 5.11: Bar chart of the SOC estimation RMSE for the discharge profiles of ten reference cycles. The performance of the model trained on cell 1 is evaluated on the data recorded from cell 2.

estimation framework, which uses CNN to achieve fast capacity estimation online and uses GPR to estimate SOC and quantify estimation uncertainty simultaneously. The capacity is first estimated using CNN and then the estimated capacity is used in the GPR to update the imprecise capacity and to estimate SOC online. Meanwhile, the estimated SOC is in turn used to update the capacity value. With the updated capacity, the SOC can be estimated more accurately, and in turn the accurate SOC estimation can further improve the accuracy of the estimated capacity. Further, new sensing measurements are utilized in SOC estimation, which is collected by FBG sensors, and the results showed that this measurement can improve the SOC estimation result and lower the estimation uncertainty, with up to 58.11% reduction on the RMSE and 42.7% reduction on the estimation standard deviation. The test results undeniably verify the performance of the proposed co-estimation framework and that the use of FBG signal is beneficial to SOC estimation. These results confirm that the capacity estimation is vital for accurate SOC estimation, and the newly involved FBG signal can further improve the SOC estimation accuracy.

Chapter 6

Conclusion and future work

6.1 Summary of work and main contributions

Li-ion batteries have been increasingly used as important energy storage devices in both grid and automotive applications on the global journey to achieve net zero by the mid of this century. Battery state estimation is one of the most important functions of BMSs to ensure safe, reliable and efficient utilization of batteries as it provides critical information for battery control, protection, optimization and maintenance. To tackle the challenges in current battery internal state estimation approaches, this thesis focuses on the development of novel frameworks and associated techniques for battery state estimation to achieve accurate and fast online estimation. The main work and contributions of this thesis are summarized as follows:

- In Chapter 1, the research background and motivations for the work carried out in this thesis are introduced. To begin with, various battery energy storage systems are reviewed, and a comparative survey of different types of batteries is presented. Then, the main functions of battery management systems are summarized, where the emphasis is given to battery internal state estimation and the potential of utilizing fiber optic sensors to acquire additional information to improve the estimation results is dis-

cussed. Thereafter, the development of state estimation techniques is reviewed. Considering the importance and difficulties of accurate battery state estimation, the motivations of this study are to develop accurate and reliable frameworks and techniques for battery state of charge (SOC) and state of health (SOH) estimation.

- To provide technical background used in each subsequent chapters, a systematic review of battery models and estimation techniques of SOC and SOH is presented in Chapter 2. The commonly used battery models including electrochemical models, equivalent circuit models, and black-box models are introduced, and various model-based and model-free state estimation techniques are comprehensively reviewed. It is worthwhile to systemically review these models and techniques to guide the selection of appropriate techniques according to practical requirements.

- For online battery SOC estimation, the model-based methods offer several advantages, such as closed-loop correction, insensitive to initial SOC error, capable of coping with uncertainties and disturbances, but require accurate battery models. To this end, a novel black-box model is established in Chapter 3 to capture the battery dynamics with satisfactory accuracy and computational efficiency, the model structure selection and parameters identification are achieved simultaneously by the fast recursive algorithm. Then the established model is combined with the coulomb counting method to form a state space model, therein, the SOC is regarded as the state variable, which is described by the state equation according to the coulomb counting method. While the established model is the measurement equation that relates the measured terminal voltage to the SOC. Subsequently, as the state space model is usually used in conjunction with filters to achieve state estimation, an introduction to the filtering algorithms including their

detailed mechanism and major formulas is presented. Thereafter, based on the formed state space model, battery SOC is estimated using different filtering algorithms, namely extended Kalman filter, unscented Kalman filter, and particle filter. The estimation results are compared in terms of accuracy and computational complexity.

- An intelligent battery capacity estimation framework is proposed in Chapter 4 to achieve fast online capacity estimation using flexible partial charging curves with a fixed length. To take full advantage of the huge amount of battery aging data, a convolutional neural network (CNN) is used in this chapter to automatically extract features and estimate capacity in one framework. Compared with conventional capacity estimation methods, the CNN-based methods are free of a priori knowledge about the battery, exempt from manually extracting health features, and have strong representational power and expressibility. This chapter starts with an introduction to the structure of CNN to illustrate its characteristics. In view of the poor performance of CNN on small datasets and the limited computing power of BMS, a preliminary introduction to transfer learning and network pruning techniques is carried out, and they are later combined to the CNN-based framework to improve the estimation results on small degradation datasets and remove redundant model structures, respectively. Afterward, detailed construction processes of the proposed estimation framework are presented step by step, including the input generation that enables the application of CNNs and network construction that integrates the transfer learning and pruning techniques into a CNN-based modelling framework. Finally, with the given implementation details and given two battery degradation datasets, the efficacy of the proposed capacity estimation framework is verified by the experimental results.

- Chapter 5 proposes a novel framework to estimate the battery SOC and capacity simultaneously based on the Gaussian process regression model and the modified CNN model. Since there exists intrinsic coupling relationships between SOC and capacity, the estimation results of SOC and capacity can be improved mutually through the joint estimation framework. Besides, fiber optic sensors are attached to the cell surface to obtain more information, and the newly involved measurement is used for SOC estimation to further improve the estimation accuracy. The experimental results have confirmed that the proposed joint estimation framework along with the utilization of the new sensing measurement provides more accurate estimation results and lower estimation uncertainty.

Collectively, the models and estimation methods developed in this thesis aim to allow for better utilization of lithium-ion batteries by monitoring their SOC and SOH accurately and timely, while requiring less or no laborious and time-consuming aging tests and characterization tests. Thus dispense with the investment of time and energy to identify parameters and manually extract features, making them promising methods to provide valuable insight to the investigations of other types of batteries with different chemistries under different application scenarios.

6.2 Future work

Although the work presented herein offers benefits over the conventional methods, there are potential extensions of this thesis that worth further investigations. The scope for future work which can lead on from this thesis are summarized below:

- The developed CNN-based estimation framework is pruned and compressed in Chapter 4, however, whether it can be implemented in the onboard BMSs with limited computational capabilities has not been verified. A practical

low-cost microcontroller-based BMS can be used for realization of the proposed algorithm. Besides, to make good use of massive data from the achieved battery datasets, the cloud-based BMSs with sufficient resources will be investigated to process the heavy computing task, which can remove the restrictions on computation power and memory sources for local processors. Therein, battery data is streamed to a cloud computer, and the model weights can be updated in real-time.

- The application of the newly introduced temperature and tension measurements acquired by the fiber optic sensors is still very limited. Considering its potential in improving the battery state estimation, the information embedded in this sensor measurement will be further mined, and the way to utilize it with various estimation techniques will be carefully studied to improve the estimation performance.

- The developed methods focus on a single battery cell. It is meaningful to extend the proposed methods to a battery pack with series and/or parallel connected cells, as battery packs are usually employed in electric vehicles and grid applications to provide sufficient power and energy.

- As set out in Section 1.1.2, apart from the battery state estimation, other functions such as charge/discharge control, cell balancing, thermal management, and safety protection are also important in a BMS to realize better utilization of the battery. Therefore, these aspects can be researched together with the proposed framework and methods in this thesis. To design optimal battery charging/discharging strategies to prevent battery from damage and prolong its cycle life, factors such as charging/discharging time and temperature rises need to be taken into account. To mitigate the inconsistencies among battery cells, cell balancing is indispensable to compensate for weaker cells, thereby improving the capacity of the battery pack and

prolonging its service life. And to operate the battery within a desirable temperature range, an efficient thermal management system is essential. Further, fault diagnosis methods will be studied to detect the premature failure of battery, such as short circuit, over-voltage and under-voltage, to prevent the battery from permanent and even catastrophic damages.

References

[1] "The race to zero emissions, and why the world depends on it, un news," Dec 2020. 2

[2] I. Tsiropoulos, D. Tarvydas, and N. Lebedeva, "Li-ion batteries for mobility and stationary storage applications scenarios for costs and market growth," *Publications Office of the European Union: Luxembourg*, p. 72, 2018. 2

[3] F. J. De Sisternes, J. D. Jenkins, and A. Botterud, "The value of energy storage in decarbonizing the electricity sector," *Applied Energy*, vol. 175, pp. 368–379, 2016. 2

[4] V. Monteiro, H. Gonçalves, J. C. Ferreira, J. L. Afonso, J. Carmo, and J. Ribeiro, "Batteries charging systems for electric and plug-in hybrid electric vehicles," in *New Advances in Vehicular Technology and Automotive Engineering*, pp. 149–168, InTech, 2012. 3

[5] K. Divya and J. Østergaard, "Battery energy storage technology for power systems—an overview," *Electric power systems research*, vol. 79, no. 4, pp. 511–520, 2009. 4

[6] S. Koohi-Fayegh and M. A. Rosen, "A review of energy storage types, applications and recent developments," *Journal of Energy Storage*, vol. 27, p. 101047, 2020. 4

[7] C. Zhang, Y.-L. Wei, P.-F. Cao, and M.-C. Lin, "Energy storage system:

Current studies on batteries and power condition system," *Renewable and Sustainable Energy Reviews*, vol. 82, pp. 3091–3106, 2018. 4, 5, 8

[8] T. Lehtola and A. Zahedi, "Solar energy and wind power supply supported by storage technology: A review," *Sustainable Energy Technologies and Assessments*, vol. 35, pp. 25–31, 2019. 4

[9] Y. Liang, C.-Z. Zhao, H. Yuan, Y. Chen, W. Zhang, J.-Q. Huang, D. Yu, Y. Liu, M.-M. Titirici, Y.-L. Chueh, *et al.*, "A review of rechargeable batteries for portable electronic devices," *InfoMat*, vol. 1, no. 1, pp. 6–32, 2019. 5

[10] G. J. May, A. Davidson, and B. Monahov, "Lead batteries for utility energy storage: A review," *Journal of energy storage*, vol. 15, pp. 145–157, 2018. 5, 6, 8, 9

[11] T. Tantichanakul, O. Chailapakul, and N. Tantavichet, "Influence of fumed silica and additives on the gel formation and performance of gel valve-regulated lead-acid batteries," *Journal of Industrial and Engineering Chemistry*, vol. 19, no. 6, pp. 2085–2091, 2013. 5

[12] C. Zou, L. Zhang, X. Hu, Z. Wang, T. Wik, and M. Pecht, "A review of fractional-order techniques applied to lithium-ion batteries, lead-acid batteries, and supercapacitors," *Journal of Power Sources*, vol. 390, pp. 286–296, 2018. 5

[13] D. Enos, "Lead-acid batteries for medium-and large-scale energy storage," in *Advances in Batteries for Medium and Large-Scale Energy Storage*, pp. 57–71, Elsevier, 2015. 5

[14] S. Dusmez, A. Cook, and A. Khaligh, "Comprehensive analysis of high qual-

ity power converters for level 3 off-board chargers," in *2011 IEEE Vehicle Power and Propulsion Conference*, pp. 1–10, IEEE, 2011. 6

[15] C. Iclodean, B. Varga, N. Burnete, D. Cimerdean, and B. Jurchiş, "Comparison of different battery types for electric vehicles," in *IOP conference series: materials science and engineering*, vol. 252, p. 012058, IOP Publishing, 2017. 6

[16] X. Hu, C. Zou, C. Zhang, and Y. Li, "Technological developments in batteries: a survey of principal roles, types, and management needs," *IEEE Power and Energy Magazine*, vol. 15, no. 5, pp. 20–31, 2017. 6, 7, 8, 9, 11

[17] Z. Yan, J. Xiao, W. Lai, L. Wang, F. Gebert, Y. Wang, Q. Gu, H. Liu, S.-L. Chou, H. Liu, *et al.*, "Nickel sulfide nanocrystals on nitrogen-doped porous carbon nanotubes with high-efficiency electrocatalysis for room-temperature sodium-sulfur batteries," *Nature communications*, vol. 10, no. 1, pp. 1–8, 2019. 6

[18] S. Schaefer, S. P. Vudata, D. Bhattacharyya, and R. Turton, "Transient modeling and simulation of a nonisothermal sodium–sulfur cell," *Journal of Power Sources*, vol. 453, p. 227849, 2020. 6

[19] W. Wang, Q. Luo, B. Li, X. Wei, L. Li, and Z. Yang, "Recent progress in redox flow battery research and development," *Advanced Functional Materials*, vol. 23, no. 8, pp. 970–986, 2013. 7

[20] C. Menictas, M. Skyllas-Kazacos, and T. M. Lim, *Advances in batteries for medium and large-scale energy storage: types and applications*. Elsevier, 2014. 7

[21] C. Minke and T. Turek, "Materials, system designs and modelling

approaches in techno-economic assessment of all-vanadium redox flow batteries–a review," *Journal of Power Sources*, vol. 376, pp. 66–81, 2018. 7

[22] B. Dunn, H. Kamath, and J.-M. Tarascon, "Electrical energy storage for the grid: a battery of choices," *Science*, vol. 334, no. 6058, pp. 928–935, 2011. 8

[23] A. Manthiram, "A reflection on lithium-ion battery cathode chemistry," *Nature communications*, vol. 11, no. 1, pp. 1–9, 2020. 8

[24] F. H. Gandoman, J. Jaguemont, S. Goutam, R. Gopalakrishnan, Y. Firouz, T. Kalogiannis, N. Omar, and J. Van Mierlo, "Concept of reliability and safety assessment of lithium-ion batteries in electric vehicles: Basics, progress, and challenges," *Applied Energy*, vol. 251, p. 113343, 2019. 9

[25] S. Nejad, *Adaptive Techniques for Estimation and Online Monitoring of Battery Energy Storage Devices*. PhD thesis, University of Sheffield, 2016. 9

[26] L. Jörissen and H. Frey, "Energy— energy storage," 2009. 9

[27] K. Ghaib and F.-Z. Ben-Fares, "A design methodology of stand-alone photovoltaic power systems for rural electrification," *Energy Conversion and Management*, vol. 148, pp. 1127–1141, 2017. 9

[28] B. K. Chakrabarti, E. Kalamaras, A. K. Singh, A. Bertei, J. Rubio-Garcia, V. Yufit, K. M. Tenny, B. Wu, F. Tariq, Y. S. Hajimolana, *et al.*, "Modelling of redox flow battery electrode processes at a range of length scales: a review," *Sustainable Energy & Fuels*, vol. 4, no. 11, pp. 5433–5468, 2020. 9

[29] T. S. of Queensland, "Types of battery energy storage," Feb 2018. 9

[30] K.-P. Kairies, "Battery storage technology improvements and cost reductions to 2030: A deep dive," *Int. Renew. Energy Agency Work*, vol. 2017, 2017. 9

[31] J. Jiang, S. Liu, Z. Ma, K. Wu, *et al.*, "Butler-volmer equation-based model and its implementation on state of power prediction of high-power lithium titanate batteries considering temperature effects," *Energy*, vol. 117, pp. 58–72, 2016. 11

[32] L. W. Sommer, P. Kiesel, A. Ganguli, A. Lochbaum, B. Saha, J. Schwartz, C.-J. Bae, M. Alamgir, and A. Raghavan, "Fast and slow ion diffusion processes in lithium ion pouch cells during cycling observed with fiber optic strain sensors," *Journal of Power Sources*, vol. 296, pp. 46–52, 2015. 14, 15

[33] A. Raghavan, P. Kiesel, L. W. Sommer, J. Schwartz, A. Lochbaum, A. Hegyi, A. Schuh, K. Arakaki, B. Saha, A. Ganguli, *et al.*, "Embedded fiber-optic sensing for accurate internal monitoring of cell state in advanced battery management systems part 1: Cell embedding method and performance," *Journal of Power Sources*, vol. 341, pp. 466–473, 2017. 14, 15, 120

[34] L. W. Sommer, A. Raghavan, P. Kiesel, B. Saha, J. Schwartz, A. Lochbaum, A. Ganguli, C.-J. Bae, and M. Alamgir, "Monitoring of intercalation stages in lithium-ion cells over charge-discharge cycles with fiber optic sensors," *Journal of the Electrochemical Society*, vol. 162, no. 14, p. A2664, 2015. 14

[35] M. Nascimento, M. S. Ferreira, and J. L. Pinto, "Real time thermal monitoring of lithium batteries with fiber sensors and thermocouples: A comparative study," *Measurement*, vol. 111, pp. 260–263, 2017. 15

[36] A. Fortier, M. Tsao, N. D. Williard, Y. Xing, and M. G. Pecht, "Preliminary study on integration of fiber optic bragg grating sensors in li-ion batteries

and in situ strain and temperature monitoring of battery cells," *Energies*, vol. 10, no. 7, p. 838, 2017. 15

[37] J. Peng, S. Jia, Y. Jin, S. Xu, and Z. Xu, "Design and investigation of a sensitivity-enhanced fiber bragg grating sensor for micro-strain measurement," *Sensors and Actuators A: Physical*, vol. 285, pp. 437–447, 2019. 15

[38] J. Peng, X. Zhou, S. Jia, Y. Jin, S. Xu, and J. Chen, "High precision strain monitoring for lithium ion batteries based on fiber bragg grating sensors," *Journal of Power Sources*, vol. 433, p. 226692, 2019. 15

[39] A. Ganguli, B. Saha, A. Raghavan, P. Kiesel, K. Arakaki, A. Schuh, J. Schwartz, A. Hegyi, L. W. Sommer, A. Lochbaum, *et al.*, "Embedded fiber-optic sensing for accurate internal monitoring of cell state in advanced battery management systems part 2: Internal cell signals and utility for state estimation," *Journal of Power Sources*, vol. 341, pp. 474–482, 2017. 15

[40] B. Rente, M. Fabian, M. Vidakovic, X. Liu, X. Li, K. Li, T. Sun, and K. T. Grattan, "Lithium-ion battery state-of-charge estimator based on fbg-based strain sensor and employing machine learning," *IEEE Sensors Journal*, vol. 21, no. 2, pp. 1453–1460, 2020. 16, 120

[41] D. Simon, *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley & Sons, 2006. 16, 62

[42] R. E. Kalman *et al.*, "A new approach to linear filtering and prediction problems," 1960. 17, 53

[43] S. C. Patwardhan, S. Narasimhan, P. Jagadeesan, B. Gopaluni, and S. L. Shah, "Nonlinear bayesian state estimation: A review of recent develop-

ments," *Control Engineering Practice*, vol. 20, no. 10, pp. 933–953, 2012. 17

[44] H. H. Afshari, S. A. Gadsden, and S. Habibi, "Gaussian filters for parameter and state estimation: A general review of theory and recent trends," *Signal Processing*, vol. 135, pp. 218–238, 2017. 17

[45] R. E. Kalman and R. S. Bucy, "New results in linear filtering and prediction theory," *Journal of basic engineering*, vol. 83, no. 1, pp. 95–108, 1961. 17

[46] M. Jouin, R. Gouriveau, D. Hissel, M.-C. Péra, and N. Zerhouni, "Particle filter-based prognostics: Review, discussion and perspectives," *Mechanical Systems and Signal Processing*, vol. 72, pp. 2–31, 2016. 18, 64

[47] A. Fotouhi, D. J. Auger, K. Propp, S. Longo, and M. Wild, "A review on electric vehicle battery modelling: From lithium-ion toward lithium–sulphur," *Renewable and Sustainable Energy Reviews*, vol. 56, pp. 1008–1021, 2016. 24

[48] Z. Wei, K. J. Tseng, N. Wai, T. M. Lim, and M. Skyllas-Kazacos, "Adaptive estimation of state of charge and capacity with online identified battery model for vanadium redox flow battery," *Journal of Power Sources*, vol. 332, pp. 389–398, 2016. 25, 30

[49] M. Doyle, T. F. Fuller, and J. Newman, "Modeling of galvanostatic charge and discharge of the lithium/polymer/insertion cell," *Journal of the Electrochemical society*, vol. 140, no. 6, p. 1526, 1993. 25

[50] T. F. Fuller, M. Doyle, and J. Newman, "Simulation and optimization of the dual lithium ion insertion cell," *Journal of the Electrochemical Society*, vol. 141, no. 1, p. 1, 1994. 25

[51] S. Abada, G. Marlair, A. Lecocq, M. Petit, V. Sauvant-Moynot, and F. Huet, "Safety focused modeling of lithium-ion batteries: A review," *Journal of Power Sources*, vol. 306, pp. 178–192, 2016. 26

[52] S. Santhanagopalan, Q. Guo, P. Ramadass, and R. E. White, "Review of models for predicting the cycling performance of lithium ion batteries," *Journal of power sources*, vol. 156, no. 2, pp. 620–628, 2006. 26

[53] D. Zhang, B. N. Popov, and R. E. White, "Modeling lithium intercalation of a single spinel particle under potentiodynamic control," *Journal of the Electrochemical Society*, vol. 147, no. 3, p. 831, 2000. 26

[54] M. Guo, G. Sikha, and R. E. White, "Single-particle model for a lithium-ion cell: Thermal behavior," *Journal of The Electrochemical Society*, vol. 158, no. 2, p. A122, 2010. 26

[55] A. M. Dubey, R. Ram, and A. K. Yadav, "Ion cell performance using single particle representation of battery electrode," *Journal of Electroanalytical Chemistry*, vol. 768, pp. 11–17, 2016. 26

[56] H. Liu, Z. Wei, W. He, and J. Zhao, "Thermal issues about li-ion batteries and recent progress in battery thermal management systems: A review," *Energy conversion and management*, vol. 150, pp. 304–330, 2017. 26

[57] J. Li, N. Lotfi, R. G. Landers, and J. Park, "A single particle model for lithium-ion batteries with electrolyte and stress-enhanced diffusion physics," *Journal of The Electrochemical Society*, vol. 164, no. 4, p. A874, 2017. 26

[58] J. Zhou, B. Xing, and C. Wang, "A review of lithium ion batteries electrochemical models for electric vehicles," in *E3S Web of Conferences*, vol. 185, p. 04001, 2020. 26

[59] M. T. Lawder, B. Suthar, P. W. Northrop, S. De, C. M. Hoff, O. Leitermann, M. L. Crow, S. Santhanagopalan, and V. R. Subramanian, "Battery energy storage system (bess) and battery management system (bms) for grid-scale applications," *Proceedings of the IEEE*, vol. 102, no. 6, pp. 1014–1030, 2014. 27

[60] P. Shrivastava, T. K. Soon, M. Y. I. B. 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, p. 109233, 2019. 27, 28, 29, 35, 38

[61] Y. Wang, J. Tian, Z. Sun, L. Wang, R. Xu, M. Li, and Z. Chen, "A comprehensive review of battery modeling and state estimation approaches for advanced battery management systems," *Renewable and Sustainable Energy Reviews*, vol. 131, p. 110015, 2020. 28

[62] B. Y. Liaw, G. Nagasubramanian, R. G. Jungst, and D. H. Doughty, "Modeling of lithium ion cells—a simple equivalent-circuit model approach," *Solid state ionics*, vol. 175, no. 1-4, pp. 835–839, 2004. 29

[63] H. He, R. Xiong, H. Guo, and S. Li, "Comparison study on the battery models used for the energy management of batteries in electric vehicles," *Energy Conversion and Management*, vol. 64, pp. 113–121, 2012. 29

[64] P. Shen, M. Ouyang, L. Lu, J. Li, and X. Feng, "The co-estimation of state of charge, state of health, and state of function for lithium-ion batteries in electric vehicles," *IEEE Transactions on vehicular technology*, vol. 67, no. 1, pp. 92–103, 2017. 29

[65] H. He, R. Xiong, and J. Fan, "Evaluation of lithium-ion battery equivalent circuit models for state of charge estimation by an experimental approach," *energies*, vol. 4, no. 4, pp. 582–598, 2011. 29

[66] A. Fotouhi, D. J. Auger, K. Propp, and S. Longo, "Accuracy versus simplicity in online battery model identification," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 2, pp. 195–206, 2016. 29

[67] 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, p. 109334, 2019. 29

[68] G. L. Plett, "Extended kalman filtering for battery management systems of lipb-based hev battery packs: Part 3. state and parameter estimation," *Journal of Power sources*, vol. 134, no. 2, pp. 277–292, 2004. 29, 36

[69] G. L. Plett, "Sigma-point kalman filtering for battery management systems of lipb-based hev battery packs: Part 2: Simultaneous state and parameter estimation," *Journal of power sources*, vol. 161, no. 2, pp. 1369–1384, 2006. 30, 36

[70] C. Zhang, W. Allafi, Q. Dinh, P. Ascencio, and J. Marco, "Online estimation of battery equivalent circuit model parameters and state of charge using decoupled least squares technique," *Energy*, vol. 142, pp. 678–688, 2018. 30

[71] X. Lai, W. Gao, Y. Zheng, M. Ouyang, J. Li, X. Han, and L. Zhou, "A comparative study of global optimization methods for parameter identification of different equivalent circuit models for li-ion batteries," *Electrochimica acta*, vol. 295, pp. 1057–1066, 2019. 30

[72] L. Kang, X. Zhao, and J. Ma, "A new neural network model for the state-of-charge estimation in the battery degradation process," *Applied Energy*, vol. 121, pp. 20–27, 2014. 31

[73] M. A. Hannan, M. S. H. Lipu, A. Hussain, M. H. Saad, and A. Ayob, "Neural network approach for estimating state of charge of lithium-ion battery

using backtracking search algorithm," *Ieee Access*, vol. 6, pp. 10069–10079, 2018. 31, 38

[74] 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. 31, 38

[75] C. Chen, R. Xiong, R. Yang, 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. 31

[76] M. Charkhgard and M. Farrokhi, "State-of-charge estimation for lithium-ion batteries using neural networks and ekf," *IEEE transactions on industrial electronics*, vol. 57, no. 12, pp. 4178–4187, 2010. 31, 36

[77] L. Zhang, K. Li, D. Du, C. Zhu, and M. Zheng, "A sparse least squares support vector machine used for soc estimation of li-ion batteries," *IFAC-PapersOnLine*, vol. 52, no. 11, pp. 256–261, 2019. 31

[78] X. Shu, G. Li, Y. Zhang, S. Shen, Z. Chen, and Y. Liu, "Stage of charge estimation of lithium-ion battery packs based on improved cubature kalman filter with long short-term memory model," *IEEE Transactions on Transportation Electrification*, 2020. 31

[79] T. Kim, Y. Wang, Z. Sahinoglu, T. Wada, S. Hara, and W. Qiao, "State of charge estimation based on a realtime battery model and iterative smooth variable structure filter," in *2014 IEEE Innovative Smart Grid Technologies-Asia (ISGT ASIA)*, pp. 132–137, IEEE, 2014. 33

[80] W.-Y. Chang, "The state of charge estimating methods for battery: A review," *ISRN Applied Mathematics*, vol. 2013, 2013. 34

[81] X. Tang, B. Liu, and F. Gao, "State of charge estimation of lifepo4 battery based on a gain-classifier observer," *Energy Procedia*, vol. 105, pp. 2071–2076, 2017. 34

[82] R. Xiong, J. Cao, Q. Yu, H. He, and F. Sun, "Critical review on the battery state of charge estimation methods for electric vehicles," *IEEE Access*, 2017. 35

[83] A. Attanayaka, J. Karunadasa, and K. Hemapala, "Estimation of state of charge for lithium-ion batteries—a review," *AIMS Energy*, vol. 7, no. 2, pp. 186–210, 2019. 35

[84] S. Lee, J. Kim, J. Lee, and B. H. Cho, "State-of-charge and capacity estimation of lithium-ion battery using a new open-circuit voltage versus state-of-charge," *Journal of power sources*, vol. 185, no. 2, pp. 1367–1373, 2008. 35

[85] L. Zhu, Z. Sun, H. Dai, and X. Wei, "A novel modeling methodology of open circuit voltage hysteresis for lifepo4 batteries based on an adaptive discrete preisach model," *Applied Energy*, vol. 155, pp. 91–109, 2015. 35

[86] Y. Xing, W. He, M. Pecht, and K. L. Tsui, "State of charge estimation of lithium-ion batteries using the open-circuit voltage at various ambient temperatures," *Applied Energy*, vol. 113, pp. 106–115, 2014. 35

[87] I.-S. Kim, "The novel state of charge estimation method for lithium battery using sliding mode observer," *Journal of Power Sources*, vol. 163, no. 1, pp. 584–590, 2006. 36

[88] X. Chen, W. Shen, Z. Cao, and A. Kapoor, "A novel approach for state of charge estimation based on adaptive switching gain sliding mode observer in electric vehicles," *Journal of Power Sources*, vol. 246, pp. 667–678, 2014. 36

[89] J. Kim and B.-H. Cho, "State-of-charge estimation and state-of-health prediction of a li-ion degraded battery based on an ekf combined with a per-unit system," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 9, pp. 4249–4260, 2011. 36

[90] B. S. Bhangu, P. Bentley, D. A. Stone, and C. M. Bingham, "Nonlinear observers for predicting state-of-charge and state-of-health of lead-acid batteries for hybrid-electric vehicles," *IEEE transactions on vehicular technology*, vol. 54, no. 3, pp. 783–794, 2005. 36

[91] M. Mastali, J. Vazquez-Arenas, R. Fraser, M. Fowler, S. Afshar, and M. Stevens, "Battery state of the charge estimation using kalman filtering," *Journal of Power Sources*, vol. 239, pp. 294–307, 2013. 36

[92] O. Barbarisi, F. Vasca, and L. Glielmo, "State of charge kalman filter estimator for automotive batteries," *Control Engineering Practice*, vol. 14, no. 3, pp. 267–275, 2006. 36

[93] Q. Yu, R. Xiong, C. Lin, W. Shen, and J. Deng, "Lithium-ion battery parameters and state-of-charge joint estimation based on h-infinity and unscented kalman filters," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 10, pp. 8693–8701, 2017. 36

[94] S. Santhanagopalan and R. E. White, "State of charge estimation using an unscented filter for high power lithium ion cells," *International Journal of Energy Research*, vol. 34, no. 2, pp. 152–163, 2010. 36

[95] J. Zhang and C. Xia, "State-of-charge estimation of valve regulated lead acid battery based on multi-state unscented kalman filter," *International Journal of Electrical Power & Energy Systems*, vol. 33, no. 3, pp. 472–476, 2011. 36

[96] H. He, R. Xiong, X. Zhang, F. Sun, and J. Fan, "State-of-charge estimation of the lithium-ion battery using an adaptive extended kalman filter based on an improved thevenin model," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 4, pp. 1461–1469, 2011. 36

[97] R. Xiong, H. He, F. Sun, and K. Zhao, "Evaluation on state of charge estimation of batteries with adaptive extended kalman filter by experiment approach," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 1, pp. 108–117, 2013. 36

[98] F. Sun, X. Hu, Y. Zou, and S. Li, "Adaptive unscented kalman filtering for state of charge estimation of a lithium-ion battery for electric vehicles," *Energy*, vol. 36, no. 5, pp. 3531–3540, 2011. 36

[99] M. Gao, Y. Liu, and Z. He, "Battery state of charge online estimation based on particle filter," in *Image and Signal Processing (CISP), 2011 4th International Congress on*, vol. 4, pp. 2233–2236, IEEE, 2011. 36

[100] Y. He, X. Liu, C. Zhang, and Z. Chen, "A new model for state-of-charge (soc) estimation for high-power li-ion batteries," *Applied Energy*, vol. 101, pp. 808–814, 2013. 36

[101] Y. Zhang, C. Zhang, and X. Zhang, "State-of-charge estimation of the lithium-ion battery system with time-varying parameter for hybrid electric vehicles," *IET Control Theory & Applications*, vol. 8, no. 3, pp. 160–167, 2013. 36

[102] R. Xiong, Q. Yu, C. Lin, *et al.*, "A novel method to obtain the open circuit voltage for the state of charge of lithium ion batteries in electric vehicles by using h infinity filter," *Applied energy*, vol. 207, pp. 346–353, 2017. 36

[103] M. Charkhgard and M. H. Zarif, "Design of adaptive $h\infty$ filter for implementing on state-of-charge estimation based on battery state-of-charge-varying modelling," *IET Power Electronics*, vol. 8, no. 10, pp. 1825–1833, 2015. 37

[104] D. N. How, M. Hannan, M. H. 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, pp. 136116–136136, 2019. 37

[105] W. He, N. Williard, C. Chen, and M. Pecht, "State of charge estimation for li-ion batteries using neural network modeling and unscented kalman filter-based error cancellation," *International Journal of Electrical Power & Energy Systems*, vol. 62, pp. 783–791, 2014. 38

[106] M. Ismail, R. Dlyma, A. Elrakaybi, R. Ahmed, and S. Habibi, "Battery state of charge estimation using an artificial neural network," in *2017 IEEE Transportation Electrification Conference and Expo (ITEC)*, pp. 342–349, IEEE, 2017. 38

[107] D. N. How, M. A. Hannan, M. S. H. Lipu, K. S. Sahari, P. J. Ker, and K. M. Muttaqi, "State-of-charge estimation of li-ion battery in electric vehicles: A deep neural network approach," *IEEE Transactions on Industry Applications*, vol. 56, no. 5, pp. 5565–5574, 2020. 38

[108] F. Yang, W. Li, C. Li, and Q. Miao, "State-of-charge estimation of lithium-ion batteries based on gated recurrent neural network," *Energy*, vol. 175, pp. 66–75, 2019. 38

[109] E. Chemali, P. J. Kollmeyer, M. Preindl, R. Ahmed, and A. Emadi, "Long short-term memory networks for accurate state-of-charge estimation of li-ion batteries," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 8, pp. 6730–6739, 2017. 38

[110] A. J. Salkind, C. Fennie, P. Singh, T. Atwater, and D. E. Reisner, "Determination of state-of-charge and state-of-health of batteries by fuzzy logic methodology," *Journal of Power sources*, vol. 80, no. 1-2, pp. 293–300, 1999. 38

[111] G. O. Sahinoglu, M. Pajovic, Z. Sahinoglu, Y. Wang, P. V. Orlik, and T. Wada, "Battery state-of-charge estimation based on regular/recurrent gaussian process regression," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 5, pp. 4311–4321, 2017. 38

[112] F. Xiao, C. Li, Y. Fan, G. Yang, and X. Tang, "State of charge estimation for lithium-ion battery based on gaussian process regression with deep recurrent kernel," *International Journal of Electrical Power & Energy Systems*, vol. 124, p. 106369, 2021. 38

[113] W. Waag, S. Käbitz, and D. U. Sauer, "Experimental investigation of the lithium-ion battery impedance characteristic at various conditions and aging states and its influence on the application," *Applied energy*, vol. 102, pp. 885–897, 2013. 38

[114] 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. 38, 39, 41

[115] 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. 39

[116] M. Ouyang, X. Feng, X. Han, L. Lu, Z. Li, and X. He, "A dynamic capacity degradation model and its applications considering varying load for a large format li-ion battery," *Applied Energy*, vol. 165, pp. 48–59, 2016. 40

[117] K. K. Sadabadi, P. Ramesh, P. Tulpule, and G. Rizzoni, "Design and calibration of a semi-empirical model for capturing dominant aging mechanisms of a pba battery," *Journal of Energy Storage*, vol. 24, p. 100789, 2019. 40

[118] D. Zhang, S. Dey, H. E. Perez, and S. J. Moura, "Real-time capacity estimation of lithium-ion batteries utilizing thermal dynamics," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 3, pp. 992–1000, 2019. 40

[119] K. Liu, C. Zou, K. Li, and T. Wik, "Charging pattern optimization for lithium-ion batteries with an electrothermal-aging model," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 12, pp. 5463–5474, 2018. 40

[120] Q. Yu, R. Xiong, R. Yang, and M. G. Pecht, "Online capacity estimation for lithium-ion batteries through joint estimation method," *Applied Energy*, vol. 255, p. 113817, 2019. 40

[121] 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. 40

[122] R. Xiong, L. Li, Z. 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. 40

[123] Y. Zheng, C. Qin, X. Lai, X. Han, and Y. Xie, "A novel capacity estimation method for lithium-ion batteries using fusion estimation of charging curve sections and discrete arrhenius aging model," *Applied Energy*, vol. 251, p. 113327, 2019. 40

[124] C. Zhang, K. Li, S. Mcloone, and Z. Yang, "Battery modelling methods for electric vehicles-a review," in *2014 European Control Conference (ECC)*, pp. 2673–2678, IEEE, 2014. 40

[125] S. Li, S. Pischinger, C. He, L. Liang, and M. Stapelbroek, "A comparative study of model-based capacity estimation algorithms in dual estimation frameworks for lithium-ion batteries under an accelerated aging test," *Applied energy*, vol. 212, pp. 1522–1536, 2018. 40

[126] C. Hu, B. D. Youn, and J. Chung, "A multiscale framework with extended kalman filter for lithium-ion battery soc and capacity estimation," *Applied Energy*, vol. 92, pp. 694–704, 2012. 40

[127] C. Chen, R. Xiong, and W. Shen, "A lithium-ion battery-in-the-loop approach to test and validate multiscale dual h infinity filters for state-of-charge and capacity estimation," *IEEE Transactions on power Electronics*, vol. 33, no. 1, pp. 332–342, 2017. 40

[128] M. Ye, H. Guo, R. Xiong, and H. Mu, "An online model-based battery parameter and state estimation method using multi-scale dual adaptive particle filters," *Energy Procedia*, vol. 105, pp. 4549–4554, 2017. 40

[129] C. Weng, Y. Cui, J. Sun, and H. Peng, "On-board state of health monitoring of lithium-ion batteries using incremental capacity analysis with support vector regression," *Journal of Power Sources*, vol. 235, pp. 36–44, 2013. 41

[130] Y. Li, M. Abdel-Monem, R. Gopalakrishnan, M. Berecibar, E. Nanini-Maury, N. Omar, P. van den Bossche, and J. Van Mierlo, "A quick on-line state of health estimation method for li-ion battery with incremental capacity curves processed by gaussian filter," *Journal of Power Sources*, vol. 373, pp. 40–53, 2018. 41

[131] C. Weng, X. Feng, J. Sun, and H. Peng, "State-of-health monitoring of lithium-ion battery modules and packs via incremental capacity peak tracking," *Applied Energy*, vol. 180, pp. 360–368, 2016. 41

[132] L. Zheng, J. Zhu, D. D.-C. Lu, G. Wang, and T. He, "Incremental capacity analysis and differential voltage analysis based state of charge and capacity estimation for lithium-ion batteries," *Energy*, vol. 150, pp. 759–769, 2018. 41

[133] 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. 396, pp. 453–458, 2018. 41

[134] Y. Li, K. Liu, A. M. Foley, A. Zülke, M. Berecibar, E. Nanini-Maury, J. Van 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, p. 109254, 2019. 41

[135] R. R. Richardson, C. R. Birkl, M. A. Osborne, and D. A. Howey, "Gaussian process regression for in situ capacity estimation of lithium-ion batteries," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 1, pp. 127–138, 2018. 42

[136] Y. Li, H. Sheng, Y. Cheng, D.-I. Stroe, and R. Teodorescu, "State-of-health estimation of lithium-ion batteries based on semi-supervised transfer component analysis," *Applied Energy*, vol. 277, p. 115504, 2020. 42

[137] M. Berecibar, F. Devriendt, M. Dubarry, I. Villarreal, N. Omar, W. Verbeke, and J. Van Mierlo, "Online state of health estimation on nmc cells based on predictive analytics," *Journal of Power Sources*, vol. 320, pp. 239–250, 2016. 42

[138] D. Yang, Y. Wang, R. Pan, R. Chen, and Z. Chen, "State-of-health estimation for the lithium-ion battery based on support vector regression," *Applied Energy*, vol. 227, pp. 273–283, 2018. 42

[139] K. Liu, Y. Li, X. Hu, M. Lucu, and W. D. Widanage, "Gaussian process regression with automatic relevance determination kernel for calendar aging prediction of lithium-ion batteries," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 3767–3777, 2019. 42

[140] X. Li, C. Yuan, X. Li, and Z. Wang, "State of health estimation for li-ion battery using incremental capacity analysis and gaussian process regression," *Energy*, vol. 190, p. 116467, 2020. 42

[141] 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. 43, 90

[142] P. Guo, Z. Cheng, and L. Yang, "A data-driven remaining capacity estimation approach for lithium-ion batteries based on charging health feature extraction," *Journal of Power Sources*, vol. 412, pp. 442–450, 2019. 43

[143] 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. 43

[144] G.-W. You, S. Park, and D. Oh, "Diagnosis of electric vehicle batteries using recurrent neural networks," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 6, pp. 4885–4893, 2017. 43

[145] Y. Li, C. Zou, M. Berecibar, E. Nanini-Maury, J. C.-W. Chan, P. van den Bossche, J. Van Mierlo, and N. Omar, "Random forest regression for online capacity estimation of lithium-ion batteries," *Applied energy*, vol. 232, pp. 197–210, 2018. 43

[146] S. Shen, M. Sadoughi, X. Chen, M. Hong, and C. Hu, "A deep learning method for online capacity estimation of lithium-ion batteries," *Journal of Energy Storage*, vol. 25, p. 100817, 2019. 43

[147] K. Kaur, A. Garg, X. Cui, S. Singh, and B. K. Panigrahi, "Deep learning networks for capacity estimation for monitoring soh of li-ion batteries for electric vehicles," *International Journal of Energy Research (2020)*, 2020. 44

[148] Y. Li, H. Sheng, Y. Cheng, D.-I. Stroe, and R. Teodorescu, "State-of-health estimation of lithium-ion batteries based on semi-supervised transfer component analysis," *Applied Energy*, vol. 277, p. 115504, 2020. 44

[149] Y. Tan and G. Zhao, "Transfer learning with long short-term memory network for state-of-health prediction of lithium-ion batteries," *IEEE Transactions on Industrial Electronics*, vol. 67, pp. 8723 – 8731, 10 2019. 44

[150] S. Shen, M. Sadoughi, M. Li, Z. Wang, and C. Hu, "Deep convolutional neural networks with ensemble learning and transfer learning for capacity estimation of lithium-ion batteries," *Applied Energy*, vol. 260, p. 114296, 2020. 44

[151] Y. Li and J. Tao, "Cnn and transfer learning based online soh estimation for lithium-ion battery," in *2020 Chinese Control And Decision Conference (CCDC)*, pp. 5489–5494, IEEE, 2020. 44

[152] M. Berecibar, I. Gandiaga, I. Villarreal, N. Omar, J. Van Mierlo, and P. Van 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. 44

[153] H. Rahimi-Eichi, U. Ojha, F. Baronti, and M. Y. Chow, "Battery management system: An overview of its application in the smart grid and electric vehicles," *IEEE Industrial Electronics Magazine*, vol. 7, pp. 4–16, June 2013. 46

[154] K. Li, J.-X. Peng, and G. W. Irwin, "A fast nonlinear model identification method," *IEEE Transactions on Automatic Control*, vol. 50, no. 8, pp. 1211–1216, 2005. 47

[155] G. Welch, G. Bishop, *et al.*, "An introduction to the kalman filter," 1995. 55

[156] R. S. Bucy, "Linear and nonlinear filtering," *Proceedings of the IEEE*, vol. 58, no. 6, pp. 854–864, 1970. 57

[157] S.J.Julier and J.K.Uhlmann, "A new extension of the kalman filter to nonlinear system," 1997. 58

[158] S. Julier, J. Uhlmann, and H. F. Durrant-Whyte, "A new method for the nonlinear transformation of means and covariances in filters and estimators," *IEEE Transactions on automatic control*, vol. 45, no. 3, pp. 477–482, 2000. 58

[159] R. Van Der Merwe, "Sigma-point kalman filters for probabilistic inference in dynamic state-space models," 2004. 58, 59, 60

[160] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, 2004. 59

[161] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, pp. 174–188, Feb 2002. 62

[162] A. Doucet and A. M. Johansen, "A tutorial on particle filtering and smoothing: Fifteen years later," *Handbook of nonlinear filtering*, vol. 12, no. 656-704, p. 3, 2009. 62, 63

[163] N. J. Gordon, D. J. Salmond, and A. F. Smith, "Novel approach to nonlinear/non-gaussian bayesian state estimation," *IEE Proceedings F (Radar and Signal Processing)*, vol. 140, no. 2, pp. 107–113, 1993. 62

[164] A. Doucet, N. De Freitas, and N. Gordon, "An introduction to sequential monte carlo methods," in *Sequential Monte Carlo methods in practice*, pp. 3–14, Springer, 2001. 63

[165] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking," *IEEE Transactions on signal processing*, vol. 50, no. 2, pp. 174–188, 2002. 64

[166] G. won You, S. Park, and D. Oh, "Real-time state-of-health estimation for electric vehicle batteries: A data-driven approach," *Applied Energy*, vol. 176, pp. 92–103, 2016. 77

[167] C. Hu, G. Jain, C. Schmidt, C. Strief, and M. Sullivan, "Online estimation of lithium-ion battery capacity using sparse bayesian learning," *Journal of Power Sources*, vol. 289, pp. 105–113, 2015. 77

[168] M. Eisen, C. Zhang, L. F. Chamon, D. D. Lee, and A. Ribeiro, "Learning optimal resource allocations in wireless systems," *arXiv preprint arXiv:1807.08088*, 2018. 79

[169] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural networks*, vol. 2, no. 5, pp. 359–366, 1989. 79

[170] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, and F. E. Alsaadi, "A survey of deep neural network architectures and their applications," *Neurocomputing*, vol. 234, pp. 11–26, 2017. 81

[171] R. Reed and R. J. MarksII, *Neural smithing: supervised learning in feedforward artificial neural networks*. Mit Press, 1999. 84

[172] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014. 84

[173] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016. 84

[174] J. Yang, M. N. Nguyen, P. P. San, X. Li, and S. Krishnaswamy, "Deep convolutional neural networks on multichannel time series for human activity recognition.," in *Ijcai*, vol. 15, pp. 3995–4001, 2015. 85

[175] Z. Cui, W. Chen, and Y. Chen, "Multi-scale convolutional neural networks for time series classification," *arXiv preprint arXiv:1603.06995*, 2016. 85

[176] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering*, vol. 22, no. 10, pp. 1345–1359, 2009. 86

[177] M. M. Ghazi, B. Yanikoglu, and E. Aptoula, "Plant identification using deep neural networks via optimization of transfer learning parameters," *Neurocomputing*, vol. 235, pp. 228–235, 2017. 86

[178] W. Zhou, S. Newsam, C. Li, and Z. Shao, "Learning low dimensional convolutional neural networks for high-resolution remote sensing image retrieval," *Remote Sensing*, vol. 9, no. 5, p. 489, 2017. 86

[179] Y. Tan and G. Zhao, "Transfer learning with long short-term memory network for state-of-health prediction of lithium-ion batteries," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 10, pp. 8723–8731, 2020. 87

[180] Y. Che, Z. Deng, X. Lin, L. Hu, and X. Hu, "Predictive battery health management with transfer learning and online model correction," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 2, pp. 1269–1277, 2021. 87

[181] S. Anwar, K. Hwang, and W. Sung, "Structured pruning of deep convolutional neural networks," *ACM Journal on Emerging Technologies in Computing Systems (JETC)*, vol. 13, no. 3, pp. 1–18, 2017. 87

[182] Z. Chen, J. Lin, S. Liu, Z. Chen, W. Li, J. Zhao, and W. Yan, "Exploiting weight-level sparsity in channel pruning with low-rank approximation," in *2019 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1–5, IEEE, 2019. 88

[183] S. Han, J. Pool, J. Tran, and W. Dally, "Learning both weights and connections for efficient neural network," in *Advances in neural information processing systems*, pp. 1135–1143, 2015. 88

[184] X. Dong, S. Chen, and S. Pan, "Learning to prune deep neural networks

via layer-wise optimal brain surgeon," in *Advances in Neural Information Processing Systems*, pp. 4857–4867, 2017. 88

[185] H. Hu, R. Peng, Y.-W. Tai, and C.-K. Tang, "Network trimming: A data-driven neuron pruning approach towards efficient deep architectures," *arXiv preprint, 2016, arXiv:1607.03250*, 2016. 88

[186] A. Bondarenko, A. Borisov, and L. Aleksejeva, "Neurons vs weights pruning in artificial neural networks," in *Proceedings of the 10th International Scientific and Practical Conference. Volume III*, vol. 22, p. 28, 2015. 88

[187] L. Cai, J. Meng, D.-I. Stroe, G. Luo, and R. Teodorescu, "An evolutionary framework for lithium-ion battery state of health estimation," *Journal of Power Sources*, vol. 412, pp. 615–622, 2019. 89

[188] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, *et al.*, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998. 92

[189] F.-F. Li, A. Karpathy, and J. Johnson, "Cs231n: Convolutional neural networks for visual recognition," *University Lecture*, 2015. 93

[190] Y. Wang, K. Li, S. Gan, and C. Cameron, "Missing data imputation with ols-based autoencoder for intelligent manufacturing," *IEEE Transactions on Industry Applications*, vol. 55, no. 6, pp. 7219–7229, 2019. 95

[191] K. A. Severson, P. M. Attia, N. Jin, N. Perkins, B. Jiang, Z. Yang, M. H. Chen, M. Aykol, P. K. Herring, D. Fraggedakis, *et al.*, "Data-driven prediction of battery cycle life before capacity degradation," *Nature Energy*, vol. 4, no. 5, pp. 383–391, 2019. 99

[192] M. E. V. Team, "A guide to understanding battery specifications," Dec 2008. 99

[193] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 249–256, 2010. 105

[194] M. Nascimento, S. Novais, M. S. Ding, M. S. Ferreira, S. Koch, S. Passerini, and J. L. Pinto, "Internal strain and temperature discrimination with optical fiber hybrid sensors in li-ion batteries," *Journal of Power Sources*, vol. 410, pp. 1–9, 2019. 118, 120

[195] E. Schulz, M. Speekenbrink, and A. Krause, "A tutorial on gaussian process regression: Modelling, exploring, and exploiting functions," *Journal of Mathematical Psychology*, vol. 85, pp. 1–16, 2018. 124

[196] G. Ozcan, M. Pajovic, Z. Sahinoglu, Y. Wang, P. V. Orlik, and T. Wada, "Online battery state-of-charge estimation based on sparse gaussian process regression," in *2016 IEEE Power and Energy Society General Meeting (PESGM)*, pp. 1–5, IEEE, 2016. 126

[197] K. Liu, K. Li, Q. Peng, and C. Zhang, "A brief review on key technologies in the battery management system of electric vehicles," *Frontiers of Mechanical Engineering*, pp. 1–18, 2018.

[198] A. Borovykh, S. Bohte, and C. W. Oosterlee, "Conditional time series forecasting with convolutional neural networks," *arXiv preprint arXiv:1703.04691*, 2017.

[199] M. U. Cuma and T. Koroglu, "A comprehensive review on estimation strategies used in hybrid and battery electric vehicles," *Renewable and Sustainable Energy Reviews*, vol. 42, pp. 517–531, 2015.

[200] N. Watrin, B. Blunier, and A. Miraoui, "Review of adaptive systems for lithium batteries state-of-charge and state-of-health estimation," in *Trans-*

portation Electrification Conference and Expo (ITEC), 2012 IEEE, pp. 1–6, IEEE, 2012.

[201] M. Hannan, M. Lipu, A. Hussain, and A. Mohamed, "A review of lithium-ion battery state of charge estimation and management system in electric vehicle applications: Challenges and recommendations," *Renewable and Sustainable Energy Reviews*, vol. 78, pp. 834–854, 2017.

[202] S. J. Julier and J. K. Uhlmann, "A general method for approximating nonlinear transformations of probability distributions," *Robotics Research Group, Department of Engineering Science, University of Oxford, Oxford, OC1 3PJ United Kingdom, Tech. Rep*, 1996.

[203] S. Haykin, *Kalman filtering and neural networks*, vol. 47. John Wiley & Sons, 2004.

[204] G. Karaduman, M. Karakose, and E. Akin, "Deep learning based arc detection in pantograph-catenary systems," in *Electrical and Electronics Engineering (ELECO), 2017 10th International Conference on*, pp. 904–908, IEEE, 2017.

[205] H. Hu, B. Tang, X. Gong, W. Wei, and H. Wang, "Intelligent fault diagnosis of the high-speed train with big data based on deep neural networks," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 4, pp. 2106–2116, 2017.

[206] T. de Bruin, K. Verbert, and R. Babuška, "Railway track circuit fault diagnosis using recurrent neural networks," *IEEE transactions on neural networks and learning systems*, vol. 28, no. 3, pp. 523–533, 2017.

[207] X. Gibert, V. M. Patel, and R. Chellappa, "Deep multitask learning for railway track inspection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 1, pp. 153–164, 2017.

[208] J. Chen, Z. Liu, H. Wang, A. Núñez, and Z. Han, "Automatic defect detection of fasteners on the catenary support device using deep convolutional neural network," *IEEE Transactions on Instrumentation and Measurement*, vol. 67, no. 2, pp. 257–269, 2018.

[209] S. Faghih-Roohi, S. Hajizadeh, A. Núñez, R. Babuska, and B. De Schutter, "Deep convolutional neural networks for detection of rail surface defects.," in *IJCNN*, pp. 2584–2589, 2016.

[210] A. Onat, P. Voltr, and M. Lata, "An unscented kalman filter-based rolling radius estimation methodology for railway vehicles with traction," *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*, vol. 232, no. 6, pp. 1686–1702, 2018.

[211] L. Kou, Y. Qin, X. Zhao, and Y. Fu, "Integrating synthetic minority oversampling and gradient boosting decision tree for bogie fault diagnosis in rail vehicles," *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*, p. 0954409718795089, 2018.

[212] I. Aydin, M. Karaköse, and E. Akin, "A new contactless fault diagnosis approach for pantograph-catenary system," in *MECHATRONIKA, 2012 15th International Symposium*, pp. 1–6, IEEE, 2012.

[213] W. Qiuting, J. Yinzhu, and L. Yunhao, "State of health estimation for lithium-ion battery based on d-ukf," *Int. J. Hybrid Inf. Technol*, vol. 8, pp. 55–70, 2015.

[214] L. Ungurean, G. Cârstoiu, M. V. Micea, and V. Groza, "Battery state of health estimation: a structured review of models, methods and commercial devices," *International Journal of Energy Research*, vol. 41, no. 2, pp. 151–181, 2017.

[215] J. Kim, S. Lee, and B. Cho, "Complementary cooperation algorithm based on dekf combined with pattern recognition for soc/capacity estimation and soh prediction," *IEEE Transactions on Power Electronics*, vol. 27, no. 1, pp. 436–451, 2012.

[216] A. Eddahech, O. Briat, N. Bertrand, J.-Y. Delétage, and J.-M. Vinassa, "Behavior and state-of-health monitoring of li-ion batteries using impedance spectroscopy and recurrent neural networks," *International Journal of Electrical Power & Energy Systems*, vol. 42, no. 1, pp. 487–494, 2012.

[217] G. L. Plett, "Dual and joint ekf for simultaneous soc and soh estimation," in *Proceedings of the 21st Electric Vehicle Symposium (EVS21), Monaco*, pp. 1–12, 2005.

[218] A. Sidhu, A. Izadian, and S. Anwar, "Adaptive nonlinear model-based fault diagnosis of li-ion batteries," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 2, pp. 1002–1011, 2015.

[219] Z. Chen *et al.*, "Bayesian filtering: From kalman filters to particle filters, and beyond," *Statistics*, vol. 182, no. 1, pp. 1–69, 2003.

[220] H. Orlande, M. Colaço, G. Dulikravich, F. Vianna, W. da Silva, H. da Fonseca, and O. Fudym, "Tutorial 10 kalman and particle filters," *Advanced spring school: Thermal measurements and inverse techniques*, p. 5, 2011.

[221] T. Orlowska-Kowalska and K. Szabat, "Neural-network application for mechanical variables estimation of a two-mass drive system," *IEEE Transactions on Industrial Electronics*, vol. 54, no. 3, pp. 1352–1364, 2007.

[222] R. Liu, G. Meng, B. Yang, C. Sun, and X. Chen, "Dislocated time series convolutional neural architecture: An intelligent fault diagnosis approach

for electric machine," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 3, pp. 1310–1320, 2017.

[223] Z. Gao, C. S. Chin, J. H. K. Chiew, J. Jia, and C. Zhang, "Design and implementation of a smart lithium-ion battery system with real-time fault diagnosis capability for electric vehicles," *Energies*, vol. 10, no. 10, 2017.

[224] R. Xiong, Q. Yu, and W. Shen, "Review on sensors fault diagnosis and fault-tolerant techniques for lithium ion batteries in electric vehicles," in *2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, pp. 406–410, IEEE, 2018.

[225] T. Seo, A. M. Bayen, T. Kusakabe, and Y. Asakura, "Traffic state estimation on highway: A comprehensive survey," *Annual Reviews in Control*, vol. 43, pp. 128–151, 2017.

[226] A. Singh, A. Izadian, and S. Anwar, "Fault diagnosis of li-ion batteries using multiple-model adaptive estimation," in *Industrial Electronics Society, IECON 2013-39th Annual Conference of the IEEE*, pp. 3524–3529, IEEE, 2013.

[227] A. Singh, A. Izadian, and S. Anwar, "Nonlinear model based fault detection of lithium ion battery using multiple model adaptive estimation," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 8546–8551, 2014.

[228] C. Yi, J. Lin, and L. Liu, "An online state estimation approach for railway axle bearing based on eemd and entropy," 2016.

[229] D. Veillard, F. Mailly, and P. Fraisse, "Ekf-based state estimation for train localization," in *SENSORS, 2016 IEEE*, pp. 1–3, IEEE, 2016.

[230] P. Li, R. Goodall, and V. Kadirkamanathan, "Parameter estimation of

railway vehicle dynamic model using rao-blackwellised particle filter," in *European Control Conference (ECC), 2003*, pp. 2384–2389, IEEE, 2003.

[231] Y. Zhao, B. Liang, and S. Iwnicki, "Friction coefficient estimation using an unscented kalman filter," *Vehicle System Dynamics*, vol. 52, no. sup1, pp. 220–234, 2014.

[232] J. Chen, Z. Liu, H. Wang, and K. Liu, "High-speed railway catenary components detection using the cascaded convolutional neural networks," in *Imaging Systems and Techniques (IST), 2017 IEEE International Conference on*, pp. 1–6, IEEE, 2017.

[233] Z. Liu, W. Liu, and Z. Han, "A high-precision detection approach for catenary geometry parameters of electrical railway," *IEEE Trans. Instrum. Meas.*, vol. 66, no. 7, pp. 1798–1808, 2017.

[234] Y. Zhao, Z. H. Guo, and J. M. Yan, "Vibration signal analysis and fault diagnosis of bogies of the high-speed train based on deep neural networks.," *Journal of Vibroengineering*, vol. 19, no. 4, 2017.

[235] Y. Fu, D. Huang, N. Qin, K. Liang, and Y. Yang, "High-speed railway bogie fault diagnosis using lstm neural network," in *2018 37th Chinese Control Conference (CCC)*, pp. 5848–5852, IEEE, 2018.

[236] K. Liang, N. Qin, D. Huang, and Y. Fu, "Convolutional recurrent neural network for fault diagnosis of high-speed train bogie," *Complexity*, vol. 2018, 2018.

[237] Y. Zheng, Q. Liu, E. Chen, Y. Ge, and J. L. Zhao, "Time series classification using multi-channels deep convolutional neural networks," in *International Conference on Web-Age Information Management*, pp. 298–310, Springer, 2014.

[238] M. Jesussek and K. Ellermann, "Fault detection and isolation for a non-linear railway vehicle suspension with a hybrid extended kalman filter," *Vehicle system dynamics*, vol. 51, no. 10, pp. 1489–1501, 2013.

[239] Z. Zhang, B. Xu, L. Ma, and S. Geng, "Parameter estimation of a railway vehicle running bogie using extended kalman filter," in *Control Conference (CCC), 2014 33rd Chinese*, pp. 3393–3398, IEEE, 2014.

[240] S. Dey, S. Mohon, P. Pisu, and B. Ayalew, "Sensor fault detection, isolation, and estimation in lithium-ion batteries," *IEEE Transactions on Control Systems Technology*, vol. 24, no. 6, pp. 2141–2149, 2016.

[241] C. Wu, C. Zhu, Y. Ge, and Y. Zhao, "A review on fault mechanism and diagnosis approach for li-ion batteries," *Journal of Nanomaterials*, vol. 2015, p. 8, 2015.

[242] R. Xiong, Q. Yu, W. Shen, C. Lin, and F. Sun, "A sensor fault diagnosis method for a lithium-ion battery pack in electric vehicles," *IEEE Transactions on Power Electronics*, 2019.

[243] Z. Liu and H. He, "Model-based sensor fault diagnosis of a lithium-ion battery in electric vehicles," *Energies*, vol. 8, no. 7, pp. 6509–6527, 2015.

[244] Z. Liu and H. He, "Sensor fault detection and isolation for a lithium-ion battery pack in electric vehicles using adaptive extended kalman filter," *Applied Energy*, vol. 185, pp. 2033–2044, 2017.

[245] C. Vidal, M. Haußmann, D. Barroso, P. M. Shamsabadi, A. Biswas, E. Chemali, R. Ahmed, and A. Emadi, "Hybrid energy storage system state-of-charge estimation using artificial neural network for micro-hybrid applications," in *2018 IEEE Transportation Electrification Conference and Expo (ITEC)*, pp. 1075–1081, IEEE, 2018.

[246] Y. Li, K. Li, S. Li, and Y. Li, "FRA and EKF Based State of Charge Estimation of Zinc-Nickel Single Flow Batteries," in *Advances in Green Energy Systems and Smart Grid*, pp. 183–191, Springer, 2018.

[247] B. Xia, Z. Sun, R. Zhang, and Z. Lao, "A cubature particle filter algorithm to estimate the state of the charge of lithium-ion batteries based on a second-order equivalent circuit model," *Energies*, vol. 10, no. 4, p. 457, 2017.

[248] J. Xu and B. Cao, "Battery management system for electric drive vehicles: Modelling, state estimation and balancing," *New Applications of Electric Drives, 1st ed.; Chomat, M., Ed*, pp. 87–113, 2015.

[249] H. He, R. Xiong, and H. Guo, "Online estimation of model parameters and state-of-charge of lifepo4 batteries in electric vehicles," *Applied Energy*, vol. 89, no. 1, pp. 413–420, 2012.

[250] 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.

[251] Y. Wang, C. Zhang, and Z. Chen, "A method for state-of-charge estimation of lifepo4 batteries at dynamic currents and temperatures using particle filter," *Journal of power sources*, vol. 279, pp. 306–311, 2015.

[252] Z. Chen, H. Zhao, X. Shu, Y. Zhang, J. Shen, and Y. Liu, "Synthetic stage of charge estimation for lithium-ion batteries based on long short-term memory network modeling and adaptive h-infinity filter," *Energy*, p. 120630, 2021.

[253] R. Xiong, F. Sun, X. Gong, and H. He, "Adaptive state of charge estimator

for lithium-ion cells series battery pack in electric vehicles," *Journal of power sources*, vol. 242, pp. 699–713, 2013.

[254] K. Li, J.-X. Peng, and E.-W. Bai, "A two-stage algorithm for identification of nonlinear dynamic systems," *Automatica*, vol. 42, no. 7, pp. 1189–1197, 2006.