



**Politecnico  
di Torino**

Master's Degree in Computer Engineering  
Master's Thesis

**Battery state of health and state of charge estimation:  
comparison between classical and  
machine learning techniques**

**Supervisors:**

Prof. Carlo NOVARA

Prof. Stefano CARABELLI

**Candidate:**

Domenico CARLUCCI

MAT. 254366

A.Y. 2020/2021  
Session October 2021



# Abstract

The topics covered in this thesis work are related to the field of electric vehicles (EVs) optimization. Thanks to their attractive properties, the majority of EVs adopt lithium-ion batteries as main energy source introducing new challenges in the car manufacturer's world. In order to guarantee the optimal management and the safety of the operations performed on the battery, a vehicle subsystem, called Battery Management System (BMS), has to estimate the state of the battery through two fundamental parameters: the State of Charge (SoC) and the State of Health (SoH). Precisely knowing these quantities in a real driving context is actually a challenging task and, for the remarkable industrial value, it has become a hot research topic in the last decade.

In the first part of the thesis report the key aspects of the problem are introduced with a top-down approach, and a state of the art analysis is performed by describing the most relevant SoC and SoH estimation approaches that exist in the literature.

In the second part, the thesis work is presented: in the context of MATLAB environment, a first principles approach (classical) estimation technique, based on Extended Kalman Filter, and a machine learning approach are developed and validated under simulation.

# Table of contents

<b>Abstract</b>	<b>I</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 An overview on the main topics</b>	<b>3</b>
2.1 Battery Management System . . . . .	5
2.2 Battery pack, modules and cells . . . . .	7
2.3 Lithium-ion cell electrochemical inspection . . . . .	10
2.4 Charging/discharging operations . . . . .	12
2.5 State of Charge and State of Health . . . . .	17
<b>3 Available approaches</b>	<b>22</b>
3.1 Data sources . . . . .	24
3.2 First principles approaches . . . . .	28
3.2.1 Coulomb Counter . . . . .	28
3.2.2 Equivalent Circuit Model based methods . . . . .	30
3.3 Data-driven approaches . . . . .	52
3.3.1 Open Circuit Voltage map . . . . .	52
3.3.2 Black-Box . . . . .	54
3.4 Summary . . . . .	64
<b>4 Application scenario</b>	<b>65</b>
4.1 Dynamic simulator . . . . .	67
4.2 Battery Model . . . . .	68
4.2.1 SoH ground truth evaluation . . . . .	72
4.2.2 Coulomb efficiency computation . . . . .	74
4.2.3 Open Circuit Voltage curve identification . . . . .	76
4.2.4 Equivalent Circuit Model identification . . . . .	81
4.2.5 Equivalent Circuit Model comparison . . . . .	87

<b>5</b>	<b>Extended Kalman Filter</b>	<b>88</b>
5.1	Initial conditions and EKF tuning . . . . .	96
5.2	Test results . . . . .	102
<b>6</b>	<b>Neural Network</b>	<b>111</b>
6.1	Dataset . . . . .	114
6.2	Training . . . . .	116
6.3	Results . . . . .	118
<b>7</b>	<b>Conclusion</b>	<b>121</b>
<b>A</b>	<b>Simulink schemes</b>	<b>123</b>
	<b>Bibliography</b>	<b>128</b>

# Chapter 1

## Introduction

The topics covered in this thesis work are related to the field of electric vehicles (EVs) optimization. Thanks to their attractive properties, the majority of EVs adopt lithium-ion batteries as main energy source introducing new challenges in the car manufacturer's world. In order to guarantee the optimal management and the safety of the operations performed on the battery, a vehicle subsystem, called Battery Management System (BMS), has to estimate the state of the battery through two fundamental parameters: the State of Charge (SoC) and the State of Health (SoH). Precisely knowing these quantities in a real driving context is actually a challenging task and, for the remarkable industrial value, it has become a hot research topic in the last decade.

The thesis work consists in developing and comparing two different estimation schemes for estimating SoC and SoH under simulation by using a dynamic simulator of a real target system.

The document is organized in the following way:

- Chapter 2 gradually introduces to the main topics with a top-down approach.
- Chapter 3 describes the most relevant estimation approaches that exist in the literature
- Chapter 4 presents the work and the adopted models

- Chapter 5 and Chapter 6 present the work with respect to the proposed estimation schemes
- Chapter 7 concludes the document by summarizing the entire work and presenting results and discussions

# Chapter 2

## An overview on the main topics

The transition from internal combustion engine vehicles toward electric vehicles contributes in solving some problems but introduces new challenges.

The main reason that can be attributed to this transition is relative to air pollution. The exhausted gases in fact, containing a lot of noxious elements (Particulate Matter, Volatile Organic Compounds, nitrogen oxides, carbon monoxide, sulfur dioxide and greenhouse gases) not only introduce human health diseases (respiratory infections, heart disease and lung cancer) but contribute to exacerbate the global warming process and more in general the climate change phenomena. To deal with these problems, based on the common objectives specified in the various international treaties (Kyoto protocol, Doha amendment, Paris agreement), Euro standards have been introduced with the aim of controlling vehicles emissions. The restrictions introduced by the standards become more and more restrictive as years pass by, bringing car manufacturers to find new solutions to the engineering challenges that arise from the respect of these norms.

By looking forward in the near future, another crucial phenomenon is the overpopulation. Based on the statistics made by United Nations, the demography growth trend shows that in the next 30 years the world population might reach about 10 billion people, and at the end of the century even 12.3 billion people [1]. This phenomenon opens the door to unpredictable scenarios and requires to be managed properly right now. The growth in population not only aggravates the impact of

human activities all around the world on the environment, as the Global Footprint Network Organization suggests [2, 3], but from a social perspective introduce the exigence to find a better way to make people and goods move.

A partial solution to these problems can be found in the adoption of full electric vehicles. Even though, from an ecological perspective, their production is still a problem, their usage allow to eliminate air pollution problem and can contribute to the realization of a real transportation revolution. In the optic of addressing overpopulation problems, electric cars drives in the right direction since their adoption can be inserted in the autonomous guide context making the transportation of people and goods more efficient and optimized.

The gradual withdrawal from fossil fuel is compensated by the adoption of a different source of energy, that is retrieved from lithium-ion batteries. These kind of batteries are taken into account since they have high density energy, light weight, long time span and low production cost, and thanks to their attractive properties are placed into the majority of electric vehicles. In order to guarantee their correct usage, an opportune vehicle subsystem, the Battery Management System (BMS), takes care to manage the operations performed on the battery. To do so, the BMS has to estimate the state of the battery by estimating two fundamental parameters: the State of Charge (SoC) and the State of Health (SoH). Knowing precisely these quantities in a real driving context is actually a difficult task but it allow to enhance the vehicle performances and to increase the vehicle driving range. For this reason SoC and SoH estimation problem has a remarkable industrial value and has become a hot research topic in the last decade.

In the next paragraphs the main aspects regarding the BMS and the battery are deepened until, in paragraph 2.5, the fundamental parameters of interest, namely the SoC and the SOH, are introduced.

## 2.1 Battery Management System

One important subsystem that can be found inside an electric vehicle is the Battery Management System (BMS). It is an embedded system that is designed to properly manage the particular Energy Storage device, which is typically a rechargeable battery, that is adopted as a source of energy for the entire vehicle. The main tasks that the BMS has to perform are:

- continuously monitor the battery by acquiring measures of voltage, current and temperature
- compute parameters that describe the state of the battery like the State of Charge (SoC) and the State of Health (SoH)
- ensure the safety by guaranteeing that every operation performed on the battery lies in the Safe Operation Area (SOA), so by avoiding: over-charge, over-discharge, over-voltage (during charge) and thermal runaway

The below figure illustrates how the main elements interact to each other by depicting a traditional BMS conceptual schema:

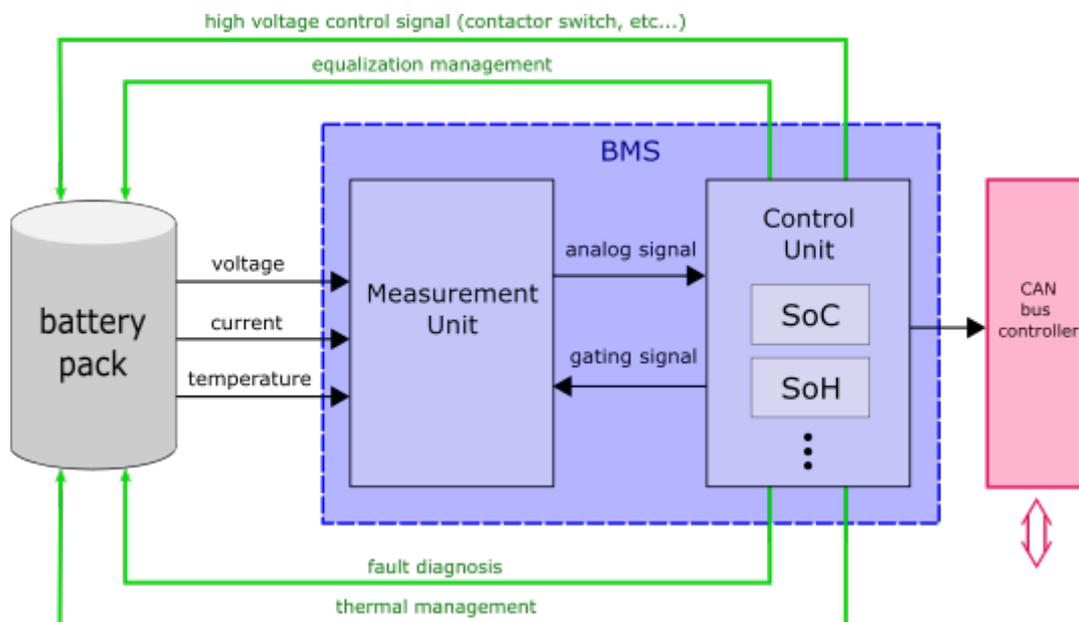


Figure 2.1: Example of a typical BMS conceptual schema

The main component of the BMS is the control unit that run special algorithms designed to compute the battery state parameters on the basis of the acquired measures gathered by the measurement unit. The computed parameters are then used to manage the battery and forwarded toward external subsystems through the network bus. More details about each functionality are discussed from the battery pack perspective in the next paragraph while the main battery state parameters (State of Charge and State of Health) are discussed in section 2.5.

## 2.2 Battery pack, modules and cells

The battery pack represents the energy storage device of an electric vehicle and is used to stock energy (typically in the form of electrochemical potential) with the purpose of providing it to the vehicle motor and other subsystems. It is managed by the BMS and depending on the choices of the manufacturer, presents a complex inner structure since is composed by many subcomponents that contribute to provide its overall functionalities. Among the possible components, the main ones consist in battery cells modules, thermal conditioning system, electric boards and other auxiliary devices that allow to interface it with an external load/charger. In order to meet the exigence in terms of voltage, capacity and current, that the battery has to provide, the number of cells modules, which are connected in series, is variable and is one of the design aspects. Moreover, every single module is composed by many cells, arranged in a parallel fashion, that represent the very atomic energy elements of the battery. Modules embed some electronics that help the BMS to monitor the state of the cells by acquiring measures of voltage, current and temperature thanks to the presence of corresponding sensors. In this context the BMS is able to act a control action in order to ensure the safety of the charge/discharge operations performed over the cells and to control the temperature thanks to the presence of a cooling/heating system. The over-charge and over-discharge control is achieved through the cell equalization mechanism: the BMS on the basis of the state of charge of the cells can actively or passively redistribute the charge level from more charged cell to the less charged one in order to have an homogenous charge distribution over all the cells. The difference between actively and passively equalization consist in the electronic devices employs in such operation. In the passive paradigm, the energy of the most charged cells is dissipated through Joule effect on a passive electric element (resistor) while in the active one the energy is distributed over the cells through an active element (transistor).

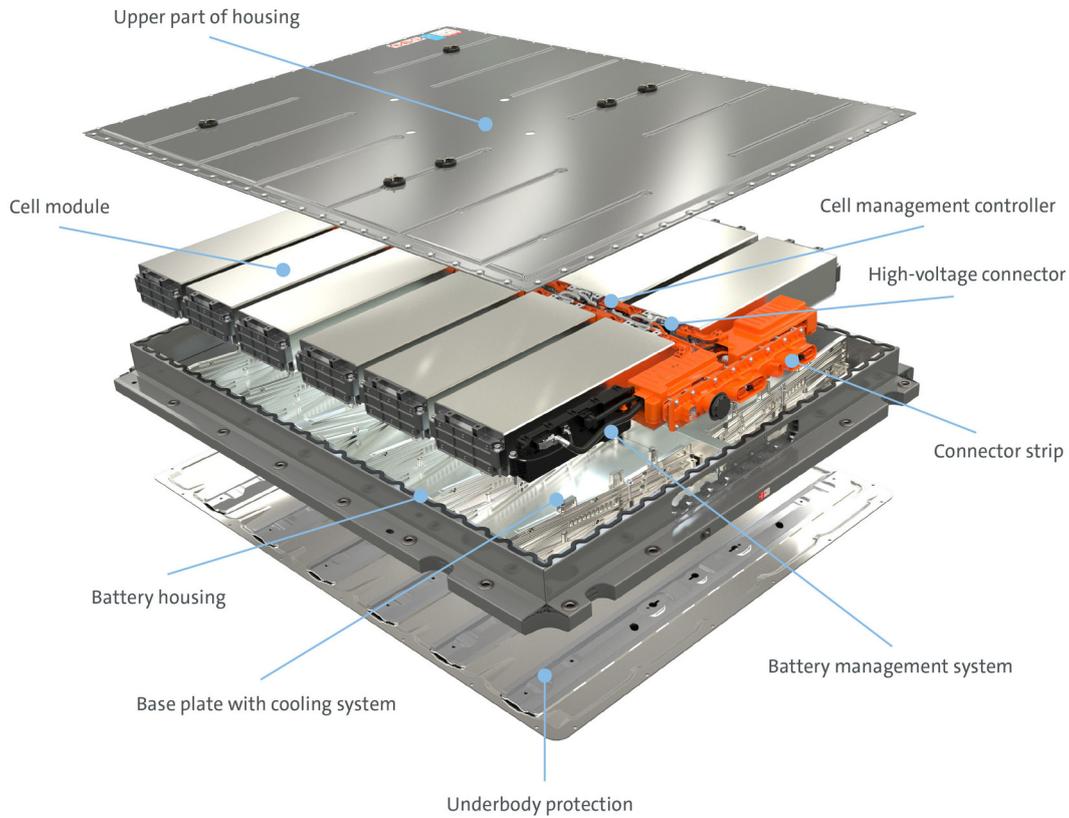


Figure 2.2: Example of a complete battery system regarding an inspection of Volkswagen Modular Electric drive matrix Battery system (MEB) platform [4]

According to the different chemical constituent nature, there exist many kind of cells that can be adopted as building element for the vary modules. Nowadays, lithium-ion batteries are considered the leading battery typology for their attractive properties since they provide high-density energy, long-life span a low production cost.

Depending on the specific battery cell product, the manufacturer can provide different type of information (often in form of tables and graphs) but in general they consist in nominal, electrical, mechanical and safety specifications. The following table, as an example, illustrates the nominal characteristics of a LG Chem lithium-ion battery cell datasheet information:

**1. General Information**

1.1 Scope

This product specification defines the requirements of the rechargeable lithium ion battery to be supplied to the customer by LG Chem.

1.2 Product classification : Cylindrical rechargeable lithium ion battery

1.3 Model name : INR 18650 M29

**2. Nominal Specification**

Item	Specification
2.1 Nominal Capacity	2,850mAh (0.2C, 2.50V discharge)
2.2 Minimum Capacity	2,750mAh (0.2C, 2.50V discharge)
2.3 Norminal Energy	10.5 Wh (0.2C, 2.50V discharge)
2.4 Nominal Voltage	3.67V (0.2C, 2.50V discharge)
2.5 Standard Charge	Method : CC-CV
	Charging Voltage : 4.20V
	Charging Current : 0.5C (1,375mA)
	Cut-off Current : 50mA
2.6 Max. Charge Current	1.0C (2,750mA)
2.7 Standard Discharge	Method : CC
	Discharging Current : 0.2C (550mA)
	Cut-off Voltage : 2.50V
2.8 Max. Discharge Current	6,000mA(for continuous discharge) 10,000mA(not for continuous discharge)
2.9 Weight	45.00 g
2.10 Operating Temperature, Charge	-10 ~ 0℃ : 0.1C(275mA)
	0 ~ 50℃
2.11 Operating Temperature, Discharge	-30 ~ 60℃ (Cut-off Temp. 70℃)
2.12 Storage Temperature (30% SOC)	1month : -30~60℃
	3months : -30~45℃
	1year : -30~24℃

Figure 2.3: A schreenshot about a section of a battery cell datasheet illustrating the main nominal characteristics of a INR 18650 M29 lithium-ion battery cell. [5]

## 2.3 Lithium-ion cell electrochemical inspection

The first commercial Lithium-ion battery was introduced by SONY in 1991 [6] and from that time has become one of the best way to store energy for many electronic devices and EVs. It is a particular kind of electrochemical cell that converts electric energy into chemical energy during discharging process (in this case is called electrolytic cell) and vice versa during charging process (in this case is called galvanic cell). At the simplest abstraction level, the cell is composed by a positive (anode) and negative (chathode) electrodes (active elements) and an electrolyte (passive element). The mechanisms of energy conversion, during charging and discharging processes happen through redox reactions (oxidation and reduction) at the active components. In these circumstances, electrons migrate from one electrode to the other by following an external circuit, while generated ions passes through the electrolyte reaching the other side electrode.

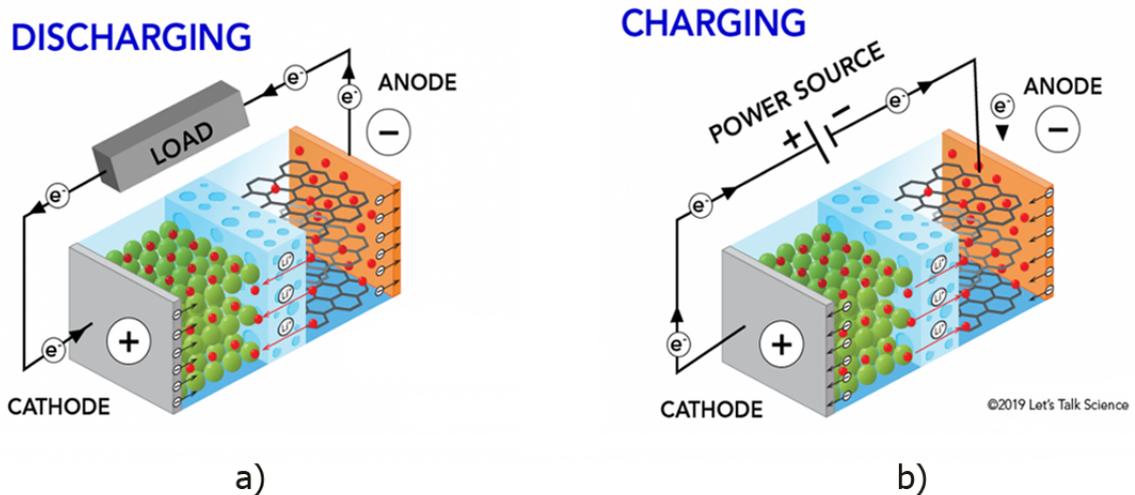


Figure 2.4: Conceptual diagram of an electrochemical cell undergoing to a discharge (a) and charge (b) processes [7, 8]

The main components of the electrochemical cell from the perspective of a Lithium-ion cell are the following:

- electrodes: are based on electrically and ionically conductive materials. The anode acts like an ion magazine and electron donor (when the cell is full

charged) and its composition is typically based on graphene. The cathode acts like an electron acceptor and (when the cell is fully discharged) stores the lithium by combining it to a metal-oxide whose elements can be Co, Ni, Fe, Mn.

- electrolytes: can be of various nature (liquid or polymeric) but the most used is based on a solution of one or many salts dissolved in one or many solvents. It allows to carry ions thanks to its high ionic conductivity.
- separator: In the case that a liquid electrolyte is adopted, a porous separator with electron insulation property is inserted in between the two electrodes in order to avoid electrodes touching each other causing a short circuit.
- current collector: Enhances the electrode by increasing the efficiency in the electrons transportation during charge and discharge processes. They do not participate in redox reactions and consist typically in good electric conductive materials like aluminium or copper.
- casing: Consists in an external enclosure that insulates the electrochemical cell by avoiding the liquid electrolyte evaporates and protects the cell from the outside world. It can be composed of plastic or metallic materials and assumes cylindrical or prismatic shapes.

## 2.4 Charging/discharging operations

A crucial aspect that has to be addressed properly regards the charging and discharging operations. In fact, an improper usage of the cells during these processes can lead to damaging the battery by triggering irreversible degrading processes in its constituent and in the worst case create dangerous situations like fires or explosions.

The reference quantity that is used to characterize the charge and discharge operation is the C-rate that indicate the amount of current employed to the related battery capacity. For instance, if a cell is stated to have a rated capacity of 2.9 Ah, a 1 C charge/discharge current rate means that to fully charge or discharge that battery a 2.9 A current is adopted and the process lasts for 1 h. Typically multiple of the rated capacity are used (nC-rates). The time necessary to fully charge or discharge the battery is inversely proportional to the nC-rate: a 4C charge/discharge rate corresponds to 15 minutes, while 0.2C (1/5C) corresponds to 5 hours.

There exist three types of charging processes performed by means of a charger device that is connected to the terminal electrodes of the cell:

1. constant current (CC): it is the simplest charging method and consist in applying a constant charging current (low value of C-rate are commonly used) to the battery cell until the cell voltage raising, reaches its upper cut-off value. The full charge state must be recognized in order to avoid over-charge issues. The recognition of the full state is obtained by means of either voltage or temperature based approaches. Some kind of batteries (ex: NiMH) when reaches its full state manifest a drop in terms of terminal voltage, so the difference in voltage potential over the time can be used in the voltage method to indicate the full charge state. On the other hand, the same principle is applied in the temperature based approach. In this case the full state charge is suggested by an increasing in temperature over time, since the energy is mostly dissipated through Joule effect when the battery is fully charged.

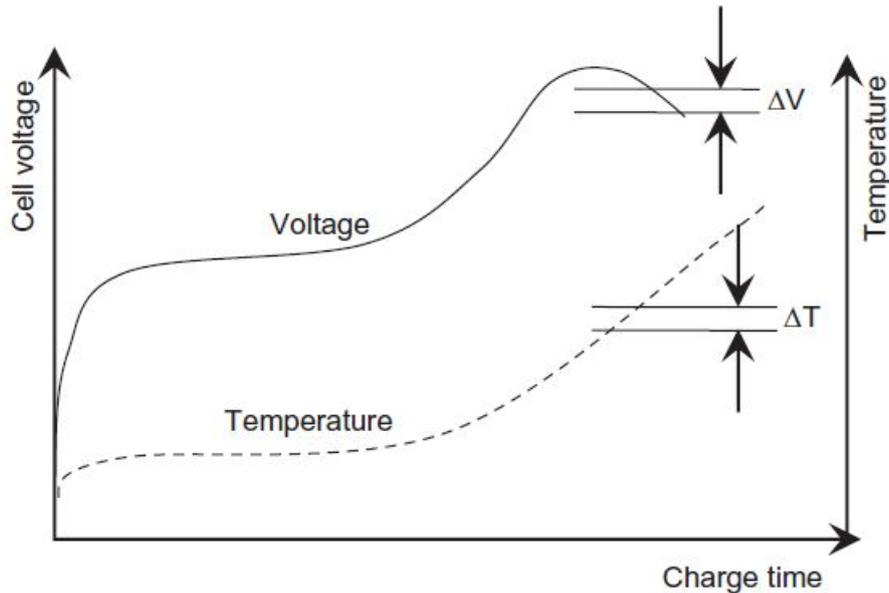


Figure 2.5: Comparison of charge control methods based on voltage and temperature approaches . [9]

2. constant voltage (CV): In the constant voltage charge method, usually the upper cut-off voltage of the battery cell (suggested by the manufacturer) is imposed allowing a maximum current to be generated to feed the battery. This method results to be less time consuming with respect to the constant charge current one.
3. constant current-constant voltage (CC-CV): It is made by mixing together constant current and constant voltage methods. The whole process, which is represented in the figure 2.6, is divided into three different phases consisting in a CC charging, followed by a CV charge and finally a rest period. In the last phase the battery undergoes to a self-discharge process characterized by a low decreasing current that ends when electrochemical equilibrium inside the cell is reached.

As mentioned before, the charging process is a delicate process and its timing must be respectful of the electrochemical timing reactions. Other important aspect regard fast charge (C-rates biggest then the rated capacity) that have to be possibly avoided

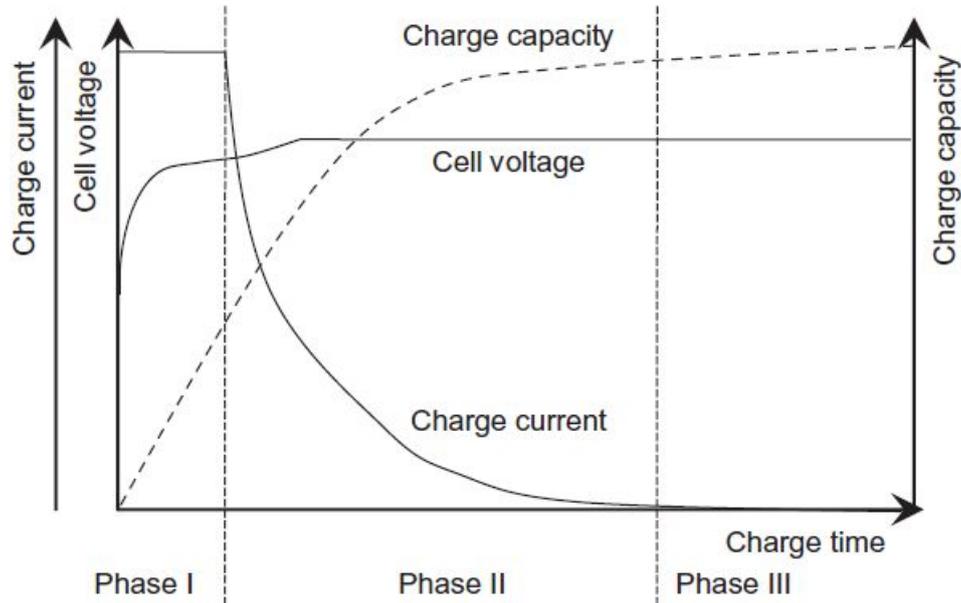


Figure 2.6: CC-CV charging approach. [10]

and the voltage applied during CV charge doesn't exceed the one suggested by the manufacturer since it can short the life time of the battery.

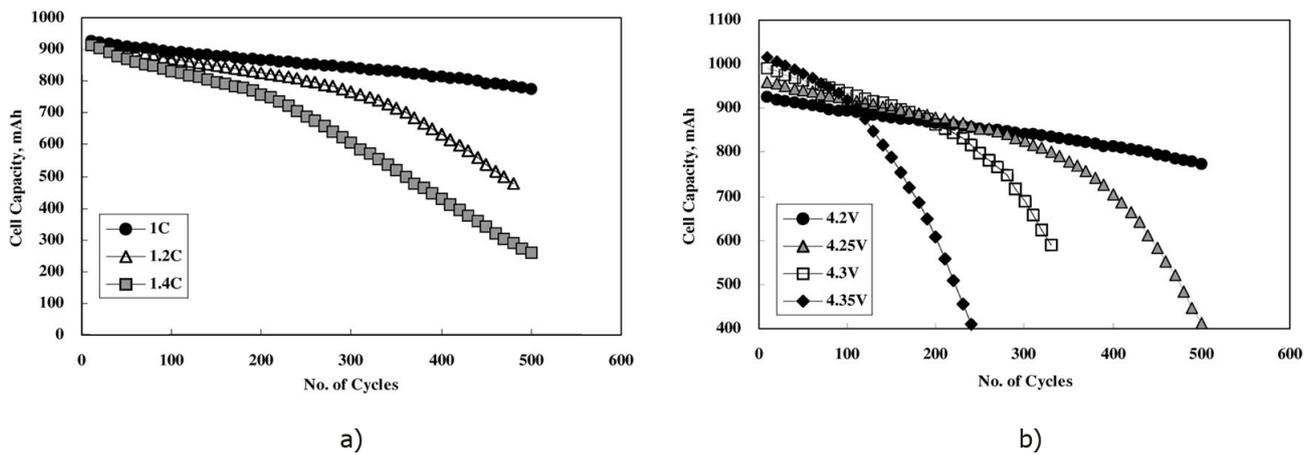


Figure 2.7: Capacity degradation induced by high C-rates (a) and over-voltage during CV (b) charge processes over different cycles. [11, 12]

When the battery cell is disconnected from any kind of external load, the measured tension between the positive and negative electrode is called Open Circuit Voltage OCV. During the discharging process, when a load is connected to the terminal of the battery cell, the voltage decreases due to internal losses. These kinds of losses are caused by different types of polarization or overpotential that occur when a current passes through the cell and consist in:

- activation polarization loss is due to the charge transfer at the electrode surface
- concentration polarization loss manifests in presence of a concentration difference of charged species between the electrode surface and the electrolyte bulk
- ohmic polarization loss is the dominant loss and is due to the internal resistance of the cell

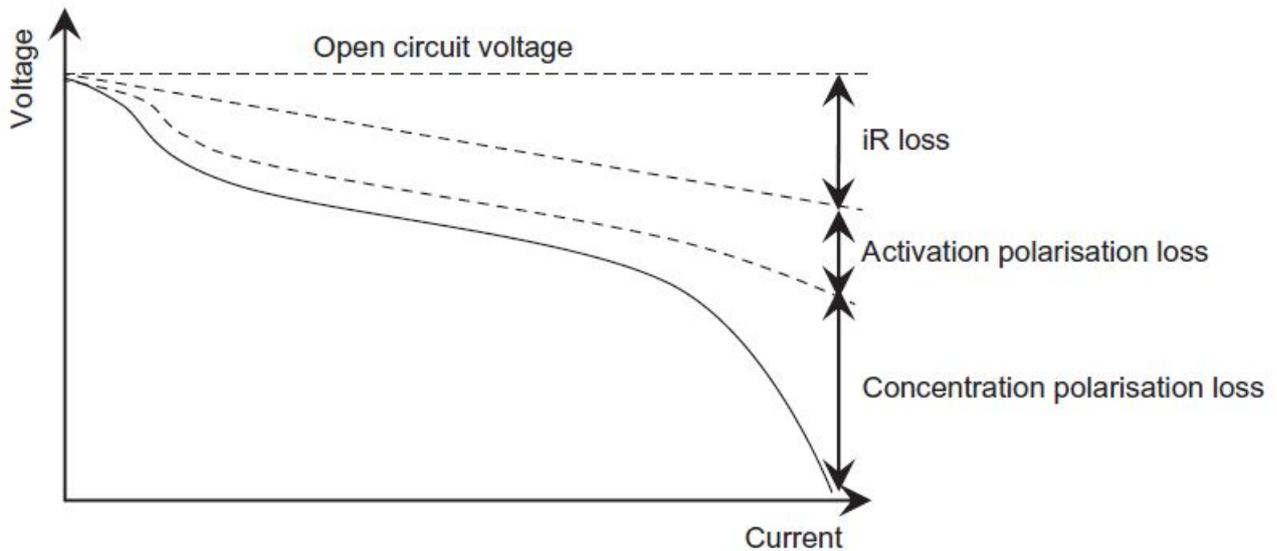


Figure 2.8: Different kind of losses during discharge. [13]

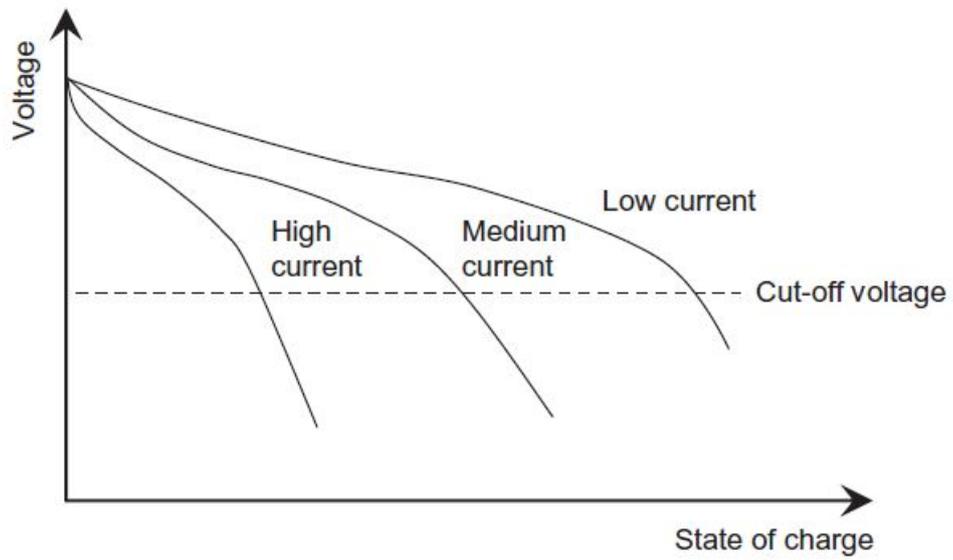


Figure 2.9: Discharge profiles at different discharges rate. [14]

## 2.5 State of Charge and State of Health

As discussed in the paragraph 2.1, the BMS by monitoring the battery has to provide information regarding its status. These information are summarized basically through two main parameters namely the State of Charge (SoC) and the State of Health (SoH).

The SoC is typically defined as the percentage of the residual capacity at a given time with respect to the maximum capacity of the battery. This concept can be formulated as following:

$$SoC(t) = \frac{Q(t)}{Q_n} \times 100\%, \quad t \geq 0 \quad (2.1)$$

The residual capacity  $Q(t)$  [A·h] corresponds to the amount of charge that has to be removed in order to bring the current battery charge state to a full discharge state. The nominal capacity  $Q_n$  [A·h] corresponds to the total amount of charge that the battery has when is in a full charge state. Often, in the literature it is referred to this quantity through its numerical complement called Depth of Discharge (DoD) that is obtained through:

$$DoD = 1 - SoC \quad (2.2)$$

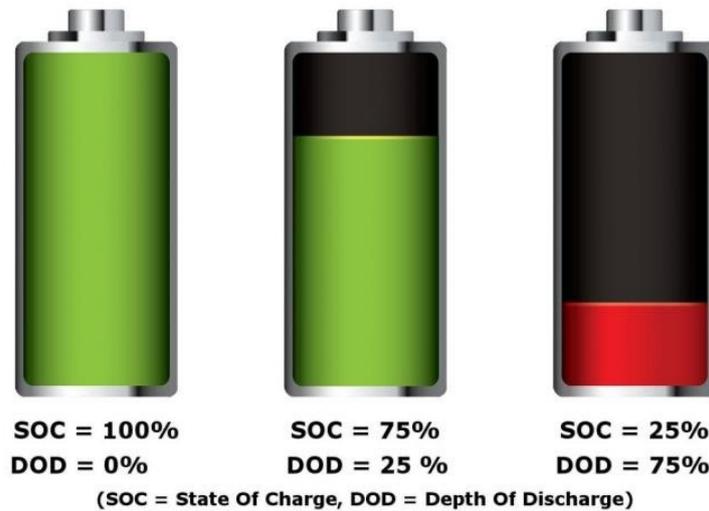


Figure 2.10: Conceptual representation of SoC and DoD [15].

The given definition (eq. 2.1) is an approximation of the real one since, in real world, SoC depends from different factors such as temperature and battery aging mechanisms. In particular, the nominal capacity tends to decrease as the battery cell ages over time.

In every traditional internal combustion engine vehicle, it is common to see the presence of a fuel gauge on the dashboard, that is the indicator of the available amount of the fuel in the tank. This device works thanks to the presence of a sensor which is able to measure the level of the remaining fuel in the tank. In an electric vehicle, a corresponding sensor, which is able to directly measure the SoC level, is not possible to realize since the SoC has complex relationships with voltage, temperature and current. For this reason the BMS has to estimate it starting from the acquired physical quantities.

In order to characterize the status of the battery from a qualitative perspective, driven by the aging effects, the other important status battery parameter, taken into account by the BMS, is the so called State of Health (SoH). It expresses, through a percentage index, the useful lifetime associated with the battery in a way that is equal to 100%, when the battery is in its begin of life (BOL), and it decreases toward 0% as the battery ages. Depending on the application for which the battery is employed, the End Of Life (EOL) corresponds to a particular threshold value of the SoH. In the automotive field the reference value for the EOL is in correspondence of the 80% SoH value.

Although the provided SOH definition is widespread, actually, there is no common agreement in the industry for the establishment of some metrics that can be used to retrieve SOH information. Based on experimental evidence, there exist different features that are sensible to the aging mechanisms, called "health indicators" (HI), that can be monitored to provide such information. Among the measurable quantities, the best HIs that can be significative to represent the SOH are the battery internal resistance and the battery capacity. According to these electrical quantities it is possible to provide different SoH definitions.

Considering the internal resistance, the SoH can be expressed as:

$$SOH = \frac{R_{int}^{EOL} - R_{int}}{R_{int}^{EOL} - R_{int}^{BOL}} \times 100\% \quad (2.3)$$

The superscript EOL and BOL indicate that the internal resistance is evaluated at the End Of Life and at the Begin Of Life respectively, while  $R_{int}$  represent the current internal resistance value. When battery ages it is observed that the internal resistance tends to increase its value and typically when the battery reaches the end of its life, it doubles. This variation is also exacerbated by the changing in the internal cell temperature, which is guided by the operating activity and by the ambient temperature, making the evaluation of the SoH a difficult task. Since in the HEV application it is important to guarantee power performance, the eq. 2.3 is frequently adopted [16].

An analogue expression is employed for the SoH definition from the perspective of the capacity:

$$SOH = \frac{Q_n - Q_n^{EOL}}{Q_n^{BOL} - Q_n^{EOL}} \times 100\% \quad (2.4)$$

The very same syntax with respect to the first expression has been employed and the capacity simply substitute the respective quantities. The difference in the minus sign in the two expression indicate that, since the SoH is a positive quantity, the capacity actually tends to decrease as the battery ages through the so called capacity fade phenomenon. Another common flavour of this SoH expression considers only the ratio between the current nominal capacity and the one corresponding to its BOL:

$$SOH = \frac{Q_n}{Q_n^{BOL}} \times 100\% \quad (2.5)$$

An important observation consists in notice how the precise knowledge of the SOH can influence the computation of the SoC since, as can be seen in the eq. 2.1, it depends strictly by the value of the nominal capacity. In contrast with equation 2.3, the relation 2.4 is quite used in the context of EV since it is important to guaranty a given range autonomy [16].

It is common to see also an heuristic definition of the SoH that is thought for the final user since it is specified in the datasheet as it provides an intuitive understanding of the current state of the battery [17]:

$$SOH = \frac{\text{residual number of charge and discharge}}{\text{maximum number of charge and discharge}} \times 100\% \quad (2.6)$$

Similarly to the case of SoC, is not possible to create a sensor that is able to measure directly the SoH and for this reason the BMS has to estimate it starting from the HIs.

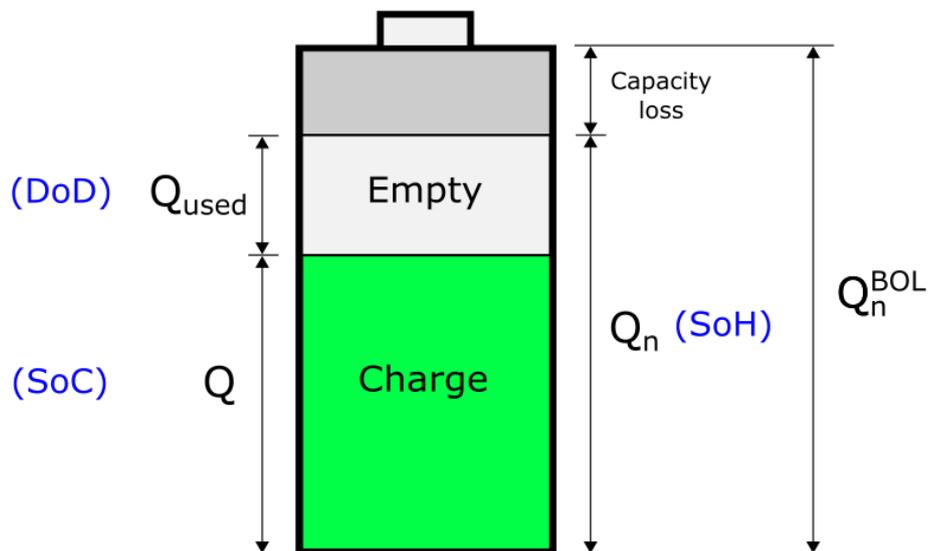


Figure 2.11: Conceptual representation about the three main status parameters (DoD, SoC and SoH) with respect to the battery capacity.

By observing the figure 2.11 it is possible to better understand the relationship between the three parameters and in particular how the SoC can vary following a limited SoH variation. This happens because the two quantities are both related to the nominal capacity that changes over time. Typically, SoC can vary rapidly according to the battery usage, while SoH is a slowly changing quantity.

The quality of the estimated value for both SoC and SoH is crucial for many reasons:

- **Performance:** SoC is related to the amount of available energy which can be used to power the engine and other vehicle's subsystems, then a good quality estimation allows to have more trust in the knowledge of the available range that a car can make on a journey. Moreover, having a reduced bounded uncertainty on the estimated value permits to better manage the request coming from the driver by full satisfying the requirement of instant power extracted from the battery in order to guarantee the right acceleration to the vehicle.
- **Longevity:** A poor quality estimation increase the possibility of either over-charge or over-discharge the battery, enabling the main mechanisms that are responsible to decrease the battery useful life.
- **Safety:** As described in previous paragraphs, over-charge and over-discharge operations are dangerous situations and have to be avoided since, in the worst cases, can cause fire or explosion hazards.
- **Reliability:** A good estimation has to be guaranteed for different driving profiles which change from different countries and depend strictly on the driving style of the driver.

In the successive chapter (3), the main estimation approaches that can be found in the literature are discussed for both SoC and SoH.

# Chapter 3

## Available approaches

In the last paragraph of the previous chapter (2.5), the main status battery parameters are introduced and the importance of finding an accurate estimation for both SoC and SoH is explained. The effort of finding a solution to the problematics that SoC and SoH introduce, has focused the interest of the scientific community over the years and currently in the research world, the argument has become a hot topic. The figure below shows the trend of the number of publications per year in the last 16 years.

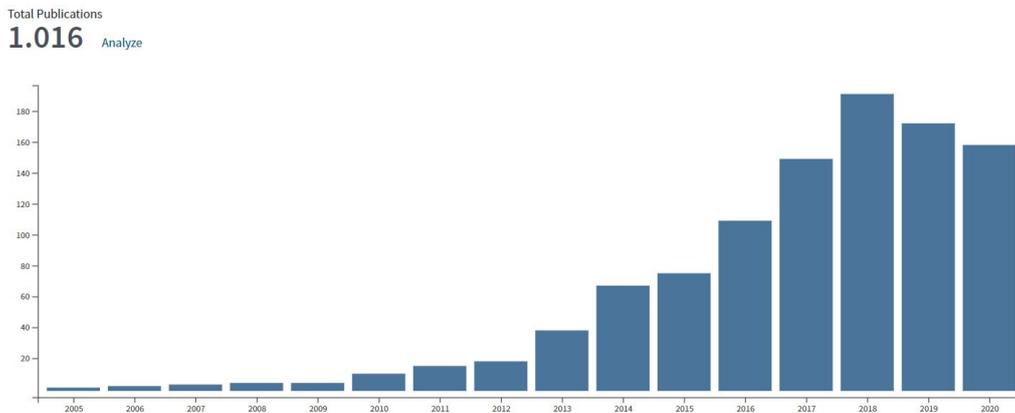


Figure 3.1: Statistics retrieved from Web of Science database regarding the number of publications with respect years. The used searching criterion contained the following keywords: "state of charge estimation", "state of health estimation", "lithium-ion battery", "electric vehicle", "hybrid electric vehicle". More details are specified at [18].

There exist many kind of approaches that can be found in the literature to deal with SoC and SoH estimation problem and they can be grouped into the following two macro categories:

- **First principles approaches:** The estimator is developed according to the physical and or chemical principles.
- **Data-driven approaches:** The estimator is a model which maps a given set of data to the target parameter without considering any physical and or chemical principle.

All the methods of these families of approaches rely on the usage of some set of data, and for this reason in the next paragraph (3.1) the possible data sources are described. Paragraphs from 3.2 to 3.3 enter into the details, for each family of approaches, about the most relevant estimation techniques and an example of their usage is provided by analyzing some research article. Finally, in the last paragraph of this chapter 3.4, a conclusive review about the analyzed estimation techniques is provided.

### 3.1 Data sources

In the automotive field, the leading data source is obtained by considering one or many dynamometer tests. In the circumstance of these tests, typically, the battery is monitored by acquiring measurement of current, voltage and temperature while the target vehicle perform a Dynamometer Drive Schedules (driving cycle). A driving cycle is a collection of speed referencies with respect to time that capture an average driving behaviour of a typical driver in different road conditions and different contexts (i.e. urban, highway, and so on). These tests are important because the resulting data can capture the peculiarity of the application scenario for which the estimator is designed. There exist a lot of driving cycles and every country or company have their own of reference. In European Union the leading one is the Worldwide Harmonized Light Vehicles Test Cycle (WLTC) which has substituted the old New European Driving Cycle (NEDC).

There exist many WLTCs and they are applicable to some specific category of vehicles which are distinguished from their power-to-mass (PMR) ratio.

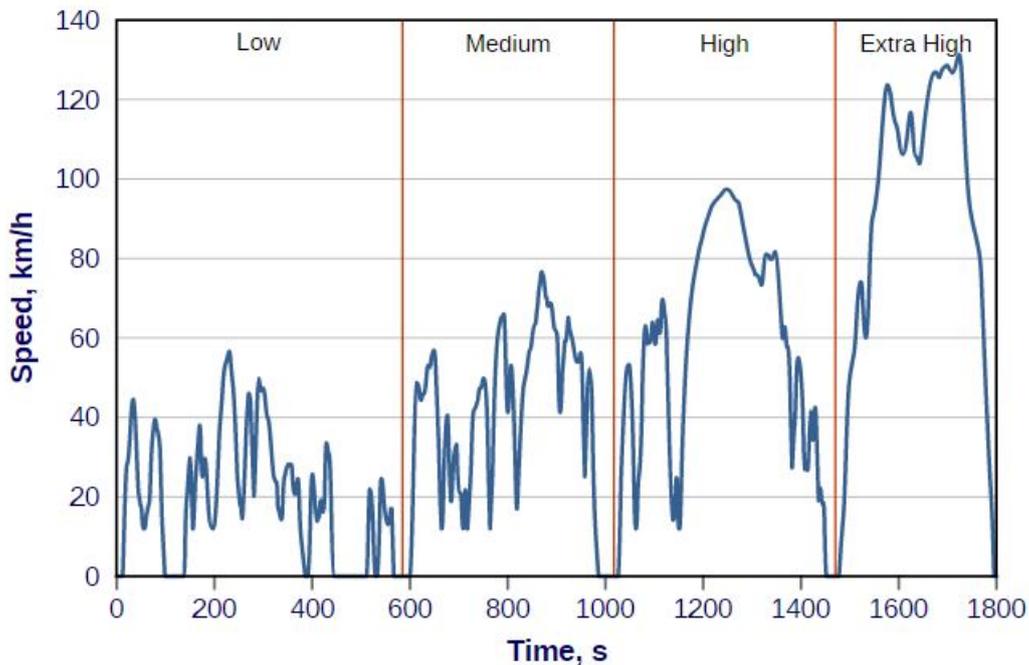


Figure 3.2: WLTC cycle for Class 3b vehicles. [19]

Other types of data can be retrieved by performing some tests directly on the battery system. These tests are typically adopted with the purpose of characterize the batteries properties and making possible to build batteries datasheets. Along with the tests (CC, CC-CV) already described in paragraph 2.4) there exist Pulse Discharge/Charge Test (PDT, PCT), Continuous Discharge/Charge Test (CDT, CCT) and Hybrid Pulse Power Characterization test (HPPC). Their execution can be performed using a cycler machine that repeats systematically the tests by alternating full charge and discharge of the battery. In this way is possible to acquire data during the entire life of the battery and make possible to track its age by measuring the capacity at a given iteration (cycle). For the seek of clarity, the following figures depict the significant signals regarding PDT, CDT and HPPC.

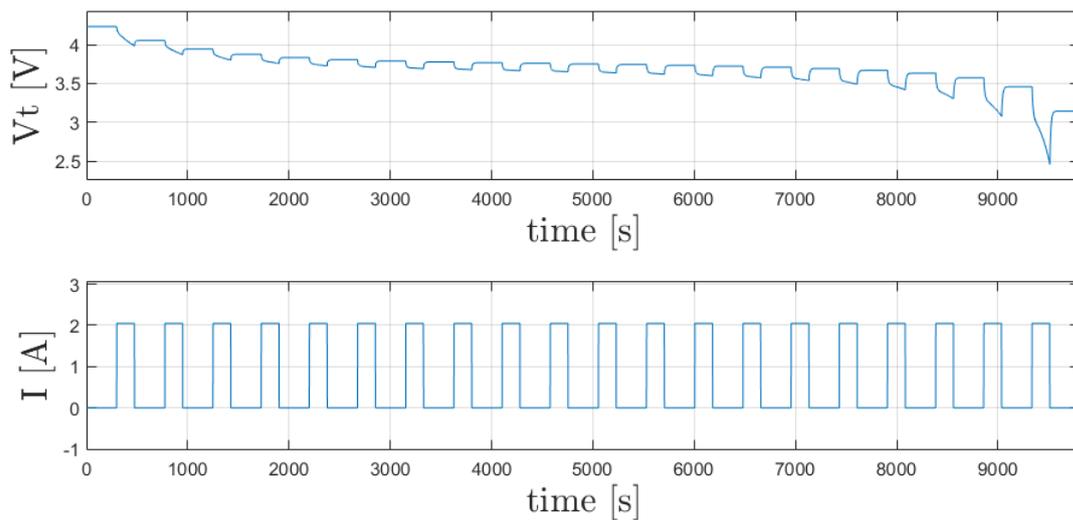


Figure 3.3: Example of PDT showing current stimuli and terminal voltage response, executed via simulation on a battery model of a Panasonic CGR18650AF. Starting from a full charged battery, discharge current pulses at 1C stimulate periodically the battery every 300 s for 35.12 s until the battery is fully discharged.

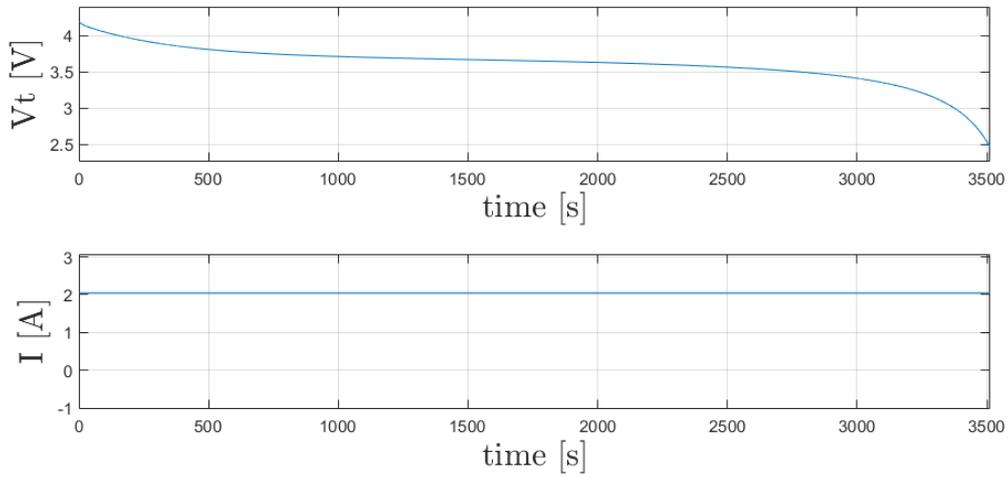


Figure 3.4: Example of CDT showing current stimuli and terminal voltage response, executed via simulation on a battery model of a Panasonic CGR18650AF. Starting from a full charged battery, a 1C continuous discharge current stimulate the battery until it reaches its fully discharged state.

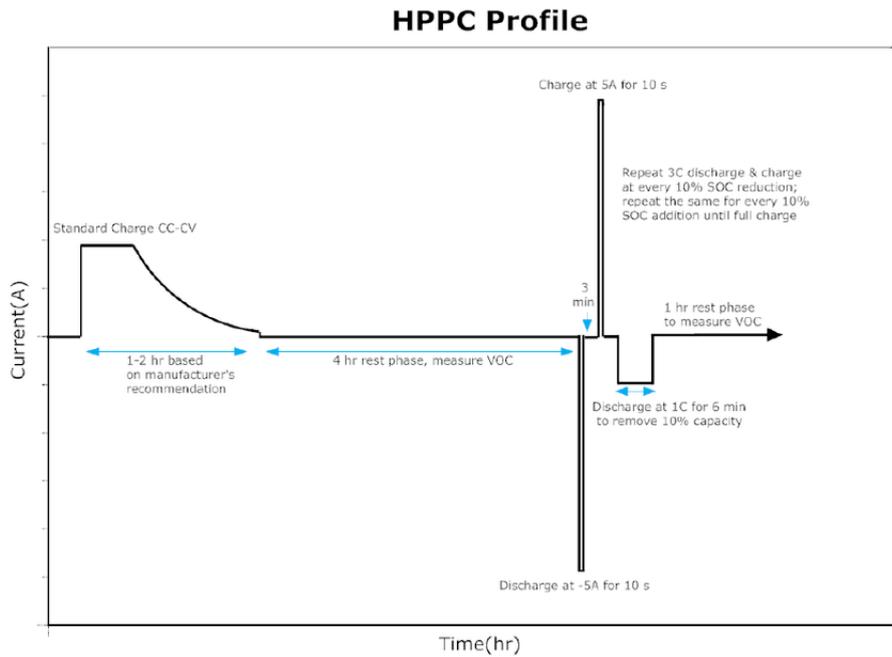


Figure 3.5: Example of HPPC showing the main phases of the test.

Data can be real-based, namely, obtained by performing tests on the real system or synthetic-based, that is via simulation by considering an accurate system model. Since self made experiment can be expensive, real-based data can be found on the web in the form of datasets which are made public by some authoritative entity like universities. A common dataset that can be found in the literature is provided by the NASA's Prognostics Center of Excellence through its Prognostics Data Repository. It contains different set of data gathered in the context of many tests performed on different lithium-ion batteries.

## 3.2 First principles approaches

### 3.2.1 Coulomb Counter

This technique allows to compute SoC by literally “counting the coulombs”, that is by integrating the battery current over time. A common formulation of the SoC expression is the following:

$$SoC(t) = SoC(t_0) + \int_{t_0}^t \frac{\eta_i \cdot I(\tau)}{Q_n} d\tau \quad (3.1)$$

This equation shows that starting from an initial condition for the SoC at a given time  $t_0$ , is possible to compute the current SoC at time  $t$  by adding over time all the contributes which come from the charging/discharging current profiles.

$Q_n[A \cdot h]$  is the nominal battery capacity while  $\eta_i$  is the coulomb efficiency which represents the ratio of the discharge capacity over the charge capacity during a discharge/charge cycle. The coulomb efficiency has to be computed empirically but for simplicity, in the literature can be found that  $\eta_i = 1$  if “I” is the discharge current, while  $\eta_i = \eta \leq 1$  if “I” is the charge current [20]. In the expression 3.1 battery capacity is considered to be stationary over time, and then aging effects are not taken into account [21]. Since the measured signals are sampled over time the 3.1 assume its discretized form:

$$\begin{cases} SoC(k+1) = SoC(k) + \frac{\eta_i \cdot \Delta t}{Q_n} \cdot I(k), & k \geq 1 \\ SoC(k) = SoC_0, & k = 0 \end{cases} \quad (3.2)$$

Where  $Q_n$  and “I” are respectively the nominal battery capacity and the charge/discharge current considered in the temporal interval  $[k\Delta t, (k+1)\Delta t]$  and  $\Delta t$  is the sampling time interval. Since it is possible to measure the current that is pumped into or requested from the battery, this method results to be simple to implement and can be used both offline and online, but in order to have high accuracy on the resulting value of the SoC it is necessary to have clear knowledge on the measured current. In fact, as it is an integral method it tends to accumulate the error over time making the SoC value drifts away from the true one. This happens especially when current sensors are pretty noisy, or even when the nominal capacity is not

updated in the circumstance that battery aging effects are neglected. A possible solution to reduce the error is to use this technique together with OCV-based methods that periodically correct the value of the SoC by looking at the existing relation between Open Circuit Voltage and SoC. This correcting technique can be used only in certain conditions that are explained in paragraph 3.3.1. Moreover, a crucial aspect regards the SoC initial condition ( $SoC_0$ ) which has to be estimated correctly, otherwise a significant error offset source in the overall estimation is introduced. To avoid issues with the initial SoC, typically the starting scenario refers to a full charged or discharged state of the battery.

The Coulomb Counting approach can be implemented also to estimate the SoH. As discussed in [22], the process is divided into the following two steps:

$$\begin{cases} Q_{discharged} = \int_0^T I(t)dt \\ SoH = \frac{Q_{discharged}}{Q_{rated}} \times 100\% \end{cases} \quad (3.3)$$

Starting from a full charged state, the battery is completely discharged and the relative capacity is obtained through the integration of the discharging current "I". Then, the SoH is computed by dividing the quantity obtained in the previous step by the rated capacity. Also in this case the accuracy depend on the goodness of the measures of current as explained for the SoC case.

Coulomb counter, hence, suffers in correspondence of a poor-quality sensors (which have to present high resolution and low SNR) and needs helps from OCV-SoC method for correction purposes. Moreover, is not always possible to completely charge/discharge the battery online, so, for these reasons, Coulomb Counter is suggested to be used as offline method in the context of a laboratory environment [23] in order to retrieve ground-truth data for other methods.

### 3.2.2 Equivalent Circuit Model based methods

In order to retrieve SoC and or SoH information, a possibility is to build an estimator based on a battery model which is described by the following general relationship:

$$V_t = f(SOC, I, T) \quad (3.4)$$

The output voltage is function of SOC, the current I and the battery temperature.

Depending on the chosen model typology and on the application purpose the function  $f$  can be different and arbitrarily complex. In particular, in scenarios where current is drawn or delivered considering fraction of the capacity rate, the usage of a simple model is justified due to a low dynamic profile, on the other hands, when high currents are considered (multiple of the capacity rate) or non constant current profiles, a more complex model is needed [24].

The most complete representation that can be defined in order to describe the behaviour of the battery system is an electrochemical model that consider very detailed phenomena happening on microscopic scale in terms of chemical reactions between active elements present into the cells. These types of models are computational expensive since are based on a system of several partial differential equation and present a lot of parameters to be identified, so are not appropriate to be elaborated online on BMS but are useful in the cell design process. Equivalent circuit models (ECM), in contrast, since consider macroscopic effects of microscopic phenomena are simplest but, at the same time, can preserve the useful information that are necessary to describe well the system dynamic. ECM can be defined at different granularity levels: starting from the finest level it can represent respectively the single cell, the single module and at the coarse level the entire battery pack. For practice purpose, in general, is desirable to work with a model defined at battery pack level. In the other cases, the overall pack model is built by put together as elementary blocks the cell/module models in the respective series/parallel fashion. In other cases the model can be seen as a cell-average model with respect to all the cells that constitute the battery pack.

Among the possible ECMs the most common models are the internal resistance (Rint) ECM, the First and the Second Order Thèvenin ECM (FOTM, SOTM), Partnership for a New Generation of Vehicles (PNGV) model and General Non-Linear (GNL) model. The circuit diagrams of each model, their respective expressions and description are reported in next discussion.

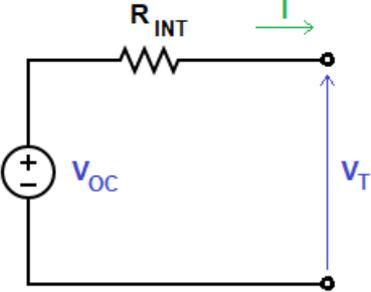
circuit schema	expressions
	$V_T = V_{OC} - R_{INT} \cdot I$

Table 3.1:  $R_{INT}$  model

The simplest way to model the battery system is to consider the series between a voltage source representing the Open Circuit Voltage ( $V_{OC}$ ), and a resistor which represents the internal battery resistance that is responsible of the voltage drop between  $V_{OC}$  and terminal voltage ( $V_t$ ). The OCV depends from many factor explained in paragraph 3.3.1, and the internal resistance value depends on it as well. According to the thesis work "Electro-Thermal Modelling of Lithium-Ion Battery" conducted by Mohammad Taffal at Politecnico di Torino, this model can be a good choice when the operations performed on the battery stimulate it according to the limit suggested by the cell manufacturer. In these circumstances, it is possible to build the entire battery pack model and starting from the information present in the cell datasheet, construct the OCV-SoC relationship and find the proper internal resistance values, for different temperature and discharge/charge condition. The low complexity of this model allows it to be easily implemented for real time applications.

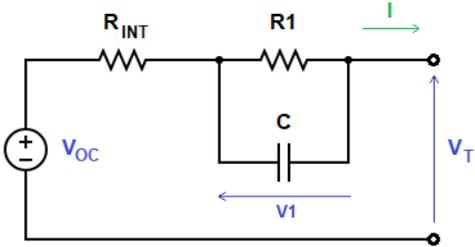
circuit schema	expressions
	$\begin{cases} \frac{dV_1}{dt} = -\frac{V_1}{\tau_1} + \frac{I}{C} \\ V_T = V_{OC} - V_1 - R_{INT} \cdot I \end{cases}$ <p>where</p> $\tau_1 = R_1 \cdot C$

Table 3.2: First order Thèvenin model

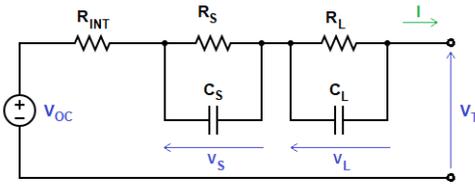
circuit schema	expressions
	$\begin{cases} \frac{dV_S}{dt} = -\frac{V_S}{\tau_S} + \frac{I}{C_S} \\ \frac{dV_L}{dt} = -\frac{V_L}{\tau_L} + \frac{I}{C_L} \\ V_T = V_{OC} - V_L - V_S - R_{INT} \cdot I \end{cases}$ <p>where</p> $\begin{cases} \tau_S = R_S \cdot C_S \\ \tau_L = R_L \cdot C_L \end{cases}$

Table 3.3: Second order Thèvenin model

The Thèvenin model starts from the Rint schema and introduce respectively in the first (table 3.2) and second order (table 3.3) versions, one and two RC networks with different time constants. These networks introduce more complexity to the model and permits to describe in more details the short-term and long-term transient behaviour of the terminal voltage. Physically RC networks model the diffusion process in the electrolyte and porous electrodes, the charge transfer and double-layer effect in the electrode [25].

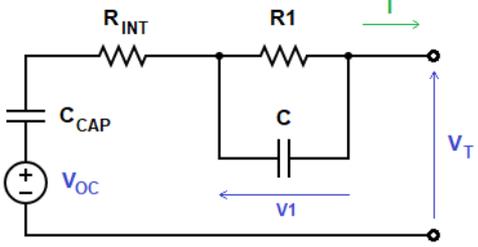
circuit schema	expressions
	$\begin{cases} \frac{dV_S}{dt} = -\frac{V_S}{\tau_S} + \frac{I}{C_S} \\ \frac{dV_L}{dt} = -\frac{V_L}{\tau_L} + \frac{I}{C_L} \\ V_T = V_{OC} - V_L - V_S - R_{INT} \cdot I \end{cases}$ <p>where</p> $\tau_1 = R_1 \cdot C$

Table 3.4: PNGV model

PNGV is builded starting from the respective Thevenin version by adding the so-called bulk capacitance ( $C_{CAP}$ ) between the voltage source and internal resistance. According to [25] the presence of an additional capacitance helps to model the variation in term of OCV due to discharge current accumulation effect. Thanks to this additional component, PNGV model reaches often better result if compared with Rint and first order Thèvenin models.

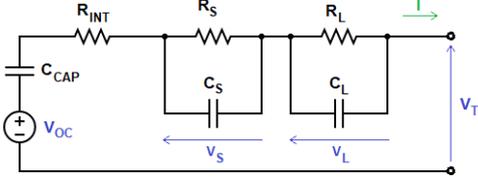
circuit schema	expressions
	$\begin{cases} \frac{dV_S}{dt} = -\frac{V_S}{\tau_S} + \frac{I}{C_S} \\ \frac{dV_L}{dt} = -\frac{V_L}{\tau_L} + \frac{I}{C_L} \\ V_{CAP} = \frac{1}{C_{CAP}} \int I d\tau \\ V_T = V_{OC} - V_L - V_S - R_{INT} \cdot I \end{cases}$ <p>where</p> $\begin{cases} \tau_S = R_S \cdot C_S \\ \tau_L = R_L \cdot C_L \end{cases}$

Table 3.5: GNL model

Finally, GNL model is obtained by adding an extra RC network to PNGV model in order to consider concentration polarization effect [25].

Every model is characterized by a certain amount of parameters that have to be identified. This activity is accomplished a-priori or dynamically, and in both cases empirical data are needed. In the a-priori strategy, parameters are found offline, namely, before the model is deployed. Therefore, at the first, data are acquired during laboratory sessions according to one or more tests described in both paragraphs 2.4 and 3.1. The Open Circuit Voltage is found according to OCV map method (paragraph 3.3.1). During tests all the interesting measures in term of temperature, voltage and current are acquired. SoC and or SoH are monitored and recorded using an offline method, typically Coulomb Counting is applied. The gathered data form a dataset that is splitted in two subsets in order to create an estimation set, used to estimate the parameters, and a validation set, used to validate the final model. A common strategy is to split the whole dataset into 2/3 – 1/3 partitions for estimation set and validation set respectively.

At this point, the model parameters can be found typically by using Least Square method. This estimation technique can be used only if the model relationship is

Linear Time Invariant (LTI) and under a certain condition explained later. In these circumstances, and considering all the estimation data, the equation 3.4 can assume the following matrix form expression:

$$y \cong \Phi\theta \quad (3.5)$$

$$y \in \mathbb{R}^{N_e}, \theta \in \mathbb{R}^n, \Phi \in \mathbb{R}^{N_e, n}, N_e \gg n$$

Where  $y$  is the output vector containing  $N_e$  measurements of the terminal voltage,  $\Phi$  is the regression matrix containing the measures of SoC, current and temperature and finally  $\theta$  is a vector containing the  $n$  model parameters to be found. The almost equal sign emphasizes the fact that the measures are affected by noise that is omitted in the expression.

The model parameters then are computed by applying the pseudo-inverse of the matrix  $\Phi$  to the measure vector and it can be performed only if the matrix  $\Phi^T\Phi$  is non singular.

$$\hat{\theta}_{LS} = (\Phi^T\Phi)^{-1}\Phi^T y \quad (3.6)$$

Where the matrix  $A_{LS} = (\Phi^T\Phi)^{-1}\Phi^T$  is the pseudoinverse ( $\Phi^\dagger$ ) of the regression matrix  $\Phi$ . Since this method involves the calculation of the inverse of a typically big matrix, LS in this form is used a-priori, namely, offline, because otherwise could be a bottleneck in term of time spent in performing its calculation. This means that the parameters estimated from data are static and they don't update during the battery operation when the system is working once deployed. Additionally, static parameters don't allow to adaptively capture any changing phenomena like battery aging effects.

For these reasons, Recursive Least Square (RLS) can be take into account for updating the model parameters in an online fashion. Starting from the LS expression is possible to retrieve the following set of equations that are needed to estimate the parameters iteratively:

$$\left\{ \begin{array}{l} \beta_{k-1} = 1 + \phi(k)^T V(k-1) \phi(k) \\ V(k) = V(k-1) - \beta_{k-1}^{-1} V(k-1) \phi(k) \phi(k)^T V(k-1) \\ K(k) = V(k) \phi(k) \\ \epsilon(k) = y(k) - \phi(k)^T \hat{\theta}_{k-1} \\ \hat{\theta}_k = \hat{\theta}_{k-1} + K(k) \epsilon(k) \end{array} \right. \quad (3.7)$$

Where  $V(k_0) = \alpha I$ ;  $k_0$  is the starting discrete time instant,  $I$  is the  $n \times n$  identity matrix,  $\alpha \in \mathbb{R}_+$  is a parameter to be tuned. If  $\alpha$  is almost equal to 1  $\hat{\theta}_k$  converges rapidly, while if  $\alpha$  is much smaller than 1 then  $\hat{\theta}_k$  converges slowly. This technique is quite used and as can be seen it doesn't involve any matrix inverse during the computation process but introduce only an additional cost in term of the number of simple operations. Other possibilities that can be considered in the parameters identification process, which can be found in the literature, consists in utilizing Non Linear Least Square (NLLS), Extended Kalman Filter (EKF), Neural Network (NN), Genetic Algorithm (GA), Fuzzy Logic (FL).

As mentioned at the beginning of this paragraph, the final SoC/SoH estimator is built on the basis of the system model. In the literature the most common adopted model-based estimators are Kalman Filter (KF), Extended Kalman Filter (EKF) and Unscented Kalman Filter (UKF). In the following subparagraphs, the theory of each filter is addressed and some example of their usage is provided by citing some paper.

### Kalman Filter

The Kalman approach, considered the optimal state linear estimator, is based on a probabilistic framework and is associated to a discrete-time LTI system (S), which can be described in term of state space model by means of the following system of equations:

$$S : \left\{ \begin{array}{l} x(k+1) = Ax(k) + Bu(k) + w(k) \\ y(k) = Cx(k) + v(k) \end{array} \right. \quad (3.8)$$

- $k \in \mathbb{N}$  is a discrete time instant

- $x(k) \in \mathbb{R}^n$ ,  $\sim (\bar{x}_k, P_k)$  is the state random vector of the system at time instant  $k$  with mean value  $\bar{x}_k$  and covariance matrix  $P_k$
- $u(k) \in \mathbb{R}^p$  is the exogenous input vector of the system at time instant  $k$
- $y(k) \in \mathbb{R}^q$  is the measured output of the system at time instant  $k$
- $A \in \mathbb{R}^{n,n}$  is the state system matrix
- $B \in \mathbb{R}^{n,p}$  is the input matrix
- $C \in \mathbb{R}^{q,n}$  is the output matrix
- $w(k) \in \mathbb{R}^n$ ,  $\sim N(0, Q)$  is a sample drawn from a multivariate normal distribution which has zero mean value and  $Q$  as covariance matrix; it represents the process noise into the system
- $v(k) \in \mathbb{R}^q$ ,  $\sim N(0, R)$  is a sample drawn from a multivariate normal distribution which has zero mean value and  $R$  as covariance matrix; it represents the noise afflicting the measurements
- Both  $w$  and  $v$  are white noises which are correlated considered the same time instant and uncorrelated with each other if considered at different time instants:  $E[w(k_1)v(k_2)^T] = V\delta(k_2 - k_1)$
- $A \in \mathbb{R}^{n,n}$ ,  $B \in \mathbb{R}^{n,p}$ ,  $C \in \mathbb{R}^{q,n}$ ,  $Q \in \mathbb{R}^{n,n}$ ,  $R \in \mathbb{R}^{q,q}$ ,  $V \in \mathbb{R}^{n,q}$  are known matrices;  $x(k=1) \sim (\bar{x}_1, P_1)$  is an unknown random vector, uncorrelated with  $w$  and  $v$ , with known  $\bar{x}_1 \in \mathbb{R}^n$  and known  $P_1 \in \mathbb{R}^{n,n}$ , and the output measurements  $y(k)$  are available for  $k = 1, 2, 3, \dots, N$

Considering the available output measurements  $y(k)$  for  $k = 1, 2, 3, \dots, N$ , the optimal estimate of the state  $x(N)$  is obtained thanks to the following set of equations describing the behaviour of the Kalman filter (KF):

$$KF : \begin{cases} \hat{x}(k+1|k) = A\hat{x}(k|k-1) + Bu(k) + K(k)e(k) & (3.9.a) \\ \hat{y}(k|k-1) = C\hat{x}(k|k-1) & (3.9.b) \\ \hat{x}(k|k) = \hat{x}(k|k-1) + K_0(k)e(k) & (3.9.c) \\ e(k) = y(k) - \hat{y}(k|k-1) & (3.9.d) \end{cases}$$

Where  $K_0(k) \in \mathbb{R}^{n,q}$  is the Kalman filter gain matrix obtained by:

$$K_0(k) = P(k)C^T[CP(k)C^T + R]^{-1} \quad (3.10)$$

This expression involves the state prediction error variance  $P(k)$ , which is computed iteratively by means of the so called Difference Riccati Equation (DRE):

$$P(k+1) = AP(k)A^T + Q - K(k)[CP(k)C^T + R]K(k)^T \quad (3.11)$$

Where  $K(k) \in \mathbb{R}^{n,q}$  is the one-step Kalman predictor gain matrix:

$$K(k) = [AP(k)C^T + V][CP(k)C^T + R]^{-1} \quad (3.12)$$

The below figure illustrates how the model and the filter interact to each other.

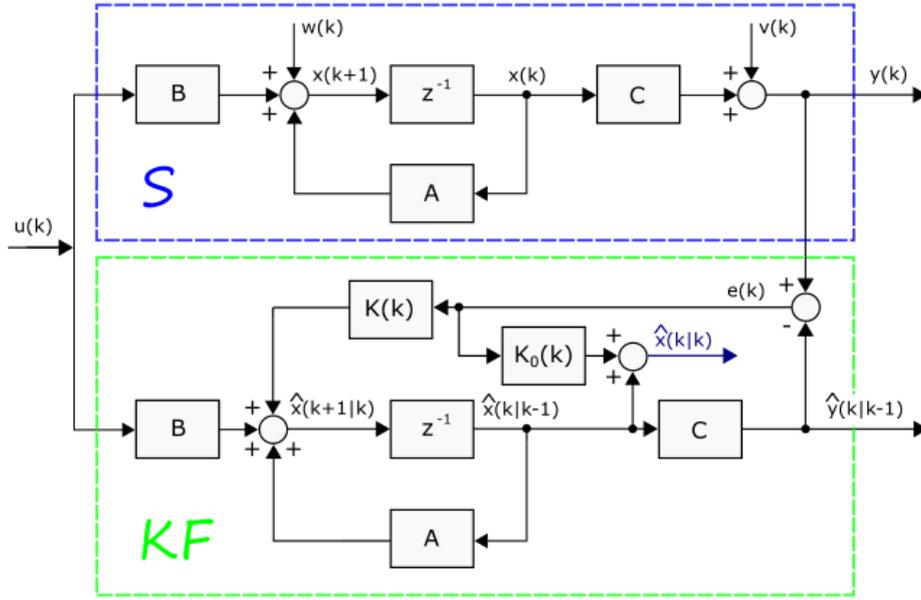


Figure 3.6: Schematic diagram illustrating how system S and Kalman Filter KF interact to each other.

The Kalman filter algorithm is implemented iteratively by performing the steps:

1. The Kalman filter gain matrix ( $K_0$ ) and the one-step Kalman predictor gain matrix ( $K$ ) are computed at time instant  $k$  thanks to equations 3.10 and 3.12 respectively
2. The innovation term  $e(k)$  is obtained by means of equations 3.9.b and 3.9.d
3. Using the state estimate at time  $k$  and the corrector element, by applying equation 3.9.c the filtered state is retrieved
4. A one-step prediction of the state is performed using 3.9.a and the new variance of the state prediction error  $P(k+1)$  is calculated through DRE using equation 3.11
5. Time is updated ( $k=k+1$ ) and the cycle is repeated starting from step 1.

Matrices  $Q$  and  $R$  can be found according to the a priori statistic information about the process and measurement noises, otherwise have to be found using a trial and error approach. Kalman Filter can estimate only SoC: from the perspective of SoH, since it is related to capacity fade or internal resistance degradation, the state vector would contain the capacity or resistance term making the system 3.8 non-linear.

In [26] is explored the usage of the Kalman filter based on the Rint model which was considered to represent a 6.8Ah fresh lithium-ion cell for both photovoltaic and Hybrid Electric Vehicle (HEV) applications. Measurements were taken in a first test by stimulating the cell for 7200s with a step current of 10% of the rated capacity, and in a successive test using a constant current profile. They achieve a maximum error of 5% in estimating SoC in photovoltaic scenario, good results in estimating the terminal voltage and resistance for the HEV application. In this circumstance, SoC was estimated using Coulomb Counting. They observed about 250s before the estimator reached a steady estimation, and found some troubles in finding suitable values for  $R$  and  $Q$  matrices of the Kalman Filter.

In [27] the continuous time version of the Kalman Filter (called Kalman-Bucy Filter) is used along with a Thevenin like ECM, to affine the SoC estimation accuracy of the Coulomb Counting online technique. Pulse charging test it has been performed offline to construct OCV-SoC map and make possible to validate the proposed method. In order to test the CC-KF method, for the tests it was used a 40Ah-14.6V Valance U1-12XP Li-ion battery pack and the simulation was performed using National Instruments' LabView 2009 SP with a portable data acquisition unit National Instruments myDAQ. Among the different tests performed, one was a pulse charge. In this test, during charge the CC-KF SoC was computed, and in the idle moments of 8 s OCV was measured. They were able to estimate the SoC with an error of  $\pm 1.76\%$  in comparison with OCV-SoC map method estimation.

Since Kalman Filter needs an accurate battery model in order to be able to properly estimate the state of the system, in [28] were analyzed at the first the possible sources that can lead to model error and proposed a joint estimation algorithm based on Kalman filter that can overcome to this issue. The method consists in adjoin to the state of the system, modelled by the first order Thèvenin ECM, a constant biased term that represents the model error. Then the resulting KF is decoupled into two parallel parts that interact to each other. The first part is a standard KF that estimate the state of the system as the model error doesn't exist, while the other part is another KF that estimate the model error. The final state estimation is corrected by computing the corrector term which comes from having estimated the model error. With the aim of validating the proposed method called Bias Corrected KF (BCKF), three different driving cycles were chosen (UDDS, US06 and NYCC) along with two different sources of error such us OCV-SOC data drift due to battery aging and voltage sensor drift. The final results were compared with the ones obtained with standard KF using RMSE as comparison criterion. In all the tests BCKF obtains better results with respect to KF. The best one was achieved under NYCC driving cycle by considering OCV-SOC drifting were BCKF obtain 0.61% and KF 22.04% in terms of RMSE referenced to SoC estimation error. The proposed method provides more robustness and accuracy in estimating the SoC but it is stated that the method is only capable to track slowly changing model error over time.

### Extended Kalman Filter

The linear assumption taken on the system in the built framework of the Kalman Filter, weakly holds, since in real world most of the systems have a non-linear dynamic. For this reason, Kalman Filter is extended to embrace the possibility to estimate the state of the system in the circumstance it is described by non-linear relationship. In particular, the state space model is rewritten in the following way:

$$S : \begin{cases} x(k+1) = f(x(k), u(k), k) + w(k) \\ y(k) = h(x(k), u(k), k) + v(k) \end{cases} \quad (3.13)$$

Where  $f$  and  $h$  are two well-known differentiable non-linear functions, and the assumptions made on  $w$  and  $v$  still hold as in the KF setting.

In order to apply KF in this scenario, EKF linearizes the functions  $f$  and  $h$  by evaluating and approximating their first order Taylor expansion around the last state estimate, input and time instant. In this way the state-transition matrix  $A$  and observation matrix  $C$  involved in the KF are obtained:

$$\begin{aligned} \hat{A}(k|k) &= \left. \frac{\partial f}{\partial x} \right|_{(\hat{x}(k|k), u(k), k)} \\ \hat{C}(k|k) &= \left. \frac{\partial h}{\partial x} \right|_{(\hat{x}(k|k), u(k), k)} \end{aligned} \quad (3.14)$$

The Extended Kalman Filter is then described by the following system of equations:

$$EKF : \begin{cases} \hat{x}(k+1|k) = f(\hat{x}(k|k-1), u(k), k) + \hat{K}(k)e(k) & (3.15.a) \\ \hat{y}(k|k-1) = h(\hat{x}(k|k-1), u(k), k) & (3.15.b) \\ \hat{x}(k|k) = \hat{x}(k|k-1) + \hat{K}_0(k)e(k) & (3.15.c) \\ e(k) = y(k) - \hat{y}(k|k-1) & (3.15.d) \end{cases}$$

Where the matrices  $\hat{K}$ ,  $\hat{K}_0$  are computed in the following way:

$$\hat{K}_0(k) = P(k)\hat{C}(k|k)^T [\hat{C}(k|k)P(k)\hat{C}(k|k)^T + R]^{-1} \quad (3.16)$$

$$P(k+1) = \hat{A}(k|k)P(k)\hat{A}(k|k)^T + Q - \hat{K}(k)[\hat{C}(k|k)P(k)\hat{C}(k|k)^T + R]\hat{K}(k)^T \quad (3.17)$$

$$\hat{K}(k) = [\hat{A}(k|k)P(k)\hat{C}(k|k)^T + V][\hat{C}(k|k)P(k)\hat{C}(k|k)^T + R]^{-1} \quad (3.18)$$

The algorithm illustrated for the Kalman Filter is modified by introducing the preliminary linearization step of the function  $f$  and  $h$  (eq. 3.14) at the beginning and by referring to respective EKF equations (3.15.a to 3.15.d instead of 3.9.a to 3.9.d, and 3.16 to 3.18 instead of 3.10 to 3.12).

The same consideration, as in the KF case, can be done about matrices  $Q$  and  $R$ . In this context both SoC and SoH can be estimated since the system state vector can contain both SoC and battery capacity or internal resistance.

In [29] a model-driven approach is implemented in order to estimate the SoC of light electric vehicles. For this purpose was chosen a battery pack produced by AllCell Technologies made by 40 LG ICR18650MG1 cells, with 2.6 Ah nominal capacity, arranged in a series of 10 modules containing a parallel of 4 cells. The battery pack model is represented by an average cell model through a second order Thèvenin model ECM. Different tests were performed in order to construct the OCV-SoC map and to characterize the model by finding the respective parameters. It is stated that the temperature dependence of the model parameters was not considered and the experiments were conducted at environment temperature. The charge transfer and diffusion parameters of the RC networks were provided by the manufacturer, while two different approaches, such as EKF and LS, were adopted and compared with the aim of identifying the average cell internal resistance. First battery discharge test was established on the fresh battery pack in order to acquire the measures of terminal tension corresponding to the respective current profile. Using EKF it was possible to identifying along with internal resistance also the nominal capacity, proving the reliability of the data provided by the manufacturer. From the estimated internal resistance signal was possible to notice the remarkable dependence from the SoC. This relationship with SoC was emphasized in the least square approach were, starting from the data it was possible to reconstruct the explicit dependency of the internal resistance from SoC achieving better results with respect to EKF. Once the parameters were found, both EKF and Adaptive EKF (AEKF) were implemented and compared on the SoC estimation task. The difference between the two algorithms lies in the staticity of the covariance matrices of the noises present in the state

space model. In the AEKF the Q matrix is adaptively updated in order to follow the estimation error while in EKF is defined offline in function of the matrix R. In both cases matrix R was found by performing a statistical analysis on the measurement error. The updating formula in the AEKF was found by means Maximum Likelihood estimation approach. This work concludes that EKF was able to contain the SoC estimation error into 5% while AEKF allow to reduce the boundary down to 1%.

In [30] the battery cell LG HG2 3.6V, 3.0 Ah was modelled using the second order Thèvenin ECM. Model parameters were identified through pulse discharge test and, by means of polynomial curve fitting, an explicit dependency on the SoC, for each parameter, was established. At this point, in order to validate EKF in estimating SoC, different simulations were performed. The first test performed was the constant current discharge test with 0.5C, 1C, 2C discharge rates. The true values, provided by Coulomb Counting technique, were compared with the estimated ones by EKF and the final estimation error was less than 3%. It is stated that other tests were conducted using Urban Dynamometer Driving Schedule (UDDC) Cycle but the performance achieved are not mentioned.

In [31] EKF was implemented along with a parameter-dependent first order Thèvenin ECM. In particular, the dependency of the RC network parameters on SoC and temperature was explicated and taken into account in the whole estimation process. A new time-varying variable, related to the RC parameters, was introduced in the state-space model by applying the gain scheduling technique. The relationship of these parameters, with respect to SoC and temperature, was established offline by applying curve fitting using LS on the basis of AC impedance measurements performed in the circumstance of 0.5 C discharge tests with a range temperature from 0°C to 45°C. Then, an OCV-SoC map was obtained by fitting Plett expression with LS, for each of the different discharge/charge tests performed at 0.02C. The OCV-SoC map was then averaged over all the experiments. Finally, EKF was applied on static-model and on the proposed parameter-dependent model. The two different estimations were subsequently compared with the ground true SoC obtained by Coulomb Counting. It was observed that EKF enhanced its accuracy in correspondence of parameter-dependent model and its estimation error remained under 2%

for the entire validation time interval; while the estimation error of the EKF with static-parameter model started to diverge from a certain instant of time.

In [32] was employed an EKF on a temperature-dependent battery model including hysteresis effects. First, they retained to adopt a generalized Thévenin ECM with an arbitrary numbers of parallel RC network since it exposes a good tradeoff between complexity and accuracy in estimating the terminal voltage. In order to find the proper number of RC networks, they set up a discharge pulse test made by 3 minutes of 6.5 A discharge current followed by 30 minutes of rest period, at the temperature of 20°C. By observing the terminal voltage data, in correspondence of the SoC range 55%-50%, it was possible to recognize and isolate, in the curve characteristic, the effects of the internal battery resistance (linear voltage drop) and the ones due to the RC network (transient response). At this point two curve models, representing respectively the first and second order RC networks with the additive internal resistance, were used to fit the transient response of the terminal voltage, and by means of RMSE criterion they showed that the second order Thévenin model achieved the highest accuracy in describing the terminal voltage. At this point the temperature-dependent model was established. The parameters were found by means of non-linear least square (NLLS) by minimizing the difference between the estimated terminal voltage and the true one obtained by discharge pulse test at 20 °C temperature. Afterward, discharge/charge pulse tests were conducted at different temperature (-20°C, -10°C, 0°C, 20°C, 25°C, 20°C, 60°C) in order to construct OCV-SoC curve. Both the tests presented 20 cycles of 1 hour rest period followed by a discharge/charge at C/10 rates which ended when the cell reached 2.7 V for discharge test and 4.2V for charge test. A gap between the OCV curves, obtained in correspondence of the same discharge/charge current and temperature, was observed and for this reason the acquired OCV data were averaged over discharge/charge tests in order to take into account the hysteresis effect. Then, the resulting data were fitted using 6-th degree polynomial for the considered different temperatures. For optimization purpose and, in order to give a continuous range of temperatures, the polynomials were substituted by a single linear interpolation that took, as reference offset, the OCV-SoC-20 °C polynomial. At this point, the proposed temperature-dependent model was enhanced by introducing the

one-state hysteresis effect. In contrast to traditional hysteresis models, which use a constant maximum hysteresis voltage, they proposed to adaptively change this value according to the SoC value and temperature. The hysteresis parameters were again identified using NLLS algorithm. The overall model (proposed model) was then validated against the method using only 2 RC networks (2 RC model) and the temperature-dependent model without hysteresis (conventional model). The validation was performed through UDDS cycle on the temperature range  $-20^{\circ}$ - $60^{\circ}$ C, using an NMC-oxide li-ion battery with 6.5Ah nominal capacity and 3.7 nominal voltage. The current profile was determined on the basis of another article which consider a vehicle roughly twice the size of the Honda Insight. The best validation on the models in predicting the terminal voltage shown how the performance increases when passing from “only 2 RC model” to the “proposed model”. Finally, the three different models were coupled with EKF and were compared. Again, the best results were achieved over all the temperature range by the proposed models which was able to estimate SoC value with an error under 1%.

### Unscented Kalman Filter

The Extended Kalman Filter has become a de facto standard method in many engineering applications thanks to its versatility in estimating the state, as well as identifying the parameters, of a non-linear dynamic system. The main characteristics of this estimator consist in approximating the state distribution with a gaussian random variable which is analytically propagated through the linearized version of the non-linear dynamic system. This aspect, along with the first order Taylor expansion used to linearize the system, can be quite good in certain cases but in others can penalize the accuracy of the computation of the posterior mean and covariance of the transformed state GRV, bringing to a sub-optimal solution and, sometimes, to the divergence of the filter. The Unscented Kalman Filter (UKF) takes into account these issues and proposes a better solution even if the computational complexity remains the same as well as in the EKF algorithm. The UKF assumes again a gaussian distribution for the system state, but represents it by collecting a finite number of special samples that are able to capture the real mean and covariance, and when propagated through the non-linear function of the system can capture the posterior mean and covariance as the EKF might do when utilizing a third order Taylor Expansion. This remarkable ability is valid for any non-linear function and is provided by the so called unscented transformation (UT):

Let consider a L dimensional random vector (r.v.)  $x$  and apply it to a generic non-linear function  $g$ .

$$\begin{aligned} y &= g(x) \\ x &\sim (\bar{x}, P_x) \end{aligned} \tag{3.19}$$

In order to compute the mean and covariance of the r.v.  $y$ , a matrix  $\mathcal{X}$  containing  $2L+1$  sigma vectors and the relative  $W_i$  coefficient are computed in the following way:

$$\begin{aligned}
 \mathcal{X}_0 &= \bar{x} \\
 \mathcal{X}_i &= \bar{x} + \sqrt{i(L+\lambda)}P_x, & i = 1, \dots, L \\
 \mathcal{X}_i &= \bar{x} - \sqrt{i(L+\lambda)}P_x, & i = L+1, \dots, 2L \\
 W_0^{(m)} &= \frac{\lambda}{L+\lambda} \\
 W_0^{(c)} &= \frac{\lambda}{L+\lambda} + (1 - \alpha^2 + \beta) \\
 W_i^{(m)} &= W_i^{(c)} = \frac{1}{2(L+\lambda)}, & i = 1, \dots, 2L
 \end{aligned} \tag{3.20}$$

Where  $\lambda = \alpha^2(L + \kappa) - L$  is a scaling parameter that is tunable by properly adjusting the values for the parameters  $\alpha$  and  $\kappa$ , which allow to control the spread of the sigma points around  $\bar{x}$ . The parameter  $\beta$  considers the prior knowledge of the distribution of  $x$ . A typical setting of this parameters is  $\alpha = 1e - 3$  (small value),  $\kappa = 0$ , and  $\beta = 2$  is considered optimal for gaussian distributions.

Once the sigma vectors are computed, they are propagated through the non-linear function  $g$ :

$$\mathcal{Y}_i = g(\mathcal{X}_i), \quad i = 0, \dots, 2L \tag{3.21}$$

Finally, the statistics in term of mean and covariance of the output vector  $\mathcal{Y}$  are given by:

$$\begin{aligned}
 \bar{y} &= \sum_{i=0}^{2L} W_i^{(m)} \mathcal{Y}_i \\
 P_y &= \sum_{i=0}^{2L} W_i^{(c)} (\mathcal{Y}_i - \bar{y})(\mathcal{Y}_i - \bar{y})^T
 \end{aligned} \tag{3.22}$$

In [33], considered with the aim of acquiring inspiration for the theoretical study of this argument, the UKF is well explained and its performances are compared against Monte Carlo method, and EKF on different time-series data.

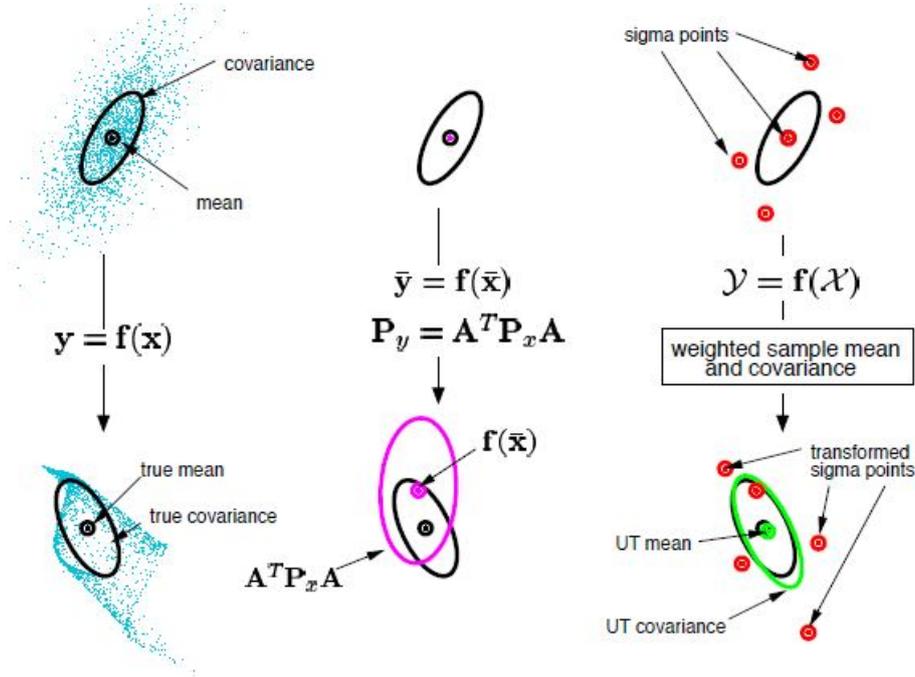


Figure 3.7: Example of the UT for mean and covariance propagation. a) actual, b) first-order linearization (EKF), c) UT. [33, Figure 1]

The UKF relies on the following discrete-time non-linear dynamic system:

$$S : \begin{cases} x(k+1) = f(x(k), u(k), k) + w(k) \\ y(k) = h(x(k), u(k), k) + v(k) \end{cases} \quad (3.23)$$

In this case, in contrast with EKF, the restriction on the additivity of the noises  $w$  and  $v$  is dropped and the general case, where they are input of the respective known non-linear functions  $f$  and  $h$ , is considered.

The algorithm can be summarized in the following steps:

**Inizialization:**

$$\begin{aligned}
 \hat{x}_0 &= \bar{x}_0 = E[x_0] \\
 P_0 &= E[(x_0 - \bar{x}_0)(x_0 - \bar{x}_0)^T] \\
 \hat{x}_0^\alpha &= \bar{x}_0^\alpha = E[x^\alpha] = [\bar{x}_0^T \quad \mathbf{0} \quad \mathbf{0}]^T \\
 P_0^\alpha &= E[(x_0^\alpha - \bar{x}_0^\alpha)(x_0^\alpha - \bar{x}_0^\alpha)^T] = \begin{bmatrix} P_0 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & P_w & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & P_v \end{bmatrix}
 \end{aligned} \tag{3.24}$$

for  $k \in [1, +\infty)$ ,

Calculate sigma points:

$$\mathcal{X}_{k-1}^\alpha = [\hat{\mathcal{X}}_{k-1}^\alpha \quad \hat{\mathcal{X}}_{k-1}^\alpha \pm \sqrt{(L + \lambda)P_{k-1}^\alpha}] \tag{3.25}$$

**Time update:**

$$\begin{aligned}
 \mathcal{X}_{k|k-1}^x &= f(\mathcal{X}_{k-1}^x, u(k), \mathcal{X}_{k-1}^w) \\
 \hat{x}_k^- &= \sum_{i=0}^{2L} W_i^{(m)} \mathcal{X}_{i,k|k-1}^x \\
 P_k^- &= \sum_{i=0}^{2L} W_i^{(c)} [\mathcal{X}_{i,k|k-1}^x - \hat{x}_k^-][\mathcal{X}_{i,k|k-1}^x - \hat{x}_k^-]^T \\
 \mathcal{Y}_{k|k-1} &= h(\mathcal{X}_{k|k-1}^x, u(k), \mathcal{X}_{k-1}^v) \\
 \hat{y}_k^- &= \sum_{i=0}^{2L} W_i^{(m)} \mathcal{Y}_{i,k|k-1}
 \end{aligned} \tag{3.26}$$

**Measurement update:**

$$\begin{aligned}
 P_{\tilde{y}_k \tilde{y}_k} &= \sum_{i=0}^{2L} W_i^{(c)} [\mathcal{Y}_{i,k|k-1} - \hat{y}_k^-] [\mathcal{Y}_{i,k|k-1} - \hat{y}_k^-]^T \\
 P_{x_k y_k} &= \sum_{i=0}^{2L} W_i^{(c)} [\mathcal{X}_{i,k|k-1} - \hat{x}_k^-] [\mathcal{Y}_{i,k|k-1} - \hat{y}_k^-]^T
 \end{aligned} \tag{3.27}$$

$$\begin{aligned}
 K &= P_{\tilde{y}_k \tilde{y}_k}^{-1} P_{x_k y_k} \\
 \hat{x}_k &= \hat{x}_k^- + K (y_k - \hat{y}_k^-) \\
 P_k &= P_k^- - K P_{\tilde{y}_k \tilde{y}_k} K^T
 \end{aligned}$$

Where  $x^a = [x^T w^T v^T]^T$  is the augmented state vector,  $\mathcal{X}^a = [(\mathcal{X}^x)^T (\mathcal{X}^w)^T (\mathcal{X}^v)^T]^T$ ,  $\lambda$  is the scaling parameter,  $L = \text{card}(x^a)$ ,  $P_w$  is the covariance matrix of the process noise  $w$ ,  $P_v$  is the covariance matrix of the measurement noise  $v$ , and the coefficients  $W_i$  are computed according to 3.20.

Similarly to the EKF case, UKF allow to estimate both SoC and SoH.

In [34] they chose an overall SoC estimation algorithm composed of a first order Thèvenin ECM, whose parameters were identified online through the RLS with forgetting factor (RLSFF), and the UKF. The test data were provided directly by the BMS installed on a real EV in correspondence of five LiFePO4 battery packs. The nominal voltage and capacity of each battery were 96V and 72Ah respectively and every packs contained a series of 30 China Aviation Lithium Battery (CALB) of 3.2 V and 72 Ah. The BMS acquired voltage, current and temperature measures and computed the SoC every 5 s. In order to set up the RLSFF algorithm they first translate the state-space model of the battery into an AutoRegressive eXogenous (ARX) representation by applying Z transform to the system and find the transfer function as a polynomials ratio expression. The model parameters were encapsulated into the coefficient of the polynomials. In this context, they approximated the OCV-SoC curve, provided by the manufacturer, with a linear curve. Afterward the RLSFF algorithm was setup and applied on gathered tested data for each battery pack in order to validate it. From this experience they noticed the highly tendency of the parameters to change with environmental and operative conditions. Finally

by applying UKF along with RLSFF they were able to estimate the SoC with an average error of 0.52%. The maximum mean error was 1.23%. They stated that the results could be better if high quality current sensors were adopted since the BMS computed SoC with accuracy of 5%.

In [35] it was proposed a theoretical double estimation structure which is able to both estimate SoC and internal battery cell temperature, but only the portion which estimate the SoC was studied. For the first piece of the estimator a first order Thévenin ECM was adopted, whose parameters depended on SoC, temperature and sign of the current. Data for both parameters identification and validation were provided by NASA Prognostic Data Repository and referred to an 18650 cylindrical cell with nominal capacity of 2.2 Ah and voltage range 4.2V-3.2V. Since the provided data referred to a small thermal excursion, the dependency of the model parameters from temperature was considered negligible. They found the OCV-SoC map using a third order polynomial fitting curve, on the basis of constant discharge current profile data. Then parameters were identified by applying RLS on the AutoRegressive Moving Average (ARMA) equivalent of the cell model, for both discharge and charge current pulse data. Finally, random uniformly distributed C-rates discharge/charge current pulse were used to validate the performance of different SoC estimation algorithm, such as EKF, UKF and Particle Filter (PF). The results showed that both UKF and PF achieved an error percentage below 1%, but UKF was considered the best one since it exposes less complexity than PF algorithm.

### 3.3 Data-driven approaches

#### 3.3.1 Open Circuit Voltage map

The simplest method, but also the most accurate one, that can be thought is to construct a one-by-one map (Lookup table) between the Open Circuit Voltage (OCV) and the SoC and or the SoH. Then, by inverting the relationship in correspondence of a specific OCV value, the desired battery state parameter is obtained. The below figure gives an idea of this curve from the perspective of the SoC.

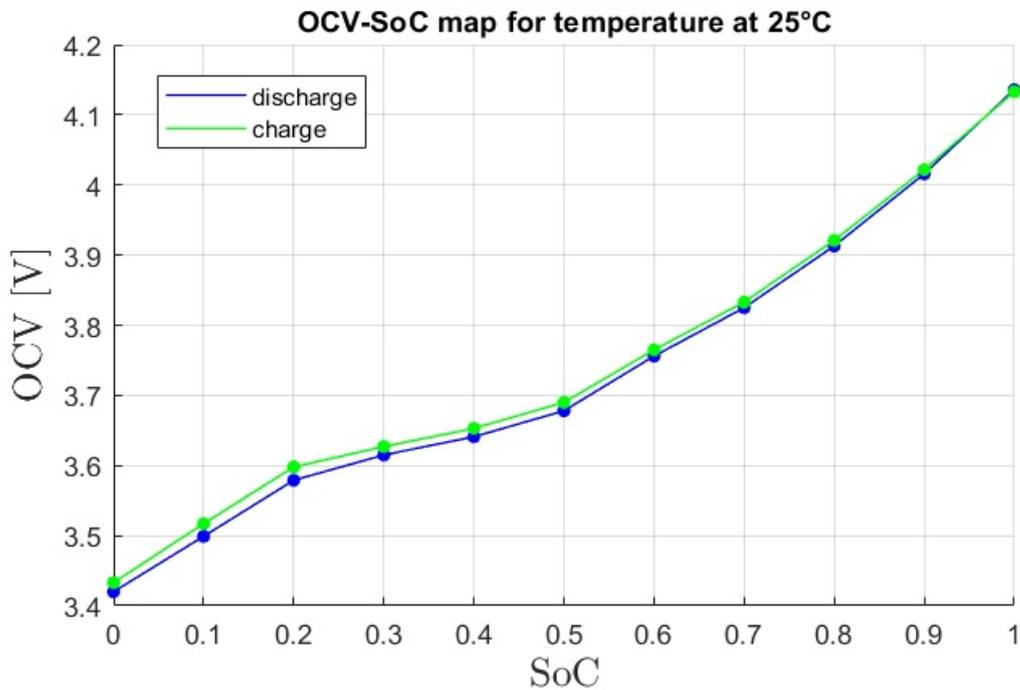


Figure 3.8: An example of OCV-SoC graph obtained by representing tabular data retrieved from the Samsung 94Ah prismatic cell datasheet.

As can be seen from the figure 3.8, the existing relationship that exists between OCV and SoC is highly non-linear and depends dramatically from many factors such as temperature, cell aging status (SoH), and current flow direction (charge/discharge), and for this reason the curve, actually, can be thought as a multidimensional surface. Depending on the cell manufacturer, the information regarding the OCV

curve can be given in different forms (tabular or graphs) and can be partial (neglecting some dimension like SoH, temperature, or current sign dependency) since the construction of the curve is very time expensive and require high quality hardware to be performed. A typical test adopted to retrieve OCV data points is the PDT (paragraph 3.1). In the context of PDT, by referring to a single cell with a specifying aging status at a given ambient temperature, without lost of generality, the acquisition of a data point happens by reading the terminal voltage once a rest time has passed. This relaxing time is compulsory because it is necessary to wait until electrochemical equilibrium, inside the cell, is reached. In fact, when no current transfer is acting into the cell, after a transient time the measured terminal voltage coincides with the Open Circuit Voltage. Typically, the waiting time needed to get into this situation can be very long (many minutes or hours) and for this reason, OCV measures consist in few points. Often, data refer to a fresh battery (100% SoH) an ambient temperature (25 °C), low C-rate discharge current (charge and discharge behaviour is assumed to be equal) and are taken for every 5 or 10% of SoC. The final curve is constructed by means of a curve fitting process which consists in polynomial regression. In the case of SoC the OCV-SoC map can be also given by the following expression:

$$V_{OC} = K_0 + K_1 SOC + \frac{K_2}{SOC} + K_3 \ln(SOC) + K_4 \ln(1 - SOC) \quad (3.28)$$

Where the parameter  $K_0, K_1, K_2, K_3, K_4$  has to be identified on the basis of the acquired data points. Since is expensive to acquire a lot of data in all the operative scenario the accuracy of this method can be penalized. For this reason, OCV method is used as offline method and typically as a complementary or corrective technique [23].

### 3.3.2 Black-Box

Lithium-ion battery is a highly complex nonlinear time varying electrochemical system [36] and for this reason is difficult to find a simple model that describe its behaviour in all the operating contexts. Black-Box methods allow to find a model regardless any physical or chemical principle by simply find a relationship that is able to associate a given input to a given output. In this way is possible to enestablish a direct relationship that relates the input measurements (current, temperature and voltage) to the target signals of interest (SoC and or SoH).

The most common black-box methods employed in the literature are based on machine learning techniques, in particular on Artificial Neural Networks (ANNs). The final estimator is obtained as output of the learning process that is performed on the basis of a given set of data:

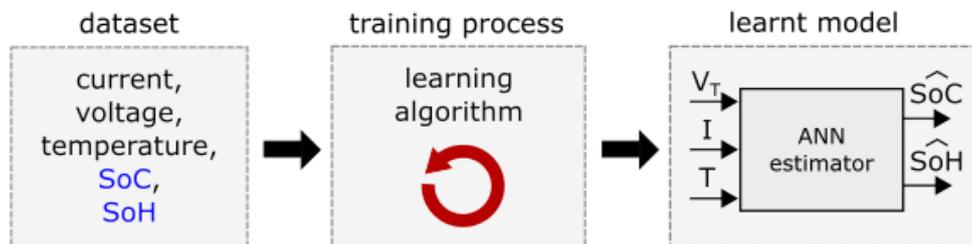


Figure 3.9: Basic conceptual schema of ANN estimator development. Blue SoC and SoH are ground truth quantities.

The learning process is performed in a supervised way, namely by providing along with input signal samples also the target signals that have to be estimated. By performing experimental tests (paragraph 3.1), the input signals (current, temperature, terminal voltage) can be acquired and then, using an offline method like Coulomb Counter (paragraph 3.2.1), ground truth SoC and SoH can be enestablished. Formally, the dataset can be seen as a collection of pair samples composed by the input

feature vector  $x$  and the ground-truth  $y$ .

$$D = \{x_i, y_i\}_i, \quad i = 1, 2, \dots, \#D \quad (3.29)$$

Typically, a pre-processing phase on the dataset is needed in order to prepare the data before applying any machine learning algorithm. Common procedures in data pre-processing consist in applying data transformations and data normalization.

The training process is never applied on the whole dataset, but on an its subset. Since the estimator, once delivered, has to work with data that has never seen during the learning phase, the training process must take into account this aspect. This problem is solved by partitioning the dataset into a training set  $D_t$  and a test set  $D_T$ . The training set is used to learn the model while the test set is used to evaluate the model.

In general, if the dataset is sufficient big a “Holdout” partition criterion is applied: the training set is typically composed of 2/3 of the whole data while the remaining 1/3 data form the test set, but other choices can be done as well by considering the sets ratio 70%-30%. The process of building the partitions consists of randomly sampling, with or without replacing, the starting dataset. Holdout can be applied iteratively on the training data in order to create new sets of training  $D_{t'}$  and validation set  $D_v$ . In this case the validation set is used to evaluate the model and for tuning purpose in the training process, while test set is used to evaluate the final performance of the final model, emulating unseen real-world data.

If the dataset has medium size, a “k-fold Cross Validation” technique can be applied: the dataset is sampled and partitioned in  $k$  sets, called folds, where  $k-1$  are used as training set while the remaining 1 is used as validation set. The training process is repeated  $k$  times in a way that, at the end, the role of validation set can be attributed at all the  $k$  partitions. The resulting performance is then the average of the performances obtained using the different  $k$  validation subsets.

In the context of the evaluation of the model, it is required to avoid overfitting by monitoring the behaviour of the generalization error, namely the difference between the training estimation error and the validation estimation error, that has to not increase over time. In fact, a scenario where the training estimation error is low

while the validation error is high is suggesting that the model is overfitting the training data, losing the possibility to perform well on the unseen data provided through the test set.

The most relevant ANN that can be found in the literature are Feed Forward Neural Network (FFNN) and Long Short Term Memory Recurrent Neural Network (LSTM-RNN). The details regarding each particular NN is discussed in the next subparagraphs, and some example of their usage is provided for the SoC estimation problem through the testimoniance of research papers.

### Feed Forward Neural Network

ANNs get inspiration from the biology world by emulating the human brain ability to learn information patterns by means of mathematical models which mimic the neurons behaviour. The Feed Forward Neural Network (FFNN) is a kind of ANN and its architecture is depicted in the following figure:

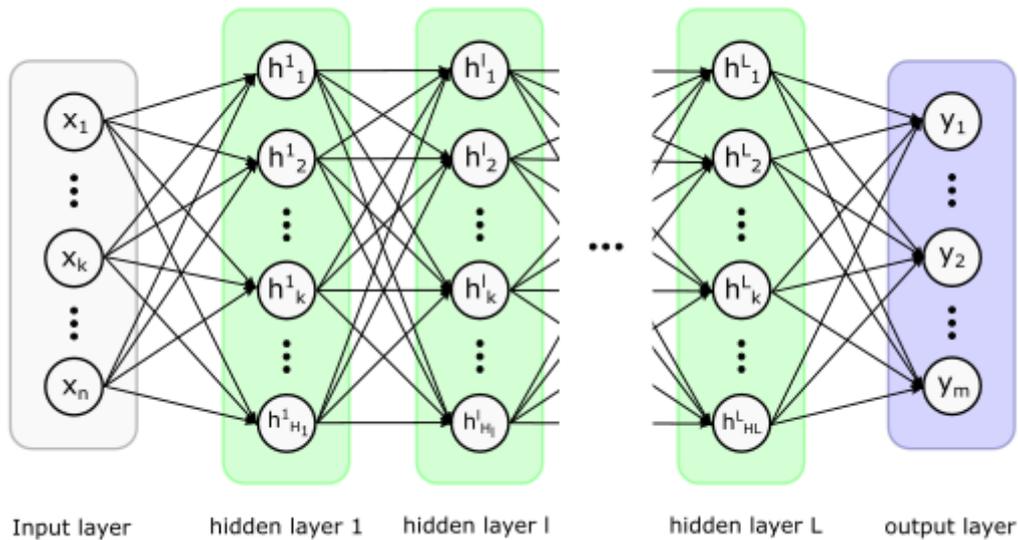


Figure 3.10: General FFNN architecture.

The structure of the network is made by an input layer, a certain number  $L$  of hidden layers and finally by an output layer. Each layer is composed by a certain

number of neurons that, excepting for the output layer, are linked to the neurons of the successive layer in a fully connected fashion. The information provided by the input layer flows through the entire network until it reaches the output layer, and for this reason the network is called feed forward. Each neuron of the network, excluding the ones of the input layer present the following inner structure:

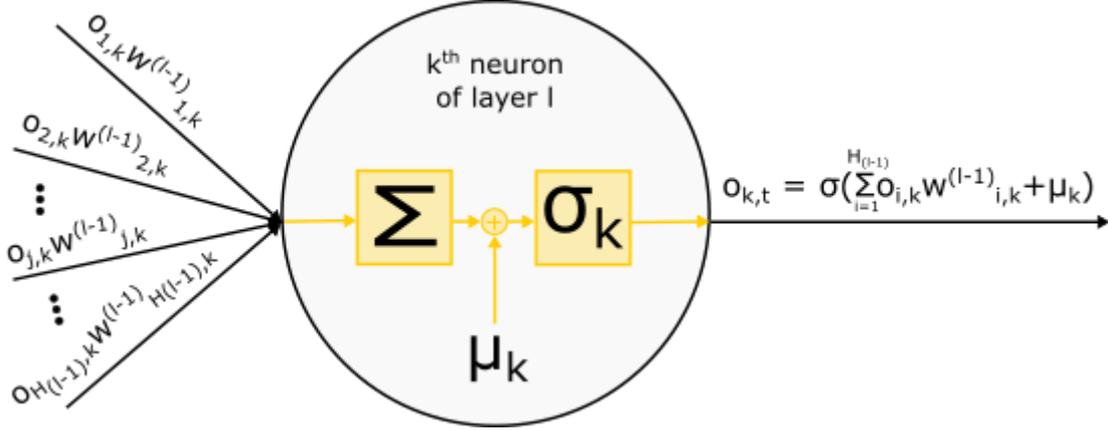


Figure 3.11: Neuron inner structure.

A generic neuron computes the weighted sum of its inputs, then adds a bias term, pass the intermediate result to a non-linear function called activation function and finally gives the resulting value as output. The activation function is quite-important and its choice is part of the design process of the network. Typical functions considered are logistic function, hyperbolic tangent, Rectified Linear Unit (ReLU), trigonometric functions and Radial Basis Function (RBF). The other parameters such us the weighs and biases terms are self-learned during the learning process. Finally, a loss function, which embeds the task that the network is supposed to do, have to be defined. The most-simple typical choice is to assume as loss function the Mean Square Error (MSE) between the ground-truth and the estimated output.

$$L = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (3.30)$$

The entire learning process so can be seen as an optimization problem. The loss function acts like an objective function that has to be minimized by finding

the proper set of network parameters with respect to the considered training data. The update of the network parameters happens thanks to the backpropagation of the error by means an optimization algorithm that are based on gradient descent. Typically, the sampled data are forwarded into the network in batch which size is another parameter to be chosen. It is common to distinguish between network parameters and hyperparameters: the first ones are characteristic of the model that the network learn and are found automatically thanks to an optimization algorithm, while hyperparameters are all the remaining parameters to be chosen for the network design (typically using trial-and-error approach or other strategies like grid search or Back Searching Algorithm (BSA)).

The learning process is composite of many steps called epochs  $\varepsilon$  and all of them consist on a pair of simple operation:

1. Forward the (batch) training data through the network and compute the estimation error through loss function
2. Update the network parameters by performing backpropagation using the optimization algorithm chosen

An epoch ends when all the training data are elaborated by the network according to the aforementioned steps. The overall learning process can be then summarized by performing the following steps:

1. Define the loss function, choice the activation function for each neuron and the optimization algorithm
2. Choice the set of hyperparameters: Define the number of hidden layer  $L$  and the quantity of neurons  $H_l$  in each layer  $l$ , number of epochs  $\varepsilon$ , batch-size  $B$ , and the optimization algorithm parameters (e.g learning rate)
3. Initialize all the internal parameters of the network to a rand value
4. Forward batch data through the network until the estimation is computed
5. Compute the loss and use backpropagation in order to update the network parameters

6. Continue from step 4. Until all the training data are propagated into the network otherwise continue from step 7
7. Advance in epoch repeating the process from step 4 until a threshold in terms of accuracy is met or if the epoch reaches a maximum value
8. Validate the final model by forwarding the validation data through the network and compute the estimation error

At the end of the process, if the result is not acceptable, is it possible to repeat the previous step by changing hyperparameters according to some strategy.

In [23] a FFNN with 1-hidden layer and ReLu as activation function was adopted as estimation algorithm. The dataset was built by performing a test procedure that combines capacity check, charge/discharge current pulse injection and accelerating aging on a NMC battery cell. The acquired data consisted in measurements of current and voltage as well as the ground-truth SoC. The model was trained for 10000 epochs using 100 and 10000 neurons inside the hidden layer. In both the circumstances the model achieved, once validated, a SoC estimation error below 2%.

Another example is provided by [37] where they adopted a deep FFNN architecture on a public domain dataset provided by University of Winsconsin-Madison. The provided dataset contains nine drive cycles and among them a “NN drive cycle” was designed for neural network training purpose. The provided cycle power profile was calculated for an electric Ford F150 truck with a 35kWh battery pack that was scaled for a single 2.9 Ah Panasonic 18650PF li-ion battery cell. Among the available data, current, voltage, temperature and ground-truth SoC, were selected to build the dataset which contained, at the end, 116982 samples that were divided respectively in training, validation and test sets. They tested different architectures but the one that obtained the best result was characterized by 2 hidden layer with ReLU activation function, 256 hidden neurons and batch-size 128. They stated that the network was able to achieve a MAE of approximately 1.60% in SoC estimation.

## Long Short Term Memory Recurrent Neural Network

Another remarkable architecture that can be found in the literature is provided by RNN. The main characteristic of RNN, in contrast with FFNN, is that it embeds in its architecture a concept of short-term memory that allows to consider, in the learning process, along with the inputs at a given time instant, also their history thanks to the presents of an hidden inner state which acts like a memory. This is a powerful concept but it introduces more complexity and it might bring to instability in the learning process by causing gradient vanishing or exploding phenomena, if not addressed properly. In its basic flavour, the RNN has one hidden layer with one hidden state which along with the input vector contributes to compute an output. At every time step, when a new input arrives, the inner state is updated by considering the current input vector and the inner state at previous time step, and then the output is computed. The structure of RNN can be figure out in the next picture, where on the left side a compact version of the RNN is presented, while on the right part the network is unfolded over time in a manner that the dependence with time is more clear.

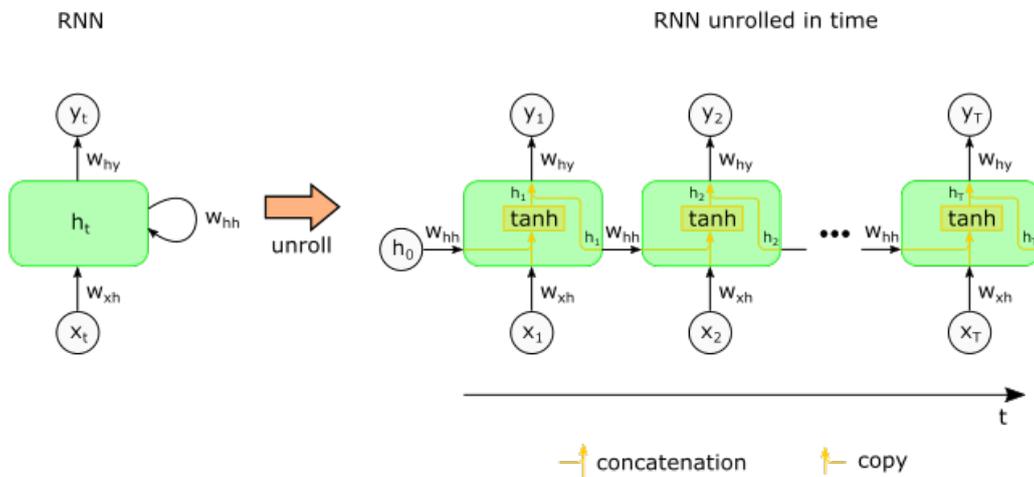


Figure 3.12: Compact version of a one hidden layer RNN structure on the left; time unrolled version of the same network on the right.

If the purpose of the task is to learn short time dependency in a sequence data, RNN is an optimal choice, but as mentioned before when sequences become longer,

the network manifests some trouble in the backpropagation step that bring to vanishing or exploding gradient problem. Moreover, train on long sequences a RNN is also an expensive task because require to unroll the network for a length correspondent to the length of the sequence introducing many intermediate layers. Last but not least, RNN tend to consider more important information saw recently “forgetting” the ones processed in the very past. To solve these issues the so called Long Short-Term Memory (LSTM) were introduced. LSTM-RNN, or simply LSTM networks are the very same as RNN but the inner layer, called cell, has a different structure. A new inner state variable is introduced which act as a very memory cell and thanks to the present of inner gates it is given to the network the ability to decide when remember an information as well as when forget it and finally how much of the memorized information to use. All these behaviours happens at every time step and are encoded through the following system of equation:

$$\left\{ \begin{array}{l} f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \\ i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \\ o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \\ \tilde{C}_t = \tanh(W_c[h_{t-1}, x_t] + b_c) \\ C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \\ h_t = o_t \odot \tanh(C_t) \end{array} \right. \quad \begin{array}{l} (3.31) \\ (3.32) \\ (3.33) \\ (3.34) \\ (3.35) \\ (3.36) \end{array}$$

Equations from 3.31 to 3.33 represent respectively the forgot, input and output gate which are computed through a sigmoid function that operates on a vector obtained by the concatenation of the past inner state and current input. In the equation 3.34 a candidate inner cell state is computed. In equation 3.35 the inner cell state is computed according to the respective gates: depending on the value the forgot gate takes (0 or 1) it decides if the past inner state has to be maintained or forgotten (the  $\odot$  symbol indicates an element-wise multiplication), while the input gate decides if the candidate inner cell state has to be promoted to participate in the building process of the inner cell state. Finally, through equation 3.36, the output gate decides how much of the inner cell state has to be exposed in the inner state. A graphical representation of the behaviour cell of a LSTM network is presented in the below figure.

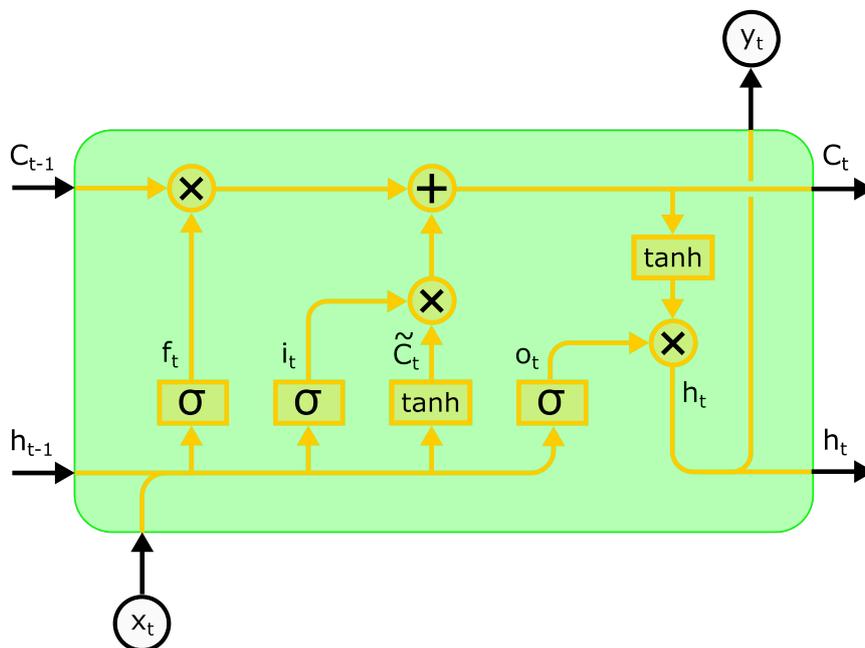


Figure 3.13: LSTM cell inner structure.

This architecture gives the ability to the network to learn long time dependencies, thanks to the presents of a new inner cell state and the presence of the gates, solving the short-term memory issue of the classic RNNs. Moreover, the presence of the simple path associated with the inner cell state (the straight line that links the previous inner cell state with the current one), makes possible at the same time to simplify the backpropagation step and to dramatically reduce the chance of having problem with vanishing or exploding gradient.

There exist many different representations of the LSTM cell that re-arrange the inner gates in a different configuration but the result remains the same. A particular reformulation, which is commonly used, is called Gated Recurrent Unit (GRU) and it uses only two gates called reset and update with the purpose of summarizing the operations that input, forget and output gates do in the classical LSTM cell.

In [38] they examined the SoC estimation performances obtained by using 1 hidden layer RNN and LSTM-RNN. The dataset was built by collecting measurements of current, temperature, voltage and ground-truth SoC with a sampling frequency

of 10Hz in the context of common discharge tests drive cycles (US06, HWFET, UDDS, LA92), with different ambient temperatures (0°C, 10°C and 25°C) on a 2.9 Ah LNMC/Graphite 18650 lithium-ion battery cell. The test set was obtained by selecting randomized portion of the data obtained in the previous mentioned driving cycles tests. The results proved that the LSTM-RNN was able to outperform the RNN by achieving an average MAE and MAX estimation percentage error over the three different temperatures, of 0.97% and 8.7%, in contrast with the RNN that obtained an average MAE and MAX estimation percentage error of 3.15% and 12.92%.

Also in [39] the usage of LSTM-RNN was investigated. The dataset was constructed by collecting data through ten unique drive cycles tests which were the results of a random mix of HWET, UDDS, LA92 and US06 drive cycles at ambient temperatures of 0°C, 10°C and 25°C. Regenerative braking was considered in the tests at temperatures above 10°C. Eight out of ten cycles were used as training data while the remaining portion as validation set. Other two tests cases were considered for model evaluation purpose that consisted in a 1 C rate charging scenario at 25°C and in an increasing temperature stress test where temperature raised from 10°C to 25°C. According to the study, the drive cycle power profiles, which were referred to an electric Ford F150 truck with a 35 kWh battery pack, were scaled to a single 2.9 Ah Panasonic 18650 NCA cell. Different performed training experiments, showed that by increasing the network time depth (increasing the historical dependency in input data) the model accuracy increased and the minimal number of driving cycles in the training set from which start to get a MAE accuracy below 1% was 3. When the network was trained using 500 time depth on the eight training cycle set at 10°C it was able to achieve in the remaining validation sets a MAE of 0.8% and 1.2% respectively, while in the charging test case it obtained a MAE of 0.68%. Afterward the network was firstly trained by considering the nine training partitions at different temperatures, using a time depth equals to 1000, and then when evaluated on the remaining dataset portion it achieved a MAE of 2.08%, 0.78%, 0.77% in correspondence of the temperatures 0°C, 10°C and 25°C. Finally the same model was evaluate on the varying temperature case and it reached 1.6% MAE.

## 3.4 Summary

In the previous paragraphs the most relevant SoC and SoH estimation approaches are analyzed and different research articles are considered as testimonial example of their usage.

From the utilization perspective, depending on which context is convenient to adopt a specific method, these estimation approaches can be grouped into offline and online categories. Offline techniques are useful to acquire data in a laboratory environment during battery tests, where high quality sensor can be used and a better control on the battery operations can be guaranteed. Moreover, they allow to support other methods by providing ground truth data. Offline approaches are Coulomb Counter (paragraph 3.2.1) and OCV map (paragraph 3.3.1). In contrast, online methods take support from offline methods in the development step, but can be implemented on a BMS of a EV vehicle. Online methods are KF, EKF and UKF from the first principle approaches family (paragraph 3.2.2), FFNN, RNN and LSTM-RNN from the data-driven family (paragraph 3.3.2).

In the literature is hard to find a clean implementation of these methods and typically they are modified in order to enhance the estimation performance.

Remarkable is the accuracy in estimating SoC and SoH achieved by machine learning techniques, that, developed starting from time series obtained from driving cycle data, are able to overwhelm classical methods (i.e. filters approaches) which, in contrast, are developed starting from battery datasheets and require additional battery tests.

# Chapter 4

## Application scenario

In the first two chapters, an overview on the main topics regarding the SoC and the SoH are analyzed using a top-down approach and the most relevant estimation techniques that exist in the literature are described.

At this point, the objective of this thesis work is presented: The goal consists in developing and comparing two estimation schemes to estimate SoC and SoH considering a system simulator. The chosen estimation techniques are taken respectively from the first principles approaches family (classical approach) through an enhanced version of the Extended Kalman Filter (EKF), and the data-driven family (machine learning approach) by using a Non-linear AutoRegressive with eXogenous input Neural Network (NARX-NN).

The target simulated system (described in the next paragraph) emulates an electric vehicle that undergoes to test bench dynamometric driving cycles according to WLTP standard at a given ambient temperature. The considered vehicle is a Fiat Panda and the target battery pack consists in a 28S1P Samsung SDI 94Ah. Among all the available information that the simulator can provide, only battery voltage, current and temperature are significative for the estimation techniques: they are used to feed the estimators in order to estimate the SoC and the SoH of the battery pack.

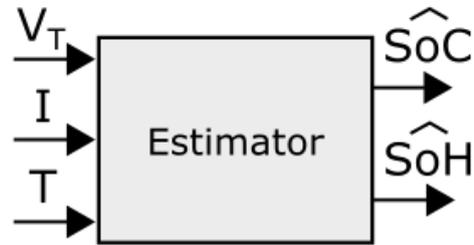


Figure 4.1: Conceptual schema of the estimator.

Before to develop the estimators is necessary to prepare the simulator to be able to work properly in different battery pack aged scenarios. All the details are discussed in paragraph 4.1 and 4.2. After the end of the simulator setup, the work is presented firstly from the perspective of the EKF (chapter 5) and then the NN is taken into consideration (chapter 6). In the conclusion chapter, the two estimation approaches are compared and conclusive considerations are presented.

## 4.1 Dynamic simulator

The dynamic simulator emulates the behaviour of a vehicle that execute a given dynamometric cycle test at a specific ambient temperature and road profile. The principles elements that are included consist in a vehicle dynamic model with a single degree of freedom, a motor model, a battery model and a thermal model of the battery and the motor. Among the possible outputs, the simulator allows to provide the signals of interest such us battery voltage, current, temperature and ground truth SoC. The Simulink schema that implements the dynamic simulator has been given and its validity has been proved outside of this thesis work. Its implementation is provided in appendix (A) at figure A.1.

One important element present in the schema is the battery model, called "datasheet battery model". As the name suggests, it is based on data that are provided through the cell datasheets. According to its documentation [40] the needed parameters are:

- Rated capacity at nominal temperature
- Open circuit voltage table data at given SoC breakpoints
- Internal resistance table data with battery temperature and SoC breakpoints
- Topology of the battery pack (number of cell in series and in parallel)
- Initial battery capacity

Unfortunately, the datasheet model doesn't support aging mechanisms and so the only way to emulate aged behaviour is to provide the needed parameters at different aged states of the battery. These information are only partially present since most of the data provided by the manufacturer refer to fresh cell. For this reason, another battery model that includes aging mechanisms is considered with the purpose of providing, at different SoH states, the required parameters for the datasheet battery model. In the next paragraph the battery model is described along with the tests performed with the aim of estimate the required parameters.

## 4.2 Battery Model

The battery pack to be modeled is composed by a single battery module with 28 prismatic Samsung SDI 94Ah cells combined in a series fashion. The main characteristics of the cells are reported below:

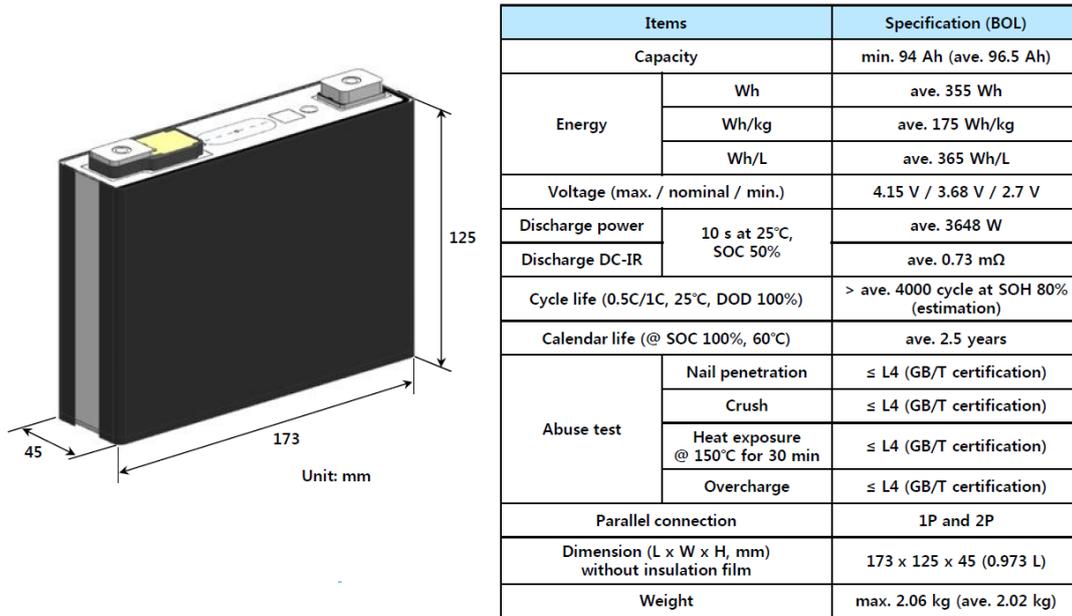


Figure 4.2: Overview on the main characteristics of the Samsung SDI 94Ah cell.

The battery model is based on a Simulink Specialized Power Systems element block called "Generic Battery Model" (GBM) which implements a generic dynamic model that represents, among the most popular types of rechargeable batteries, also lithium-ion batteries. According to the GBM documentation, the model shows a maximum error of 5% during experimental validation: when SOC is between 10% and 100%, for the charge when current is between 0 and 2 C and for the discharge when the current is between 0 and 5 C. The model is based on a  $R_{INT}$  ECM (paragraph 3.2.2) and can model temperature and aging effects. It accepts as input a current stimuli and gives as output terminal voltage, SoC, maximum battery capacity, the age of the battery (Equivalent full cycle) and the internal battery temperature.

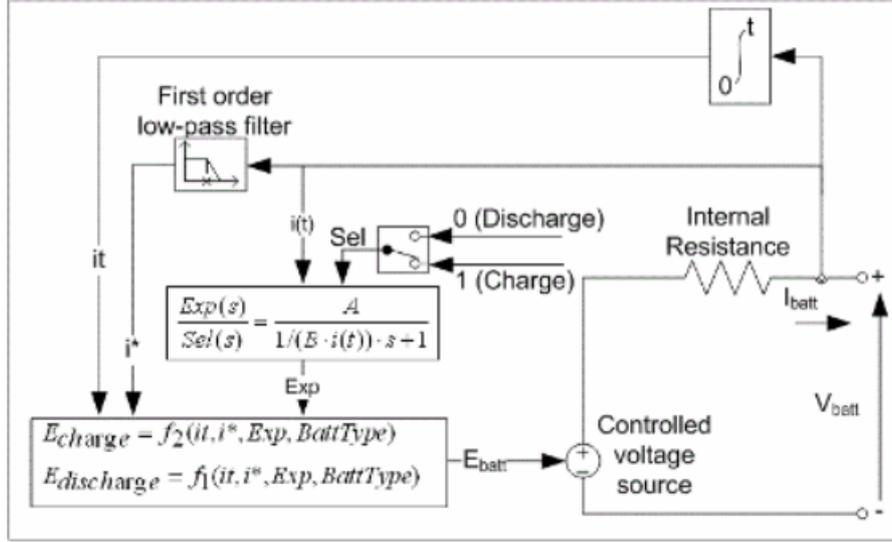


Figure 4.3: Electric model implemented inside the GBM. [41]

The main assumptions of the model declared in the documentation [41] are the following:

- The internal resistance is assumed to be constant during the charge and discharge cycles and does not vary with the amplitude of the current.
- The parameters of the model are derived from the discharge characteristics. The discharging and charging characteristics are assumed to be the same.
- The capacity of the battery does not change with the amplitude of the current (there is no Peukert effect).
- The self-discharge of the battery is not represented. It can be represented by adding a large resistance in parallel with the battery terminals.
- The battery has no memory effect.

The adopted equations that describe the particular working mechanisms of the battery model are reported into its documentation [41].

The parameters needed to the GBM can be taken from the cell datasheets, through tabular data and discharge curve graphs at different temperature. The information then can be adapted in order to represent a given battery pack with a given cell topology.

In the following table, all the parameters needed to the GBM to model the considered battery pack are reported:

TAB	PARAMETERS	CELL	PACK 28S1P	UNITY
Parameters	Nominal Voltage	3.68	103.04	V
	Rated capacity	94	94	Ah
	Initial SoC	100	100	%
	Battery response time	855	855	s
Discharge	Maximum capacity	94	94	Ah
	Cut-off voltage	2.7	75.6	V
	Fully charged voltage	4.15	116.2	V
	Nominal discharge current	31.33	31.33	A
	Internal resistance	0.00073	0.02044	$\Omega$
	Capacity at nominal voltage	85.0087	85.0087	Ah
	Exponential zone [V, Q]	[3.69, 38.96]	[103.25, 39]	[V, Ah]
Temperature	Initial cell temperature	25	25	$^{\circ}\text{C}$
	Nominal ambient temperature T1	25	25	$^{\circ}\text{C}$
	Second ambient temperature T2	-25	-25	$^{\circ}\text{C}$
	Maximum capacity	76.26	76.26	Ah
	Initial discharge voltage	4.14	115.92	V
	Voltage @ 90% max. capacity	3.68	103.04	V
	Exponential zone [V, Q]	[3.97, 4.62]	[111.32, 4.62]	[V, Ah]
	Thermal resistance, cell-to-amb.	0.06	0.06	$^{\circ}\text{C}/\text{W}$
	Thermal time const., cell-to-amb.	1000	1000	s
	Heat loss difference [charge vs discharge]	0	0	W
Aging	Initial battery age (Equivalent full cycle)	0	0	cycle
	Aging model sampling time	1.8e4	1.8e4	s
	Ambient temperature Ta1	25	25	$^{\circ}\text{C}$
	Capacity @ EOL	75.2	75.2	Ah
	Internal resistance @ EOL	0.00146	0.04088	$\Omega$
	Charge current [Ic, Icm <sub>ax</sub> ]	[47, 270]	[47, 270]	[A, A]
	Discharge current [Id, Id <sub>max</sub> ]	[94, 413]	[94, 413]	[A, A]
	Cycle life @ 100% DOD, Ic and Id	4000	4000	cycle
	Cycle life @ 25% DOD, Ic and Id	28000	28000	cycle
	Cycle life @ 100% DOD, Ic and Id <sub>max</sub>	3551	3551	cycle
	Cycle life @ 100% DOD, Icm <sub>ax</sub> and Id	3900	3900	cycle
	Ambient temperature Ta2	5	5	$^{\circ}\text{C}$
	Cycle life @ 100% DOD, Ic and Id	3793	3793	cycle

Table 4.1: Generic battery model parameters. See GBM documentation for parameters details [41].

The parameters relative to the cell in table 4.1 are partially taken from the provided Samsung SDI 94Ah datasheets, and partially from the information contained in a given CNR report which is about experimental tests on a battery module of the same battery cell with 10S1P configuration. Missing parameters that are neither present in datasheet nor in the report are replaced with the default values provided by the GBM. The parameters relative to the battery pack 28S1P are computed starting from the cell ones and according to the GBM documentation.

The Simulink schema that implements the battery model is provided in Appendix A, figure A.2. In order to evaluate the goodness of the GBM, using its Simulink implementation, a discharge test is performed and the results are compared with the ones present in the CNR report according to the same test procedure:

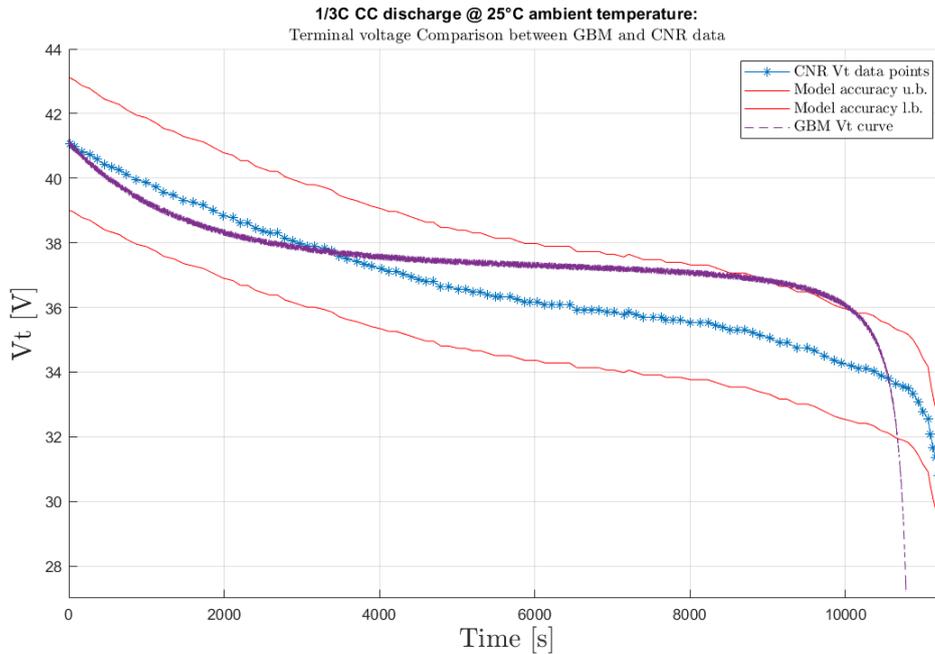


Figure 4.4: The terminal voltage response produced by the GBM is adapted to the 10S1P configuration in order to be comparable with the one provided by the CNR. Moreover, the data points of the CNR terminal voltage response have been extracted from the relative noisy curve (depicted in an graph) by using a self-made software tool.

The obtained results, according to the characteristics and the stated accuracy of the GBM, are good enough to confirm the validity of the parameters setting.

As mentioned in the previous paragraph (4.1), the purpose of the GBM is to retrieve information that are needed to the datasheet model of the dynamic simulator, to work properly at different aged states. In particular, in order to obtain OCV and internal resistance values for different temperature, SoC and SoH, a First Order Thévenin (FOTM) ECM is identified through the acquired data obtained performing some tests on the GBM. The obtained parametric model is also adopted to support the development of the EKF which is addressed in chapter 5.

The tests are performed using the following sets of ambient temperatures and equivalent full cycles (Efc) which represent a specific battery aged status in terms of the number of complete charge-discharge cycles (0 represent battery BOL and 4000 is the EOL according to datasheet data):

$$\begin{aligned} T_a &\in \{15,20,25,30,35,40\} \quad (^\circ C) \\ Efc &\in \{0,400,800,\dots,4000\} \quad (cycle) \end{aligned} \tag{4.1}$$

The next paragraphs enter into the details on how the needed parameters are obtained.

### 4.2.1 SoH ground truth evaluation

As discussed in paragraph 2.5, when battery ages, the maximum capacity tends to decrease over time under the so called capacity fade phenomenon. So, one possibility to evaluate the SoH at a given time is to apply equation 2.5. But, since the battery capacity changes also with battery temperature, the 2.5 must be contextualized to a given temperature range.

Using the Simulink schema A.2, a 1C constant discharge tests is executed on the GBM according to all the combination of the chosen ambient temperature and Efc, which are reported in 4.1. The battery is in thermal equilibrium with the ambient temperature and start in its full charge state (SoC = 100%) at a given Efc.

The test procedure ends when the terminal voltage reaches its cut-off value. At the end of the tests, the maximum capacity given by the battery model for each

combination of temperatures and Efc's are mapped into the relative SoH reference by applying the 2.5.

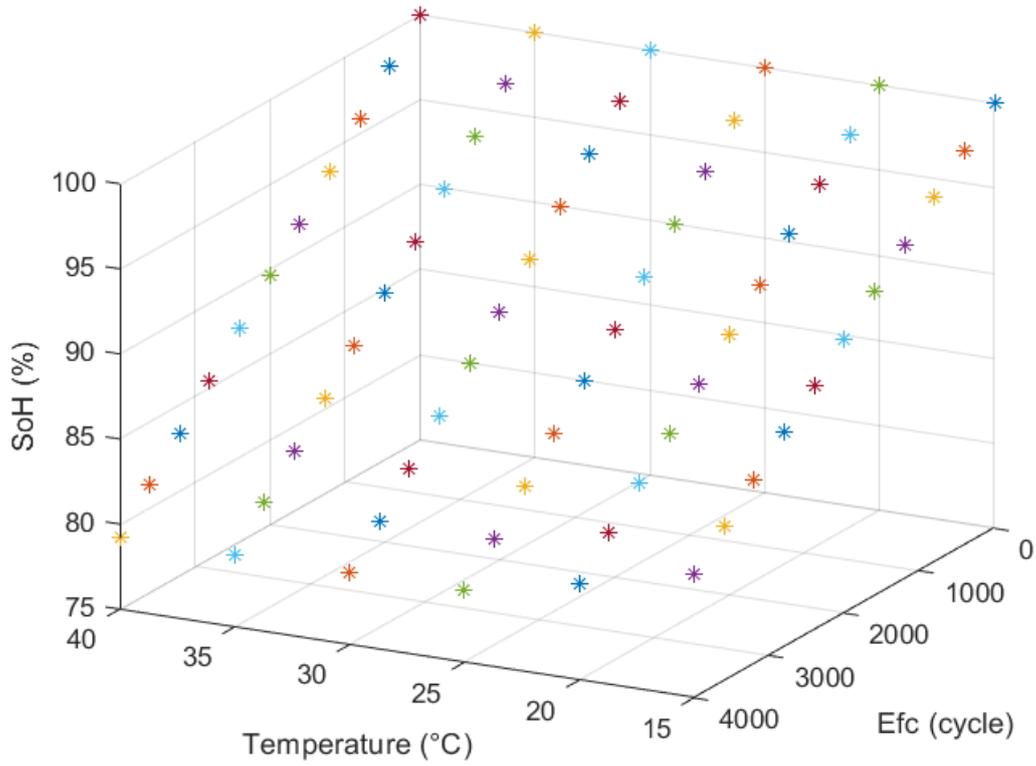


Figure 4.5: SoH reference points for each combination of temperature and Efc. The obtained SoH value are assumed to be constant in the temperature range  $[T_{amb}, T_{amb} + 5]$  °C.

## 4.2.2 Coulomb efficiency computation

Given a fully charged/discharged cell battery of  $Q_n$  [Ah] capacity, it takes  $Q_n/c$  hours to fully discharge/charge when it is stimulated by a constant discharge/charge current at  $c$  C-rate. Actually, this is not always true and the factor  $\eta$  (coulomb efficiency) has to be found in order to determine how long the battery takes to fully discharge/charge at different ambient temperature and age conditions. This parameter is important to allow to compute the SoC or SoH through the Coulomb Counter technique (paragraph 3.2.1) in order to provide ground truth data.

The parameter  $\eta$  depends on the temperature and particular aged status of the battery. So, In order to find its values at different temperatures as the battery ages, different constant discharge tests (CDT) are performed on the battery model by considering the simulink schema A.2 in appendix A.

In particular the test consists in applying a 1C constant discharge current to the battery, at a given ambient temperature and at specific aged status in terms of equivalent full cycles (4.1). The battery is put in thermal equilibrium with the ambient temperature, and start at its full-charged state and at a specific Efc. The test ends when the battery terminal voltage reaches its cut-off value.

Starting from equation 3.1, the formula for computing the Coulomb efficiency is derived accordingly to the performed choices during the tests ( $I(t) = Q_n * 1 C$ ,  $t \geq 0$  and  $SoC(0) = 100\%$ ,  $T = T_i$ ,  $Efc = Efc_j$ ).

$$\begin{aligned}
 SoC(t) &= SoC(0) - \frac{\eta(T, Efc)}{Q_n} \int_0^t Q_n * 1C d\tau \\
 SoC(t) &= SoC(0) - \eta(T, Efc) \int_0^t d\tau \\
 SoC(t) &= SoC(0) - \eta(T, Efc)t \\
 SoC(t_{end}) &= SoC(0) - \eta t_{end} \\
 \eta(T, Efc) &= \frac{SoC(0) - SoC(t_{end})}{t_{end}}
 \end{aligned} \tag{4.2}$$

Knowing the relation between Efc and SoH from the previous paragraph, the obtained values of  $\eta$  are fitted with a one degree polynomial in the SoH variable for each temperature reference, by using Least Square. The obtained results are the following:

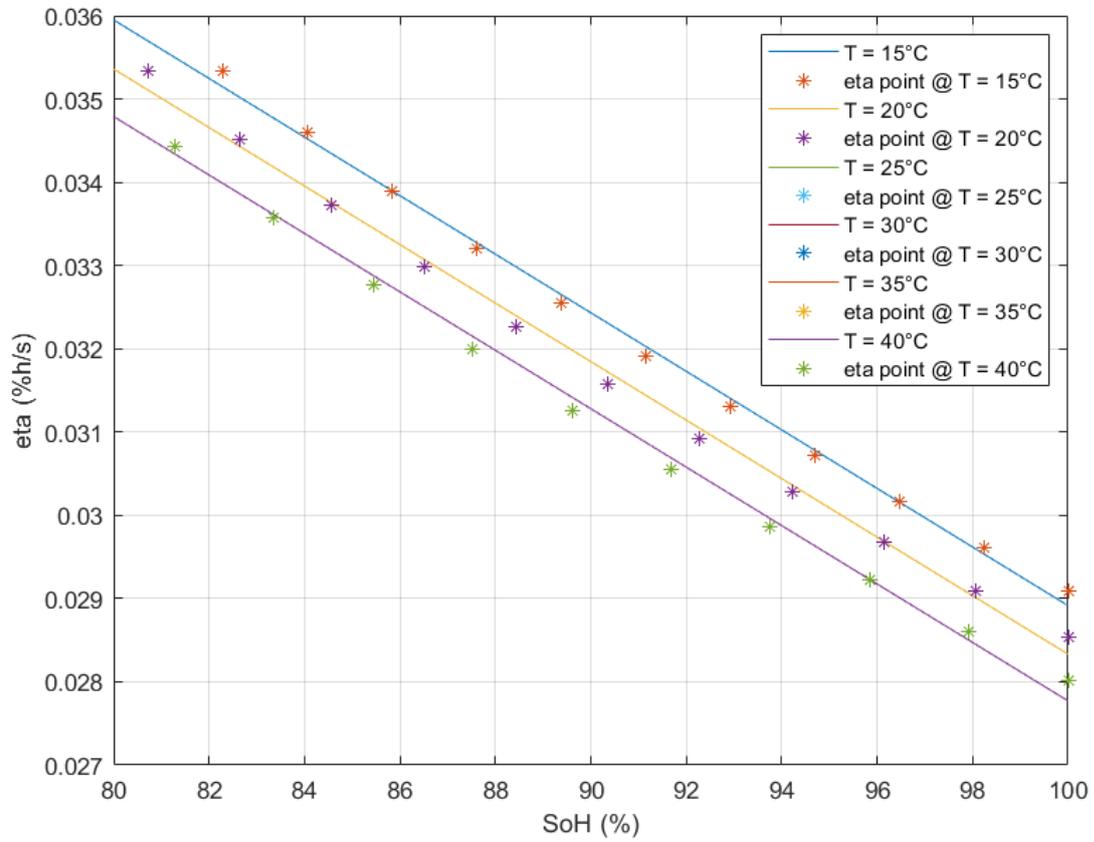


Figure 4.6: Coulomb efficiency points and fitted polynomials for different temperature. The eta points and the relative polynomials are assumed to be constant in the temperature range  $[T_{amb}, T_{amb} + 5]$  °C.

### 4.2.3 Open Circuit Voltage curve identification

According to the simulink schema A.5 in appendix A, a pulse discharge test (PDT) is performed for each combination of temperatures and Efc<sub>s</sub> defined in 4.1, in order to acquire OCV data points at different SoC percentages level. The chosen SoC resolution to acquire OCV points is 5% and the C-rate is 0.5C. The current profile according to the test is obtained by alternating constant discharge current at a given C rate, to relaxing period:

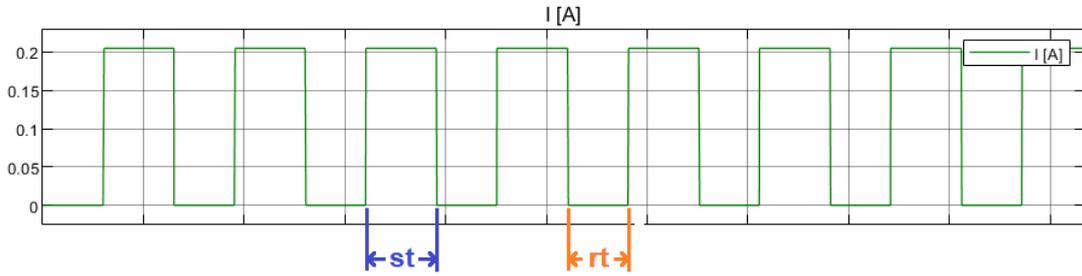


Figure 4.7: Example of a Pulse discharge current profiles. st: stimuli time, rt: rest time.

The rest time has to be sufficient long in order to make the transient terminal voltage response ends, and then, make possible to acquire the relative OCV points. It has been chosen to be 30 minutes long. The stimuli time, used to guarantee a decrease in term of SoC according to the SoC resolution, is found according to the following equations:

Given two time instants  $t_2 > t_1 > 0$ :

$$SoC(t_2) = SoC(0) - \frac{\eta}{Q_n} \int_0^{t_2} Q_n * 1C d\tau \quad (4.3)$$

$$SoC(t_1) = SoC(0) - \frac{\eta}{Q_n} \int_0^{t_1} Q_n * 1C d\tau \quad (4.4)$$

By subtracting each member of eq. 4.4 from eq. 4.3:

$$\begin{aligned}
 \Delta SoC &= SoC(t_2) - SoC(t_1) = -\frac{\eta}{Q_n} \int_0^{t_2} I(\tau) d\tau + \frac{\eta}{Q_n} \int_0^{t_1} I(\tau) d\tau \\
 \Delta SoC &= -\frac{\eta}{Q_n} \int_0^{t_2} I(\tau) d\tau - \frac{\eta}{Q_n} \int_{t_1}^0 I(\tau) d\tau \\
 \Delta SoC &= -\frac{\eta}{Q_n} \int_{t_1}^{t_2} I(\tau) d\tau \\
 \Delta SoC &= -\frac{\eta}{Q_n} \int_{t_1}^{t_2} Q_n * Crated\tau \\
 \Delta SoC &= -\eta Crate \int_{t_1}^{t_2} d\tau \\
 \Delta SoC &= -\eta Crate(t_2 - t_1) = -\eta Crate \Delta t \\
 \Delta t &= -\frac{\Delta SoC}{\eta Crate}
 \end{aligned} \tag{4.5}$$

Where  $\Delta t$  is the stimuli time and  $\Delta SoC$  is the resolution.

As an example, The following images refers to one of the performed tests, which is relative to 25 °C ambient temperature, Efc 1600 (corresponding to 91.68% of SoH):

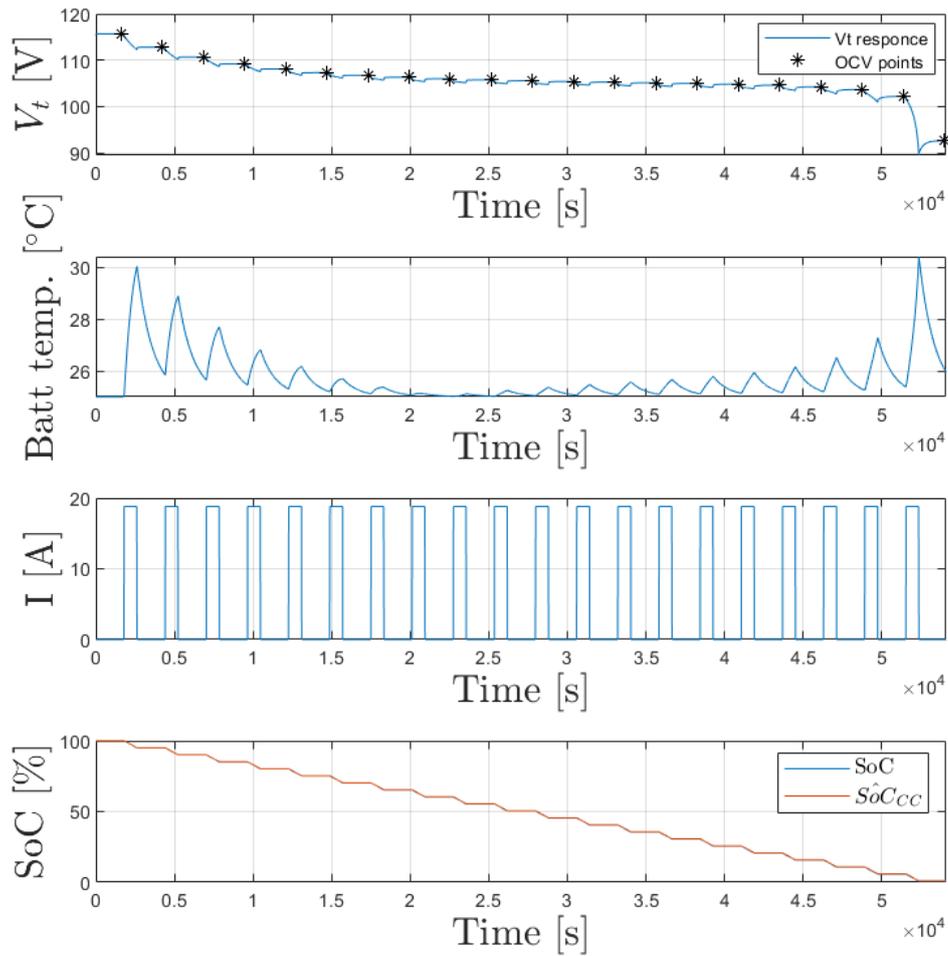


Figure 4.8: The acquired sampled (1Hz) data during PDT. During the tests the coulomb counter estimation technique estimates SoC which is compared with the one produced by the battery model.

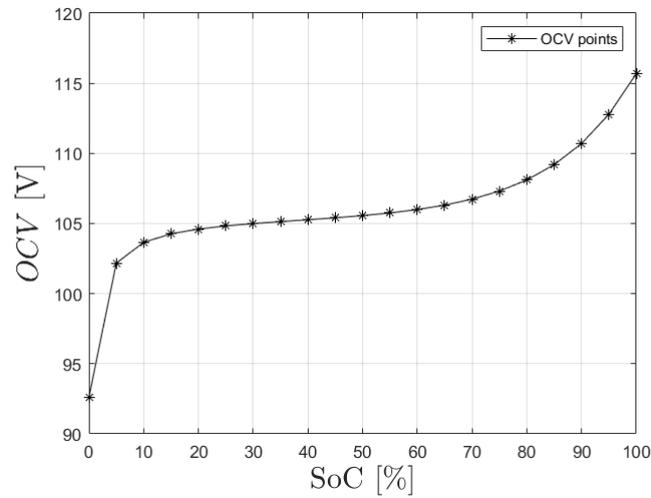


Figure 4.9: Linearly interpolated OCV data points vs SoC percentage.

The obtained OCV data points at different temperature, Efc, and SoC are used to find a two variable polynomial surface for different temperature. The chosen degree is 11 and it has been found using "polyfitn" function [42].

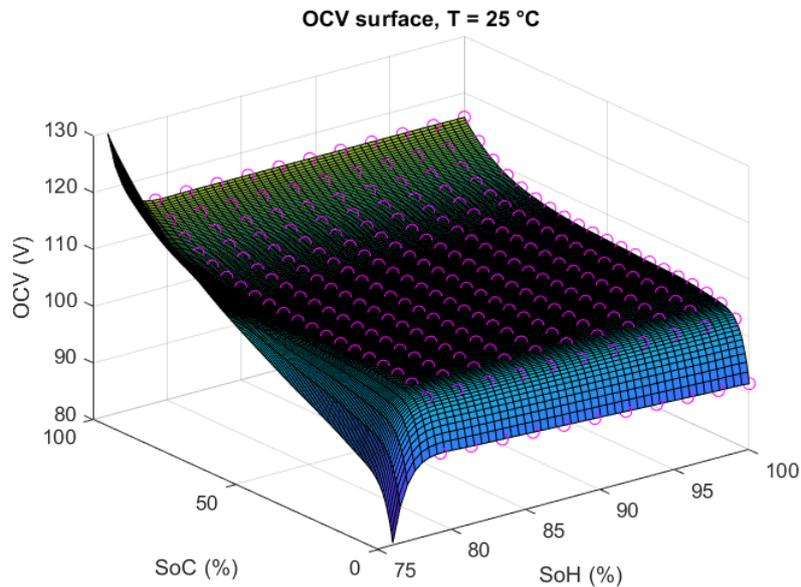


Figure 4.10: The OCV points and the relative surfaces are assumed to be constant in the temperature range  $[T_{amb}, T_{amb} + 5]$  25°C.

The obtained OCV points are adapted and compared with the ones provided by the cell manufacturer:

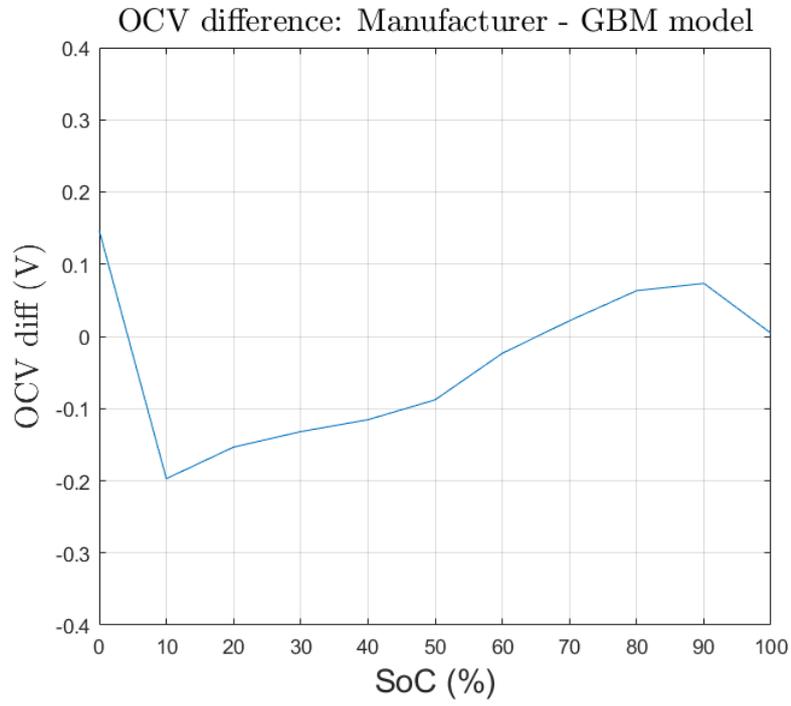


Figure 4.11: Difference between identified OCV data points and the ones provided by the cell manufacturer at BOL condition, temperature of 25°C and 10 % SoC resolution.

#### 4.2.4 Equivalent Circuit Model identification

As mentioned in paragraph 4.2, the First Order Thévenin Model is the chosen ECM. The system of equations that describe its behaviour is:

$$FOTM : \begin{cases} C \frac{dV_1}{dt} + \frac{V_1}{R_1} = I \\ V_T = OCV(SoC) - V_1 - R_{int}I \end{cases} \quad (4.6)$$

By applying Laplace transform, the first equation of 4.6 becomes:

$$V_1 = \frac{I}{C(s + \frac{1}{R_1C})} \quad (4.7)$$

Substituting 4.8 in the second equation of 4.6:

$$V_D = V_T - OCV(SoC) = -\frac{I}{C(s + \frac{1}{R_1C})} - R_{int}I \quad (4.8)$$

Then, the transfer function is:

$$\frac{V_D}{I} = -\frac{R_{int}s + \frac{R_{int}+R_1}{CR_1}}{s + \frac{1}{R_1C}} \quad (4.9)$$

By applying the forward rule and passing to z domain ( $s = (z-1)/\Delta t$ ) the 4.9 can be rewritten as:

$$\frac{z-1}{\Delta t}V_D + \frac{V_D}{R_1C} = -\frac{R_{int}}{\Delta t}(z-1)I - \frac{R_{int}+R_1}{R_1C}I \quad (4.10)$$

By anti-transforming from z domain to discrete time, the 4.10 become an ARX representation of the system:

$$V_D(k+1) + \left(\frac{\Delta t}{\tau} - 1\right)V_D(k) = -R_{int}I(k+1) + \left(R_{int} - \frac{\Delta t}{\tau}(R_{int}+R_1)\right)I(K) \quad (4.11)$$

Where k is a discrete time instant,  $\Delta t$  sampling interval and  $\tau = R_1C$ . By changing variable and parametrizing 4.11:

$$V_D(k) + a_1V_D(k-1) = b_1I(k) + b_2I(K-1) \quad (4.12)$$

where:

$$\begin{cases} a_1 = \frac{\Delta t}{\tau} - 1 \\ b_1 = -R_{int} \\ b_2 = R_{int} - \frac{\Delta t}{\tau}(R_{int} + R_1) \end{cases} \quad (4.13)$$

Equation 4.14 can be rearranged as following:

$$\begin{aligned} V_D(k) &= -a_1 V_D(k-1) = b_1 I(k) + b_2 I(k-1) \\ y(k) &= [-V_D(k-1)I(k-1)I(k)][a_1 b_1 b_2]^T \\ y &= A * \theta \end{aligned} \quad (4.14)$$

Eq. 4.14 can be used along with Least square algorithm in order to identify parameters  $a_1, b_1, b_2$  and consequently  $R_{int}, R_1, C$ , by using  $N$  sampled data.

$$\theta = (A^T A)^{-1} A^T y \quad (4.15)$$

$$\begin{cases} \tau = \frac{\Delta t}{a_1 + 1} \\ R_{int} = -b_1 \\ R_1 = (R_{int} - b_2) \frac{\tau}{\Delta t} - R_{int} \\ C = \frac{\tau}{R_1} \end{cases} \quad (4.16)$$

The ECM parameters  $R_{INT}, R_1, C$  and  $\tau$  are dependent from battery temperature, SoH and SoC. In order to find their values for all the combination of temperature, SoH and SoC different pulse discharge tests (PDT) at 1C are performed. Similarly to the previous tests, the temperature and Efc are chosen according to 4.1, while SoC goes from 100 % to 0 % with a resolution of 5 %. Before applying the test, the battery is in thermal equilibrium with the ambient temperature, start from its full-charged state and at a specific aged status (Efc). For each test, the stimuli time is computed according to equation 4.5.

The necessary data consist in current, terminal voltage samples (acquired at 1Hz) and OCV which is find at the previous paragraph. Each SoC percentage value is

associated with a batch of the acquired data and the relative OCV. Batches of data are chosen in the following way: The first batch at 100 % SoC corresponds to the discharge part of the voltage response, while all the other batches at different SoC level, correspond to the raising part of the voltage response. The first batch starts  $rt/9$  seconds before the current raising part and ends when the current pulse ends. The other batches start on the falling part of the current pulse and ends after  $8/9$   $rt$  seconds.

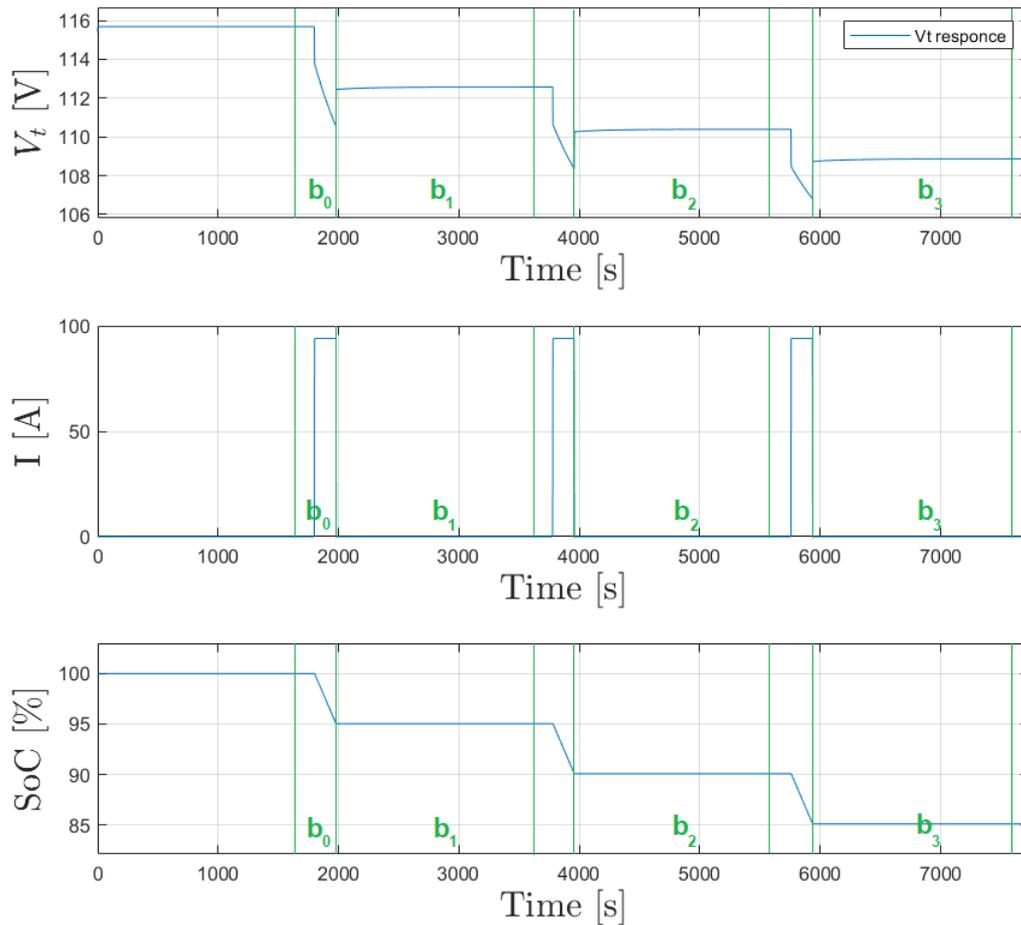


Figure 4.12: Example of batches partitioning for SoC interval [85, 100] %. The signals refers to a 1C PDT at 25°C and 100% SOH.

Inside each batch, the voltage response, current stimuli and the computed OCV are used along with equations 4.14, 4.15 and 4.16 to find the ECM parameters in the relative context.

The obtained parameters at different temperature, Efc, and SoC are used to find a two variable polynomial surface for different temperature by using "polyfitn" function [42]. The chosen polynomial degrees are: 8 for  $R_{INT}$ , 11 for  $R_2$ , 2 for C and 8 for  $\tau$ .

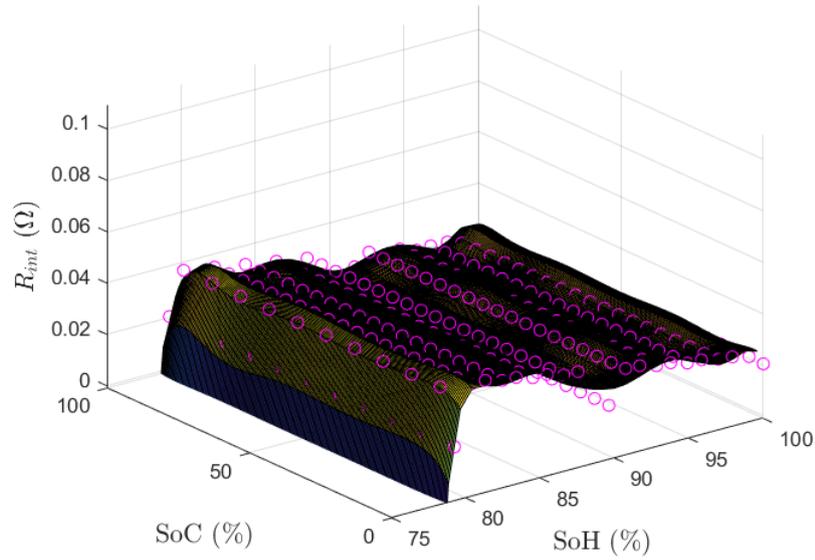


Figure 4.13: The  $R_{INT}$  points and the relative surface at 25°C.

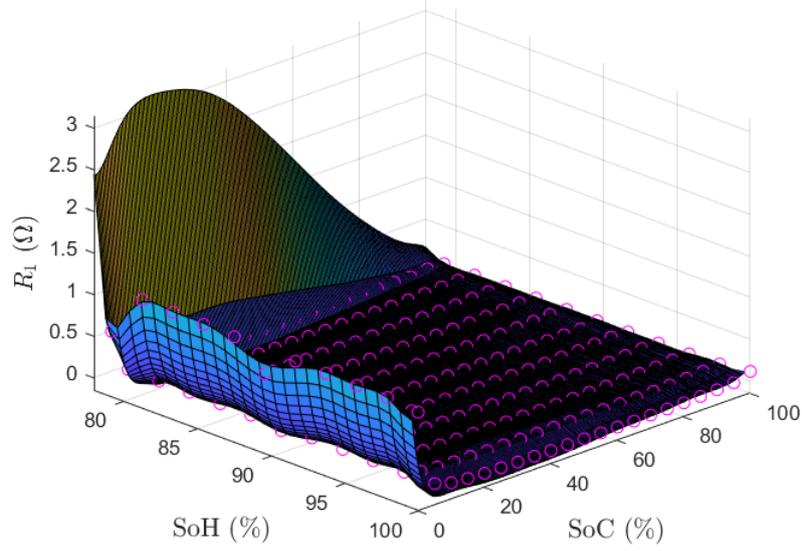


Figure 4.14: The  $R_1$  points and the relative surface at 25°C.

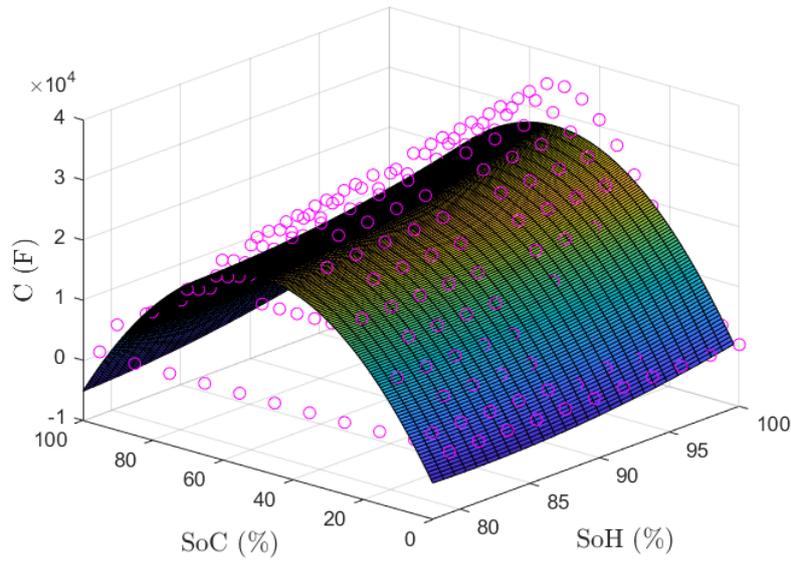


Figure 4.15: The  $C$  points and the relative surface at 25°C.

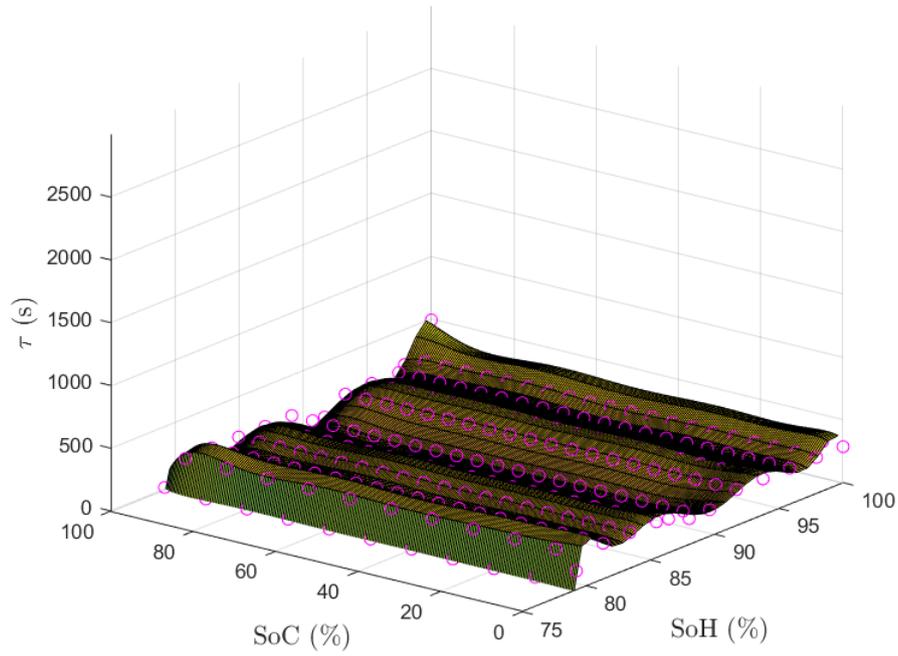


Figure 4.16: The  $\tau$  points and the relative surface at 25°C.

The parameters data are assumed to be constant in the temperature range  $[T_{amb}, T_{amb} + 5]$ .

### 4.2.5 Equivalent Circuit Model comparison

By using simulink schema A.7 in appendix A the identified ECM is compared with the GBM under a 1C CDT test at ambient temperature of 25 °C and 100% SoH. The voltage responses are acquired for both the models and compare to each other.

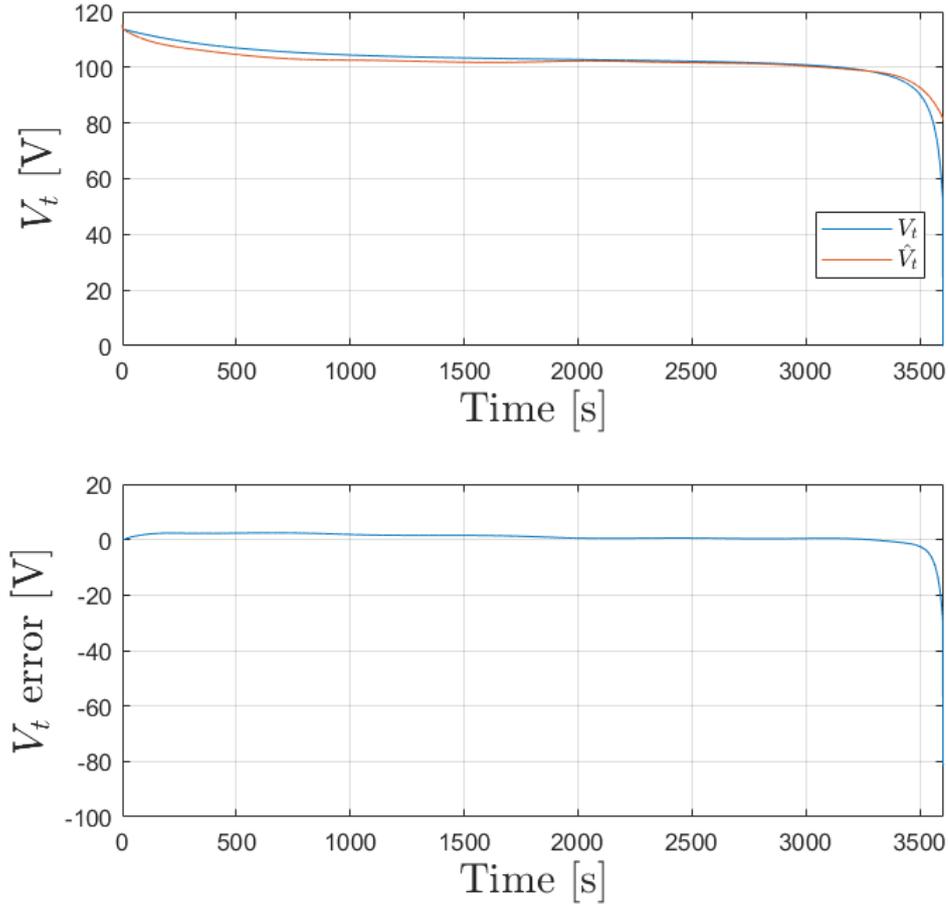


Figure 4.17: Comparison between FOTM and GBM under 1C CDT @ 25 °C and 100% SoH.

The voltage response of the ECM is pretty similar to the one produced by the GBM but become worse when SoC is near to 0. The average error is 0.7368 V, minimum error -81.5258 (SoC=0%) and the maximum error is 2.4375 V.

# Chapter 5

## Extended Kalman Filter

In paragraph 3.2.2 the first principles estimation approaches based on ECMs are addressed and the theory of the Extended Kalman Filter is explained. In this chapter the EKF is considered as classical approach and is enhanced with OCVmap technique 3.3.1 to better estimate SoC and SoH.

One important aspect regarding the EKF is the system model on which is based. For its development, the First Order Thévenin (FOTM) ECM is taken into account since it provides a good compromise between complexity and accuracy in representing the electrical behaviour of a given battery. The parameters of the model have been estimated in paragraph 4.2.4.

In order to derive the EKF algorithm, it is necessary to bring the FOTM equations into the following state-space representation:

$$S : \begin{cases} x_{k+1} = f(x_k, u_k) + w_k \\ y_k = h(x_k, u_k) + v_k \end{cases} \quad (5.1)$$

The considered FOTM is described by the following system of equations:

$$FOTM : \begin{cases} \frac{dV_1}{dt} + \frac{V_1}{\tau_{RC}} = \frac{I}{C} \\ V_T = OCV - V_1 - R_{int}I \\ \tau_{RC} = R_1C \end{cases} \quad (5.2)$$

The chosen system state vector is:

$$x_k = \begin{bmatrix} SoH_k \\ SoC_k \\ V_{1_k} \end{bmatrix} \quad (5.3)$$

The expression for  $SoH_k$  is obtained by modelling the decreasing behaviour of the SoH in time according to the calendar life provided by the cell manufacturer:

$$SoH(t) = -\frac{20}{78840000}t + 100 \quad (5.4)$$

Where 78840000 is the equivalent in seconds of 2.5 years, namely, the time estimated by the manufacturer that a cell need to wait until it reaches its EOL starting from its BOL, in worst operating conditions.

By applying derivative and aproximating it with finite difference:

$$SoH(k + 1) = -\frac{20}{78840000}T + SoH(k) \quad (5.5)$$

where T is considered as sampling time.

The expression for  $SoC_k$  is derived from the Coulomb Counter expression:

$$SoC(t) = SoC(t_0) - \frac{\eta}{Q_n} \int_{t_0}^t I(\tau) d\tau \quad (5.6)$$

Passing to discrete time the following substitution are performed:

$$\begin{cases} t = (k' + 1)T \\ t_0 = k'T \end{cases} \quad (5.7)$$

Where  $k'$  is a generic discrete time instant and T is the sampling time. The 5.6 become:

$$SoC((k' + 1)T) = SoC(k'T) - \frac{\eta}{Q_n} \int_{k'T}^{(k'+1)T} I(\tau) d\tau \quad (5.8)$$

By changing variable  $\tau = k'T + \sigma$ ,  $0 \leq \sigma \leq T$ , the previous eq. is rewritten as:

$$SoC((k' + 1)T) = SoC(k'T) - \frac{\eta}{Q_n} \int_0^T I(k'T + \sigma) d\sigma \quad (5.9)$$

Assuming a zero-order holder  $I(k'T + \sigma) = I(k'T)$  and then:

$$SoC((k' + 1)T) = SoC(k'T) - \frac{\eta}{Q_n} \int_0^T I(k'T) d\sigma \quad (5.10)$$

By solving the expression:

$$SoC((k' + 1)T) = SoC(k'T) - \frac{\eta}{Q_n} I(k'T)T \quad (5.11)$$

Finally, by changing variable  $k = k'T$ :

$$SoC(k + 1) = SoC(k) - \frac{\eta T}{Q_n} I(k) \quad (5.12)$$

By adding SoH dependency to the Coulomb Counter expression, making SoC and SoH directly dependent, the previous expression become:

$$SoC(k + 1) = SoC(k) - \frac{100\eta T}{SoH(k)Q_{BOL}} I(k) \quad (5.13)$$

The expression for  $V_{1_k}$  is obtained by integrating  $V_1$  in time  $[t_0, t]$  from 5.2.

$$V_1(t) = V_1(t_0)e^{\frac{t_0-t}{\tau RC}} + \frac{e^{\frac{-t}{\tau RC}}}{C} \int_{t_0}^t e^{\frac{\tau}{\tau RC}} I(\tau) d\tau \quad (5.14)$$

Passing to discrete time the following substitution are performed:

$$\begin{cases} t = (k' + 1)T \\ t_0 = k'T \end{cases} \quad (5.15)$$

The 5.14 become:

$$V_1((k' + 1)T) = V_1(k'T)e^{\frac{-T}{\tau RC}} + \frac{e^{\frac{-(k'+1)T}{\tau RC}}}{C} \int_{k'T}^{(k'+1)T} e^{\frac{\tau}{\tau RC}} I(\tau) d\tau \quad (5.16)$$

By changing variable  $\tau = k'T + \sigma$ ,  $0 \leq \sigma \leq T$ , the previous eq. is rewritten as:

$$V_1((k' + 1)T) = V_1(k'T)e^{\frac{-T}{\tau RC}} + \frac{e^{\frac{-(k'+1)T}{\tau RC}}}{C} \int_0^T e^{\frac{k'T+\sigma}{\tau RC}} I(k'T + \sigma) d\sigma \quad (5.17)$$

By Applying zero-order holder  $I(k'T + \sigma) = I(k'T)$  and then:

$$V_1((k' + 1)T) = V_1(k'T)e^{\frac{-T}{\tau_{RC}}} + \frac{e^{\frac{-(k'+1)T}{\tau_{RC}}}}{C} \int_0^T e^{\frac{k'T+\sigma}{\tau_{RC}}} I(k'T) d\sigma \quad (5.18)$$

$$V_1((k' + 1)T) = V_1(k'T)e^{\frac{-T}{\tau_{RC}}} + \frac{e^{\frac{-T}{\tau_{RC}}}}{C} I(k'T) \int_0^T e^{\frac{\sigma}{\tau_{RC}}} d\sigma \quad (5.19)$$

By solving equation 5.19 and by performing some simplification:

$$V_1((k' + 1)T) = V_1(k'T)e^{\frac{-T}{\tau_{RC}}} + R_1(1 - e^{\frac{-T}{\tau_{RC}}})I(k'T) \quad (5.20)$$

By changing variable  $k=k'T$ , the final expression is obtained:

$$V_1(k + 1) = V_1(k)e^{\frac{-T}{\tau_{RC}}} + R_1(1 - e^{\frac{-T}{\tau_{RC}}})I(k) \quad (5.21)$$

So passing to discrete time, equations of the FOTM model (5.2) become:

$$FOTM : \begin{cases} V_{1_{k+1}} = V_{1_k} e^{\frac{-T}{\tau_{RC}}} + R_1(1 - e^{\frac{-T}{\tau_{RC}}})I_k \\ V_{T_k} = OCV(SoC_k) - V_{1_k} - R_{int}I_k \\ \tau_{RC} = R_1C \end{cases} \quad (5.22)$$

At this point by considering equations 5.5, 5.13 and 5.22 the state space representation of the system is:

$$\begin{cases} \begin{bmatrix} SoH_{k+1} \\ SoC_{k+1} \\ V_{1_{k+1}} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & e^{\frac{-\Delta T}{\tau_{RCk}}} \end{bmatrix} \begin{bmatrix} SoH_k \\ SoC_k \\ V_{1_k} \end{bmatrix} + \begin{bmatrix} -\frac{20\Delta T}{78840000I_k} \\ -\frac{100\eta_k\Delta T}{SoH_k Q_{BOLk}} \\ R_{1_k}(1 - e^{\frac{-\Delta T}{\tau_{RCk}}}) \end{bmatrix} I_k + w_k \\ y_k = OCV_k - V_{1_k} - R_{int_k}I_k + v_k \end{cases} \quad (5.23)$$

where  $\Delta T$  is the sampling time,  $\tau_{RCk} = \tau_{RC}(T_k, SoH_k, SoC_k)$ ,  $\eta_k = \eta(T_k, SoH_k)$ ,  $Q_{BOL} = Q_{BOL}(T_{amb})$ ,  $\mathbb{R}_{1_k} = \mathbb{R}_1(T_k, SoH_k, SoC_k)$  and  $OCV_k = OCV(T_k, SoH_k, SoC_k)$

The system is non-linear and Jacobian matrices  $F_k$ ,  $H_k$  has to be computed:

$$\begin{cases} F_k = \frac{\partial f(x,u)}{\partial x} \Big|_{(x,u)=(\hat{x}_k,u_k)} \\ H_k = \frac{\partial h(x,u)}{\partial x} \Big|_{(x,u)=(\hat{x}_k,u_k)} \end{cases} \quad (5.24)$$

According to 5.23, the matrices  $F_k$  and  $H_k$  can be computed as:

$$F_k = \begin{bmatrix} 1 & 0 & 0 \\ f_{2,1} & 1 & 0 \\ f_{3,1} & f_{3,2} & e^{\frac{-\Delta T}{\tau_{RC}(T_k, \hat{S}\hat{o}H_k, \hat{S}\hat{o}C_k)}} \end{bmatrix} \quad (5.25)$$

Where:

$$f_{2,1} = -\frac{100\Delta T I_k}{Q_{BOL}(T_{amb})} \frac{\frac{\partial \eta(T_k, \hat{S}\hat{o}H_k)}{\partial S\hat{o}H} S\hat{o}H_k - \eta(T_k, \hat{S}\hat{o}H_k)}{S\hat{o}H_k^2} \quad (5.26)$$

$$\begin{aligned} f_{3,1} = & \Delta T V_{1k} e^{-\frac{\Delta T}{\tau_{RC}(T_k, \hat{S}\hat{o}H_k, \hat{S}\hat{o}C_k)}} \frac{\frac{\partial \tau_{RC}(T_k, \hat{S}\hat{o}H_k, \hat{S}\hat{o}C_k)}{\partial S\hat{o}H}}{\tau_{RC}(T_k, \hat{S}\hat{o}H_k, \hat{S}\hat{o}C_k)^2} + \\ & + I_k \frac{\partial R_1(T_k, \hat{S}\hat{o}H_k, \hat{S}\hat{o}C_k)}{\partial S\hat{o}H} (1 - e^{-\frac{\Delta T}{\tau_{RC}(T_k, \hat{S}\hat{o}H_k, \hat{S}\hat{o}C_k)}}) + \\ & - \Delta T I_k R_1(T_k, \hat{S}\hat{o}H_k, \hat{S}\hat{o}C_k) \frac{\frac{\partial \tau_{RC}(T_k, \hat{S}\hat{o}H_k, \hat{S}\hat{o}C_k)}{\partial S\hat{o}H}}{\tau_{RC}(T_k, \hat{S}\hat{o}H_k, \hat{S}\hat{o}C_k)^2} e^{-\frac{\Delta T}{\tau_{RC}(T_k, \hat{S}\hat{o}H_k, \hat{S}\hat{o}C_k)}} \end{aligned} \quad (5.27)$$

$$\begin{aligned} f_{3,2} = & \Delta T V_{1k} e^{-\frac{\Delta T}{\tau_{RC}(T_k, \hat{S}\hat{o}H_k, \hat{S}\hat{o}C_k)}} \frac{\frac{\partial \tau_{RC}(T_k, \hat{S}\hat{o}H_k, \hat{S}\hat{o}C_k)}{\partial S\hat{o}C}}{\tau_{RC}(T_k, \hat{S}\hat{o}H_k, \hat{S}\hat{o}C_k)^2} + \\ & + I_k \frac{\partial R_1(T_k, \hat{S}\hat{o}H_k, \hat{S}\hat{o}C_k)}{\partial S\hat{o}C} (1 - e^{-\frac{\Delta T}{\tau_{RC}(T_k, \hat{S}\hat{o}H_k, \hat{S}\hat{o}C_k)}}) + \\ & - \Delta T I_k R_1(T_k, \hat{S}\hat{o}H_k, \hat{S}\hat{o}C_k) \frac{\frac{\partial \tau_{RC}(T_k, \hat{S}\hat{o}H_k, \hat{S}\hat{o}C_k)}{\partial S\hat{o}C}}{\tau_{RC}(T_k, \hat{S}\hat{o}H_k, \hat{S}\hat{o}C_k)^2} e^{-\frac{\Delta T}{\tau_{RC}(T_k, \hat{S}\hat{o}H_k, \hat{S}\hat{o}C_k)}} \end{aligned} \quad (5.28)$$

The matrix  $H_k$  is computed as:

$$H_k = [h_1 \quad h_2 \quad -1] \quad (5.29)$$

Where:

$$h_1 = \frac{\partial OCV(T_k, SoH_k^p, SoC_k^p)}{\partial SoH} - \frac{\partial V_1}{\partial SoH} - \frac{\partial R_{int}(T_k, SoH_k^p, SoC_k^p)}{\partial SoH} I_k \quad (5.30)$$

and

$$\begin{aligned} \frac{\partial V_1}{\partial SoH} &= \frac{V_{1_{k+1}} - V_{1_k}}{SoH_{k+1}^p - SoH_k^p} = \\ &= \frac{V_1 e^{\frac{-T}{\tau_{RC}(T_k, SoH_k^p, SoH_k^p)}} + R_1(T_k, SoH_k^p, SoH_k^p) (1 - e^{\frac{-T}{\tau_{RC}(T_k, SoH_k^p, SoH_k^p)}}) I_k - V_{1_k}}{\left(-\frac{20\Delta T}{78840000} + SoH_k^p\right) - SoH_k^p} \\ &= -\frac{78840000}{20\Delta T} \left(1 - e^{\frac{-T}{\tau_{RC}(T_k, SoH_k^p, SoH_k^p)}}\right) (R_1(T_k, SoH_k^p, SoH_k^p) I_k - V_{1_k}) \end{aligned} \quad (5.31)$$

$$h_2 = \frac{\partial OCV(T_k, SoH_k^p, SoC_k^p)}{\partial SoC} - \frac{\partial V_1}{\partial SoC} - \frac{\partial R_{int}(T_k, SoH_k^p, SoC_k^p)}{\partial SoC} I_k \quad (5.32)$$

and

$$\begin{aligned} \frac{\partial V_1}{\partial SoC} &= \frac{V_{1_{k+1}} - V_{1_k}}{SoC_{k+1}^p - SoC_k^p} = \\ &= \frac{V_1 e^{\frac{-T}{\tau_{RC}(T_k, SoH_k^p, SoH_k^p)}} + R_1 \left(1 - e^{\frac{-T}{\tau_{RC}(T_k, SoH_k^p, SoH_k^p)}}\right) I_k - V_{1_k}}{\left(SoC_k^p - \frac{\eta(T_k, SoH_k^p) \Delta T}{Q_n} I_k\right) - SoC_k^p} \\ &= -\frac{Q_n}{\eta(T_k, SoH_k^p) \Delta T I_k} \left(1 - e^{\frac{-T}{\tau_{RC}(T_k, SoH_k^p, SoH_k^p)}}\right) (R_1(T_k, SoH_k^p, SoH_k^p) I_k - V_{1_k}) \end{aligned} \quad (5.33)$$

In order to make more robust the estimation, it has been chosen to make the EKF adaptive by updating iteratively the process noise covariance matrix ( $Q$ ) on the basis of the innovation terms [43]. Moreover, in order to enhance both SoH and SoC estimation the OCVmap technique is applied along with EKF on the basis of the filtered estimation of the state. In the following figure the overall algorithm is presented:

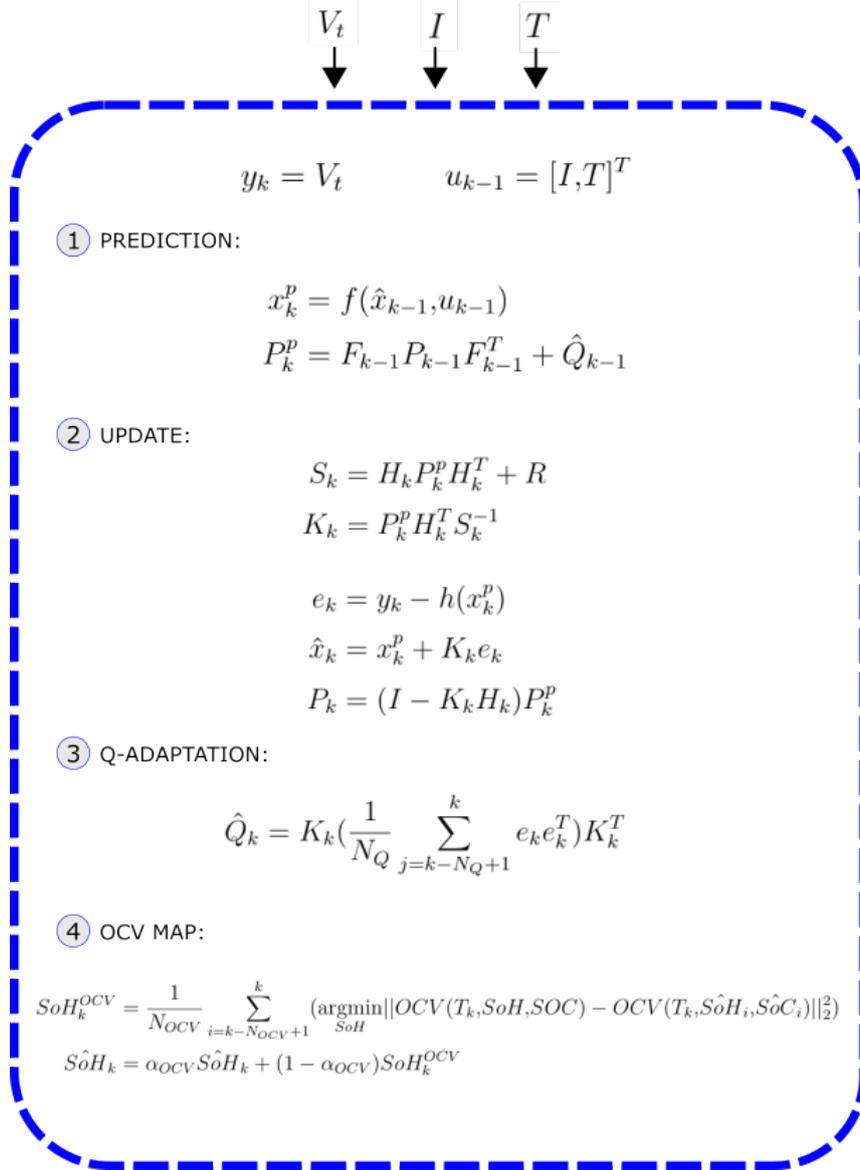


Figure 5.1: Overview of the EKF estimation algorithm for a  $k$ -esime iteration.

The first two set of equations illustrated in figure 5.3 represent the usual prediction and update step of the EKF, while the equation in step 3 implements a Q-adaptation based on a sliding window of size  $N_Q$ . Step 4 implements the OCV map strategy to update the filtered version of the SoH: using a sliding window of size  $N_{OCV}$ , the SoH obtained in corrispondence of the minimum OCV error (computed by comparing the a-priori OCV data points and the computed OCV using filtered state) are collected; so at a given time the filtered value of the SoH is substituted with the exponential mean of the current filtered SoH and the mean value obtained through OCV map.

Intial values of the windows for both step 3 and 4 are the zero vector and their sizes ( $N_Q, N_{OCV}$ ) along with  $\alpha_{OCV}$  become parameters to be tuned. Step 3 and step 4 are introduced to make more robust the estimation for both SoH and SoC.

The implementation of the EKF is presented in figures A.8 and A.9 in appendix A.

## 5.1 Initial conditions and EKF tuning

The system is supposed to start from a full charged state and after a long time is passed from its last usage, while its SoH is supposed to be unknown. The estimate of the initial condition is chosen randomly:

$$\hat{x}_0 = \hat{x}(k=0) = \begin{bmatrix} \hat{S}o\hat{H}_0 \\ \hat{S}o\hat{C}_0 \\ \hat{V}_{10} \end{bmatrix} = \begin{bmatrix} 91.9459 \\ 95.6547 \\ 0 \end{bmatrix} \begin{matrix} \% \\ \% \\ V \end{matrix} \quad (5.34)$$

The true system initial condition is unknown and is modelled as a multivariate gaussian distributed random vector:

$$x_0 = x(k=0) \sim \mathcal{N}(\mu_{x_0}, \Sigma_{x_0}) \quad (5.35)$$

Although the initial condition is unknown, some assumption on its statistical parameters can be done. In particular the SoH is supposed to belong to the interval  $[80, 100]\%$  with 99.73% of probability. This can be traduced in the following interval  $[\mu_{SoH_0} - 3\sigma_{SoH_0}, \mu_{SoH_0} + 3\sigma_{SoH_0}]$ , where  $\mu_{SoH_0}$  is 90 % and  $\sigma_{SoH_0} = 10/3\%$ . SoC is supposed to belong to an interval equal to  $[\mu_{SoC_0} - 3\sigma_{SoC_0}, \mu_{SoC_0} + 3\sigma_{SoC_0}]$ , where  $\mu_{SoC_0}$  is 95 % and  $\sigma_{SoC_0} = 5/3\%$ . In this way, with 99.73% of probability, the SoC initial condition of the system belong to the interval  $[90, 100] \%$ . Similarly to the previous cases,  $V_{10}$  is supposed to be near to 0. Its variation belong to the interval  $[\mu_{V_{10}} - 3\sigma_{V_{10}}, \mu_{V_{10}} + 3\sigma_{V_{10}}]$ , where  $\mu_{V_{10}}$  is 0.1 V and  $\sigma_{V_{10}} = 0.1/3$  V.

So the statistical parameters attributed to the state initial condition are known by assumption:

$$\mu_0 = \begin{bmatrix} 90 \\ 95 \\ 0.1 \end{bmatrix} \begin{matrix} \% \\ \% \\ V \end{matrix} \quad (5.36)$$

$$\begin{aligned}
 \Sigma_{x_0} &= E \left[ (x_0 - E[x_0])(x_0 - E[x_0])^T \right] = \\
 &= E \left[ (x_0 - \mu_{x_0})(x_0 - \mu_{x_0})^T \right] = \\
 &= \begin{bmatrix} \sigma_{SoH_0}^2 & \sigma_{SoH_0SoC_0} & \sigma_{SoH_0V_{1_0}} \\ \sigma_{SoC_0SoH_0} & \sigma_{SoC_0}^2 & \sigma_{SoC_0V_{1_0}} \\ \sigma_{V_{1_0}SoH_0} & \sigma_{V_{1_0}SoC_0} & \sigma_{V_{1_0}}^2 \end{bmatrix} = \\
 &= \begin{bmatrix} \sigma_{SoC_0}^2 & \rho_{HC}\sigma_{SoH_0}\sigma_{SoC_0} & \rho_{HV}\sigma_{SoH_0}\sigma_{V_{1_0}} \\ \rho_{HC}\sigma_{SoC_0}\sigma_{SoH_0} & \sigma_{SoC_0}^2 & \rho_{CV}\sigma_{SoC_0}\sigma_{V_{1_0}} \\ \rho_{HV}\sigma_{V_{1_0}}\sigma_{SoH_0} & \rho_{CV}\sigma_{V_{1_0}}\sigma_{SoC_0} & \sigma_{V_{1_0}}^2 \end{bmatrix}
 \end{aligned} \tag{5.37}$$

Where the parameters  $\rho_{HC}, \rho_{HV}$  and  $\rho_{CV}$  are the correlation coefficients and are found by trial and error approach. The EKF matrices  $R$ ,  $Q_0$  and  $P_0$  are chosen in the following way:

- $R = E[(\tilde{y}_k - E[\tilde{y}_k])(\tilde{y}_k - E[\tilde{y}_k])^T]$ ,  $\tilde{y} = y_k - \hat{y}_k$  is chosen according to the statistic properties of the measurement error which is described next.
- $P_0 = E[(\tilde{x}_0 - E[\tilde{x}_0])(x_0 - E[x_0])^T]$ ,  $\tilde{x}_0 = x_0 - \hat{x}_0$  is equal to  $\Sigma_{x_0}$ .

According to the simulink schema in figure A.1, the WLTP cycle test is performed at 25 °C and for all the SoH reference conditions {100, 97.92, 95.84, 93.76, 91.68, 89.6, 87.52, 85.44, 83.36, 81.28, 79.20} % on the dynamic simulator. The sampled (1Hz) acquired data in terms of battery voltage, current, and temperature are then corrupted by measurement random noise uniformly distributed, in order to introduce disturbances by emulating measurement sensors. The characteristics of the noises for each measurement are the following:

- $\max(w_V) = 0.3$  V,  $\min(w_V) = -0.3$  V,  $\mu_V = 0$
- $\max(w_I) = 0.05$  A,  $\min(w_I) = -0.05$  A,  $\mu_I = 0$
- $\max(w_T) = 1$  °C,  $\min(w_T) = 0$  °C,  $\mu_T = 0$

The following figure depicts the signals for one sample WLTP test performed at 100 % SoH:

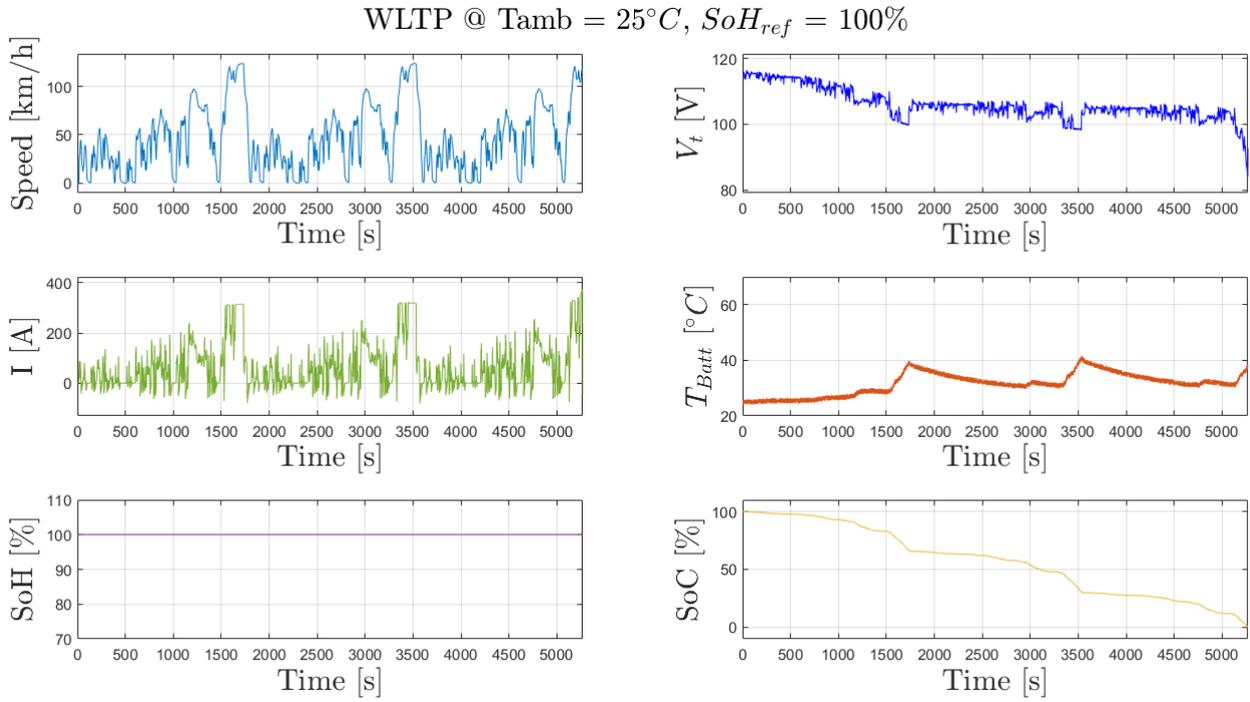


Figure 5.2: Signals contextualized at one WLTP test performed wltpt at  $25^{\circ}C$  ambient temperature and 100% SoH.

Referring to the scenario with 100% SoH, the simulink schema A.6 is used to find the measurement error of the FOTM model and then a suitable value for the R matrix of the EKF.

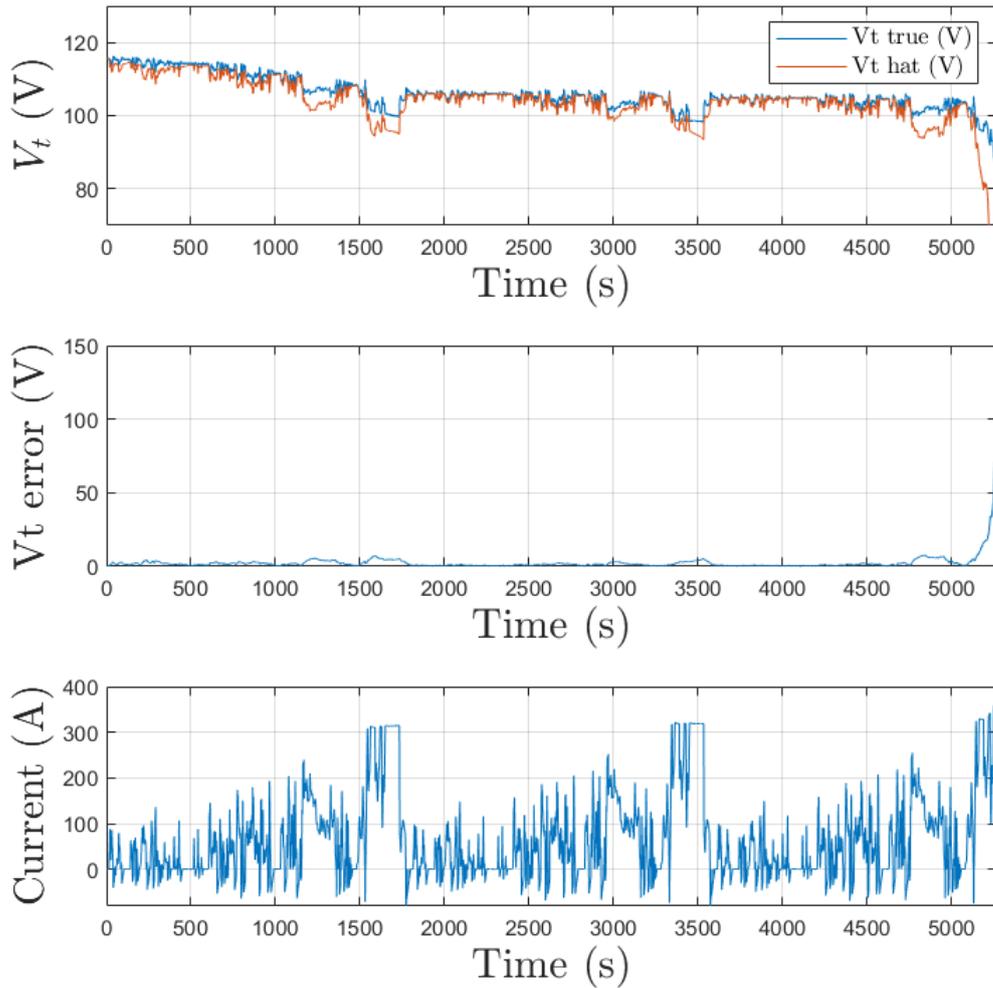


Figure 5.3: Signals obtained by comparing data produced by the dynamic simulator and the FOTM model by performing wltip at 25 °C ambient temperature and 100% SoH.

Referring to the previous figure, the standard deviation of the measurement error

is  $\sigma_{\bar{y}} = 6.978(V)$ , then the matrix  $R$  is equal to  $\sigma_{\bar{y}}^2$ . The chosen tuning scenario for the EKF parameters refers to the performed test at SoH references  $\{100, 97.92, 93.76, 89.6, 87.52, 83.36, 79.20\}$  %. According to the simulink schema A.8 and to the tuning scenario data, the EKF parameters are found by trial and error approach.

In the following picture a sample of the typical signals used in the tuning process are presented:

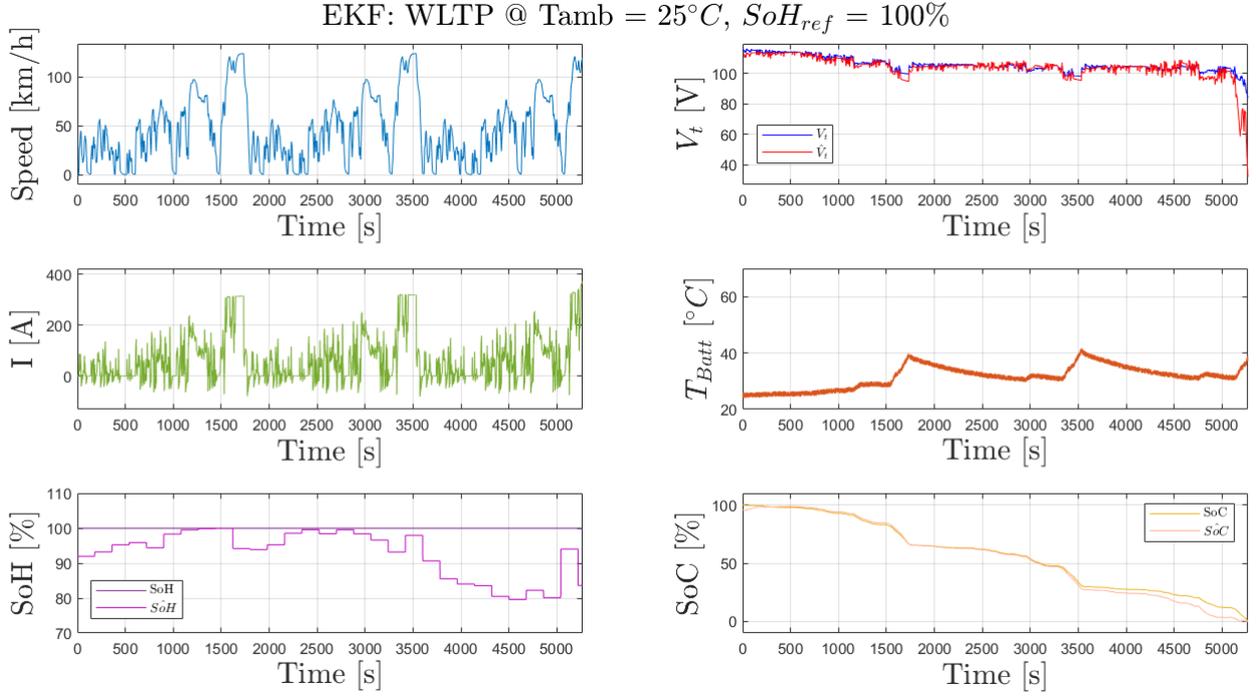


Figure 5.4: Tuning scenario signals for 100% SoH.

The final parameters found for the proposed EKF algorithm are:

Parameter	value
$\rho_{HC}$	0.3
$\rho_{HV}$	0.001
$\rho_{CV}$	0.831
$Q_0$	$0.1 I_3$
$N_Q$	60 s
$N_{OCV}$	180 s
$\alpha_{OCV}$	0.3

## 5.2 Test results

The EKF is initialized according to the parameters found in the previous paragraph and, by considering the Simulink schema A.8, the algorithm is validated on the WLTP tests conducted on the dynamic simulator A.1 at  $25\text{ }^{\circ}\text{C}$  ambient temperature and SoH references that are not used in the tuning process  $\{95.84, 91.68, 85.44, 81.28\}$  %.

The following figures summarize the results obtained at each test.

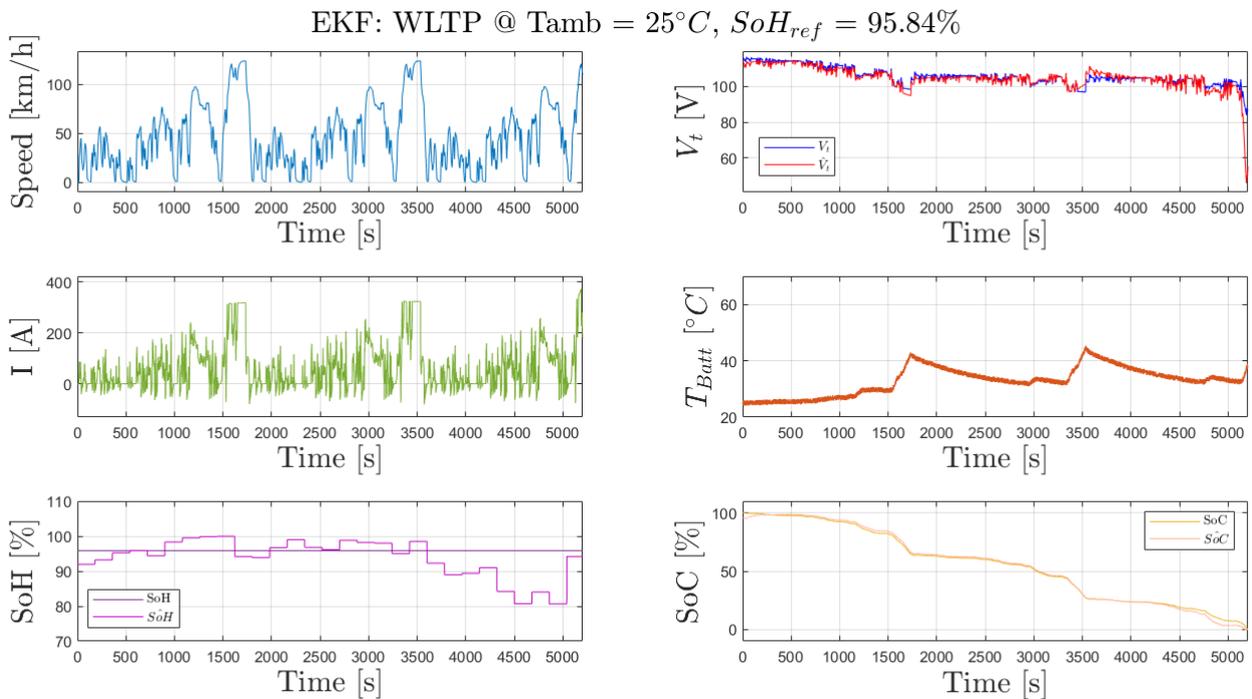


Figure 5.5: Test 1 - signals

EKF: WLTP @  $T_{amb} = 25^{\circ}C$ ,  $SoH_{ref} = 95.84\%$

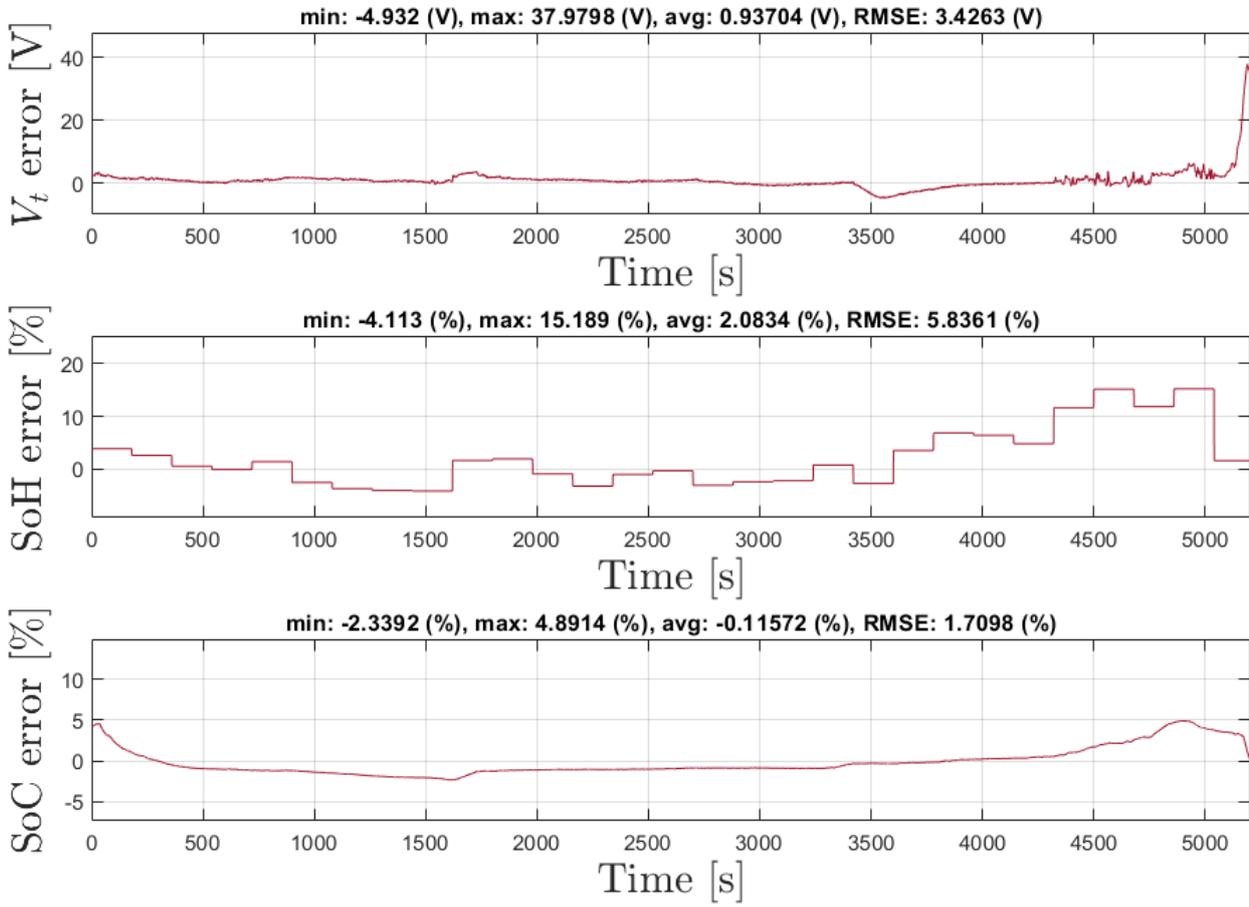


Figure 5.6: Test 1 - errors

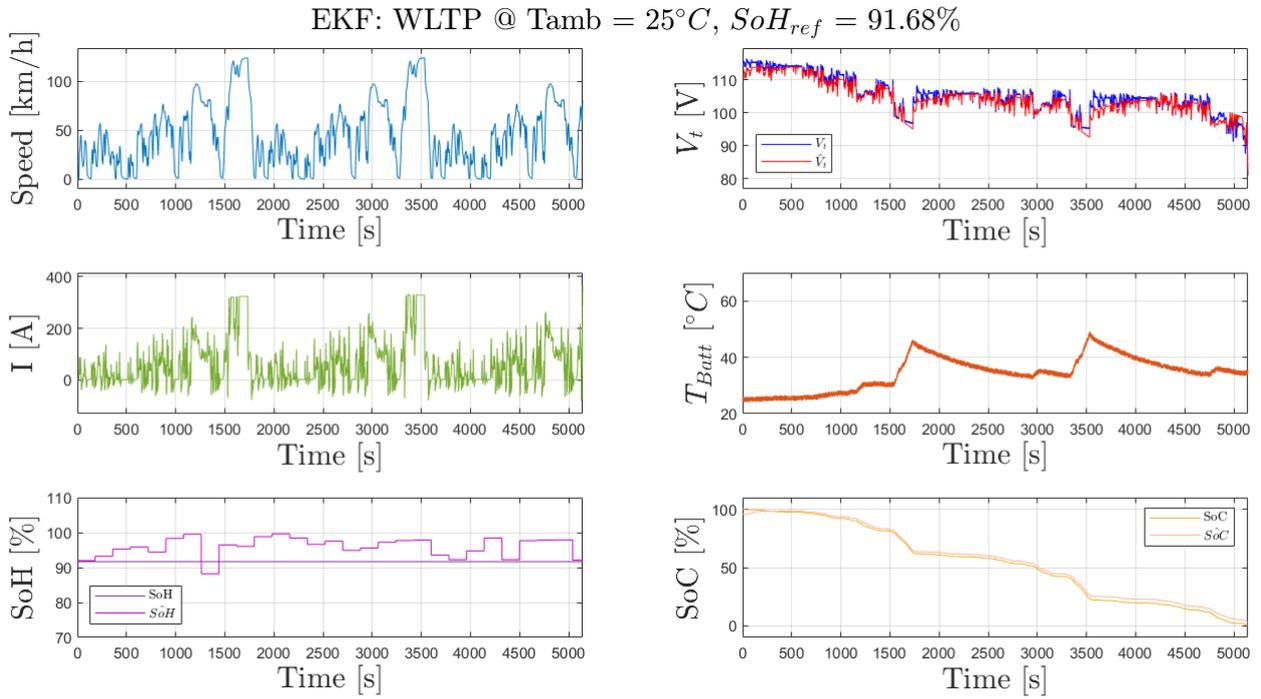


Figure 5.7: Test 2 - signals

EKF: WLTP @  $T_{amb} = 25^{\circ}C$ ,  $SoH_{ref} = 91.68\%$

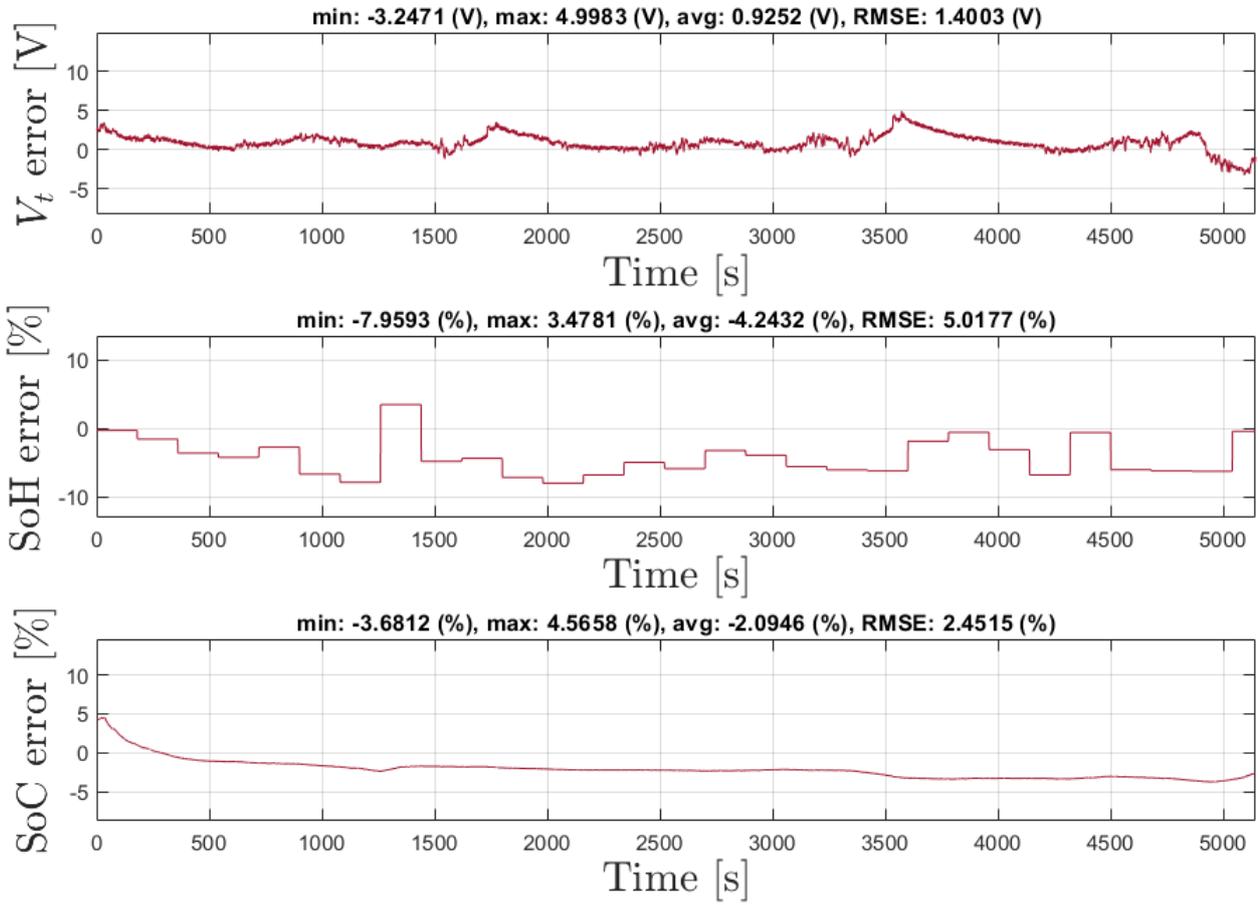


Figure 5.8: Test 2 - errors

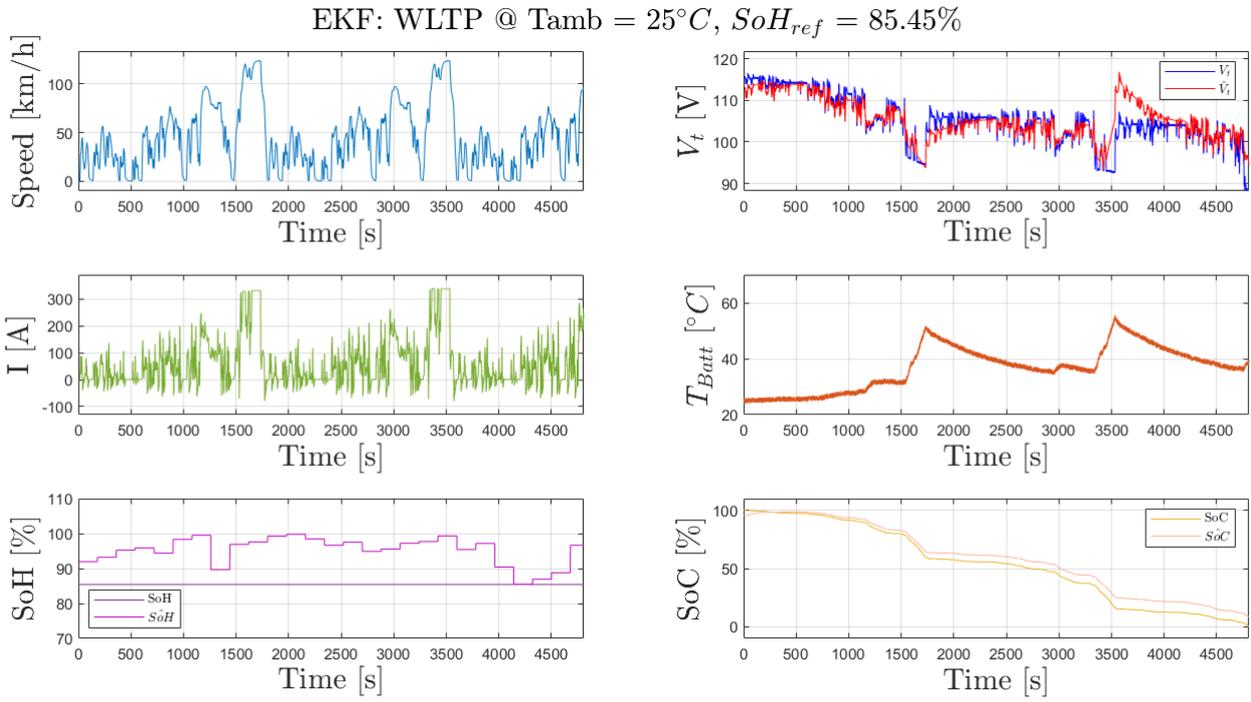


Figure 5.9: Test 3 - signals

EKF: WLTP @  $T_{amb} = 25^{\circ}C$ ,  $SoH_{ref} = 85.45\%$

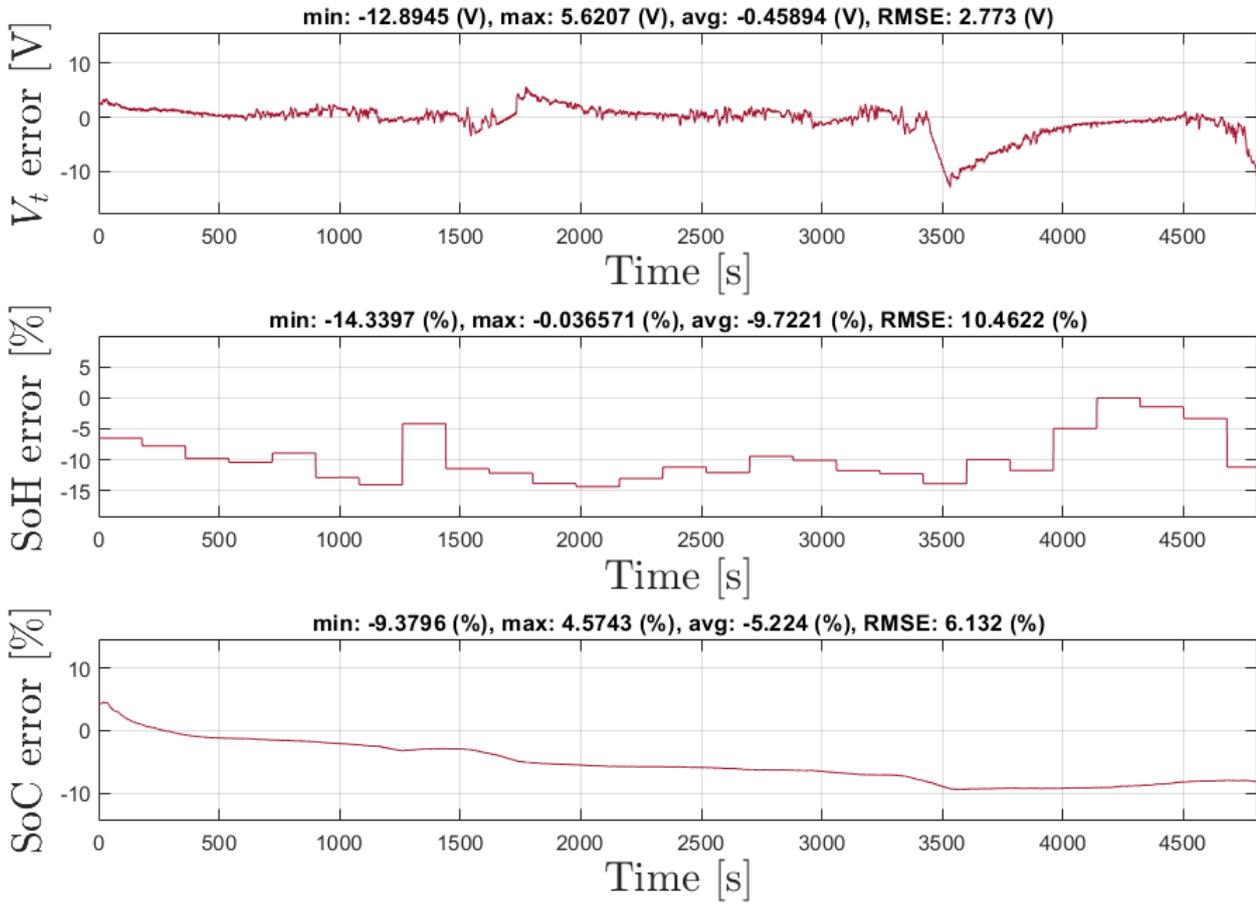


Figure 5.10: Test 3 - errors

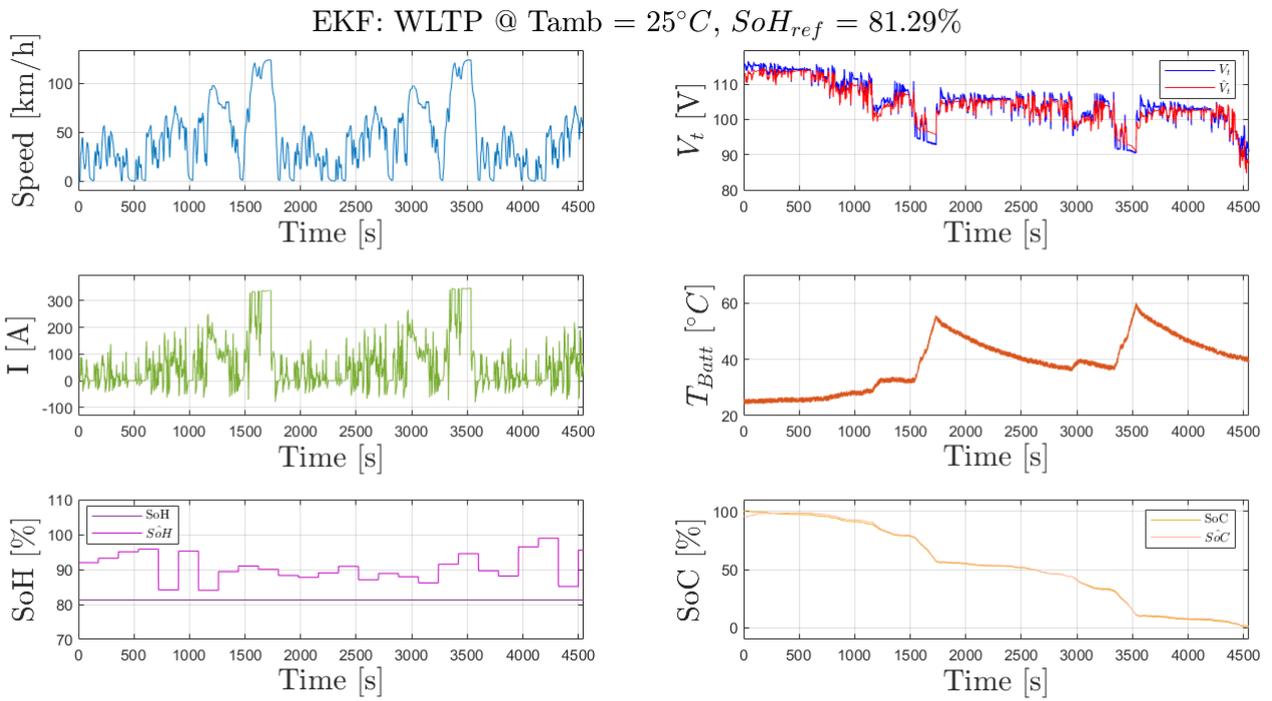


Figure 5.11: Test 4 - signals

EKF: WLTP @  $T_{amb} = 25^{\circ}C$ ,  $SoH_{ref} = 81.29\%$

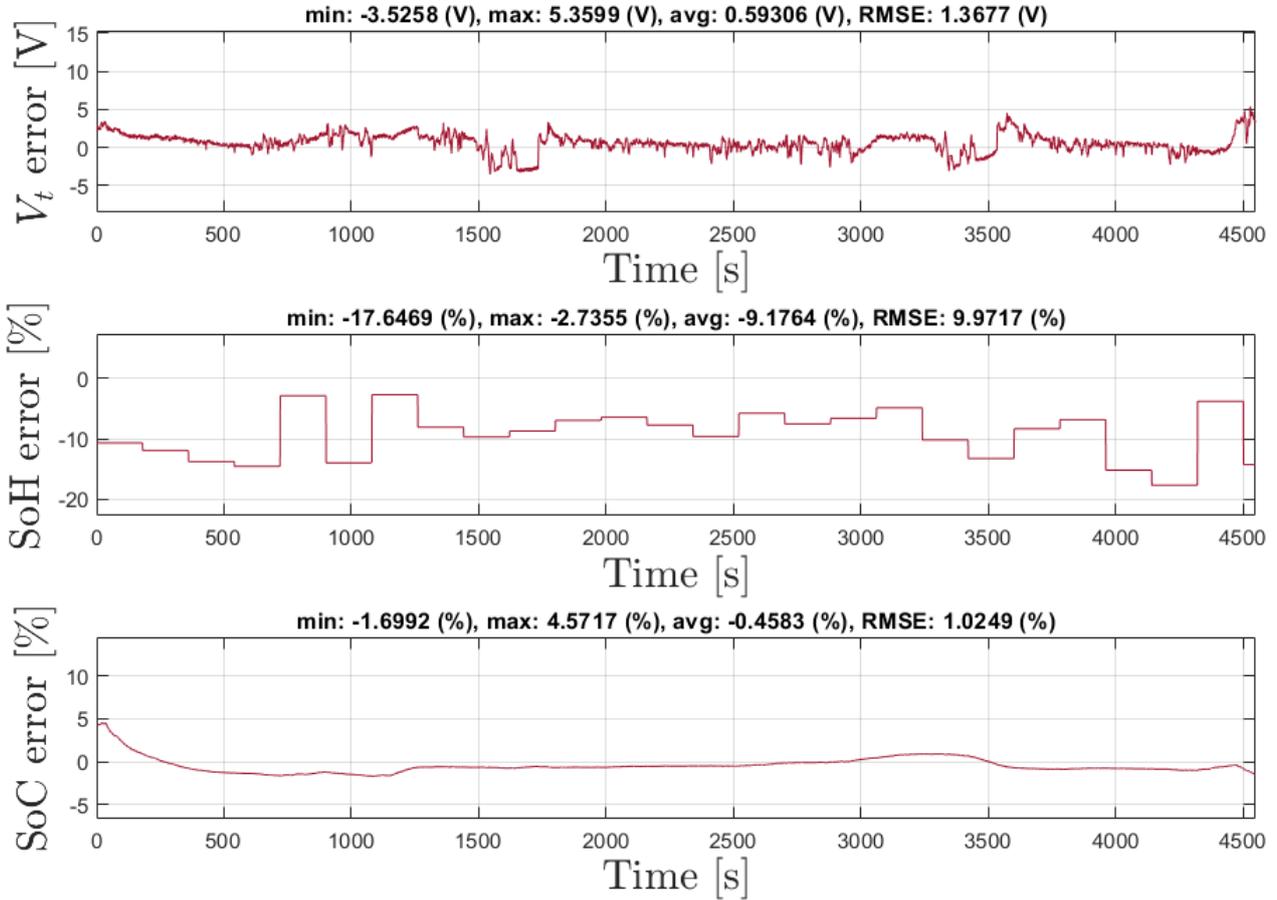


Figure 5.12: Test 4 - errors

As can be observed by looking at the results, when the estimate of the SoH approaches the truth one the error in the SoC estimation reduces thanks to the adaptivity of the algorithm. Even if the starting initial condition of the state estimate is different from the truth one, the SoC estimate tends to the reference in a finite time. The average maximum absolute error for SoC estimate over all the tests is under 5 %, while the worse is about 10% which is achieved in the Test 3. The best average error is -0.12% (Test 1) while the worst is -5.22 % (Test 3). It is observed that the SoH estimate fluctuates near the reference but is not always accurate. This can be due to the presence of the noise in the measurements and the precision of

the OCV curve that can penalize the SoH update in time. The best average error in SoH estimation is about 2 % (Test 1) while the worse is -9.7 % (Test 3).

# Chapter 6

## Neural Network

In chapter 3.3 the most relevant data-driven estimation approaches that exist in the literature are addressed. In this chapter, a machine learning (Black-box) approach based on Non-linear AutoRegressive with eXogenous input Neural Network (NARX-NN) is adopted to estimate SoC and SoH in the context of WLTP dynamometric tests data acquired by using the provided dynamic simulator (paragraph 4.1).

The principle on which the network is based relies on the assumption that there exist an unknown non-linear relationship between the time delayed inputs of the system, the past history of the output and the output parameters to be predicted. In particular the network model this relationship according to the following equation:

$$y(t) = f(y(t-1), y(t-2), \dots, y(t-d_y), u(t-1), \dots, u(t-d_u)) \quad (6.1)$$

Where:

- $u$  is the input vector containing the measures of voltage, current and battery temperature
- $y$  is the output vector containing SoH and SoC to be estimated
- $d_u$  is the maximum time delay for the input
- $d_y$  is the maximum time delay for the output

The following figure represent the architecture of the proposed NN:

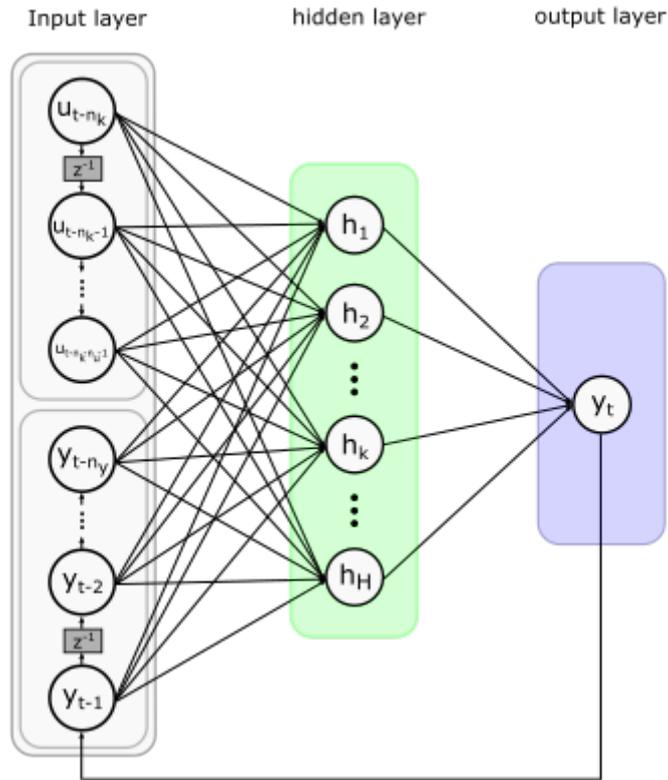


Figure 6.1: Architecture of the proposed NARX-NN.

The overall network is based on closed-loop structure which reflects the relationship 6.1 and contains three layers called respectively input, hidden and output layer. Except for the input layer all the remaining ones contains a given number of neurons. According to the figure, the hidden layer contains  $H$  neurons while the output layer contains only one neuron.

Each neuron elaborates its input data and gives a certain output according to the following figure:

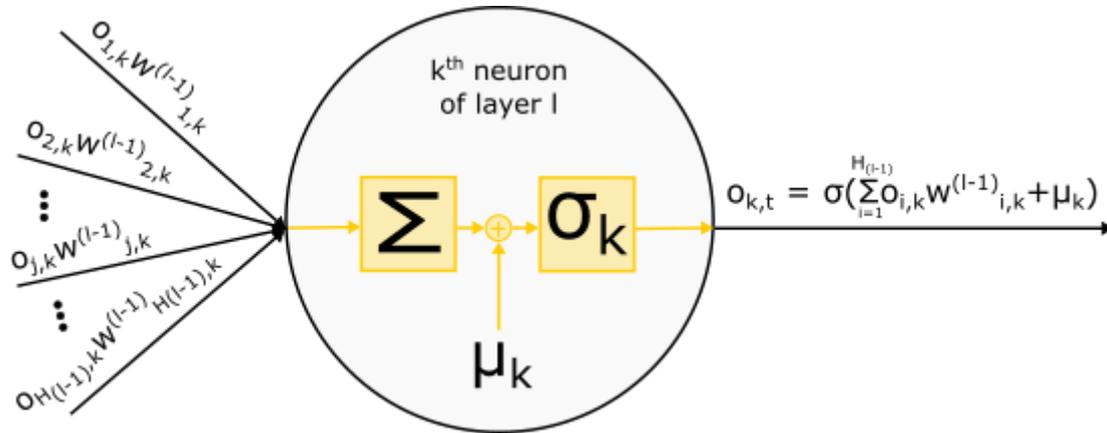


Figure 6.2: Neuron inner structure.

For more details about the meaning and the behaviour of the neuron, the reader is suggested to read paragraph 3.3.2 where different types of neural networks are discussed.

## 6.1 Dataset

In order to make the network able to learn the unknown function 6.1, a dataset containing time-series data for both inputs and output is needed.

By considering the Simulink schema A.1, a WLTP test procedure is performed at different aged status of the battery pack and ambient temperature of 25 °C. The SoH references are taken from the set {100, 97.92, 95.84, 93.76, 91.68, 89.6, 87.52, 85.44, 83.36, 81.28, 79.20 }. The acquired data in terms of battery voltage, current, and temperature are sampled at 1Hz and then corrupted by measurement random noise uniformly distributed, in order to introduce disturbances by emulating measurement sensors. The characteristics of the noises for each measurement are the following:

- $\max(w_V) = 0.3 \text{ V}$ ,  $\min(w_V) = -0.3 \text{ V}$ ,  $\mu_V = 0$
- $\max(w_I) = 0.05 \text{ A}$ ,  $\min(w_I) = -0.05 \text{ A}$ ,  $\mu_I = 0$
- $\max(w_T) = 1 \text{ }^\circ\text{C}$ ,  $\min(w_T) = 0 \text{ }^\circ\text{C}$ ,  $\mu_T = 0$

The following figure depicts the signals for one sample WLTP test performed at 93.76 % SoH:

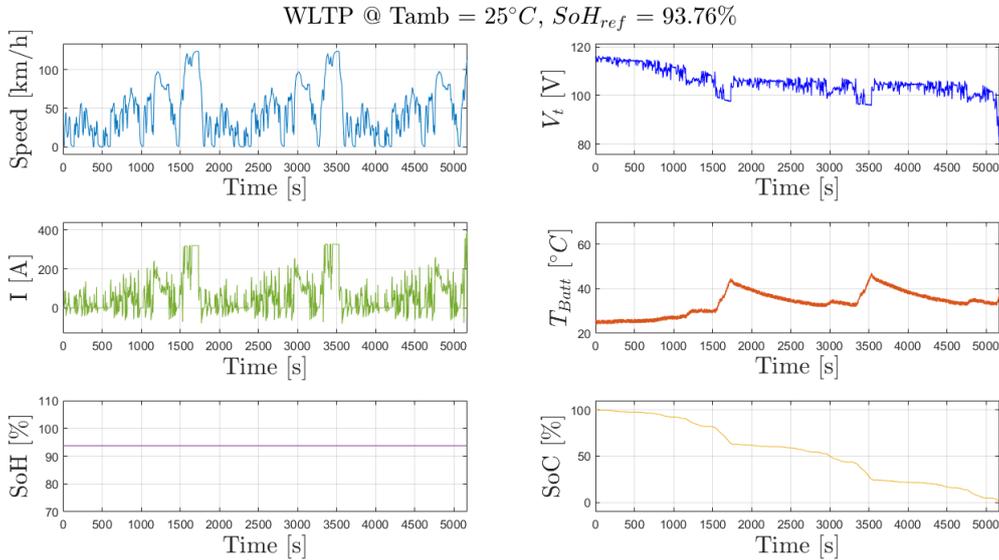


Figure 6.3: Signals contextualized at one WLTP test performed wltpl at 25 °C ambient temperature and 93.76% SoH.

The ground truth output is based on the ground truth references of SoH and ground truth SoC provided by the dynamic simulator. The dataset is build by combining all the acquired data for each SoH reference in such a way to depict an aging behaviour of the battery pack in time. For visualization simplicity only the targets output are reported in the next figure 6.4. The total number of available samples in the dataset is 54654 and they are splitted randomly (preserving causality) into training, validation and test sets with respectively proportions 70%, 15% and 15%.

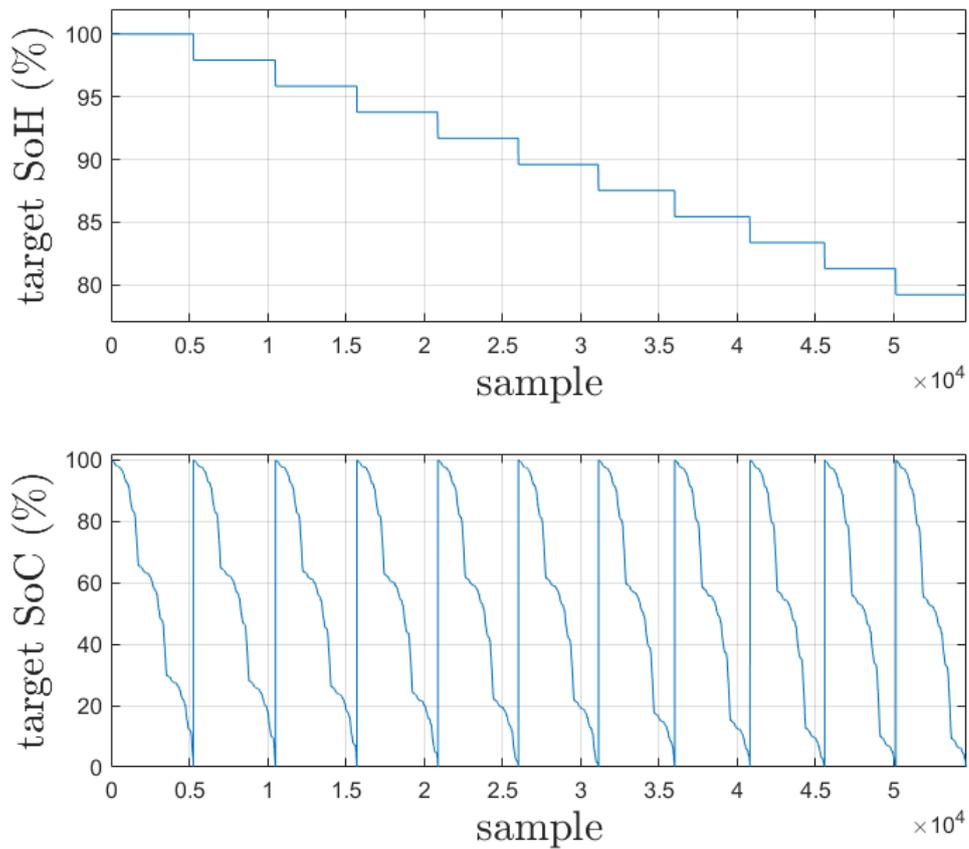


Figure 6.4: Targets samples obtained by combining all the performed WLTP tests.

## 6.2 Training

The training process is performed on the dataset described in the previous paragraph in a supervised way and by adopting an open-loop strategy: since the inputs and output targets are all available in the dataset, they are used to fill the delayed relative quantities during the training process. By opening the loop, the network become a purely feed forward NN and an efficient training algorithm can be used. This strategy allow to get better results with respect the closed-loop one which is based on the time delayed estimations of the output during the training.

The whole process is performed by using MATLAB Deep Learning tool box that provides usefull commands to prepare data according to the chosen training policy, to perform training and visualize results.

The default training algorithm which is suggested by the toolbox for this kind of problems, is called Levenberg-Marquardt and is based on gradient-descent algorithm. It resolves at each training iteration an optimization problem by minimizing the MSE between estimated target and the ground truth one. The maximum number of training epochs is 1000.

The activation functions for each neuron of the hidden layer is the sigmoid function, while for the output neuron is the identity function. The number of hidden neurons along with the maximum time delay step for input and output are tuning parameters.

The training procedure is performed automatically and in an optimal way by the toolbox which stops the process when there is a maximum number of 6 failures in the improvement of the validation performances, or when the maximum number of epochs is reached.

Several configurations of the tuning parameters are tested and is observed that the training process time grows drammatically as the the delays and number of neurons increase.

In order to make the training process feasible and to avoid to increase the complexity of the network, the number of neurons and delay time steps are chosen to be adequately small. Moreover input and output delayed time step are chosen to be equal in value. It is performed a grid search by considering the combination of

different delays and number of neurons and the best set of parameters found is:

- $d = d_y = d_u = 20$
- $H = 10$

The results are presented in the next paragraph.

### 6.3 Results

By considering a maximum delay time step of 20 s for both input and output, and a number neurons of the hidden layer equal to 10, the following results are obtained after the end of the training of the proposed NARX-NN:

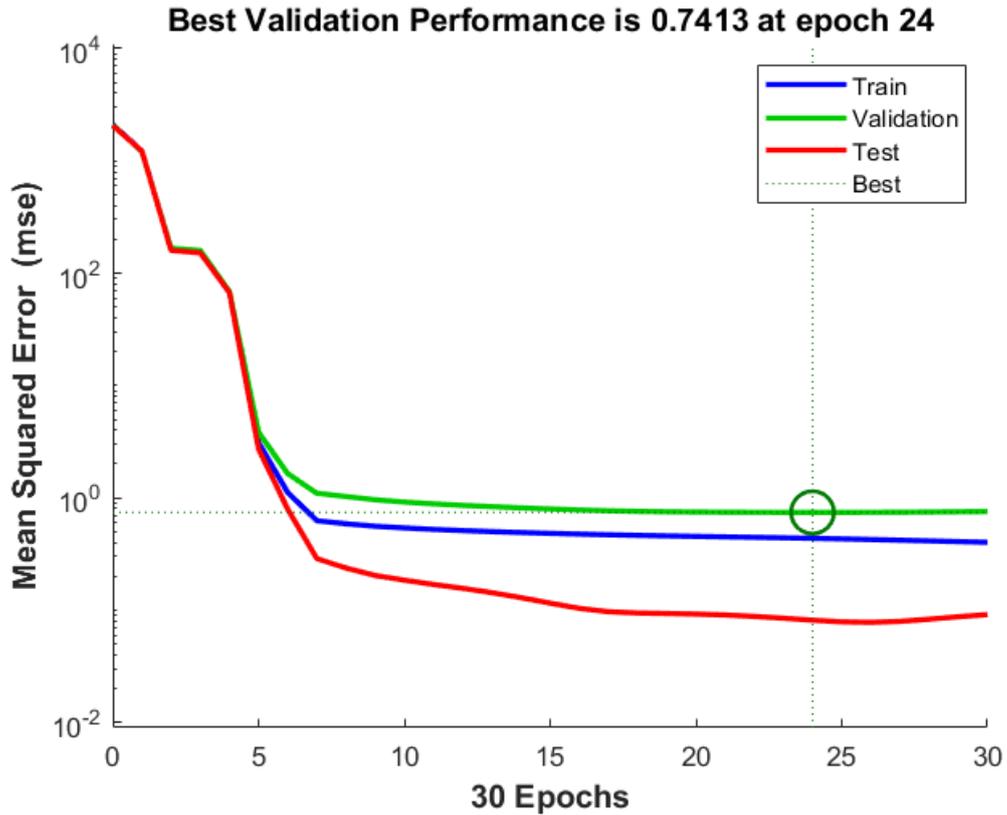


Figure 6.5: Performances obtained during training process respectively on the train, validation and test sets.

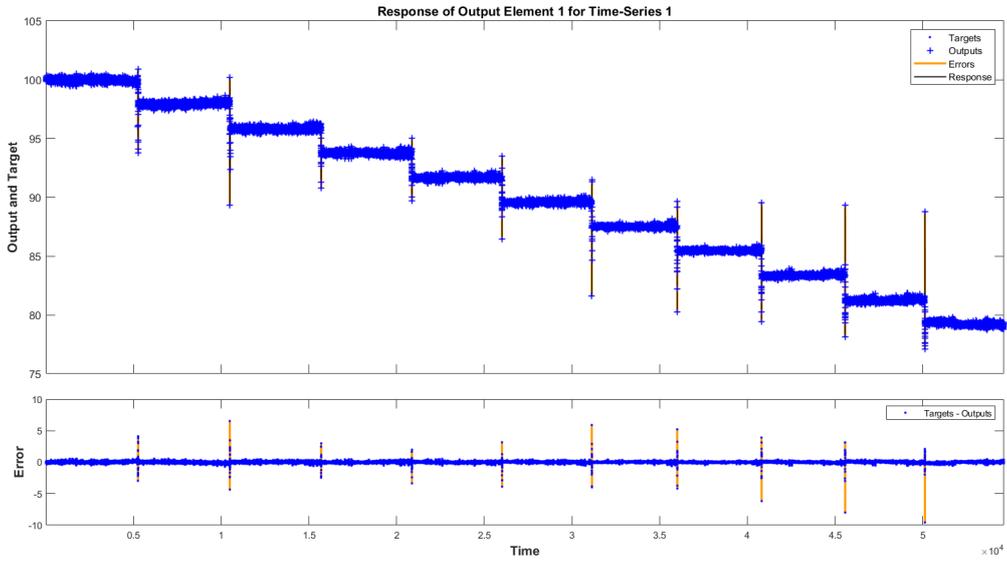


Figure 6.6: Comparison between target test set and network predicted output with respect target SoH.

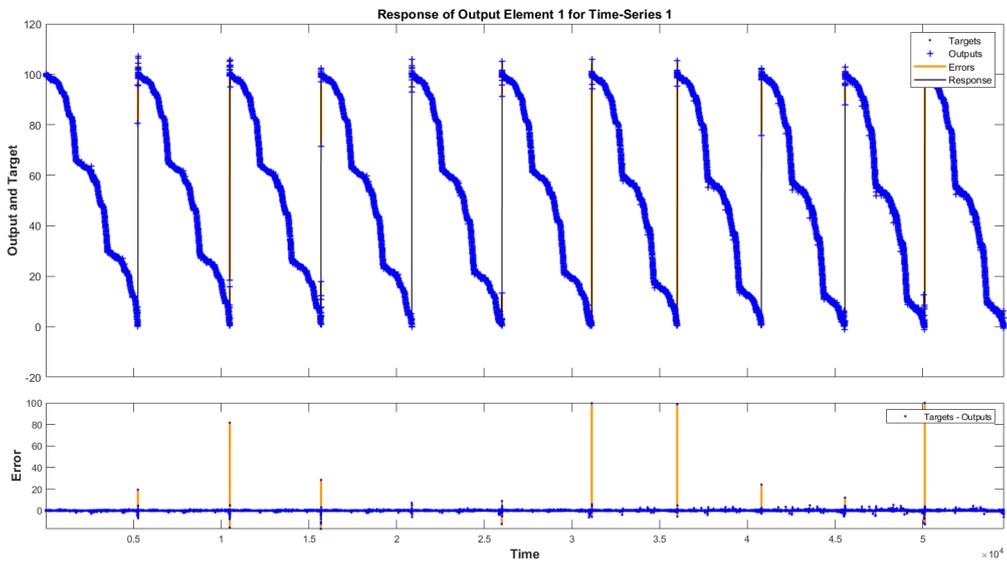


Figure 6.7: Comparison between target test set and network predicted output with respect target SoC.

As can be observed by looking at figure 6.5, the network achieves the best results on the validation set at epoch 24 by obtaining the smallest MSE value that is equal to 0.7413. The process is stopped at that epoch in order to avoid to overfit training data, allowing to maintain generalization flexibility to unseen data. In the diagram is reported also the performance obtained on the test set which is not taken into consideration in the training process but is purely introduced as proof of the generalization ability of the network. The MSE obtained on the test set at epoch 24 is approximately equal to 0.07. Finally, from figures 6.6 and 6.7 it can be seen that the network is able to adequately predict the target parameters.

# Chapter 7

## Conclusion

In this thesis work two proposed estimation schemes are considered to estimate two important battery state parameters: the State of Charge (SoC) and the State of Health (SoH). Both the estimators are developed and validated under simulation, in the context of a MATLAB environment, according to WLTP dynamometric test performed through a dynamic simulator of a real target system. The simulator is composed by different models and the principle ones are used to emulate the road profile, the vehicle dynamic of a Fiat Panda first series, an electric motor, a 28S1P Samsung SDI 94Ah battery pack and a thermal model of the battery. The dynamic simulator has been given and its validity was proved outside of this thesis work. The synthetic time series data obtained by the tests are contextualized to different aged states of the battery and consist in noise corrupted measurements of battery voltage, current and battery temperature sampled at 1 Hz.

The first estimator is taken from the family of the first principles estimation approaches and consists in an online classical technique called Extended Kalman Filter. It has been proposed in its Q-adaptive version and has been enhanced by using a data-driven estimation approach called Open Circuit Voltage map. This approach mainly relies on the usage of an Electric Circuit Model of the battery that has been identified according to some preliminary tests. The validation tests, which have been performed at different aged states of the battery, prove that in

many circumstances the estimator is able to follow the ground truth target parameter references. In particular it has been observed that the estimation of the SoC is strictly related to the SoH estimation one, in fact the goodness of the SoC estimation increases as the estimation of the SoH approaches the ground truth one. When the estimation of the SoH becomes worse, also the SoC seems to be affected. The maximum absolute average error obtained through the simulated tests is about 5 % for the SoC, and 9.72 % for the SoH, but, some times the error can reach 10 % for SoC and even 17 % for the SoH.

The second estimation technique is based on a machine learning approach. In particular it relies on the usage of an Artificial Neural Network called Nonlinear AutoRegressive with eXogenous input Neural Network. In contrast with the first approach it needs only time series data in order to be developed. In fact, by performing many tests on the dynamic simulator, at different aged states of the battery, the acquired data have been rearranged as to build a dataset in order to make possible to learn a model that is used to establish a direct mapping between inputs and the target parameters. The training process has been performed splitting the dataset into training, validation and test sets. In correspondence of the best time delay step and number of hidden neurons found, (respectively 20 s and 10), the final performance results achieved on the test set in terms of MSE is about 0.07.

By comparing the two approaches it is possible to conclude that the machine learning technique is a better approach to address the problem since can achieve better estimation results and require a relative minor development effort.

A possible proposal for a future work can be to test the proposed techniques on experimental data or to prove other estimation strategies.

# Appendix A

## Simulink schemes

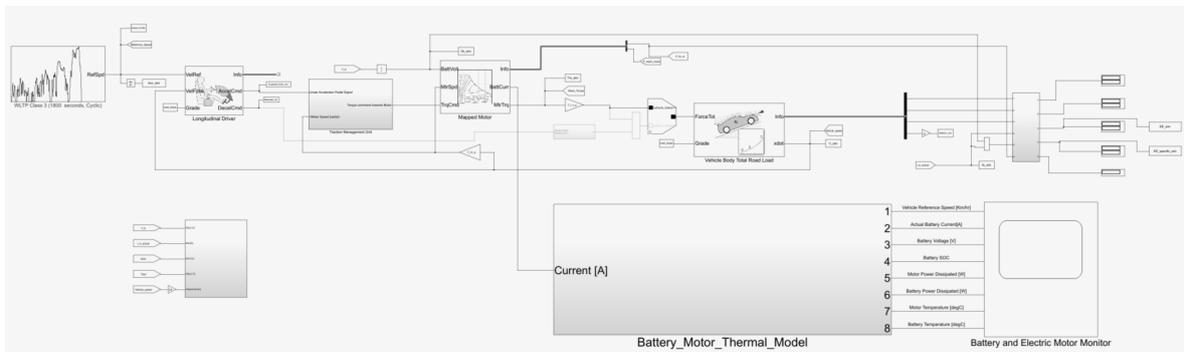


Figure A.1: Simulink schema adopted with the aim of performing tests on the simulated system.

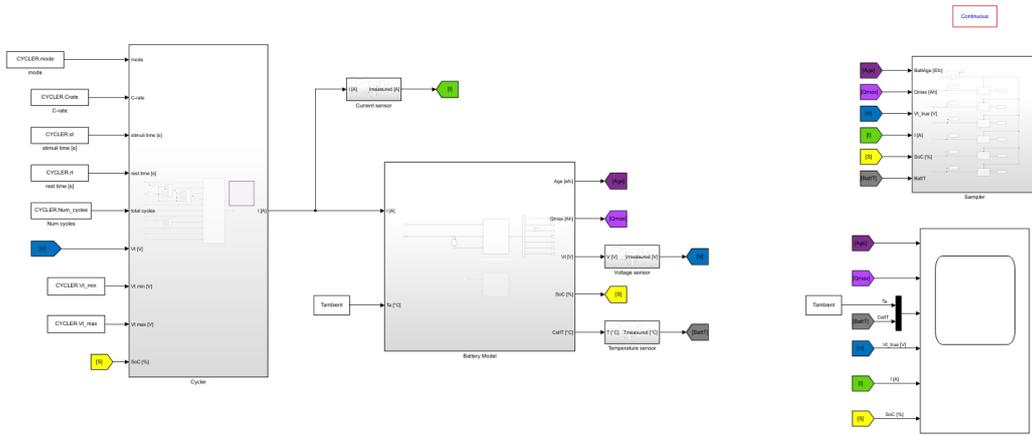


Figure A.2: Simulink schema adopted with the aim of performing tests on the battery model.

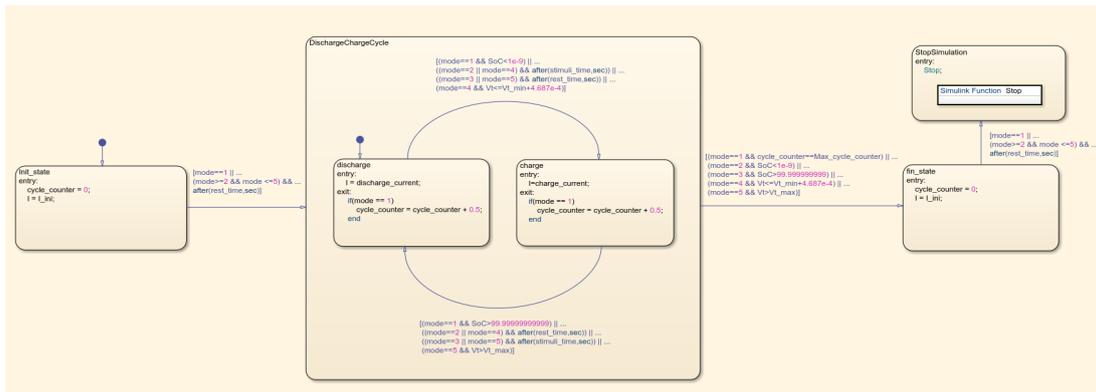


Figure A.3: Cycler inner finite state machine.

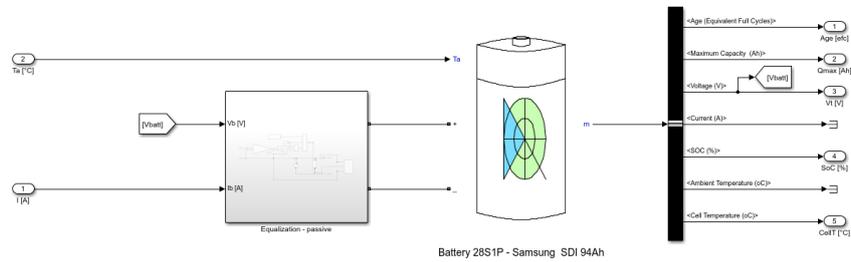


Figure A.4: Battery model inner structure.

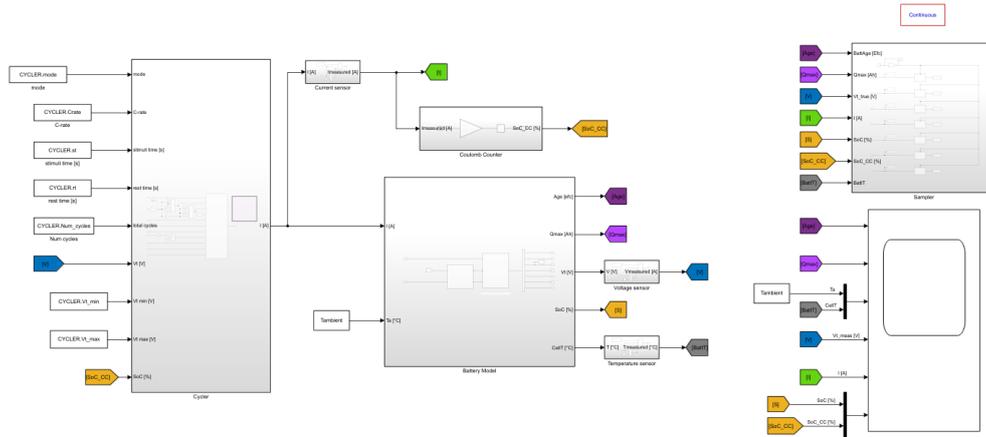


Figure A.5: Simulink schema adopted with the aim of performing tests on the battery model using Coulomb Counter estimation technique.

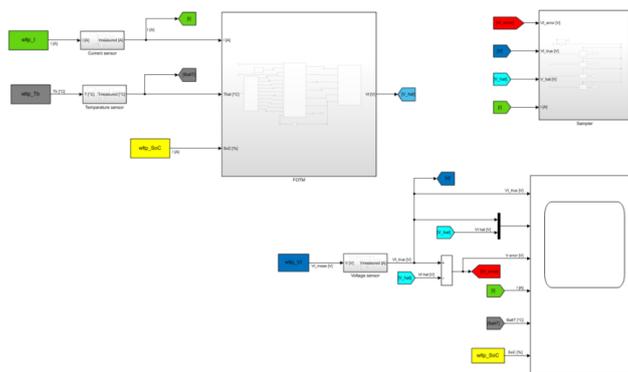


Figure A.6: Simulink schema of the FOTM ECM.

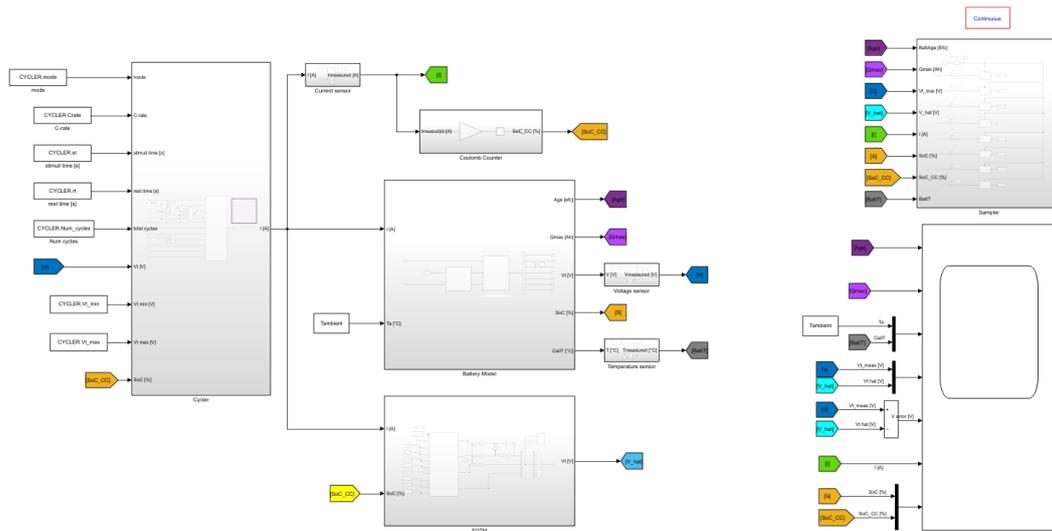


Figure A.7: Simulink schema adopted with the aim of comparing battery model and FOTM ECM.

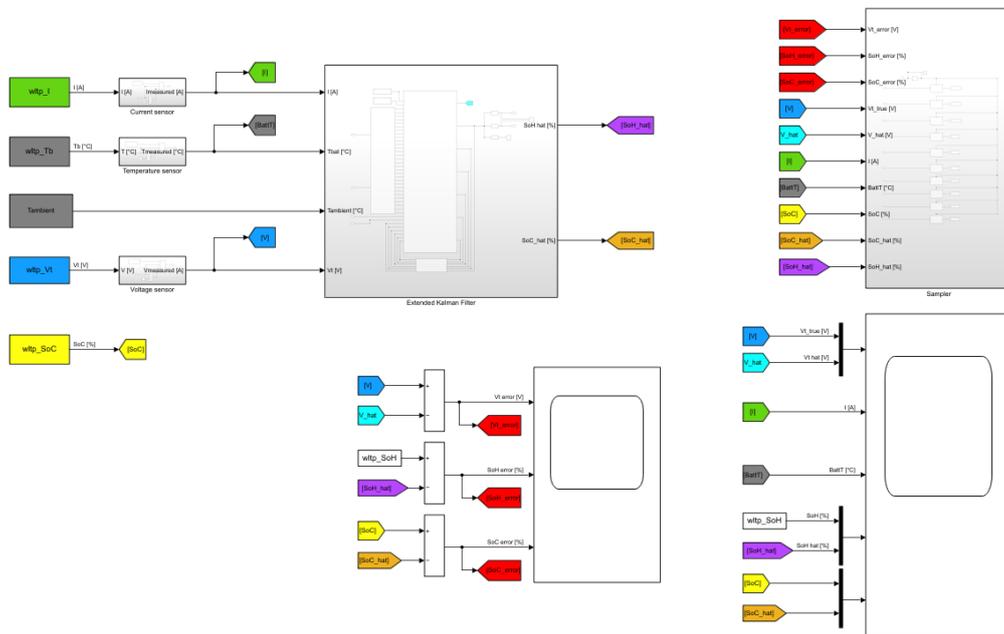


Figure A.8: Simulink schema implementing EKF estimator.

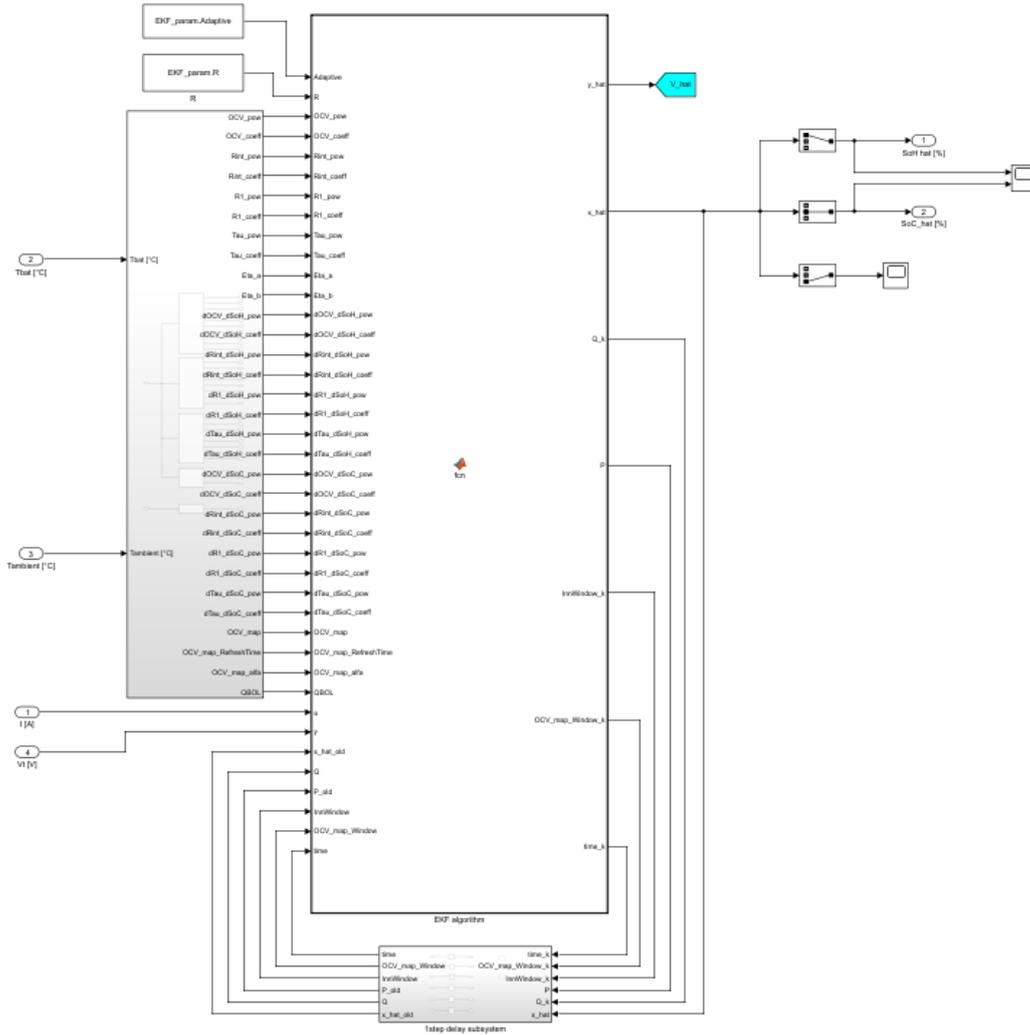


Figure A.9: EKF inner structure.

# Bibliography

- [1] United Nations Department of Economic and Social Affairs. World population prospects 2019. URL: <https://population.un.org/wpp/Graphs/DemographicProfiles/Line/900>.
- [2] Global Footprint Network. Open data platform. URL: [https://data.footprintnetwork.org/?\\_ga=2.79056983.233411667.1621272074-1349089283.1621272074#/](https://data.footprintnetwork.org/?_ga=2.79056983.233411667.1621272074-1349089283.1621272074#/).
- [3] Focus Elisabetta Intini. Ecologia overshoot day: il nostro debito con la terra, 20 september 2012. URL: <https://www.focus.it/ambiente/ecologia/oggi-e-l-overshoot-day-siamo-ufficialmente-in-debito-con-il-pianeta>.
- [4] Volkswagen AG. The components of the meb battery system, 12 may 2019. [Online; Image No: DB2019NR01227]. URL: [https://uploads.volkswagen-newsroom.com/system/production/media/31847/images\\_file\\_en/4b9b341bfff98ce0fee11b7da13d07640c515966/DB2019NR01227\\_medium.jpg?1581018055&disposition=attachment](https://uploads.volkswagen-newsroom.com/system/production/media/31847/images_file_en/4b9b341bfff98ce0fee11b7da13d07640c515966/DB2019NR01227_medium.jpg?1581018055&disposition=attachment).
- [5] LG Chem. Nominal specification of inr 18650 m29 lithium ion cell battery datasheet. [Screenshot]. URL: [http://akkuplus.de/mediafiles/Datenblatt/LG/LG\\_INR18650M29.pdf](http://akkuplus.de/mediafiles/Datenblatt/LG/LG_INR18650M29.pdf).
- [6] Wikipedia. Lithium-ion battery. URL: [https://en.wikipedia.org/wiki/Lithium-ion\\_battery](https://en.wikipedia.org/wiki/Lithium-ion_battery).
- [7] Let's Talk Science. Lithium-ion\_battery\_discharging. [Online; adaptation]. URL: [https://letstalkscience.ca/sites/default/files/styles/x\\_large/public/2019-10/Lithium-ion\\_battery\\_discharging.png?itok=yRq65X9B](https://letstalkscience.ca/sites/default/files/styles/x_large/public/2019-10/Lithium-ion_battery_discharging.png?itok=yRq65X9B).
- [8] Let's Talk Science. Lithium-ion\_battery\_charging. [Online; adaptation]. URL: <https://letstalkscience.ca/sites/default/files/styles/>

- x\_large/public/2019-10/Lithium-ion\_battery\_charging.png?itok=WeF5vd\_m.
- [9] Sweden Helena Berg, AB Libergreen. *Batteries for Electric Vehicles Materials and Electrochemistry*, chapter 6, page 173. Cambridge University Press, August 2015. Figure 6.2.
- [10] Sweden Helena Berg, AB Libergreen. *Batteries for Electric Vehicles Materials and Electrochemistry*, chapter 6, page 174. Cambridge University Press, August 2015. Figure 6.3.
- [11] image2016-7-28\_13-19-20. [Online; adaptation]. URL: [https://accubattery.zendesk.com/hc/en-us/article\\_attachments/205420125/image2016-7-28\\_13-19-20.png](https://accubattery.zendesk.com/hc/en-us/article_attachments/205420125/image2016-7-28_13-19-20.png).
- [12] image2016-7-28\_11-57-28. [Online; adaptation]. URL: [https://accubattery.zendesk.com/hc/en-us/article\\_attachments/205420025/image2016-7-28\\_11-57-28.png](https://accubattery.zendesk.com/hc/en-us/article_attachments/205420025/image2016-7-28_11-57-28.png).
- [13] Sweden Helena Berg, AB Libergreen. *Batteries for Electric Vehicles Materials and Electrochemistry*, chapter 1, page 34. Cambridge University Press, August 2015. Figure 1.23.
- [14] Sweden Helena Berg, AB Libergreen. *Batteries for Electric Vehicles Materials and Electrochemistry*, chapter 1, page 35. Cambridge University Press, August 2015. Figure 1.25.
- [15] Adeepa Palihawadana. Design of the propulsion system for an electric formula car - scientific figure on researchgate. [Online; "accessed 20 May, 2021"; "Ashish 2017"]. URL: [https://www.researchgate.net/figure/State-of-Charge-and-Depth-of-Discharge-Ashish-2017\\_fig57\\_322116711](https://www.researchgate.net/figure/State-of-Charge-and-Depth-of-Discharge-Ashish-2017_fig57_322116711).
- [16] Electropaedia. State of health (soh) determination. URL: <https://www.mpoweruk.com/soh.htm>.
- [17] Ruomei Zhou, Shasha Fu, and Weiwen Peng. A review of state-of-health estimation of lithiumion batteries: Experiments and data. In *2020 Asia-Pacific International Symposium on Advanced Reliability and Maintenance Modeling (APARM)*, pages 1–6, 2020. doi:10.1109/APARM49247.2020.9209548.
- [18] Web of Science. Number of publications vs years. [Online; "accessed 24 May, 2021"]. URL: <http://isiknowledge.com>.

- [19] Dieselnet.com. Figure 1 - wltc cycle for class 3b vehicles. [Online; "accessed 8 Aug, 2021"]. URL: <https://dieselnet.com/standards/cycles/wltp.php>.
- [20] Mohammad Charkhgard and Mohammad Farrokhi. State-of-charge estimation for lithium-ion batteries using neural networks and ekf. *IEEE Transactions on Industrial Electronics*, 57(12):4178–4187, 2010. doi:10.1109/TIE.2010.2043035.
- [21] Mingyue Zhang and Xiaobin Fan. Review on the state of charge estimation methods for electric vehicle battery. *World Electric Vehicle Journal*, 11(1), 2020. URL: <https://www.mdpi.com/2032-6653/11/1/23>, doi:10.3390/wevj11010023.
- [22] Lucian Ungurean, Gabriel Cârstoiu, Mihai V. Micea, and Voicu Groza. Battery state of health estimation: a structured review of models, methods and commercial devices. *International Journal of Energy Research*, 41(2):151–181, 2017. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1002/er.3598>, arXiv:<https://onlinelibrary.wiley.com/doi/pdf/10.1002/er.3598>, doi:<https://doi.org/10.1002/er.3598>.
- [23] Weizhong Wang, Nicholas W Brady, Chenyao Liao, Youssef A Fahmy, Ephrem Chemali, Alan C West, and Matthias Preindl. High-fidelity state-of-charge estimation of li-ion batteries using machine learning. *arXiv preprint arXiv:1909.02448*, 2019.
- [24] Ala Al-Haj Hussein and Issa Batarseh. An overview of generic battery models. In *2011 IEEE Power and Energy Society General Meeting*, pages 1–6, 2011. doi:10.1109/PES.2011.6039674.
- [25] Jinhao Meng, Guangzhao Luo, Mattia Ricco, Maciej Swierczynski, Daniel-Ioan Stroe, and Remus Teodorescu. Overview of lithium-ion battery modeling methods for state-of-charge estimation in electrical vehicles. *Applied sciences*, 8(5):659, 2018.
- [26] Matthieu Urbain, Stephane Raël, Bernard Davat, and Philippe Desprez. State estimation of a lithium-ion battery through kalman filter. In *2007 IEEE Power Electronics Specialists Conference*, pages 2804–2810. IEEE, 2007.
- [27] Mori W Yatsui and Hua Bai. Kalman filter based state-of-charge estimation for lithium-ion batteries in hybrid electric vehicles using pulse charging. In *2011 IEEE Vehicle Power and Propulsion Conference*, pages 1–5. IEEE, 2011.

- [28] Weihua Wang and Jiayi Mu. State of charge estimation for lithium-ion battery in electric vehicle based on kalman filter considering model error. *IEEE Access*, 7:29223–29235, 2019.
- [29] Carlo Taborelli and Simona Onori. State of charge estimation using extended kalman filters for battery management system. In *2014 IEEE International Electric Vehicle Conference (IEVC)*, pages 1–8. IEEE, 2014.
- [30] M Satish Kumar, Thumpiri Reddy Manasa, B Raja, and K Selvajyothi. Estimation of state of charge and terminal voltage of li-ion battery using extended kalman filter. In *2020 6th IEEE International Energy Conference (ENERGY-Con)*, pages 515–520. IEEE, 2020.
- [31] M Oya, Kyotsugu Takaba, Lei Lin, R Ishizaki, N Kawarabayashi, and Masahiro Fukui. Accurate soc estimation of lithium-ion batteries based on parameter-dependent state-space model. In *2015 15th International Symposium on Communications and Information Technologies (ISCIT)*, pages 37–40. IEEE, 2015.
- [32] Eunseok Choi and Sekchin Chang. A temperature-dependent state of charge estimation method including hysteresis for lithium-ion batteries in hybrid electric vehicles. *IEEE Access*, 8:129857–129868, 2020.
- [33] Eric Wan and Ronell Merwe. The unscented kalman filter for nonlinear estimation. volume 153-158, pages 153 – 158, 02 2000. doi:10.1109/ASSPCC.2000.882463.
- [34] Edoardo Locorotondo, Luca Pugi, Lorenzo Berzi, Marco Pierini, and Giovanni Lutzemberger. Online identification of thevenin equivalent circuit model parameters and estimation state of charge of lithium-ion batteries. In *2018 IEEE International Conference on Environment and Electrical Engineering and 2018 IEEE Industrial and Commercial Power Systems Europe (EEEIC/I&CPS Europe)*, pages 1–6. IEEE, 2018.
- [35] Desham Mitra and Siddhartha Mukhopadhyay. Ukf based estimation of soc and core temperature of a lithium ion cell using an electrical cell model. In *2018 15th IEEE India Council International Conference (INDICON)*, pages 1–6. IEEE, 2018.
- [36] Muhammad Umair Ali, Amad Zafar, Sarvar Hussain Nengroo, Sadam Hussain, Muhammad Junaid Alvi, and Hee-Je Kim. Towards a smarter battery management system for electric vehicle applications: A critical review of lithium-ion

- battery state of charge estimation. *Energies*, 12(3):446, 2019.
- [37] Alexandre Barbosa De Lima, Maurício BC Salles, and José Roberto Cardoso. State-of-charge estimation of a li-ion battery using deep forward neural networks. *arXiv preprint arXiv:2009.09543*, 2020.
- [38] Chaoran Li, Fei Xiao, Yaxiang Fan, Guorun Yang, and Weiwei Zhang. A recurrent neural network with long short-term memory for state of charge estimation of lithium-ion batteries. In *2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*, pages 1712–1716. IEEE, 2019.
- [39] Ephrem Chemali, Phillip J Kollmeyer, Matthias Preindl, Ryan Ahmed, and Ali Emadi. Long short-term memory networks for accurate state-of-charge estimation of li-ion batteries. *IEEE Transactions on Industrial Electronics*, 65(8):6730–6739, 2017.
- [40] MathWorks. Datasheet battery model documentation. [Online; "accessed 2 Oct, 2021"]. URL: [https://www.mathworks.com/help/autoblks/ref/datasheetbattery.html?searchHighlight=Datasheet%20battery%20model&s\\_tid=srchtitle](https://www.mathworks.com/help/autoblks/ref/datasheetbattery.html?searchHighlight=Datasheet%20battery%20model&s_tid=srchtitle).
- [41] MathWorks. Generic battery model documentation. [Online; "accessed 2 Oct, 2021"]. URL: <https://www.mathworks.com/help/physmod/sps/powersys/ref/battery.html>.
- [42] MathWorks John D’Errico. Polyfitn function. [Online; "accessed 2 Oct, 2021"]. URL: <https://www.mathworks.com/matlabcentral/fileexchange/34765-polyfitn>.
- [43] Mili Mukherjee, Manasi Das, and Smita Sadhu. Q-adaptive extended kalman filter for nonlinear systems with unknown noise statistics. In *2018 15th IEEE India Council International Conference (INDICON)*, pages 1–6, 2018. doi: 10.1109/INDICON45594.2018.8987109.