

# POLITECNICO DI TORINO

Master's Degree in Automotive Engineering



**Politecnico  
di Torino**

Master's Degree Thesis

## Model-based State-of-Charge Estimation of Tesla Model 3 Battery using Extended Kalman Filter

Supervisors

Prof. Ezio SPESSA

Federico MIRETTI, PhD

Matteo ACQUARONE, M.Eng.

Candidate

Alexander Fernando LAUVANDY

November 2024



## Abstract

Retrieving an accurate battery's state-of-charge is crucial for dependable operation in a variety of use cases as the era of vehicle electrification advances. Traditional methods of state-of-charge estimation often struggle with the underlying non-linear behavior and the battery's temperature effect on the estimation's accuracy. This work suggests a model-based method using the Extended Kalman Filter (EKF) to estimate the battery's state-of-charge under multiple tests while accounting for the temperature effect to improve its robustness and accuracy. The testing dataset from the Tesla Model 3 NCA Li-ion battery, which McMaster University made available online, supports the study.

The methodology starts with developing a second-order continuous dual polarization (DP) equivalent circuit model. This model is used to estimate ECM parameters based on various characteristic tests and on multiple test temperatures (-20, -10, 0, 10, 25, and 40°C) given by the dataset. Once all parameters are acquired and validated, the discretized model is then developed on the base of its continuous counterpart. The EKF algorithm is then employed on top of the discretized DP model. The root mean square errors between the measured SOC and simulated SOC are calculated to quantify the performance of the EKF across multiple driving cycles.

The work demonstrates a significant increase in ECM accuracy (up to 66.3% on a highway driving cycle including the effect of road grade) by incorporating the real-time battery temperature data into the model. The proposed EKF-based SOC estimation achieves excellent accuracy and robustness at higher temperatures with an acceptable limited error band. The EKF, however, couldn't perform adequately at low temperatures (-20°C and -10°C) due to the increasing hysteresis phenomenon which is uncaptured by the ECM model. Several solutions are proposed to tackle the problem, leaving the work open for further improvement.

*Keyword:* State-of-charge, lithium-ion battery, equivalent circuit model (ECM), dual-polarization (DP) model, Extended Kalman Filter (EKF), parameter extraction, electric vehicles.



# Acknowledgements

First and foremost, I would like to express my deepest gratitude to God, whose blessings and guidance have given me the strength, resilience, and clarity to complete this thesis. Through every challenge, His love strengthens me and reminds me to always come back to Him.

I am incredibly grateful to my professors, Prof. Spessa, Prof. Miretti, and particularly Matteo Acquarone, for their invaluable guidance, expertise, and encouragement throughout the journey. I thank them for the time, guidance, and understanding given to me, even outside of working hours and during the holidays.

To my family, my heartfelt thanks for your constant love and encouragement. Your belief in me has been my source of strength, and your unwavering support has been a steady foundation on which I have built my academic and personal achievements. Thank you for always being by my side no matter what. This accomplishment would not have been possible without your love, sacrifices, and encouragement.

Lastly, I want to thank my girlfriend, all my friends and colleagues, for always being there for me, through every high and low. The level of understanding, patience, and encouragement have been invaluable, helping me stay focused and motivated even during the toughest moments of this journey. Thank you to everyone who has been a part of this journey, directly or indirectly, for helping me reach this point.

*"witing mulyo jalaran wani rekoso"*

Alexander Fernando Lauvandy



# Table of Contents

<b>List of Tables</b>	VI
<b>List of Figures</b>	VII
<b>1 Introduction</b>	1
1.1 Background . . . . .	1
1.2 Thesis Objective . . . . .	2
1.3 Thesis Organization . . . . .	3
<b>2 Methodology</b>	4
2.1 Battery Diagnostic Test and Dataset Analysis . . . . .	4
2.1.1 Characterization Test . . . . .	5
2.1.2 Driving Cycle Test . . . . .	6
2.2 Methodology Framework . . . . .	8
<b>3 Dual-Polarization Equivalent Circuit Model</b>	11
3.1 One-state dual polarization ECM . . . . .	11
3.1.1 Simulink model representation . . . . .	13
3.1.2 Parameters extraction for initial value estimation . . . . .	14
3.1.3 Initial calculation of the RC pairs . . . . .	18
3.1.4 Parameter Estimation and Optimization . . . . .	21
3.1.5 Model Validation on HPPC Test and Driving Cycles . . . . .	26
3.2 Two-state dual polarization ECM . . . . .	33
3.2.1 Model description . . . . .	33
3.2.2 Benchmark result: Fixed input temperature simulation . . . . .	39
3.2.3 Model Validation on Driving Cycle . . . . .	43
<b>4 Extended Kalman Filter Model</b>	47
4.1 One-state discrete model . . . . .	47
4.2 One-state Extended Kalman Filter . . . . .	51
4.2.1 Extended Kalman Filter development: six-step method . . . . .	51

4.2.2	Parameters Initialization and Tuning . . . . .	54
4.2.3	One-state EKF Model Validation on the HPPC and Driving Cycle Tests . . . . .	54
4.3	Two-state Extended Kalman Filter . . . . .	66
4.3.1	Model Description . . . . .	66
4.3.2	Two-state EKF Model Validation on Driving Cycles . . . . .	68
4.3.3	Two-state EKF Model Robustness Test on Driving Cycles . . . . .	72
<b>5</b>	<b>Attempts to enhance the ECM performance at low temperatures</b>	<b>75</b>
5.1	Tuning ECM parameters on concatenated HPPC test . . . . .	75
5.2	Tuning ECM parameters on concatenated Driving Cycle test . . . . .	78
5.3	Direct use of the OCV-SOC curve . . . . .	81
5.4	Hysteresis Voltage Modeling . . . . .	83
<b>6</b>	<b>Conclusion and Suggestion</b>	<b>85</b>
	<b>Bibliography</b>	<b>87</b>

# List of Tables

2.1	Dataset test case organization . . . . .	4
2.2	Arbin Cell Tester Specifications . . . . .	5
2.3	Capacity at different temperatures for 40°C C/20 rate . . . . .	6
3.1	First estimate value of capacitors . . . . .	21
3.2	HPPC average battery temperature . . . . .	37
3.3	REORDERED1 average battery temperature . . . . .	41
3.4	HWGRADE1 average battery temperature . . . . .	41
3.5	RMSE values for different test temperatures under REORDERED1 and HWGRADE1 driving cycles . . . . .	42
4.1	Temperature RMSE for REORDERED1 and HWGRADE in normal and wrong initialization test cases . . . . .	69
4.2	Temperature RMSE for REORDERED1 and HWGRADE under Current Offset Tests . . . . .	73
5.1	RMSE Comparison of Base Model and Model with Hysteresis by Bodnar [31] . . . . .	84

# List of Figures

2.1	HPPC Test procedure . . . . .	7
2.2	Voltage, SOC, and Current of HPPC test for m80 load case under 25°C test temperature . . . . .	7
2.3	Methodology Workflow . . . . .	10
3.1	Dual-Polarization Equivalent Circuit Model . . . . .	12
3.2	Equation governing dual-polarization equivalent circuit model expressed in Simulink environment . . . . .	14
3.3	Equation governing resistor-capacitor pairs expressed in Simulink environment . . . . .	15
3.4	OCV dependence on SOC obtained from 25°C capacity test . . . .	16
3.5	HPPC discharge and charge current pulse profiles and the voltage response at the n-th pulse . . . . .	17
3.6	Ohmic Resistance $R_0$ initial estimation as a function of SOC . . . .	18
3.7	Plett R-C pair parameter extraction from HPPC test [5] . . . . .	19
3.8	Initialized datapoints of $R_1 + R_2$ . . . . .	20
3.9	Constant initialized value of $C_1$ and $C_2$ over SOC for 25°C dataset .	22
3.10	OCV-SOC profile comparison between the initial extracted data from the capacitance test (Initial dataset) and the parameter estimation result (Estimated dataset) for 25°C . . . . .	24
3.11	$R_0$ profile comparison between the initial extracted data from the capacitance test (Initial dataset) and the parameter estimation result (HPPC optimized) for 25°C dataset . . . . .	25
3.12	$R_1$ and $R_2$ profile from the parameter estimation result for 25°C dataset . . . . .	26
3.13	$C_1$ and $C_2$ profile from the parameter estimation result for 25°C dataset . . . . .	27
3.14	ECM test on 25°C HPPC Test . . . . .	28
3.15	ECM validation on 25°C REORDERED1 Driving Cycle . . . . .	29

3.16	Enlarged section of ECM validation on 25°C REORDERED1 Driving Cycle. Notice the fluctuations of error are caused by the uncaptured dynamics . . . . .	29
3.17	HWGRADE1 Driving Cycle (25°C data) ECM result . . . . .	30
3.18	ECM validation on 25°C HWGRADE1 Driving Cycle . . . . .	31
3.19	SOC-segmented RMSE result for 25°C HWGRADE1 dataset . . . . .	31
3.20	Real-time battery temperature of HWGRADE 25°C . . . . .	32
3.21	OCV-SOC profile for various temperatures . . . . .	34
3.22	$R_0$ profile as a function of SOC for various temperatures . . . . .	35
3.23	$R_1$ and $R_2$ profile as a function of SOC for various temperatures . . . . .	36
3.24	$C_1$ and $C_2$ profile as a function of SOC for various temperatures . . . . .	37
3.25	Main equation of 2D ECM model using 2D LUT for OCV and $R_0$ that takes SOC and temperature as inputs . . . . .	38
3.26	Example of OCV look-up table composed of OCV grid on various temperatures and SOC levels . . . . .	38
3.27	HPPC real-time battery temperature for -20°C . . . . .	39
3.28	HPPC real-time battery temperature for 25°C . . . . .	39
3.29	REORDERED1 real-time battery temperature for -20°C . . . . .	40
3.30	HWGRADE1 real-time battery temperature for -20°C . . . . .	40
3.31	Two-state ECM with fixed temperature input validated on HWGRADE1 -20°C dataset. Voltage RMSE: 661.40 mV . . . . .	42
3.32	Two-state ECM with fixed temperature input validated on HWGRADE1 -10°C dataset. Voltage RMSE: 464.12 mV . . . . .	43
3.33	REORDERED1 driving cycle RMSE in mV comparison between fixed temperature input and real-time temperature input on two-state DP ECM . . . . .	44
3.34	HWGRADE1 driving cycle RMSE in mV comparison between fixed temperature input and real-time temperature input on two-state DP ECM . . . . .	44
3.35	Two-state ECM with real-time temperature input validated on HWGRADE1 -20°C dataset. Voltage RMSE: 222.41 mV . . . . .	45
3.36	Two-state ECM with real-time temperature input validated on HWGRADE1 -10°C dataset. Voltage RMSE: 153.46 mV . . . . .	46
3.37	Hysteresis example. Retrieved from Plett [5] . . . . .	46
4.1	Gridded Interpolant approach as a look-up table for DP ECM parameters in a given SOC . . . . .	49
4.2	Discrete model voltage output comparison . . . . .	50
4.3	Voltage discrepancy between discrete model and continuous model (above) and discrete model and measurement data (below) . . . . .	50

4.4	SOC comparison between EKF model and measured data on HPPC with correct initialization . . . . .	56
4.5	EKF wrong initialization (80%) test on HPPC . . . . .	56
4.6	EKF wrong initialization (50%) test on HPPC . . . . .	57
4.7	Current offset test for HPPC with 0.1 and 0.5 A offset . . . . .	57
4.8	Adaptive behaviour of EKF model under 0.1 A offset current test . . . . .	58
4.9	Adaptive behaviour of EKF model under 0.5 A offset current test . . . . .	58
4.10	EKF performance under REORDERED 1 Driving cycle . . . . .	59
4.11	Voltage discrepancy between the 1D discrete model and the Simulink (above) and the measured value (below). The drift between the discrete model and the measured value results in poor estimation for the EKF . . . . .	60
4.12	EKF wrong initialization (80%) test on REORDERED1 . . . . .	60
4.13	EKF wrong initialization (50%) test on REORDERED1 . . . . .	61
4.14	EKF result on REORDERED 1 with 0.1 A current offset . . . . .	62
4.15	EKF result on REORDERED 1 with 0.5 A current offset . . . . .	62
4.16	SOC comparison between EKF model and measured data on HW-GRADE 1 with correct initialization . . . . .	63
4.17	SOC comparison between EKF model and measured data on HW-GRADE 1 with 80% initial SOC . . . . .	63
4.18	SOC comparison between EKF model and measured data on HW-GRADE 1 with 50% initial SOC . . . . .	64
4.19	EKF result on HWGRADE 1 with 0.1 A current offset . . . . .	65
4.20	EKF result on HWGRADE 1 with 0.5 A current offset . . . . .	65
4.21	Two-state <i>griddedInterpolant</i> on MATLAB . . . . .	66
4.22	Calling the 2D <i>griddedInterpolant</i> . . . . .	67
4.23	Old tuning 2D EKF result on -10°C HWGRADE1 driving cycle . . . . .	67
4.24	Best scenario for EKF model under REORDERED1 25°C with 0.38% RMSE . . . . .	70
4.25	Challenging scenario for EKF model under HWGRADE -20°C due to low ECM accuracy. SOC RMSE = 2.10% . . . . .	70
4.26	Agile EKF performance under 25°C REORDERED1 with 80% initial SOC . . . . .	71
4.27	Failing EKF under HWGRADE1 -20°C dataset with 50% initial SOC . . . . .	72
4.28	Two-state EKF performance under 10°C REORDERED1 with 0.1 A current offset . . . . .	73
4.29	Failing two-state EKF robustness test for 10°C REORDERED1 with 0.5 A current offset . . . . .	74
5.1	Concatenated HPPC test . . . . .	76
5.2	Optimized OCV and $R_0$ from HPPC tests concatenated . . . . .	77

5.3	Optimized RC pairs parameters from HPPC tests concatenated . . .	77
5.4	Voltage RMSE for concatenated HPPC tuning on REORDERED1 . . .	78
5.5	Voltage RMSE for concatenated HPPC tuning on HWGRADE1 . . .	78
5.6	Voltage RMSE for concatenated driving cycles tuning on REORDERED1 . . .	79
5.7	Voltage RMSE for concatenated driving cycles tuning on HWGRADE1 . . .	79
5.8	Voltage RMSE for concatenated driving cycles tuning on HPPC . . .	80
5.9	Huge voltage offset on HPPC test with model using the parameters extracted from concatenated driving cycles . . . . .	80
5.10	OCV-SOC profile calculated using Equation 5.1 . . . . .	82
5.11	REORDERED1 -20°C result for direct OCV usage attempt . . . . .	82
5.12	HWGRADE1 -20°C result for direct OCV usage attempt . . . . .	83
5.13	SOC-averaged hysteresis as a function of temperature. Retrieved from Plett [5] . . . . .	84

# Chapter 1

## Introduction

### 1.1 Background

The necessity of battery engineering infrastructure to guarantee safety and dependability during real-time operation has been highlighted by the increasing demands of lithium-ion batteries (LIB) for grid storage, electric vehicles (EV), and aviation applications. In the electric vehicle sector alone, Global EV Outlook 2024 reported rising demand for batteries as a result of the increase in EV sales, which is maintaining the increasing trend seen in recent years [1]. Although the annual growth rate decreased marginally from 2021 to 2022, the demand for EV batteries increased by 40% to over 750 GWh in 2023, of which around 95% of this rise is attributable to electric vehicles sales. These surging demands for EV batteries are followed by competition among manufacturers and battery suppliers for safe and reliable batteries, which is translated into a set of specific requirements. One of those is to prevent overcharging or deep draining and thus, promote a longer lifespan, considering the cost of the battery itself constitutes around one-third of the cost of the vehicle [2]. To achieve it, an accurate state-of-charge (SOC) estimation is vital for maintaining the dependability, security, and lifespan of battery systems. State-of-charge is an internal parameter that reflects the remaining usable capacity in the battery, which would be the basis of other important control logic employed in the vehicle [3].

Unfortunately, the capacity of the battery (which lies the foundation of the state-of-charge definition) in a given time is not a quantity that can be measured directly. Batteries are complicated electrochemical systems that are both nonlinear and dynamic. Their properties change over time, and their internal electrical parameters change both during charging and discharging and over the course of their lifetime. Due to this complex behaviour, traditional and direct methods of SOC calculation such as Coloumb counting often struggle to predict the battery's

state. This method is based on the integration of current over time, which is very sensitive to initial conditions and the error accumulates over time [4], raising the need for a more advanced and accurate SOC prediction.

Over the years, various solutions have been proposed to accurately estimate the battery's SOC, mainly divided into 2 domains: physics-based model and data-driven model. Physics-based models contain differential equations to capture a set of specific behavior of the battery. Normally, the model is complex and requires a deep understanding of the cell's characteristics and electrochemistry of the battery for accurate parameterization [5]. On the opposite side, data-driven models utilize machine learning methods to generate the model straight from a set of training data, containing the measured quantity from prior testing. The model becomes a black box with input and output based on the training dataset. However, machine learning algorithms require huge datasets to properly train the model and to include multiple scenarios and use cases [6]. To solve this problem, the equivalent circuit model (ECM) is widely implemented for real-time application on-board. ECM doesn't model the internal dynamic inside the battery, rather, they are represented in an equivalent electrical circuit. It provides a good compromise between accuracy and complexity. Nonetheless, ECM's prediction is imprecise and is still highly sensitive to measurement error and noises which deteriorates the model's accuracy [7]. A common approach to solve this problem is using Kalman filtering-based method, such as the extended Kalman filter (EKF), as reported in multiple journals [5, 8, 9, 10, 11, 12].

In the previous works provided, the temperature effect on model's accuracy is often overlooked. One strong work from Khanum [13] incorporated temperatures in developing Kalman filters built on top of second-order ECM battery. It was tested on several dynamic cycles with a temperature range from -10 to 40 °C. This paper, however, is focused on developing the MATLAB tool instead of analyzing the temperature effect on the battery's performance. In addition, the validation test was done only on the LA92 driving cycle, a supplementary driving cycle to the Federal Test Procedure. Our work's aim is to develop an Extended Kalman Filter based on a two-state Dual Polarization equivalent circuit model while accounting for the temperature effect and testing the algorithm on multiple driving cycles. The dataset for parameterization and validation is the Tesla Model 3 NCA Li-ion battery testing, conducted and made available online by McMaster University [14].

## 1.2 Thesis Objective

The main focuses of this work are summarized as follows:

1. Developing a two-state dual polarization equivalent circuit model (ECM).

2. Developing a two-state Extended Kalman Filter (EKF) on the base of the ECM.
3. Validating the ECM and EKF model on multiple driving cycles and on multiple temperature datasets.

## **1.3 Thesis Organization**

This work is structured as follows:

1. Chapter 2 explains the framework of the thesis, along with the explanation of what is inside the dataset and which are the driving cycles used to validate the model.
2. Chapter 3 starts with a literature review around battery modeling. The main topic focuses on developing the one- and two-state dual polarization equivalent circuit model from the parameter extraction to the model's validation.
3. Chapter 4 reports how the extended Kalman filter is modeled and the SOC estimation results on multiple tests.
4. Chapter 5 gives the reader an idea of efforts conducted to make the equivalent circuit model accurate around the low temperature.
5. The last chapter concludes the thesis and gives the reader some suggestions for future exploration based on this work.

# Chapter 2

## Methodology

### 2.1 Battery Diagnostic Test and Dataset Analysis

The reference for our work is an open-source dataset of Tesla Model 3 battery packs, of which the test was conducted by McMaster Automotive Research Center (MARC) [14]. Given hundreds of SOC estimation methods proposed over the year, McMaster developed a blind modeling tool as a standardized comparison method for collaborators and developers to test their algorithm. Some of the datasets are open to public for SOC estimation parameterization (for ECM-based algorithm) or training data (data-driving/ machine learning based tool).

The structure of the dataset consists of multiple test cases, and each of the test cases consists of the corresponding characterization (diagnostic) test and the driving cycle test. There were four battery cells under test, each for every test case. The test cases are composed of varying payload mass (in addition to the vehicle curb mass of 1612 kg) and HVAC load as given in the following Table 2.1.

Cell number	Payload mass	HVAC load	Dataset name
Cell 1	80 kg	ON	m80
Cell 2	448 kg	ON	m448
Cell 3	448 kg	OFF	m448-N
Cell 4	1000 kg	ON	m1000

**Table 2.1:** Dataset test case organization

The battery cell under test is a cylindrical Panasonic 2170 Lithium-ion battery with Nickel Cobalt Aluminium (NCA) cathode. Based on the given datasheet, the upper cut-off voltage is 4.2 V and the lower cut-off voltage is 2.8 V.

On each test case, the evaluation was carried out in 16  $ft^3$  Envirotronics SH16C thermal chamber with various condition temperatures:  $-20^\circ\text{C}$ ,  $-10^\circ\text{C}$ ,  $0^\circ\text{C}$ ,  $10^\circ\text{C}$ ,  $25^\circ\text{C}$ , and  $40^\circ\text{C}$ . The testing tool used is the Arbin with specifications as shown in Table 2.2 below. Although Arbin could handle sampling frequencies up to 2000 Hz, the time arrays of all the testing are resampled to 1 Hz.

Specification	Details
Voltage	0V to 5V
Current	60A per channel
Number of channels	8
Parallel operation	2 to 8 sequential channels can be operated in parallel
Series operation	Series connections cannot be made
Input impedance	50 M $\Omega$
Current Range	+/- 60A, 5A, 500mA, 20mA
Control Accuracy	+/- 24mA, 2mA, 200 $\mu\text{A}$ , 8 $\mu\text{A}$ (0.04%)
Voltage Accuracy	+/- 2mV (0.04%)
Max command rate	5ms
Max system log rate	2000 samples per second
Temperature Sensing	16 channels, type T thermocouples
Control Software	Arbin MITS 8.0

**Table 2.2:** Arbin Cell Tester Specifications

### 2.1.1 Characterization Test

A diagnostic test or reference performance test (RPT) is a standardized set of procedures to characterize the battery aging process and track the performance of battery cells. In the battery's lifetime, the RPTs are commonly performed at regular intervals, either monthly or every 100 full equivalent cycles to assess the battery performance. Another RPT is usually conducted at the end-of-life (EoL) of the battery to assess its characteristics after a defined lifetime [15]. There are mainly 3 types of diagnostic tests conducted: the capacity test, EIS, and HPPC test. Inside this dataset, only the capacity and HPPC tests are present, from which the battery parameters are derived.

The capacity test is mainly used to assess the discharged capacity and consequently, the dependence of OCV against SOC. The test is done by discharging the battery at a very low constant current (CC) of C/20 C-rate which is measured at 40°C before the beginning of the test at each temperature. As mentioned in the previous section, the capacity highly depends on temperature. Knowing the proper discharge capacity is crucial in determining the SOC. The complete list of the battery’s capacity and the associated temperature is given in Table 2.3.

40°C C/20 Capacity [Ah]	Associated Temperature [°C]
4.6768	40
4.5339	25
4.2885	10
4.3009	0
4.2629	-10
4.2112	-20

**Table 2.3:** Capacity at different temperatures for 40°C C/20 rate

Figure 2.1 illustrates the HPPC test performed on each battery cell. The HPPC tests were performed with multiple C-rates for every SOC level. They consist of four ten-second discharge and charge pulses in the following order, with 0.3C pulses in between to return SOC to the prior value. The 0.3C return pulses ensure that each 10s charge and discharge pulse is performed at the same SOC. The same HPPC tests were repeated for every test case and every test temperature. Figure 2.2 presents the voltage, SOC, and current pulses characteristic. A negative current indicates the discharge phase. We could appreciate that four discharge phases with different C-rate levels are present in each SOC level. The SOC data plotted in the figure comes from the Arbin testing tool and is attached to the dataset. Focusing on the SOC level, the HPPC test is performed until the SOC is almost depleted. Nevertheless, such a condition is not necessarily true for all tests. On some test temperatures (particularly the lower temperatures), the HPPC tests don’t reach a low SOC level. This might hinder our characterization process which requires some modifications as explained later in Chapter 3.

### 2.1.2 Driving Cycle Test

The driving cycle is the main test for validating our model. A total of 16 driving cycles are tested on each cell. Some of the tests are blind tests, meaning they are not shared publicly and are only used for validation purposes by McMaster University on the SOC estimation algorithm. The complete list of battery experiments and

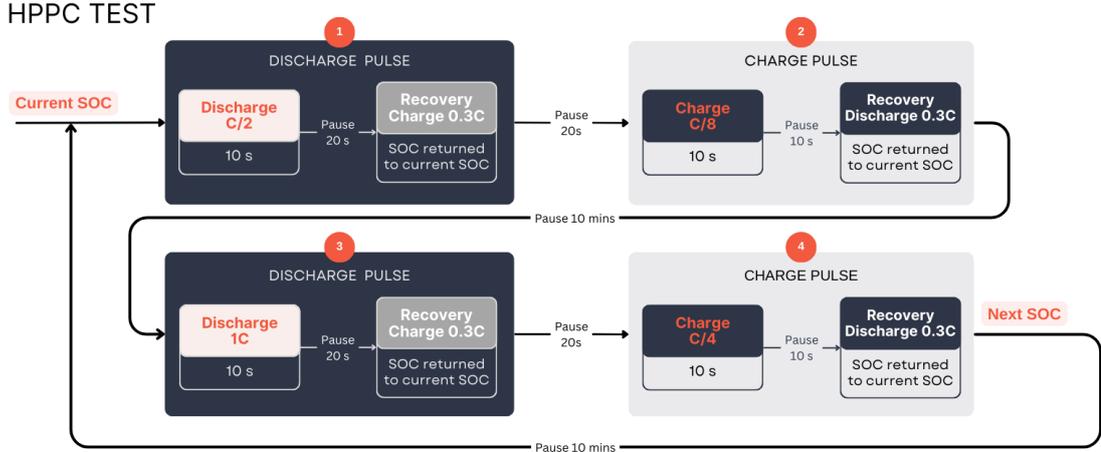


Figure 2.1: HPPC Test procedure

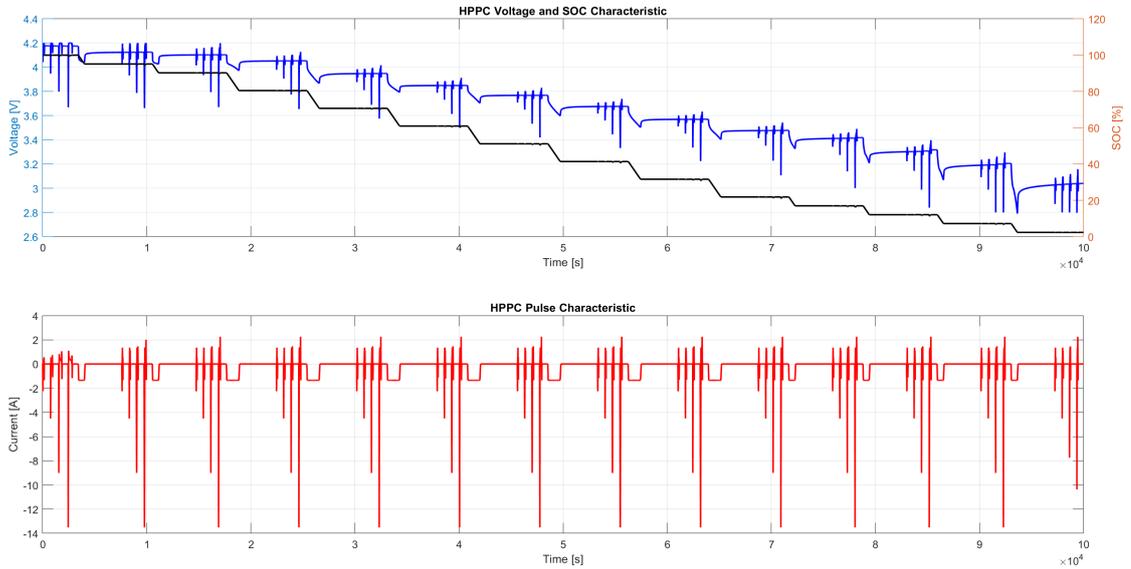


Figure 2.2: Voltage, SOC, and Current of HPPC test for m80 load case under 25°C test temperature

their corresponding status can be found in the published journal [14]. Inside the open dataset, eight REORDERED cycles, made up of randomized parts of the standardized cycles: UDDS, HWFET, LA92, and US06 cycles, are available for the public and serve as one of our main validation tests. To simulate driving across mountain passes, there are additionally HWCUST1, HWCUST2, HWGRADE1, and HWGRADE2, which are unique highway drive cycles with speeds ranging from 100 to 130 km/h with recurring steep ascents and descents. Only HWCUST1 and

HWGRADE1 are open to the public while the rest are dedicated to blind testing. Each of the driving cycles was tested repeatedly until the cell achieved the cut-off capacity, defined as the point at which the calculated pack power capability, based on the HPPC test results and the number of cells in the vehicle pack, equals 60 kW. Considering the random nature of REORDERED cycles, only one of them, REORDERED1, is used for our validation step. We argue that REORDERED1 and REORDERED8 have similar characteristics, so validating one is enough. For the highway and mountainous terrain, only HWGRADE1 is used as a validator.

## 2.2 Methodology Framework

In the following section, the workflow of this thesis is presented in Figure 2.3. In the initial phase, the continuous equivalent circuit model is developed. We started with the simple one-dimension dual-polarization equivalent circuit model for each of the temperature datasets, developed using the Simulink tool to represent the mathematical model. From the HPPC dataset, initial parameter guess is extracted on every temperature set to obtain the rough estimation of the ECM parameters which will reduce the time needed for our next step: Parameter Estimation. In this phase, optimization process is performed on the base of data fitting between the simulation output and the voltage measurement provided in the dataset. The outputs are sets of arrays for each parameter:  $OCV$ ,  $R_0$ ,  $R_1$ ,  $R_2$ ,  $C_1$ , and  $C_2$  as a function of SOC. Subsequently, these parameters' profiles are loaded on the model along with driving cycle data measurement and the model is run. This is the validation process, and the root mean square error between the simulated voltage from the model and the measured voltage is quantified. A similar procedure is conducted on all the temperature datasets until all the parameters' profiles are obtained. Six grids containing these profiles are arranged to form a 2D look-up table, with temperature and SOC as the breakpoints.

The next step is to develop the 2D dual-polarization ECM in a continuous domain using the defined parameters grid. The idea is strictly similar to our previous 1D model, the only difference is the usage of the 2D look-up table rather than 1D, which allows the model to have an additional input of the temperature as will be explained in the later chapter. As in the previous procedure, this model is then validated on the driving cycles by taking into account the temperature of the battery. This model marks the end of the ECM phase.

Coming into the second phase of our workflow, we started to develop the Extended Kalman Filter. Initially, the discrete model needs to be built as it serves as the basis of our Kalman filter. We started from the 1D discrete model and its corresponding Extended Kalman filter before moving on to the 2D model, in which all of them are written in MATLAB environment. In each of the discrete

model's procedures, the voltage of the discrete model, continuous model, and the real voltage measurement value are compared to check the discretization error (between discrete and continuous model) and the model's error (between discrete and measured data). As soon as the discrete model is validated on all temperature's HPPC and Driving cycles, the corresponding 1D EKF is matured. The important part of this section is the tuning of the EKF parameters, which consists of the initial state covariance, process noise covariance, and measurement noise covariance. The objective of the tuning is to obtain a robust filter, capable of performing under normal conditions, and under robustness tests, including incorrectly initialized state, and offset current input on all of the driving cycle tests. Once the tuning results in an acceptable SOC error under all cases mentioned above, we move on to the development of a 2D Extended Kalman Filter which has the same idea as the previous. Similarly, the only difference lies in the additional input for the model to take into account the battery's temperature.

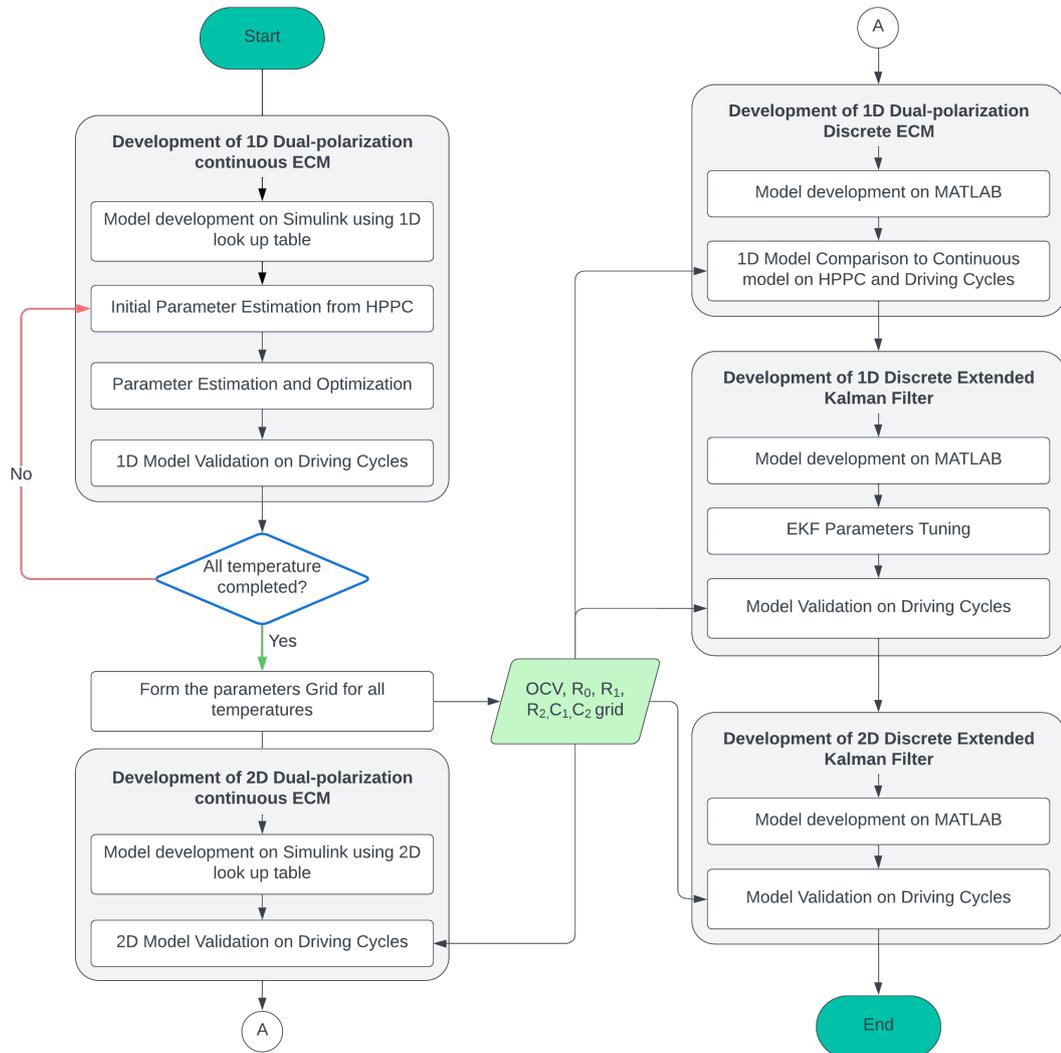


Figure 2.3: Methodology Workflow

## Chapter 3

# Dual-Polarization Equivalent Circuit Model

### 3.1 One-state dual polarization ECM

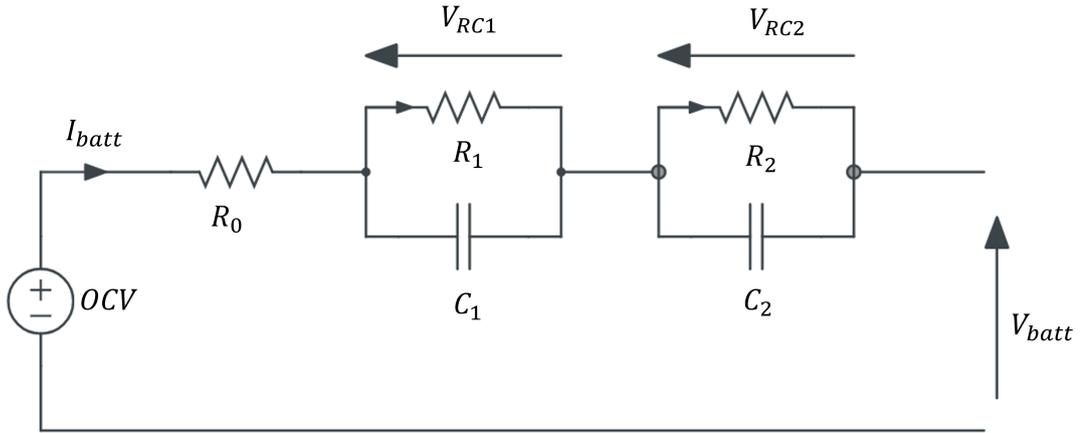
Lithium battery technology has advanced rapidly and steadily since Sony first introduced it to the market in 1991, making it a key component of EV innovation. Lithium-based batteries are now considered the best option for powering present and future generations of electric vehicles (EVs) due to their extremely effective energy storage capacities.

Efficient battery management requires precise State of Charge (SOC) estimates. Direct measurement methods and model-based estimate methods are the two primary categories into which SOC estimating techniques fall. Direct measurement techniques, such as Open Circuit Voltage (OCV) method [16] or the Coulomb counting (Ah) approach [17], are based on directly measurable battery characteristics and are typically open-loop and simpler to apply. These methods are sensitive to measurement errors which might result in inaccuracies.

The Dual-polarization (DP) model is one modeling approach considered suitable for automotive system-level complexity [18]. It was first developed from the Thevenin-based electric equivalent circuit (EEC) models with an additional RC pair to model the battery dynamic further. The equivalent circuit model is made up of a voltage source in series with a resistor and two RC blocks which model two stages of the dynamic process of the battery. These models are widely utilized because of their simplicity, ease of understanding, and standard test instrumentation requirements [18].

DP ECM model is represented by several core components:

1. Open-circuit voltage (*OCV*): The voltage represents the ideal condition of



**Figure 3.1:** Dual-Polarization Equivalent Circuit Model

the battery with no load. This parameter is highly dependent on the battery's SOC and could be extracted through a capacitance test.

2. Internal resistance ( $R_0$ ): A resistor mounted in series models the ohmic losses of the battery. When a current is flowing through, this parameter models the instantaneous voltage drop. It is composed of various contributions from the electrolyte, active materials, cell connectors, and current collectors [18, 5].
3. Resistor-Capacitor branch: The dynamic behaviors of the battery is modelled using two resistor-capacitor (RC) pairs:
  - (a) The first RC pair ( $R_1$  and  $C_1$ ) models charge transfer (activation) polarization, which represents the voltage involved in increasing the rate of the chemical reactions which then causes an increase in voltage needed to surpass the chemical activation barrier [5, 19, 20].
  - (b) The second RC pair ( $R_2$  and  $C_2$ ) models concentration or diffusion polarization, which accounts for slower dynamic changes in the battery's voltage due to ion diffusion. The pair represents the voltage involved in the concentration gradients of the charge carriers in the electrolyte. This slow phenomenon is thus fitted by the larger time constant RC block [5, 19, 20]:.

These RC pairs are what define a dual-polarization model: based on two battery polarization phenomena which are modeled in the two pairs.

These components create a nonlinear relationship between the terminal voltage, current, and SOC described in the following mathematical representations.

### Mathematical Representation of the Dual-Polarization ECM

All of the parameters mentioned above are correlated with one another in a main mathematical expression as follows:

$$V_{batt} = OCV - I_0 \cdot R_0 - V_{RC1} - V_{RC2} \quad (3.1)$$

with  $V_{RC1}$  and  $V_{RC2}$  being the voltage drop across the charge-transfer and diffusion RC pair, respectively. The equations to describe voltage drop across RC pairs are expressed below.

$$V_{RC1} = I_{batt} \cdot R_1 - I_{batt} \cdot R_1 \cdot \left( e^{-\frac{t}{C_1 \cdot R_1}} \right) \quad (3.2)$$

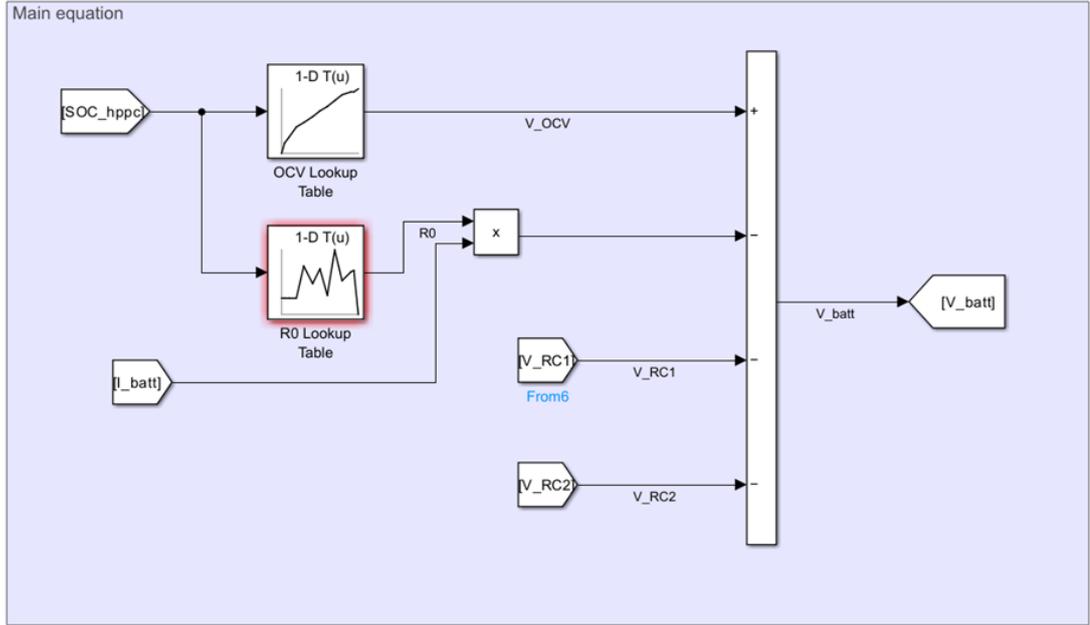
$$V_{RC2} = I_{batt} \cdot R_2 - I_{batt} \cdot R_2 \cdot \left( e^{-\frac{t}{C_2 \cdot R_2}} \right) \quad (3.3)$$

It should be noted that operating conditions have a significant impact on all of the model parameters presented. The main factors influencing OCV,  $R_0$ , and the RC blocks are SOC and temperature. In addition, RC blocks are also sensitive to load current (C-rate) and its direction [18]. For the balance of model accuracy and complexity, the C-rate is neglected, leaving just SOC as the main predictor of the parameters' value. Temperature, on the other hand, is accounted for by extracting the parameters from all 6 temperature datasets. In this section, the 1D DP model will be exhaustively described which later will lay a foundation for the 2D DP model.

#### 3.1.1 Simulink model representation

The mathematical equation presented in Eq. 3.1 which governs the main DP model and Eq. 3.2 - Eq. 3.3, which express the dynamics of the RC pairs, are translated into graphical model using Simulink blocks as shown in Fig. 3.2 and Fig. 3.3, respectively.

The Simulink model's inputs for the simulation are the current array  $I_{batt}$  and the SOC array found in the battery testing dataset. Even though the SOC could be self-calculated by integrating the current over time, given the battery capacity and initial condition, the SOC data given has already taken into consideration the degrading capacity (state-of-health) from running the tests which gives more accuracy to the model and more convenience to the user. On the other hand, the simulation output is the simulated voltage. Our model aims to compare the simulated voltage and the measured voltage, hence defining a simulation error. Having the OCV,  $R_0$ ,  $R_1$ ,  $R_2$ ,  $C_1$ , and  $C_2$  as a function of SOC requires a 1-D Lookup table for each of the parameters. The 1D LUT takes the value of the actual SOC given by the dataset to interpolate the parameters' value at each time step. These six parameter values over SOC were initially extracted from the discharge test and HPPC test and then further optimized using a Simulink Parameter Estimation as explained in the following section.



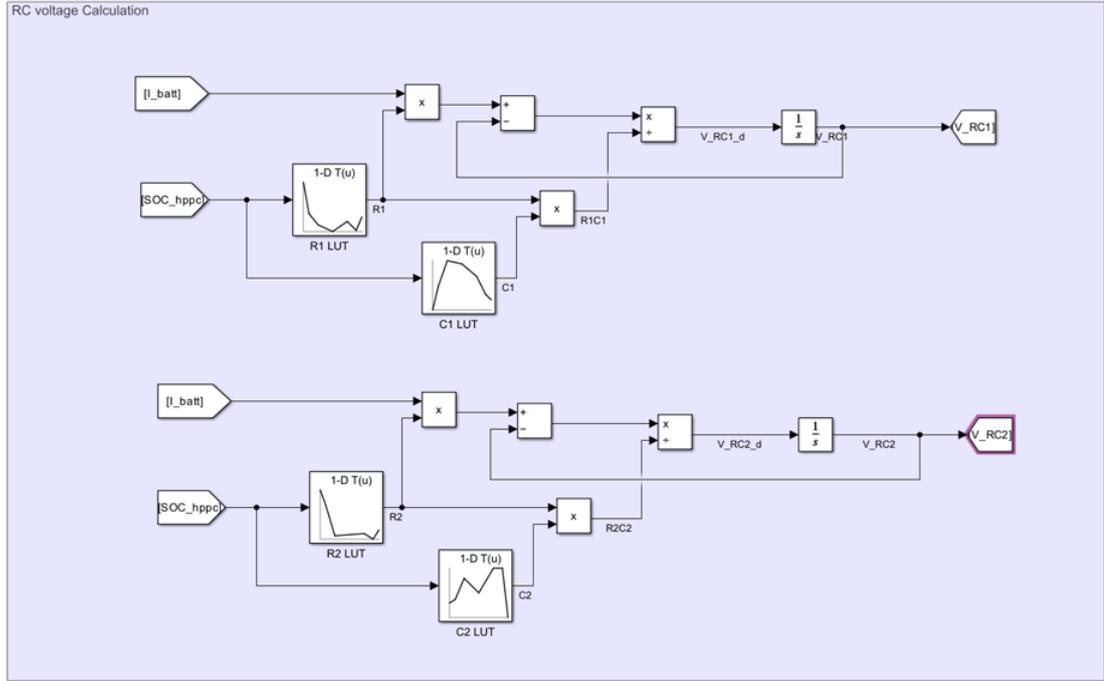
**Figure 3.2:** Equation governing dual-polarization equivalent circuit model expressed in Simulink environment

### 3.1.2 Parameters extraction for initial value estimation

Looking at the Simulink model, it is immediately understood that the accuracy of the model lies in how well the parameter arrays are constructed. As explained in Chapter 2, we will first estimate the initial value to prevent the optimization performed later using Simulink Parameter Estimation showing the local minima and unreasonable range, and to speed up the optimization time. In general, the OCV-SOC relationship is estimated from the Discharge test, while internal ohmic resistance  $R_0$  and the resistors in RC pairs ( $R_1$  and  $R_2$ ) are extracted from the HPPC test. We will initially explain the initial estimation and the following optimization steps only for 25 °C and the same procedure will be repeated for the rest of the dataset.

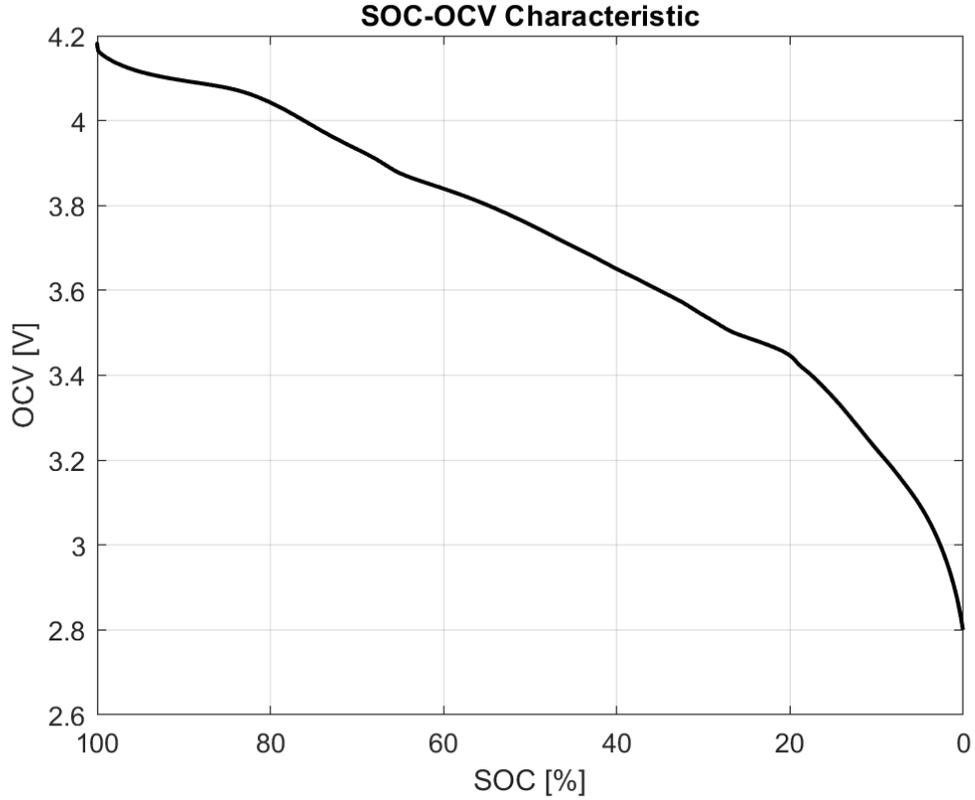
#### Discharge OCV-SOC characteristic

Open circuit voltage has a strict static relation to the actual capacitance and temperature [5]. Given the capacity test, discharge OCV dependence of SOC could be derived from it by looking at the voltage profile while the battery is emptied which is commonly referred to as the pseudo-OCV calculation method [21]. Initially, the State-of-Charge of the battery should be computed from the amount of capacity



**Figure 3.3:** Equation governing resistor-capacitor pairs expressed in Simulink environment

available throughout the testing time. For illustration purposes, the first test case (80 kg payload with HVAC ON) is evaluated. The nominal capacity  $Q_n$  for 25 °C is given in the data set as 4.5339 Ah. Inserting the known nominal capacity and the current at a given time step  $I_{batt}(t)$ ; and imposing the initial SOC as 100 %, the one-to-one relation between capacitance and SOC at any given time throughout the test is obtained. Plotting the voltage against the computed SOC will give us the voltage profile as a function of SOC. Considering the discharge rate is trivial (C/20), we could regard this characteristic as OCV's dependency profile on SOC. Figure 3.4 illustrates the OCV-SOC behaviour for the first test case under 25°C climatic temperature. Although the OCV-SOC relationship is somewhat static, as we will see later, slight variation still could occur. Accordingly, this profile only serves as an initial estimation before casting it into Simulink Parameter Estimation and tuning it based on the HPPC results. Following the same manner on every test case and test temperature of interest, we would obtain a bigger picture of the OCV dependence to SOC.



**Figure 3.4:** OCV dependence on SOC obtained from 25°C capacity test

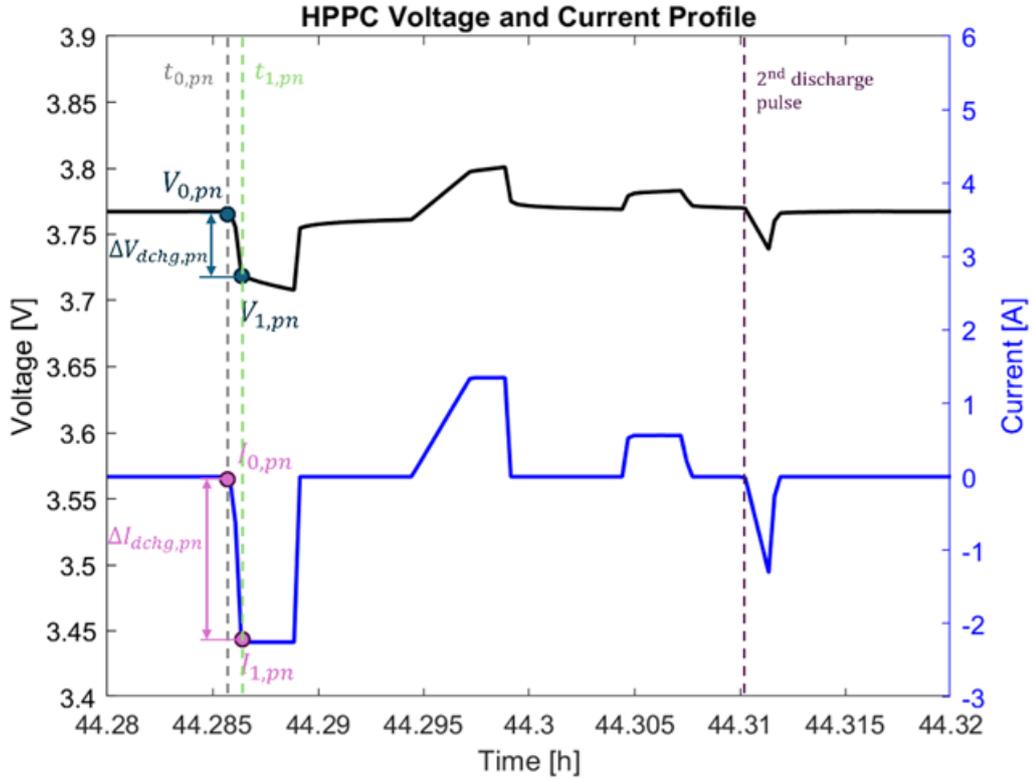
### Discharge internal ohmic resistance $R_{0,dchg}$

In this section, the internal ohmic resistance during the discharging phase is extracted from the HPPC test on every SOC level. Ha et al. [21] describes deliberately the procedure to obtain both the charge and discharge ohmic resistance from the HPPC test. The idea is to measure the delta of voltage on the leading edge when a determined current pulse is given. Figure shows the current pulses profile and the produced voltage response, equipped with the labels for  $V_{0,pn}$ ,  $V_{1,pn}$ ,  $I_{0,pn}$ , and  $I_{1,pn}$  annotation for the  $n^{th}$  pulse sequence.

The calculation for  $R_{0,dchg}$  are described by the following equation:

$$R_{0,dchg,1} = \left| \frac{V_{1,pn} - V_{0,pn}}{I_{1,pn} - I_{0,pn}} \right| \quad (3.4)$$

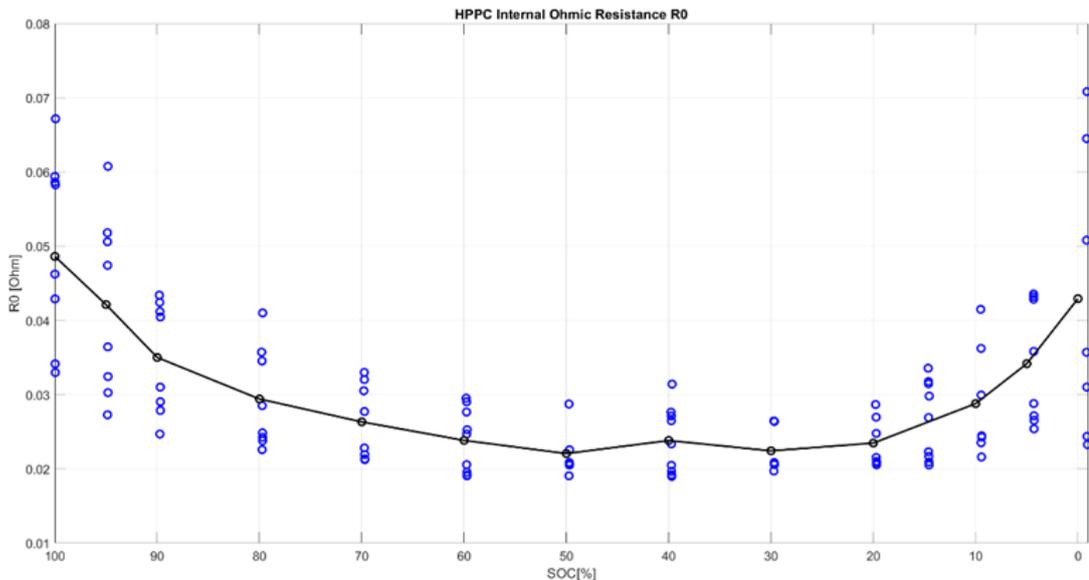
Due to the different nature of the HPPC profile, some modifications are needed from the original algorithm provided by Ha et al. The current profile of the HPPC test given by the dataset has multiple pulses for both discharge and charge for each



**Figure 3.5:** HPPC discharge and charge current pulse profiles and the voltage response at the  $n$ -th pulse

SOC. The code first locates at what index the negative slope (discharging pulse) occurs. Knowing that a negative pulse could indicate both the leading edge of a discharging pulse or the trailing edge of a charging pulse, a subsequent step to filter only the indices of discharging current pulses is needed, characterized by a negative slope delta between 2 timesteps ( $t_0$  and  $t_1$  and a zero value of the current timestep ( $t_0$ ). Looking at Figure 3.5, we could easily recognize 2 discharge pulses for each pulse sequence, giving us  $R_{0,dchg,1}$  and  $R_{0,dchg,2}$  with the same computation procedure as explained in the equation and annotation above. The code then averages the  $R_{0,dchg}$  for a defined list of SOC levels: 0%, 5%, 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, 90%, 95%, 100%. The scattered data points on each SOC and the corresponding average value are depicted in the following figure, which defines the  $R_{0,dchg}(\text{SOC})$  characteristic. Notice how some SOC points lie slightly lower or higher than the corresponding SOC levels, particularly those data points lower than 0% SOC. These values occur due to mainly discrepancies in the testing device measurement and numerical error, which then are rounded up and

interpolated accordingly. Averaging the scattered points on the closest SOC points and connecting the average points gives us Figure 3.6, which shows the profile of  $R_0$  as a function of SOC that is aligned to what is commonly found in the literature [5, 11, 21]. The resistance profiles are often found minimized around the mid-level SOC such as what we obtain from parameter estimation.



**Figure 3.6:** Ohmic Resistance  $R_0$  initial estimation as a function of SOC

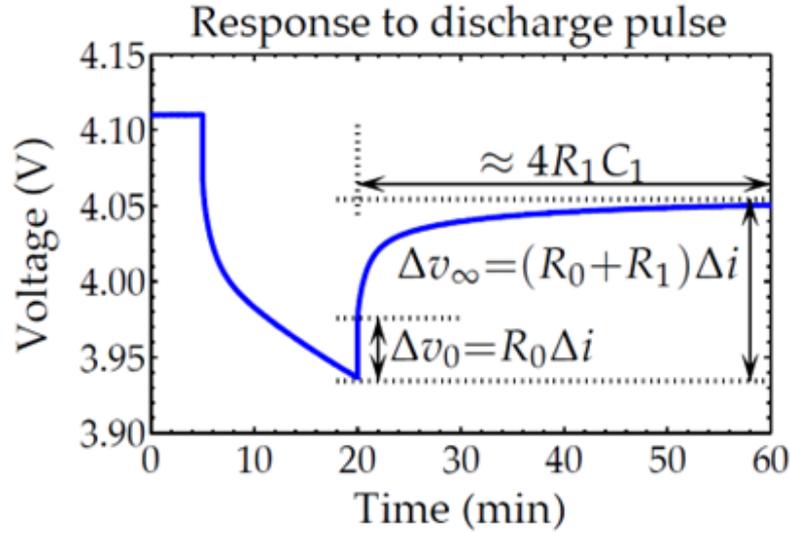
### 3.1.3 Initial calculation of the RC pairs

The approach of calculating the initial value guess for resistors and capacitors in the DP model is slightly different than the internal resistance calculation. While for the latter, we are primarily and cautiously extracting the parameters from the HPPC test through deliberate calculation; the initial values estimation of  $R_1$ ,  $R_2$ , and particularly  $C_1$ , and  $C_2$  as a function of SOC relies heavily on the Parameter Estimation process described on the following section. This approach is considered time-effective while being accurate enough to avoid local minima. Several methods of parameter extraction from other test procedures have been deliberately explained by Barai et al. [22]. It might come in handy when the test dataset (such as EIS) is available, which is not the case with ours. In the case of RC pairs, the SOC level data points are reduced to only: 0%, 10%, 25%, 50%, 75%, 90%, and 100%. The decision was made mainly to have a comparative parameter to Wassiliadis [11] and Cittanti [18], and also to prevent the fluctuations in the parameters' value found

during the several iterative studies. Later at the end of the parameter estimation, a comparative study of early findings of each capacitor and resistor profile as a function of SOC is conducted as a measure of sanity check.

### Resistor values extraction

The resistors ( $R_1$  and  $R_2$ ) are extracted in a similar manner as discharge internal ohmic resistance ( $R_{0,dchg}$ ). The main difference lies in the position of extraction of current pulse and voltage response. Knowing that these resistances oversee the model's dynamic behaviors along with the capacitors, both resistor values are observable during the transient responses. Plett [5] explained the parameter extraction of a single parallel resistor-capacitor subcircuit from the HPPC test.



**Figure 3.7:** Plett R-C pair parameter extraction from HPPC test [5]

Referring to Figure 3.7, as the discharge current pulse is terminated at 20 minutes time mark, the immediate change in voltage must equal the product of the instantaneous change in current and the series resistance  $R_{0,dchg}$ . It occurs since the voltage across the capacitor cannot change instantaneously, and the state of charge remains unchanged when the current is zero. The voltage response relaxation continues to its steady state value indicated at the 60-minute time mark. Knowing the value of  $\Delta I$  (equal definition as in  $R_{0,dchg}$  calculation) and  $R_{0,dchg}$ , we could compute the value of  $R_1$  as follows:

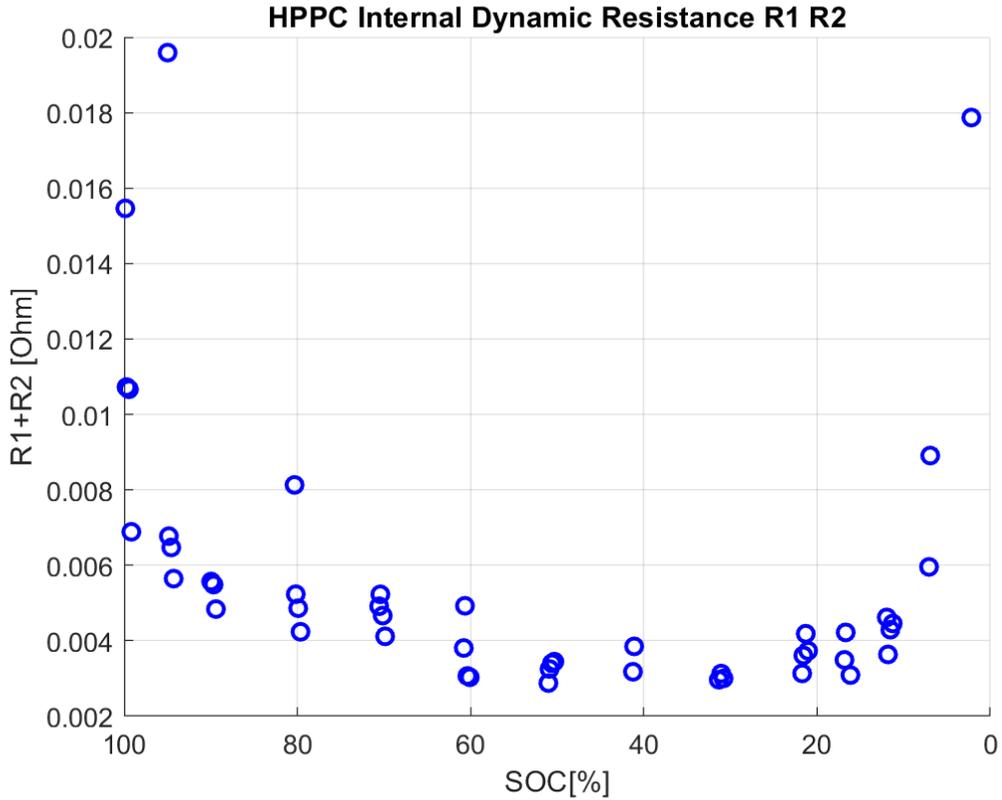
$$R_1 = \left| \frac{\Delta V_\infty}{\Delta I} \right| - R_0 \quad (3.5)$$

Plett mentioned how this method is suitable only as a rough estimation and applies only to single RC value pairs. Suppose the model uses multiple parallel RC pairs (as found in our model), this simple approach will not work, and a Simulink Design Optimization-based estimation should be done instead. Nevertheless, we argue that the approach could still be useful as a first estimate of the Simulink Design Optimization, to avoid floating and local minima convergence.

We theorized that in the dual-polarization model, the effect of  $R_1$  and  $R_2$  combined are much greater than the  $R_0$ , hence giving us the permission to omit it from the calculation and thus simplify a lot the model. The equation becomes:

$$R_1 + R_2 = \left| \frac{\Delta V_\infty}{\Delta I} \right| \quad (3.6)$$

Following the same procedure as described in the  $R_{0,dchg}$  calculation, we would get the  $R_1 + R_2$  profile as pictured in Figure 3.8.



**Figure 3.8:** Initialized datapoints of  $R_1 + R_2$

For the sake of simplicity and considering the identical order of magnitude for

$R_1$  and  $R_2$ , the proportion of each component is considered to be equal, meaning the value of  $R_1 + R_2$  is calculated by halving the  $R_1 + R_2$  as our first estimates.

### Capacitor value initialization

Differently compared to the resistors and OCV, the capacitor values are obtained through literature reviews. Plett [5] mentioned in his work on the methodology to calculate the capacitance based on the known resistance  $R_1$  and  $R_2$  by determining the resistor-capacitor time constants, referring to the subspace system identification by van Overschee and De Moor [23]. Nonetheless, the writer feels that the work in calculating the capacitances is unjustified considering how little time is saved during parameter estimation, the accuracy and uncertainty of the calculation, and the effort needed to develop the code and algorithm to extract the value. As mentioned, considering that initialization serves only as the first estimate for Simulink Parameter Estimation (SPE), it is sufficient to pass the right order of magnitude as an estimate and let SPE optimize the value based on the voltage difference between measurements and ECM result. From the references found [11, 18], we understand that normally  $C_2$  has 10-100 times the order of  $C_1$ . The value arbitrarily chosen was as follows:

Capacitor	Value [F]
$C_1$	1000
$C_2$	10000

**Table 3.1:** First estimate value of capacitors

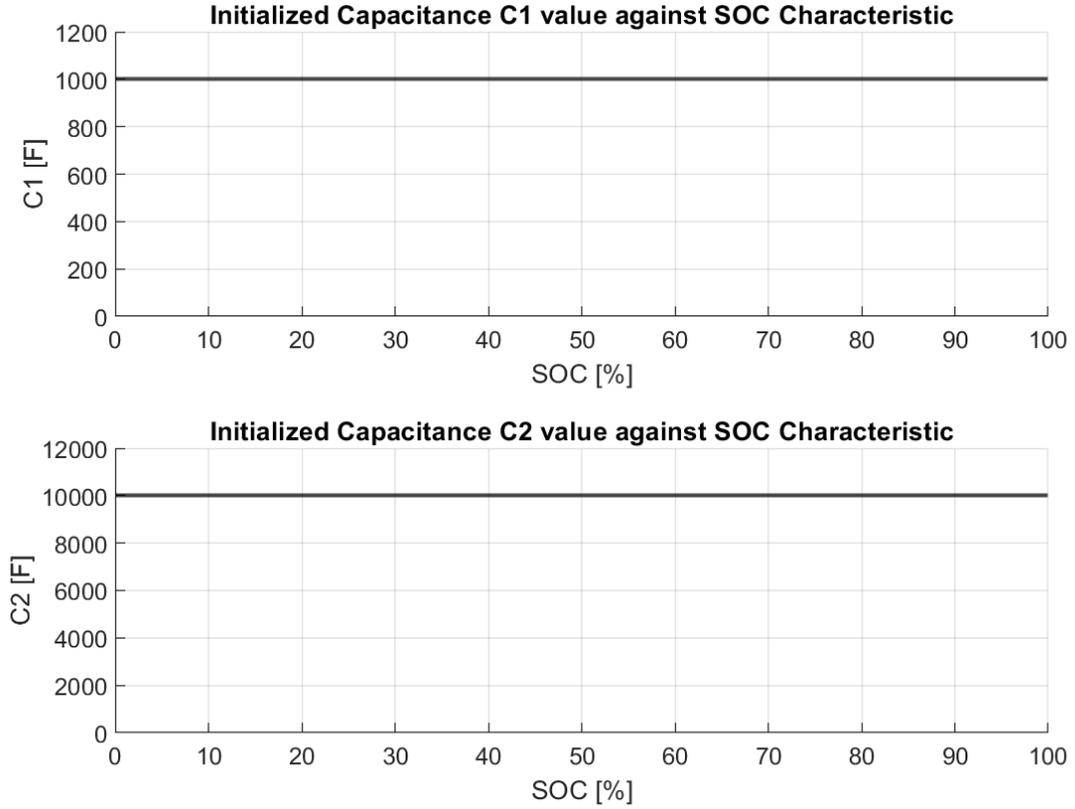
The selected values are constant across the SOC as shown in Figure 3.9.

### 3.1.4 Parameter Estimation and Optimization

Once the initial parameter estimates from the HPPC tests are extracted, they form the basis for creating an Equivalent Circuit Model (ECM). How well these parameters and the OCV-SOC curve reflect the actual battery behavior will determine how accurate this model is. Consequently, further optimization of these values is needed to guarantee accurate voltage predictions under various scenarios.

#### Optimization problem formulation

There are numerous approaches to choose from to further refine the ECM parameters. The Simulink Parameter Estimation toolbox is a widely used method. In essence, the refining procedure is an iterative cycle of simulating the output voltage and optimizing the defined parameters to achieve the least minimum error between



**Figure 3.9:** Constant initialized value of  $C_1$  and  $C_2$  over SOC for 25°C dataset

the measured quantity and the simulated output of the model. Initial estimates extracted from HPPC explained in the previous section serve as a good foundation for the first guess of this iteration, as a means to prevent local minima and enhance computational time. The simulation is then compared to the known measurement data and the cost function of a certain optimization algorithm is calculated. Simulink Parameter Estimation tries to modify the parameters on each specified SOC level within the range of set boundaries. The objective is to find the parameters that fit the measurement data, meaning with the minimized cost function, following the selected optimization algorithm.

Prior to the optimization process, an objective function and a minimizing algorithm must be selected. The objective function is the error between the measured and simulated output voltages. The error is commonly expressed as the sum of squared errors (SSE):

$$SSE = \sum_{i=1}^n (V_{meas}(i) - V_{sim}(i))^2 \quad (3.7)$$

where  $V_{meas}(i)$  is the measured voltage at time step  $i$ ,  $V_{sim}(i)$  is the voltage predicted by the ECM with the current set of parameters at that time step, and  $N$  is the total number of points in a test data. The goal of optimization is to minimize this SSE by adjusting the parameters of ECM.

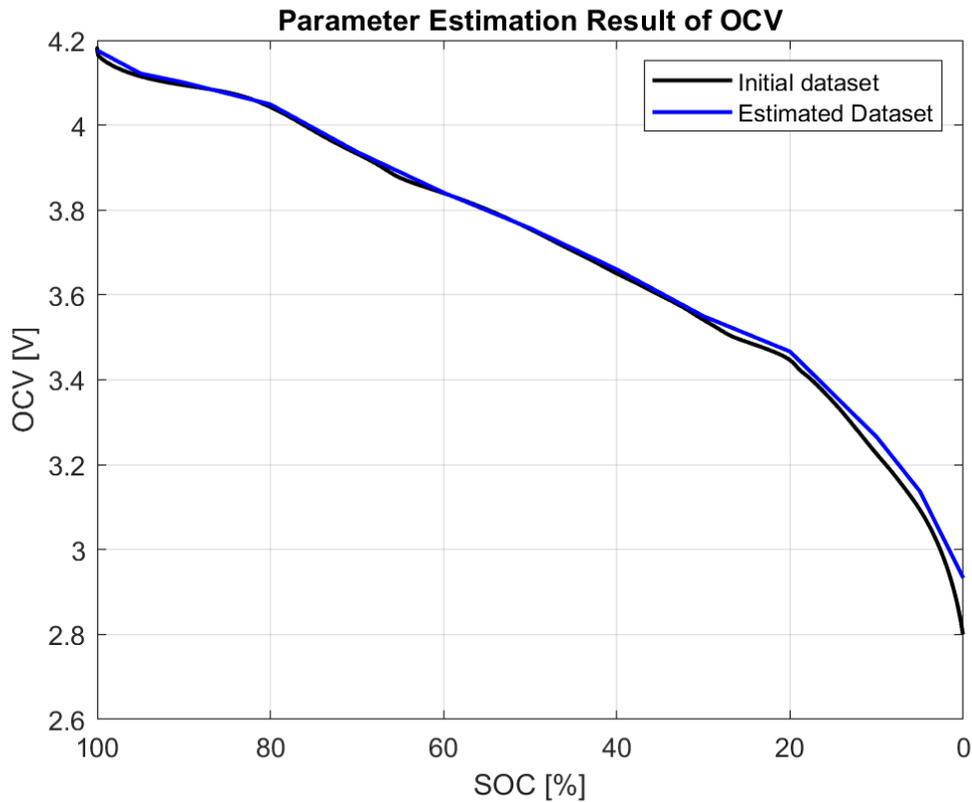
Knowing that our ECM model is nonlinear, the optimization algorithm must be able to accommodate correspondingly. Simulink's parameter estimation toolbox has several non-linear optimization algorithms such as Gradient Descent Algorithm, Non-linear Least Squares, Genetic algorithm, or Nelder-Mead (Simplex) to carry out this process. In this case, the non-linear least square method is preferable mainly due to its capability to handle non-linear models, combined with its simplicity which subsequently resulted in a faster convergence. Nevertheless, the algorithm's performance depends heavily on the quality of the initial parameter estimates, hence the reason why parameter extraction explained in the previous section shall be done prior.

### Parameter boundaries set-up

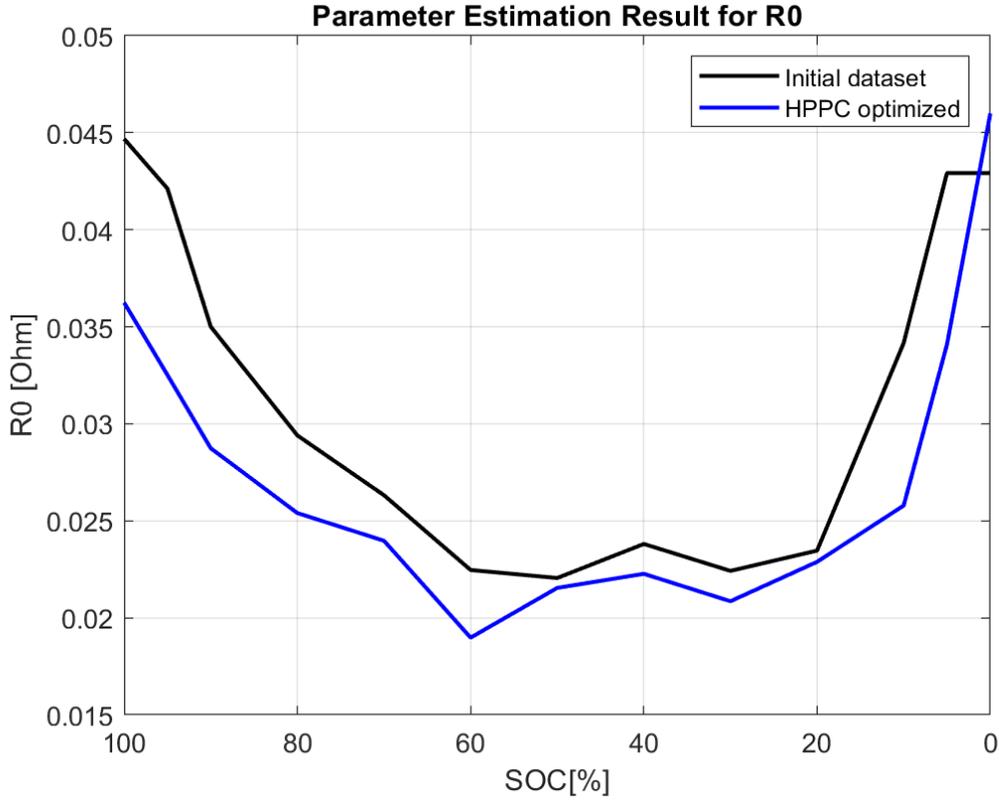
In the Simulink Parameter Estimation environment, it is important to set the boundary of each selected parameter to be estimated, which in this case happens to be  $OCV$ ,  $R_0$ ,  $R_1$ ,  $C_1$ ,  $R_2$ , and  $C_2$ . By default, Simulink sets the boundary as  $[-\text{inf}, +\text{inf}]$ , which would take a considerably longer time to converge and with even more coarse parameter modification in each iteration. Consequently, the solver might skip the minimum point of our optimization problem. Imposing the boundary, hence, is an important step. The upper and lower values of the boundary are selected as the reasonably lowest and highest values for that specific parameter. In some cases, imposing this boundary is a trivial problem, like what is encountered in the  $OCV$ . The datasheet of the battery from the manufacturer specifies meticulously the upper and lower values allowable for the battery, which are 2.8 and 4.2 V, respectively. In other cases like  $R_0$ ,  $R_1$ ,  $C_1$ ,  $R_2$ , and  $C_2$ , the upper and lower boundaries could be found through comparative literature study, by selecting the exhaustive upper and lower boundaries for the particular battery type, and through trial and error during the Simulink parameter estimation process. The latter point indicates that the parameter estimation is not done just once, since an iterative procedure is needed to increase or lower the boundary depending on whether or not there might be a minimal solution above or below the previously imposed boundary value. Generally, such occurrence could be detected by looking if in one of the SOC datapoints, the value is limited by or closely reaches the boundary. In this case, an increase or reduction of boundary is permitted as long as the value is within the reasonable range. As an example, let's say that in the first iteration, the imposed upper boundary value for the capacitor  $C_1$  is 1500, and later on, we found that the result of the Simulink parameter estimation for SOC

level 50 and 60% both reach 1500. In this case, an increase in the boundary for 50 and 60% might be needed. The idea is that the value 1500 is the local minimum in that iteration and by opening up a higher range of  $C_2$ , it could get us closer to our global minimal point of an acceptable range. The iteration continues until the  $C_1$  value of 50 and 60% stays the same even if the upper boundary is increased OR if the upper range becomes unreasonable, like reaching the commonly found value of  $C_2$ , which is rarely seen based on the literature. This step is important to have proper and representative parameters for our model since each battery type has its boundaries and characteristics, especially when the parameter is not initially extracted, like the case with the capacitors.

### Simulink Parameter Estimation result for 25°C dataset



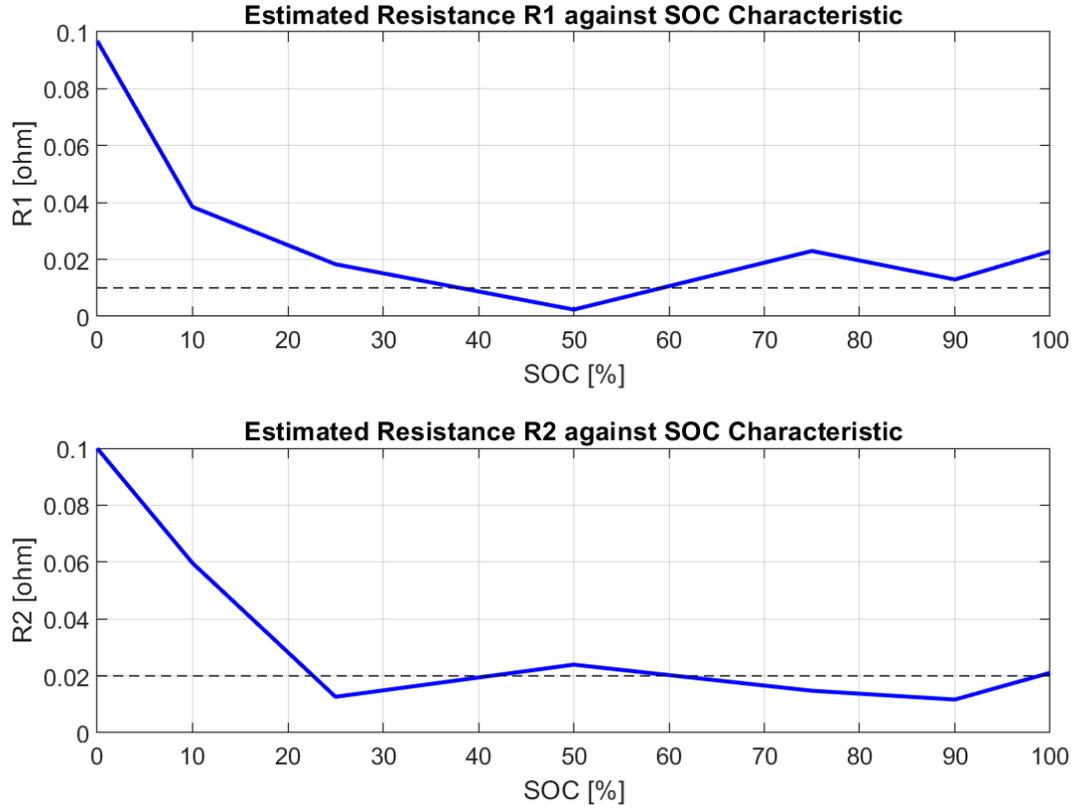
**Figure 3.10:** OCV-SOC profile comparison between the initial extracted data from the capacitance test (Initial dataset) and the parameter estimation result (Estimated dataset) for 25°C



**Figure 3.11:**  $R_0$  profile comparison between the initial extracted data from the capacitance test (Initial dataset) and the parameter estimation result (HPPC optimized) for 25°C dataset

Comparison between the initial extraction parameters and the parameter estimation result is presented in Figure 3.10 to 3.13. Regarding OCV parameters, we could see only slight differences are present between the initial dataset and the result of SPE (estimated dataset), meaning we could conclude that the initial extraction (labeled as the Initial dataset in Figure 3.10) from the capacitance test is quite accurate to predict the OCV. Notice how the trendline for not only 25°C, but also other temperature sets, is to have the estimated dataset slightly higher at the lower SOC region compared to the initially extracted data. Part of this phenomenon is caused by the considerably higher current on the HPPC rather than the capacity test (with C/40-rate), which caused the component  $I_{batt} \cdot R_0$  to augment as well. The other part is due to the higher phenomena of polarization inside the battery for the HPPC test (which affects Simulink Parameter Estimation result) at low SOC than the capacitance test which is done at a such low C-rate.

Parameter Estimation modified slightly the  $R_0$  profile by initial extraction for



**Figure 3.12:**  $R_1$  and  $R_2$  profile from the parameter estimation result for 25°C dataset

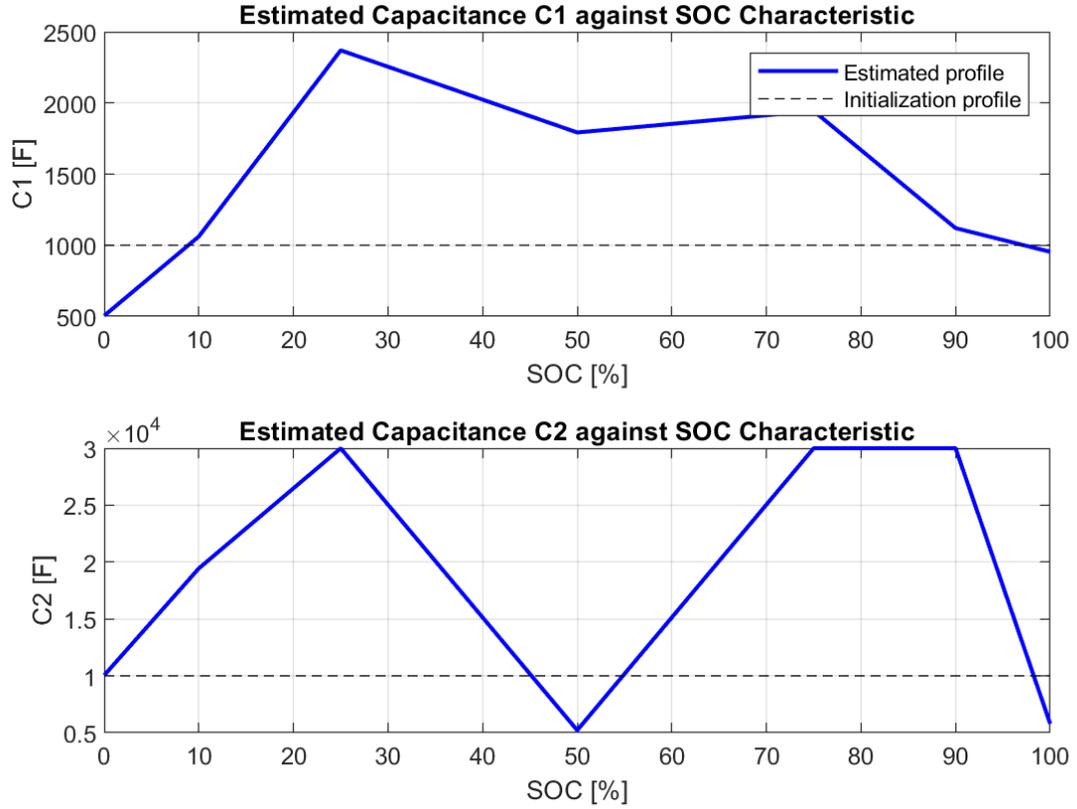
25°C. The minimum  $R_0$  value is consistently found around the mid-SOC, which is a fair profile to have, and the trendline remains consistent from the original dataset.

The resistances for the RC pairs show a commonly found profile, with high resistance at lower SOC and plateauing to a certain value as shown in Figure 3.12. This result is comparable to Cittanti's and Wassilidis' publications. In Figure 3.13,  $C_1$  and  $C_2$  profile are shown.

The same procedure of parameter extraction, estimation, and optimization is conducted on each of the dataset temperatures which will give us parameters' profiles for our DP ECM from -20 to 40°C. These profiles are then combined to form a grid which will be discussed next.

### 3.1.5 Model Validation on HPPC Test and Driving Cycles

Loading up the model with the parameters' values from Simulink Parameter Estimation, the next step is to validate our result by running it on different tests.



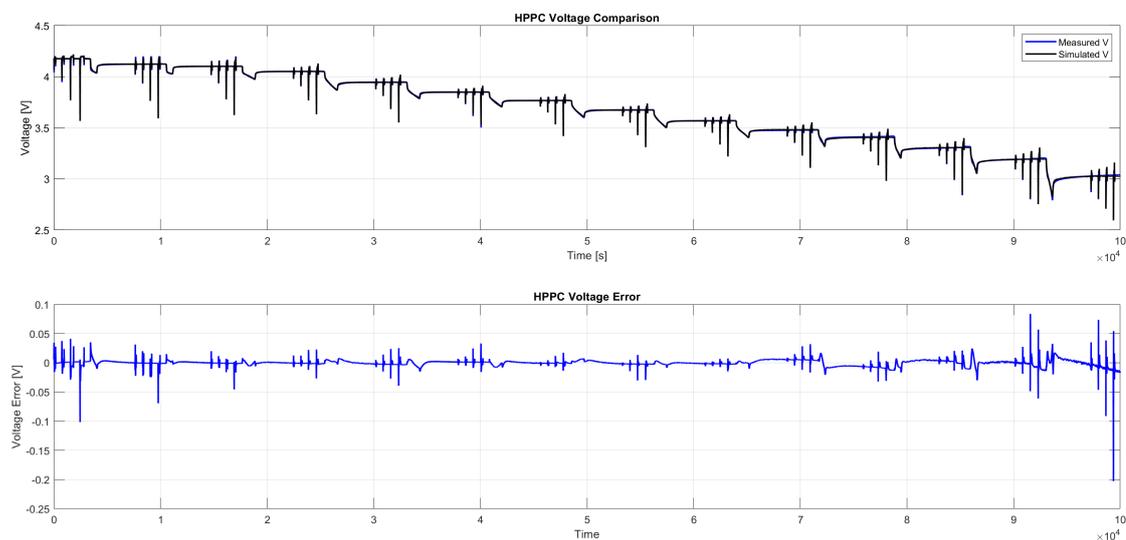
**Figure 3.13:**  $C_1$  and  $C_2$  profile from the parameter estimation result for 25°C dataset

Initially, the model will be tested on the HPPC test. We would expect an excellent result on this, considering that it is based on this HPPC test that we retrieved all the parameters. This step is important as a base comparison between temperature datasets. Given that Parameter Estimation will always be done based on the HPPC test, we would be able to judge whether the ECM model could equally predict with the same accuracy on every range of temperature.

As a validation step, the ECM model will be tested on several driving cycles. There are numerous driving cycles inside the open dataset given, and even more on the blind dataset, which could be categorized into two main classifications: the reordered drive cycles and the highway drive cycle. In order to have quick and comprehensive validation, we opted to test our model on one of each category, hence, the REORDERED 1 Driving cycle and the HWGRADE 1 Driving cycle are selected. This section presents the model's simulation result on the HPPC test, the REORDERED 1, and the HWGRADE 1.

## Model Testing on HPPC Test

Figure 3.14 presents the measured voltage quantity and the simulated voltage for 25°C dataset on the upper subplot, while the difference between the two, so-called the voltage error, is presented in the lower subplot. In this particular dataset temperature, the ECM could mimic the behaviour of the battery accurately, indicated by a low RMSE of 4.9 mV. Two major points to highlight from this dataset: first, there are some uncaptured dynamics indicated by the instantaneous spikes on the voltage error. Such behavior is expected considering the limit of the two RC pairs model. Second note, there is a tendency of degrading ECM accuracy on the lower as we will see also in the driving cycle validation. This propensity is also found in other papers, namely Cabello et. al [24]. and Ouyang [25] which characterize the limitation of the equivalent circuit model due to strong linearity around the low SOC region. The accuracy of the ECM drops when the SOC falls below 20% and decreases evidently under 10% [25] [26] [27].

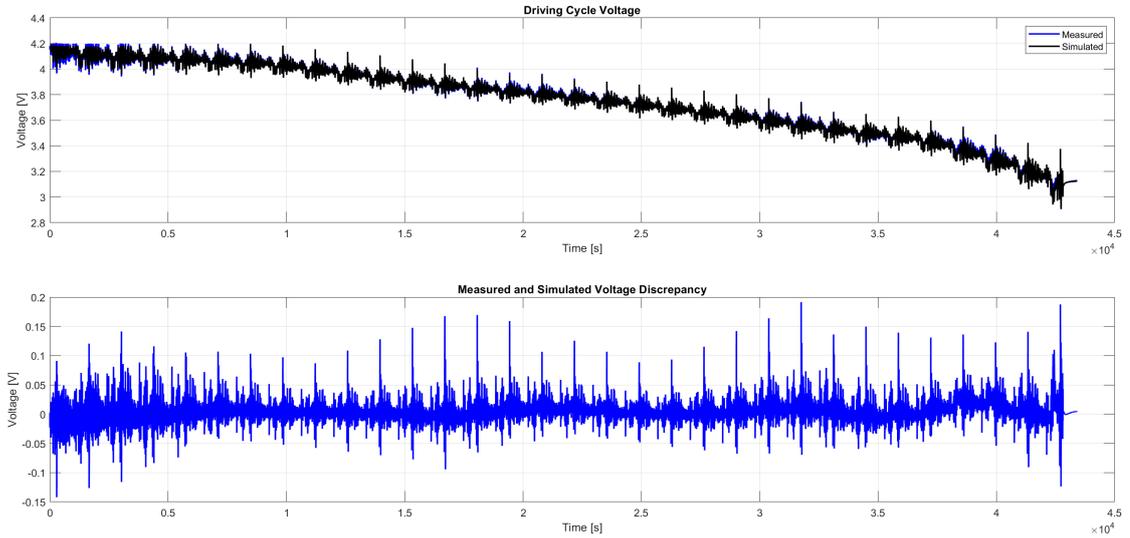


**Figure 3.14:** ECM test on 25°C HPPC Test

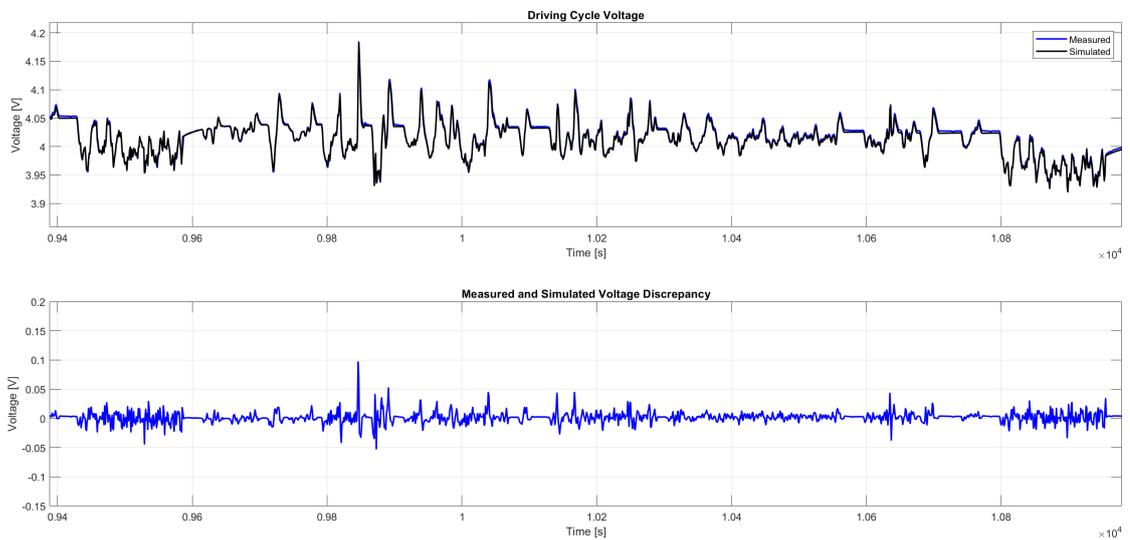
## Model Validation on REORDERED1 Driving Cycle

Immediate impression perceived looking at Figure 3.15 is how severe the dynamics of the driving cycle are compared to the HPPC test. With the dual-polarization model's two RC pairs, it can be seen that we couldn't capture all the dynamic details, as seen in the lower subplot and further enlarged in Figure 3.16 on the mid-SOC section, where some error spikes occur every time the input current fluctuates. Nevertheless, highlighting the limited error band of  $\pm 0.2$  V, we could

conclude that our model works well in following the measured voltage data with an RMSE of 14.37 mV.



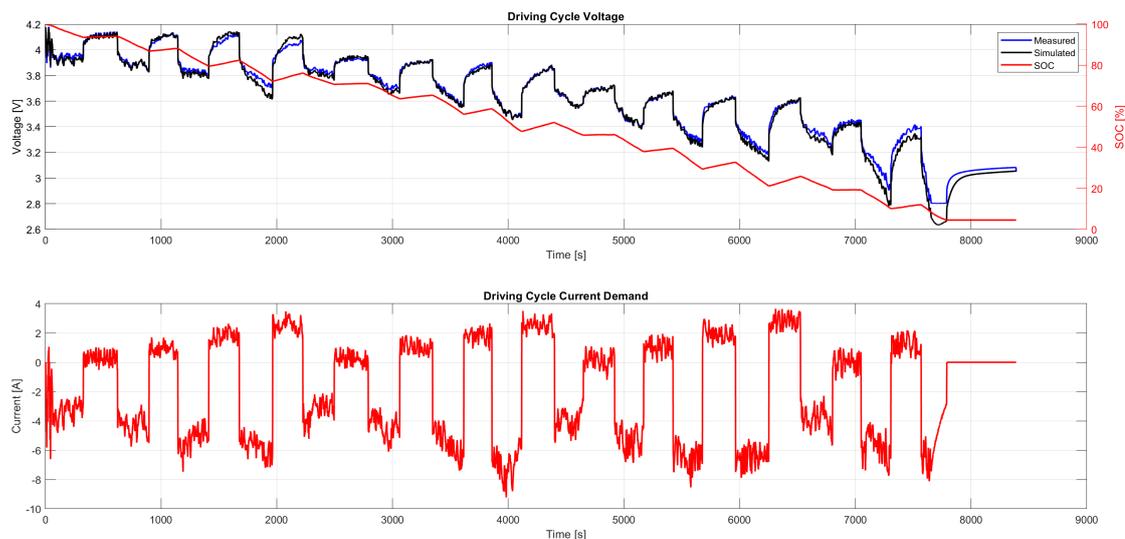
**Figure 3.15:** ECM validation on 25°C REORDERED1 Driving Cycle



**Figure 3.16:** Enlarged section of ECM validation on 25°C REORDERED1 Driving Cycle. Notice the fluctuations of error are caused by the uncaptured dynamics

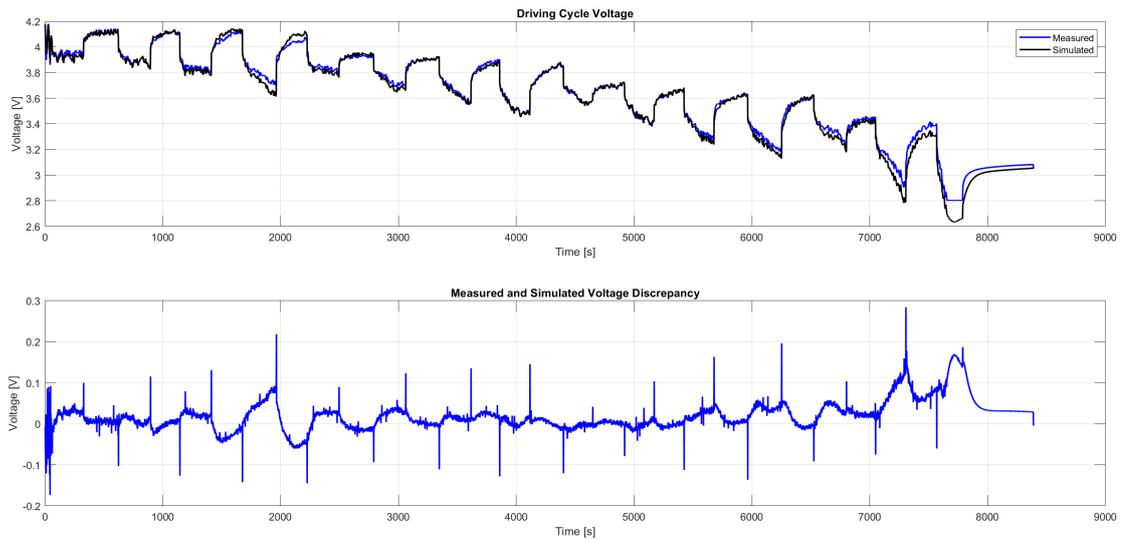
## Model Validation on HWGRADE1 Driving Cycle

As an initial analysis phase, Figure 3.17 is presented, where the measured and simulated voltage are plotted along the SOC and current demand from the HWGRADE1 driving cycle. Some notes regarding this driving cycle: the battery is, in some instances, charged as shown by the ramping up SOC and positive current demand. Notice also how the driving cycle is much shorter and more severe compared to the REORDERED1. The shorter driving cycle would cause the plot to appear with more error between the voltages compared to the REORDERED1, of which the subplot on Figure 3.18 is presented to give the magnitude of the discrepancy.

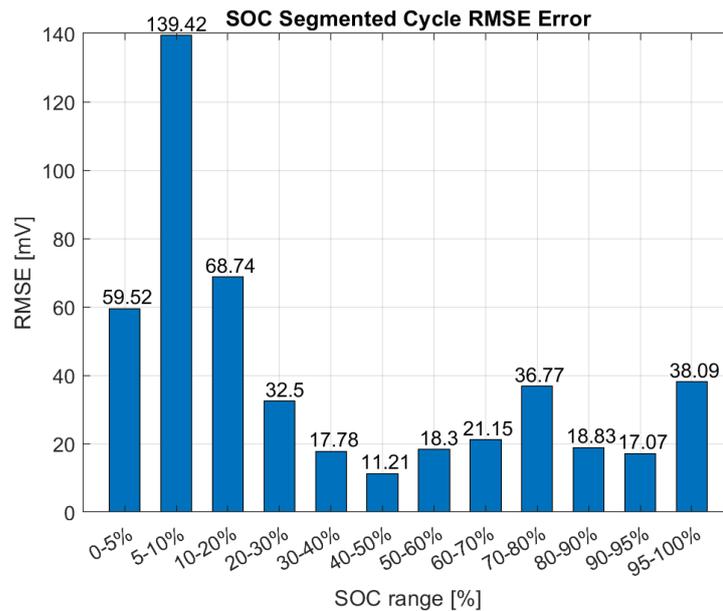


**Figure 3.17:** HWGRADE1 Driving Cycle (25°C data) ECM result

Analyzing the result in-depth, Figure 3.18 shows that there are some parts of the driving cycle where the simulated voltage has an offset to the reference measured voltage (e.g. around the 2000 s time mark) while the ECM could follow the dynamic profile of the reference voltage. Knowing that the parameters to run this driving cycle are retrieved from the HPPC, we could argue that there should be some disparity in between the two. Some possible causes are the hysteresis affecting the battery caused by the severe change in the current demand, the aging of the battery that caused the parameters to change, and the temperature discrepancy between the HPPC test and the Driving cycle test. The two formers are less likely to cause this phenomenon, as the hysteresis is considerably low around this temperature and is more dominant around the low temperature [5] and the aging of the battery is already accounted in the lowering of nominal capacity. The latter, however, has a strong reasoning as the real-time battery temperature exceeds 25°C and thus causes an inaccurate output. By comparing to Figure 3.20, we could



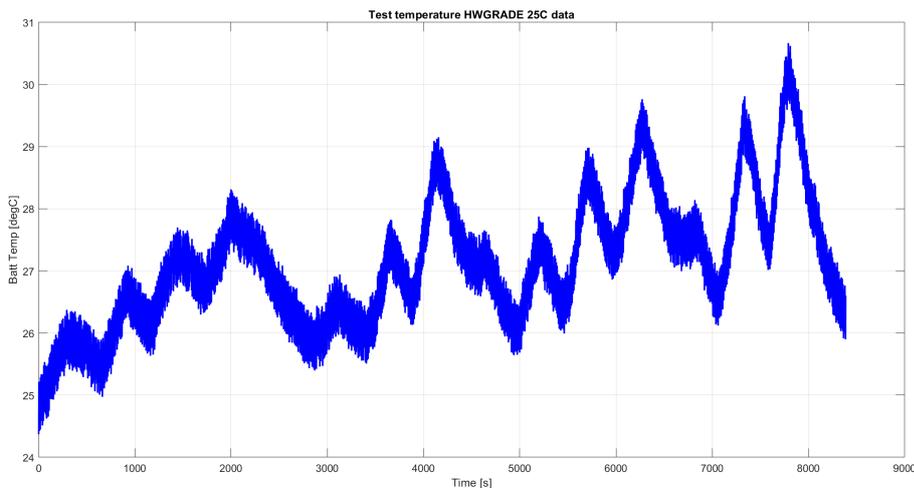
**Figure 3.18:** ECM validation on 25°C HWGRADE1 Driving Cycle



**Figure 3.19:** SOC-segmented RMSE result for 25°C HWGRADE1 dataset

highlight the 2000-second time mark and, especially, the 8000-second time mark where the temperature reaches over 30°C, the model struggles to fit the measured voltage. Such behaviour reiterates the need to take into account the temperature in our ECM model as explained later in the next section. Another interesting point is the voltage-bottoming phenomena found at the end of the simulation, where

the measured voltage is flat while the simulated dips down to follow the current demand. The discrepancy between the two could be explained by knowing that the lowest voltage specification for the corresponding battery is 2.8 V, of which the measured voltage bottoms down at that voltage. Meanwhile, we model the OCV 1D LUT to allow extrapolation, hence explaining the lowering of the simulated voltage.



**Figure 3.20:** Real-time battery temperature of HWGRADE 25°C

As previously discussed in the HPPC validation subsection, the result in HWGRADE1 reaffirms our suggestion that on low SOC, the ECM’s accuracy drops. Figure 3.19 shows the RMSE error calculated in several SOC brackets. As shown, the RMSE increased significantly under 20% SOC, although for 5-10% SOC, the spike in RMSE occurs due to the lower voltage bottoming as mentioned earlier. Regardless, this finding reaffirms our hypothesis on the limitation of ECM on low SOC.

Considering the severeness of the model, the limitation of single-state DP ECM, and the RMSE of 40.36 mV we got from the simulation, we argue that the model could satisfactorily fit the measured voltage and thus complete the validation phase for the 25 °C dataset.

## 3.2 Two-state dual polarization ECM

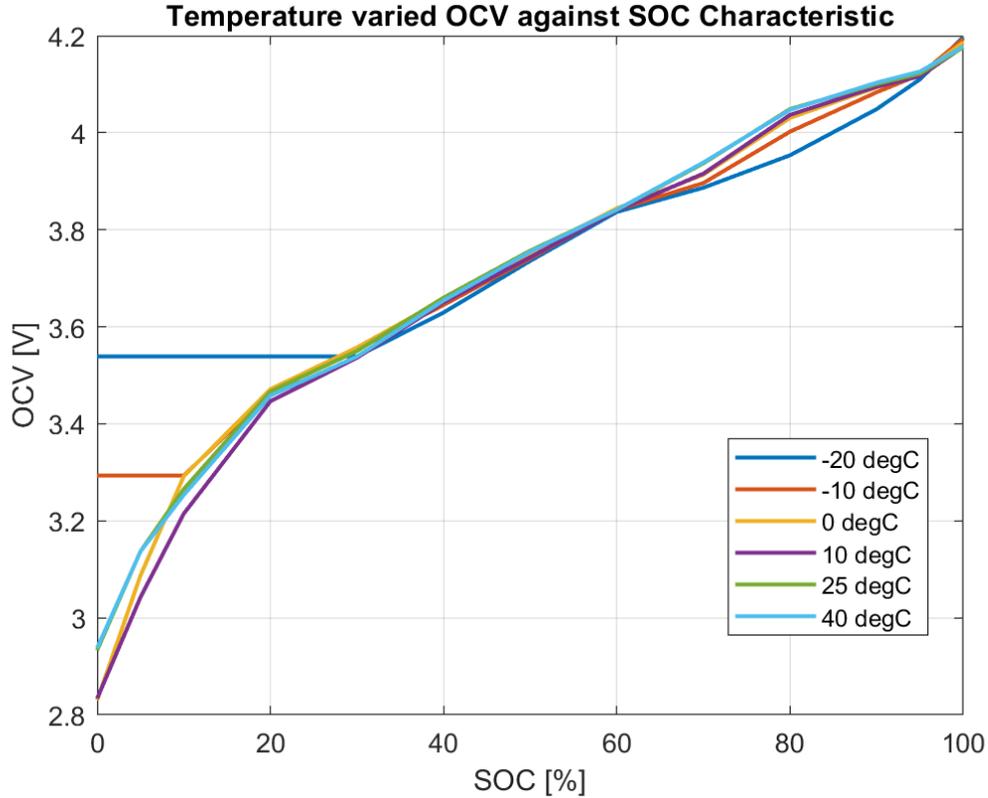
### 3.2.1 Model description

Having completed all the parameter estimations leaves us with six complete arrays of  $OCV$ ,  $R_0$ ,  $R_1$ ,  $C_1$ ,  $R_2$ , and  $C_2$  profiles for each dataset temperature. Subsequently, our next step is incorporating these temperature effects into our ECM model. In order to do that, some changes are needed in our Simulink model, mainly in the look-up table. Instead of using 1D LUT for each parameter as we did in the 1D model, the 2D LUT is used to take into account both SOC and temperature as the breakpoint of the search, with the parameter's value as the output as shown in Figure 3.25. Ideally, the temperature breakpoint list should be the intended test temperature, as stated in the naming of the dataset. This is not necessarily true as discovered later when analyzing the battery temperature. This modification brings another requirement: the 2D look-up table needs a gridded matrix instead of an array, hence we need to make parameter grids that combine all temperature profiles in a single matrix. The gridded parameters from all temperature datasets are presented in Figure 3.21 to 3.24.

Looking at Figure 3.21, we could notice some straight lines. These lines indicate the non-existing SOC value for that particular dataset. As we know, we retrieved these parameters from the HPPC test. On some temperatures, particularly the low ones, the tests don't fully deplete the SOC, hence, the characteristic of each parameter below that SOC couldn't be retrieved. In order to prevent misunderstandings, these values are assigned equal to the last value of which SOC is available to be retrieved. This decision to hold the value is based on the extrapolation method used in the 2D LUT, which is 'Clip' extrapolation, meaning it is only logical to also plot the extrapolated value as clipped. In the OCV-SOC plot, particularly around low SOC and 70-90% SOC level, an interesting characteristic could be identified. As we look closer, the lines separates with a trend: the lower the temperature, the lower the OCV. Farmann [28] mentioned similar behaviour observed on several batteries under 45°C and -20°C test temperatures with a pronounced discrepancy of OCV around 60% SOC, while in our case, it is around 80% SOC.

The internal resistor shows an expected profile (Fig. 3.22), where its value increases exponentially as the temperature drops as explained by Plett [5] and affirmed by various publications [29]. Another important characteristic of the resistors is that for each temperature, the resistor value is minimized around mid-SOC as we could also see in our result, thus affirming the sanity check of the parameter estimation result. These characteristics could also be found in the resistors inside the RC pairs as shown in Figure 3.23.

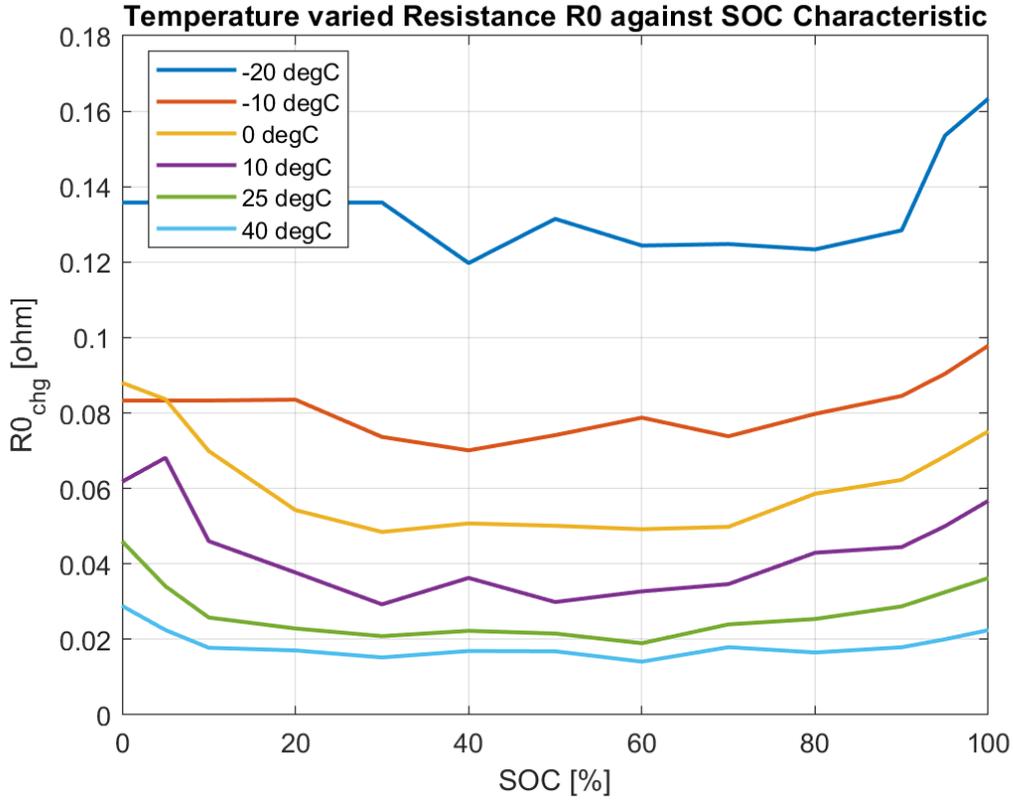
In Figure 3.24, the capacitors profiles are shown. The  $C_1$  gives us a clear image of the trendline, in accordance with the literature and common findings from other



**Figure 3.21:** OCV-SOC profile for various temperatures

publications [5, 29]. The resistor-capacitor time constants tend to get slower with increasing temperature [5, 11, 30], which might seem counterintuitive as we might expect the internal dynamics to increase at higher temperatures. Consequently, considering the resistors' characteristics at different temperatures, the capacitors must have an increasing value with increasing temperature as we see clearly in the upper subplot. On the other, similar behaviour could be spotted in  $C_2$  profile as well, although not as clear as in the  $C_1$  due to the fluctuations.

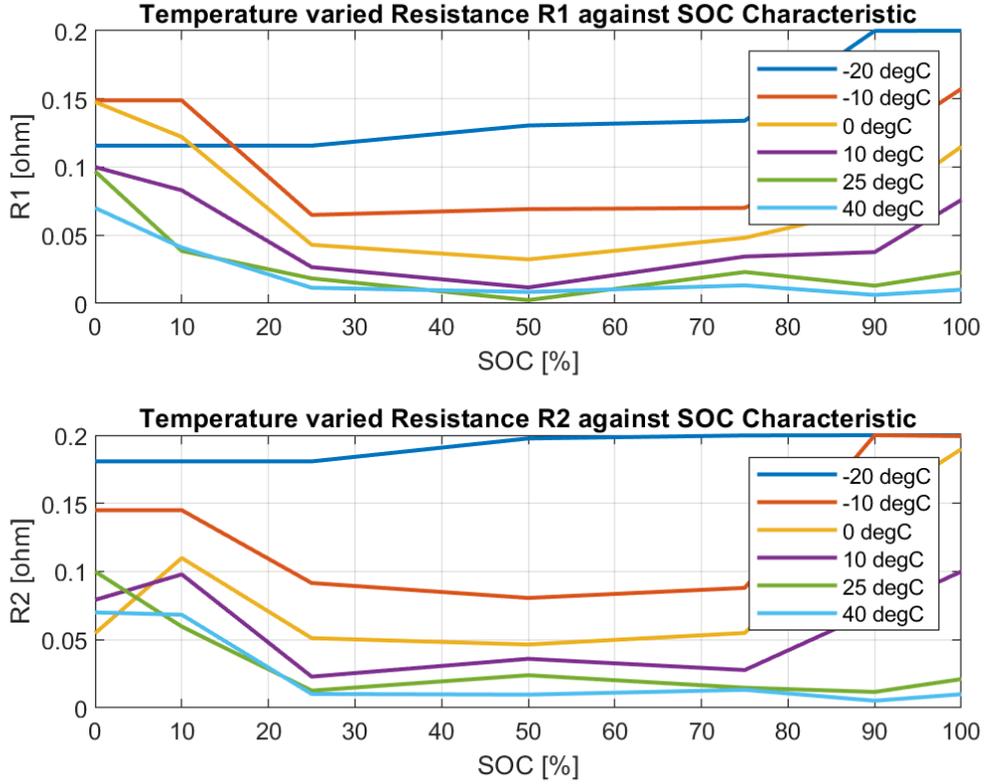
The second change revolves around the definition of the temperature as an input to the look-up table. Throughout our journey working on this project, we initially passed the temperature input for the look-up table as a fixed list of dataset temperature = [-20, -10, 0, 10, 25, 40] °C. We discovered later that along with the dataset, the real-time battery temperature in Celcius of the battery under test is also attached. By looking at these data, we understand how on several tests the temperatures fluctuate and even drift away from the designed temperature, although the tests were conducted inside a controlled climatic chamber. Figure 3.27 and 3.28 show the HPPC temperatures for -20°C and 25°C respectively. The



**Figure 3.22:**  $R_0$  profile as a function of SOC for various temperatures

lowest temperature profile exhibit a fluctuating (even with a huge spike up to 25°C for -20°C dataset) and drifting tendency. This is the worst examples of HPPC test temperatures which are most likely caused by the incapability of the climatic chamber to control the low temperature to remain constant and measurement error. On the other hand, the rest of the HPPC test has a battery temperature similar to what is shown on the 25°C test (Figure 3.28) with negligible fluctuation around its set temperature. The average battery temperature is tabulated in Table 3.2 to give the reader an insight into the temperature test profile. Combined with the information conveyed in Figure 3.28, we could appreciate that the temperature conditioning chamber struggles in the lower temperature, as the fluctuation increases. Another thing to pinpoint is how the average temperature is no longer following the dataset temperature. For example, the -20°C HPPC test has an average temperature of -17.08°C (2.92°C discrepancy between the designed temperature and the true average temperature).

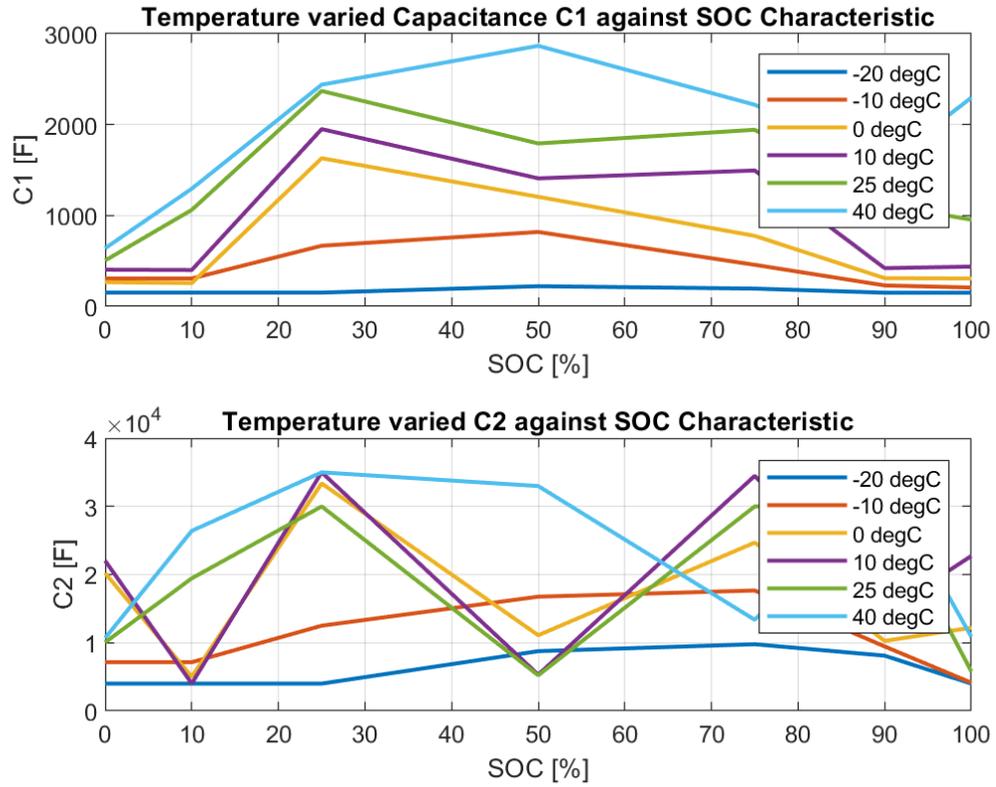
Even worse battery temperature fluctuations are found in the driving cycle, which is normal considering the severe nature of the test. Figure 3.29 to 3.30



**Figure 3.23:**  $R_1$  and  $R_2$  profile as a function of SOC for various temperatures

shows the fluctuation in REORDERED1 and HWGRADE1 driving cycles for  $-20^{\circ}\text{C}$  dataset. The average temperatures for REORDERED1, and HWGRADE1 are given in Table 3.3 and 3.4, respectively. Notice how the HWGRADE1 dataset for  $-20^{\circ}\text{C}$  has an average temperature closer to the next dataset, which is  $-10^{\circ}\text{C}$ . The trendlines are clear, first, the lower the temperature, the higher the discrepancy between the intended temperature and the real temperature of the battery. In addition, limited fluctuation occurs for higher temperatures compared to lower temperatures. Understanding these phenomena, it is not logical to use a fixed value of the dataset temperature list as the input. Consequently, the real-time battery temperature is used as the input, indicated as [temp] on the left side of Figure 3.25

Based on these preliminary analyses, two major decisions are made: incorporating this real-time battery temperature is a mandatory requirement to have an accurate ECM model, and using the true average temperature as the breakpoint for the look-up tables is a logical solution instead of utilizing the dataset (or intended) test temperature. Addressing the latter problem, since parameter estimation for each temperature dataset is based on the HPPC test, the average temperature list of

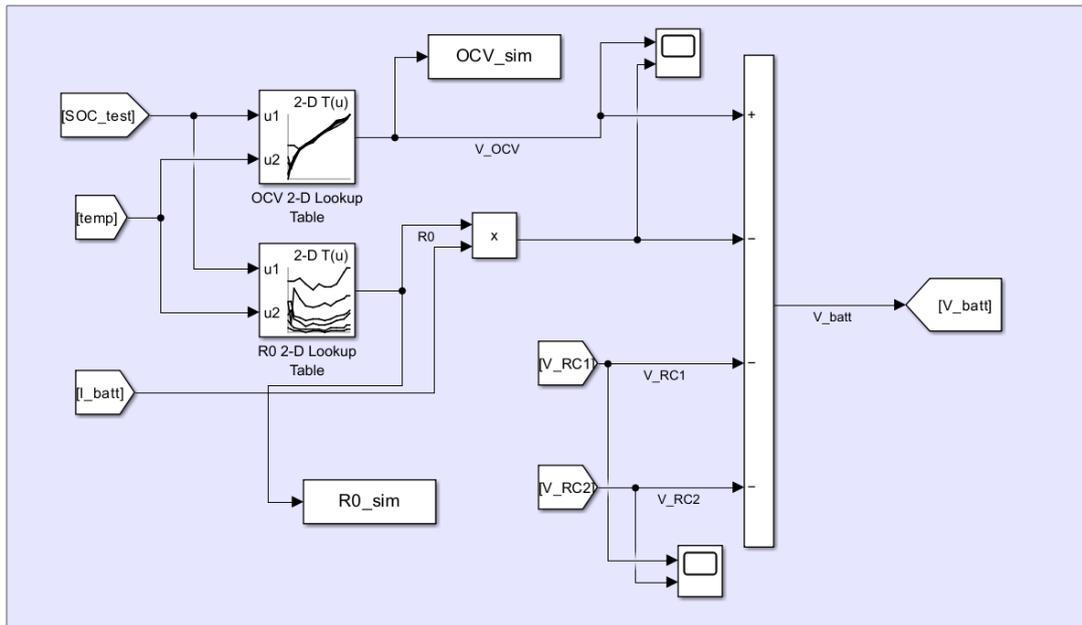


**Figure 3.24:**  $C_1$  and  $C_2$  profile as a function of SOC for various temperatures

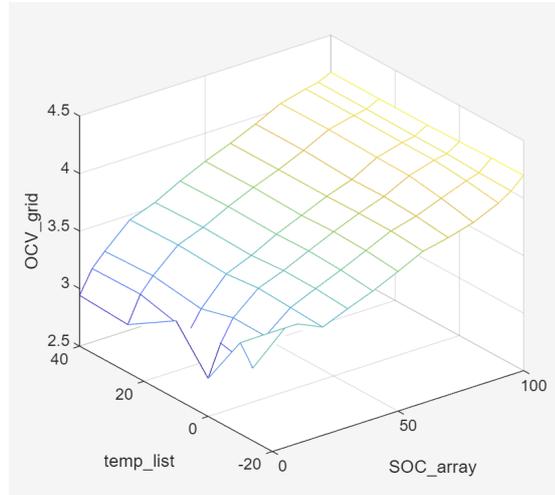
Test Temperature (°C)	HPPC Average Battery Temperature (°C)	Discrepancy (°C)
-20	-17.08	-2.92
-10	-7.49	-2.51
0	1.34	-1.34
10	11.06	-1.06
25	24.96	0.04
40	40.13	-0.13

**Table 3.2:** HPPC average battery temperature

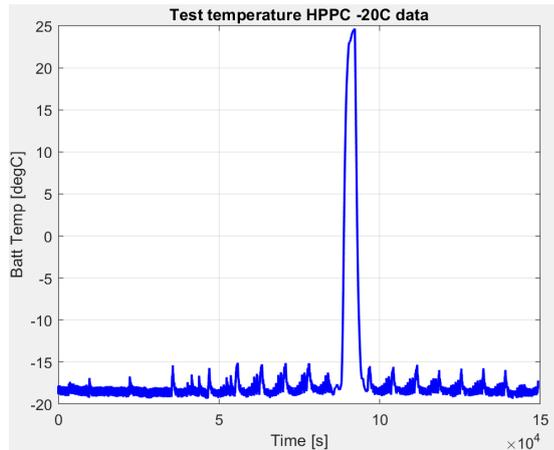
the HPPC dataset mentioned in Table 3.2 will be the breakpoint of our 2D look-up table.



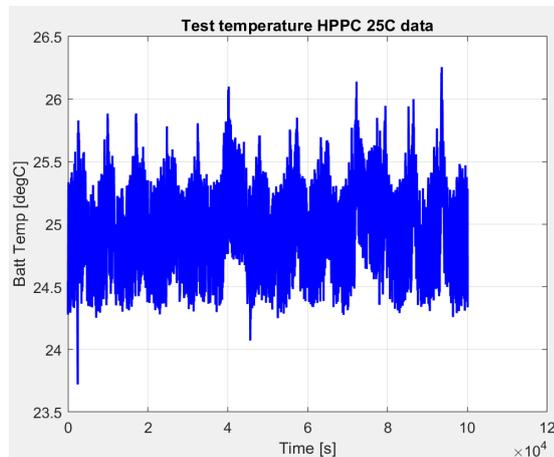
**Figure 3.25:** Main equation of 2D ECM model using 2D LUT for OCV and R0 that takes SOC and temperature as inputs



**Figure 3.26:** Example of OCV look-up table composed of OCV grid on various temperatures and SOC levels



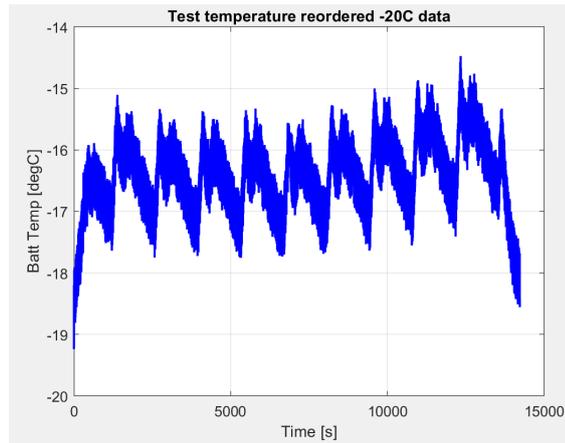
**Figure 3.27:** HPPC real-time battery temperature for  $-20^{\circ}\text{C}$



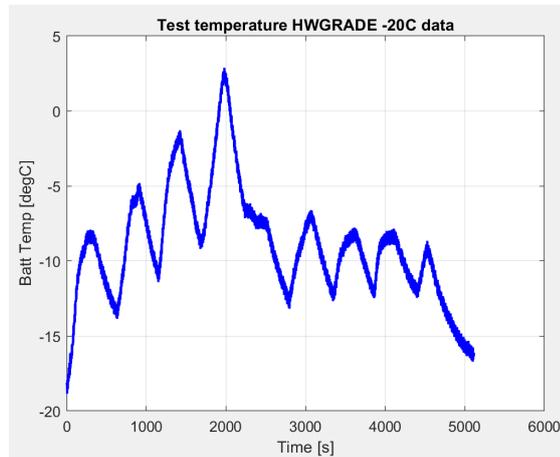
**Figure 3.28:** HPPC real-time battery temperature for  $25^{\circ}\text{C}$

### 3.2.2 Benchmark result: Fixed input temperature simulation

As earlier indicated, the more reasonable choice is to use the real-time battery temperature as the input to the 2D look-up tables. Nevertheless, it is also important to show the RMSE result of the model which input is an array of fixed temperatures that follows the temperature dataset naming, especially since we described only the  $25^{\circ}\text{C}$  result of the single-state DP model. In its entirety, it allows us to compare the effect of using real-time battery temperature on the model's accuracy and which test is most influenced by the temperature. Since the model is validated on the driving cycles, Table 3.5 presents the RMSE over REORDERED1 and HWGRADE1 driving cycles.



**Figure 3.29:** REORDERED1 real-time battery temperature for  $-20^{\circ}\text{C}$



**Figure 3.30:** HWGRADE1 real-time battery temperature for  $-20^{\circ}\text{C}$

At a glance, it is obvious that our two-state equivalent circuit model's output fits the reference value with higher accuracy on REORDERED1 compared to HWGRADE1 as the RMSEs are significantly less than on the HWGRADE1 driving cycle. The immediate reasoning for this could be found by simply looking at Table, with augmenting temperature discrepancy on the lower test temperature, the RMSE on HWGRADE1 also increases. Meaning there must be a clear correlation between this temperature discrepancy and the RMSE it produced. Focusing our attention on HWGRADE1 results, two worst examples ( $-20^{\circ}\text{C}$  and  $-10^{\circ}\text{C}$  dataset) are analyzed further.

Figure 3.31 plots the two-state ECM performance with fixed input on the  $-20^{\circ}\text{C}$  HWGRADE1 dataset, while Figure 3.32 for the  $-10^{\circ}\text{C}$  dataset. In both figures,

Test Temperature (°C)	Reordered1 Average Battery Temperature (°C)	Discrepancy (°C)
-20	-16.42	-3.58
-10	-7.49	-2.51
0	2.01	-2.01
10	11.12	-1.12
25	25.10	-0.10
40	39.79	0.21

**Table 3.3:** REORDERED1 average battery temperature

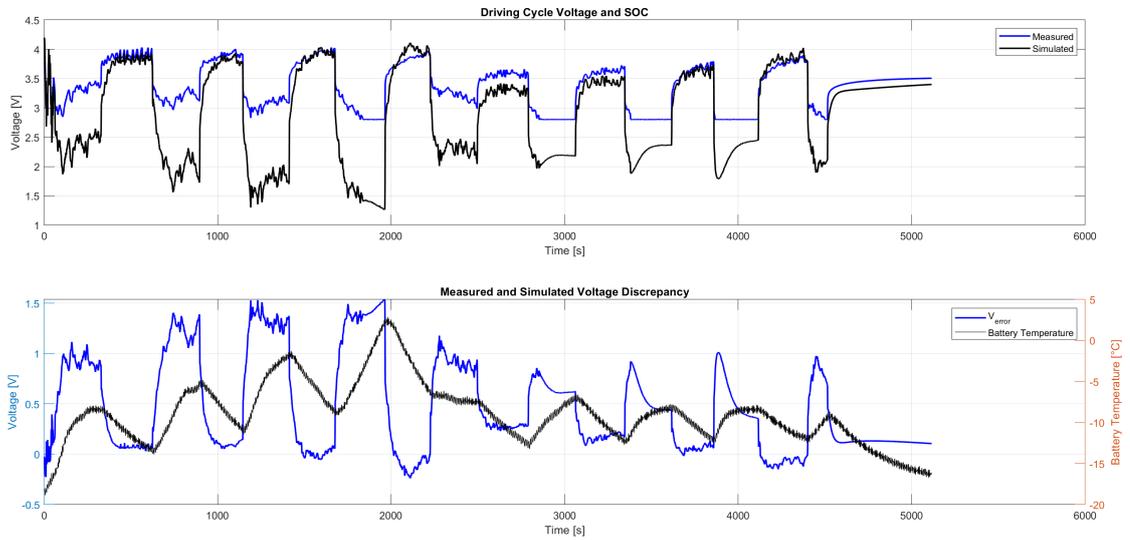
Test Temperature (°C)	HWGRADE1 Average Battery Temperature (°C)	Discrepancy (°C)
-20	-8.96	-11.04
-10	-0.57	-9.43
0	7.18	-7.18
10	14.90	-4.90
25	27.11	-2.11
40	41.80	-1.80

**Table 3.4:** HWGRADE1 average battery temperature

we could appreciate the offset between the measured and simulated voltage. To understand this issue, we need to look at the average temperature and how we assign the temperature input for the 2D LUT. The average temperatures for HWGRADE -20°C and -10°C datasets are -8.96°C and -0.57°C, which are extremely far from the intended temperature. They are even close to the next intended temperature. Consequently, casting a temperature input according to the dataset (-20°C and -10°C) is not a wise option since the parameters retrieved in the HPPC test apply only to that specific temperature (-17.08°C and -7.49°C according to Table 3.2). A further observation of the lower plot of the figures gives us an idea of the discrepancy in the fitting accuracy. At a glance, it seems like the error is growing on the positive phase of the temperature gradient, meaning when the temperature is ramping up, the error is also escalating. This is partially true. Temperature is the indication of the discharging and charging phase of the cycle. When the battery is discharged, the temperature ramps up, and vice versa. The charge and discharge nature emphasizes the hysteresis which we didn't model in the DP ECM. The hysteresis particularly becomes important for lower temperature, which explains why there are disparities between the charging and discharging phase during the HWGRADE1 particularly on these two datasets and not on high temperatures.

Fixed Input Temperature RMSE over Driving Cycles (mV)		
Test Temperature [°C]	REORDERED1	HWGRADE1
-20	60.46	661.40
-10	45.69	464.12
0	18.47	190.96
10	11.90	106.70
25	8.12	39.09
40	9.81	64.94

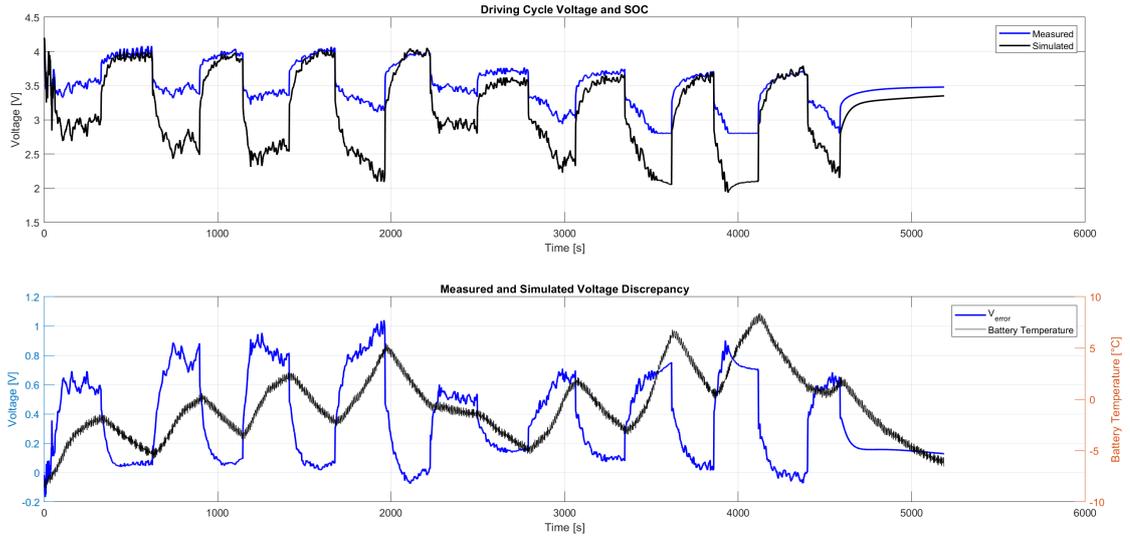
**Table 3.5:** RMSE values for different test temperatures under REORDERED1 and HWGRADE1 driving cycles



**Figure 3.31:** Two-state ECM with fixed temperature input validated on HWGRADE1 -20°C dataset. Voltage RMSE: 661.40 mV

Bottom line, our ECM could not model the large hysteresis that occurs at lower temperatures. Further information regarding hysteresis will be discussed in the next subsection and in the next chapter where the hysteresis modeling is proposed to tackle this problem.

All in all, this finding once more highlights the need to use real-time battery temperature as one of the inputs to the ECM model, apart from the second contribution of error, the hysteresis, which is out of the scope of our model. In real applications, such fluctuating conditions are extremely common and the solution is relatively simple to implement. The temperature data from the battery could be fed directly to the ECU responsible for estimating the SOC.



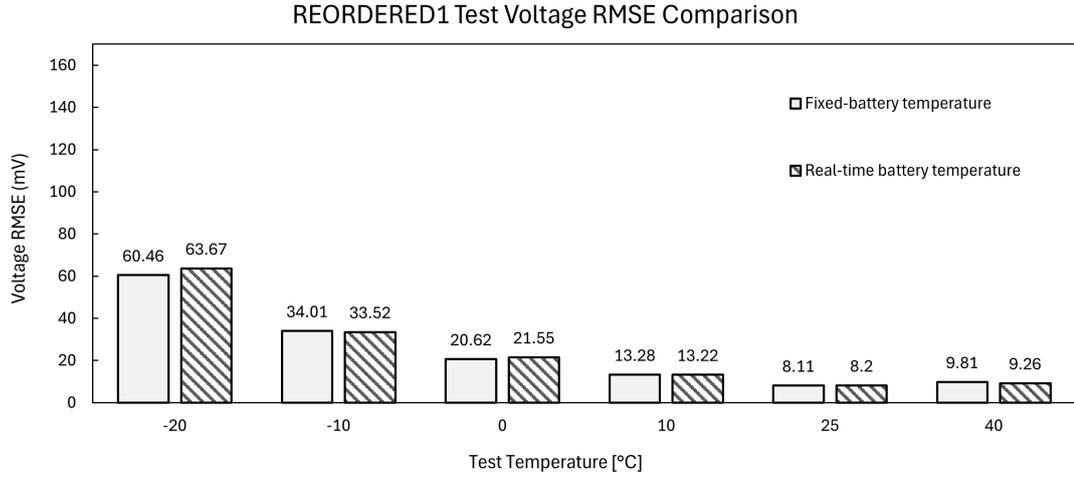
**Figure 3.32:** Two-state ECM with fixed temperature input validated on HW-GRADE1 -10°C dataset. Voltage RMSE: 464.12 mV

### 3.2.3 Model Validation on Driving Cycle

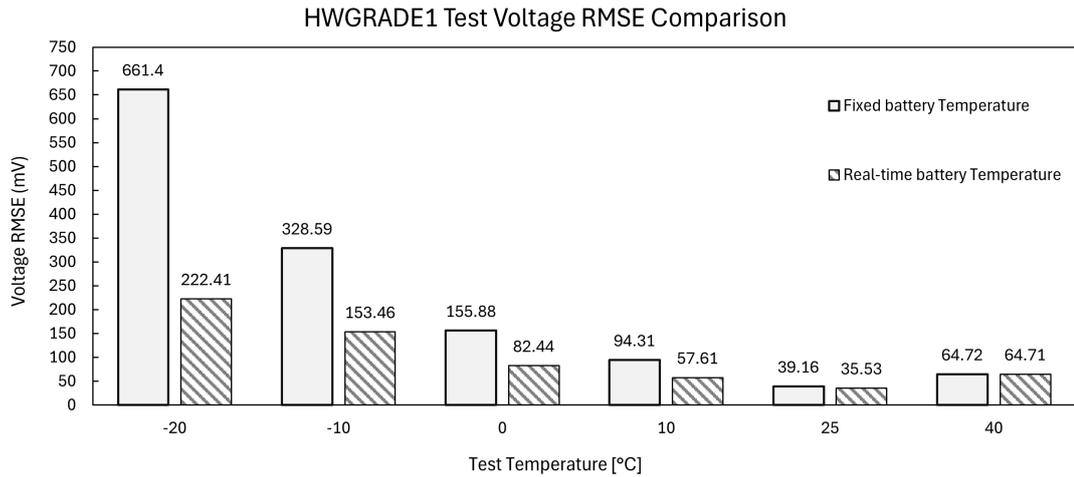
The two-state DP ECM with real-time temperature input to the 2D look-up table needs to be tested against the driving cycles. Initially, the comparative voltage RMSE between fixed and real-time battery for REORDERED1 and HWGRADE1 are summarized in Figure 3.33 and 3.34, respectively for all the temperature datasets given, to give the reader a sense of significance of having a real-time temperature input.

In the REORDERED 1 driving cycle, the use of real-time battery temperature as the secondary input (after SOC) of the parameters' 2D LUT is negatively impacting the RMSE, especially around the lower temperature. Our hypothesis is that the REORDERED1 datasets have a more stable battery temperature than the HWGRADE1. The fluctuations that occurred in these driving cycles are predominantly caused by noises and measurement errors, resembling to what happens also in the HPPC dataset. By feeding the noises inside the real-time battery temperature to the 2D LUT, we are degrading the result. Another major cause is due to unstable HPPC temperature. As shown in Figure 3.27, there is a huge spike until 25°C in the middle of the HPPC test, which significantly increases its average temperature and thus makes it not representative to the HPPC test temperature. Nevertheless, the worsening RMSE is negligible compared to what we will see next in the HWGRADE1.

The most significant improvement happens in the HWGRADE1. As we expected by looking at the high-temperature fluctuation, in addition to the significant offset



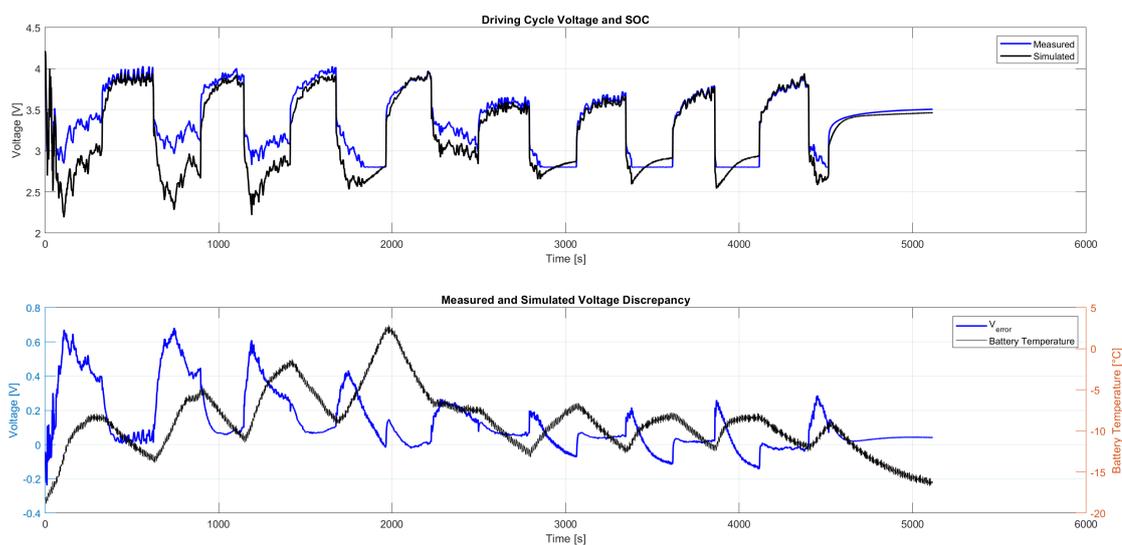
**Figure 3.33:** REORDERED1 driving cycle RMSE in mV comparison between fixed temperature input and real-time temperature input on two-state DP ECM



**Figure 3.34:** HWGRADE1 driving cycle RMSE in mV comparison between fixed temperature input and real-time temperature input on two-state DP ECM

between the average battery temperature and the intended test temperature, the usage of real-time battery temperature massively and positively impacts the RMSE. In the lowest temperature dataset (-20°C), the RMSE is cut down to 222.41 mV, equivalent to 66.3% improvement. This advantage is then reduced as we pass to the higher temperature dataset. The reason for this is due to the temperature variance as presented in Table 3.4. The more discrepancy in the average temperature, the bigger the advantage of using real-time temperature on our model.

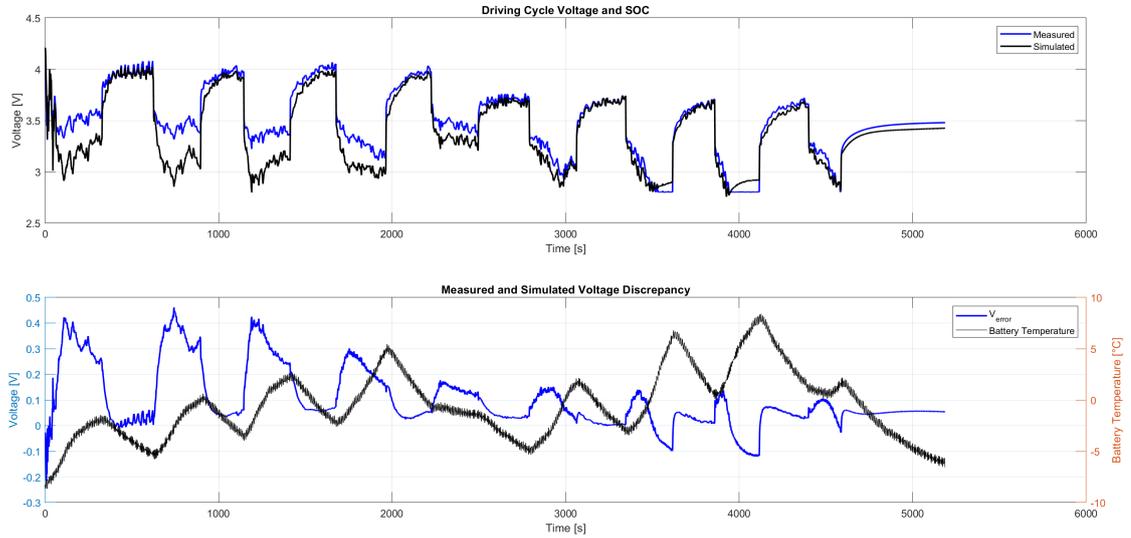
Apart from our effort to raise the model's accuracy by incorporating real-time temperature, in the end, our attempt is insufficient to reduce the error below a satisfactory level. As a means to understand the phenomenon, we will further analyze the lowest two temperature datasets (the  $-20^{\circ}\text{C}$  and  $-10^{\circ}$  dataset) and compare them to the fixed-temperature input solution presented in Figure 3.31 and Figure 3.32.



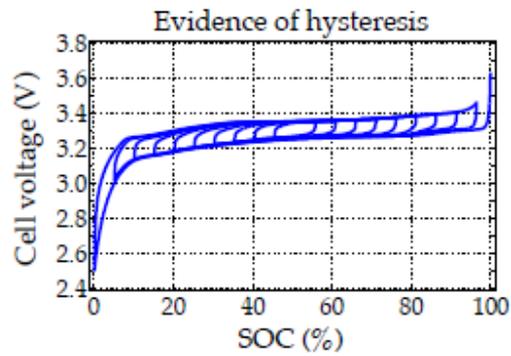
**Figure 3.35:** Two-state ECM with real-time temperature input validated on HWGRADE1  $-20^{\circ}\text{C}$  dataset. Voltage RMSE: 222.41 mV

Compared to Figure 3.31, the output voltage on  $-20^{\circ}\text{C}$  dataset fits the measurement voltage better although there are still huge errors unsolved between the two, especially in the discharging phase. This disparity is caused by voltage hysteresis. Plett [5] mentioned that voltage hysteresis is speeding up at lower temperatures and decreasing in magnitude as temperature increases. The hysteresis effect becomes more significant due to the slower diffusion of lithium ions at lower temperatures according to Bodnar [31]. The illustration of hysteresis as retrieved from Plett's book is presented in Figure 3.37.

An important remark on this SOC estimation, the accuracy of the equivalent model is crucial for the accuracy of the Extended Kalman Filter, as the latter is built upon the discretized version of the ECM. One of the inputs of the SOC estimation is the voltage resulted by the ECM, meaning if the ECM couldn't fit the measured model, EKF's accuracy would also be limited by the ECM accuracy. These errors related to the modeling process, in the EKF, will be considered as process noise.



**Figure 3.36:** Two-state ECM with real-time temperature input validated on HWGRADE1 -10°C dataset. Voltage RMSE: 153.46 mV



**Figure 3.37:** Hysteresis example. Retrieved from Plett [5]

# Chapter 4

## Extended Kalman Filter Model

### 4.1 One-state discrete model

In the early section, the continuous model of the dual-polarization equivalent circuit model has been developed on Simulink. Considering the purpose of our work, which is to develop an Extended Kalman Filter for State-of-Charge estimation of the battery, and the intended real-time application on-board of the vehicle, a fast and reliable discrete model shall be developed instead of continuous-time. The discrete model is implemented in MATLAB code with a time step following the given data array, which is 1 s (1 Hz sampling frequency). The continuous ECM equations given in Equation 3.1, 3.2, and 3.3, are translated to the discrete-time domain in the state-space system format.

In general, the discrete equation is given by:

$$x_{k+1} = f(x_k, u_k) \quad (4.1)$$

$$y_k = g(x_k, u_k) \quad (4.2)$$

where  $x_k$  is the state vector of the system at a given timestep  $k$ ,  $u_k$  describes the current input to the model, and  $y_k$  is the terminal voltage output from the model. Equation 4.1 describes the state space transition from one timestep to the next one, while Equation 4.2 describes the measurement output equation. The state vector  $x_k$  is defined as:

$$x_k = \begin{bmatrix} \text{SOC} \\ V_{RC1} \\ V_{RC2} \end{bmatrix} \quad (4.3)$$

Correlating these general equations to the corresponding physical relationships in discrete time, we would get:

$$\mathbf{f}(x_k, u_k) = \begin{pmatrix} SOC + \frac{dt}{Q_n} \cdot i_k \\ V_{RC1} e^{-\frac{dt}{R_1 C_1}} + \left(1 - e^{-\frac{dt}{R_1 C_1}}\right) R_1 \cdot i_k \\ V_{RC2} e^{-\frac{dt}{R_2 C_2}} + \left(1 - e^{-\frac{dt}{R_2 C_2}}\right) R_2 \cdot i_k \end{pmatrix} \quad (4.4)$$

where  $dt$  is the time step, and  $Q_n$  is the nominal capacity of the battery at a certain condition (temperature and aging). The output of the model  $g(\cdot)$  are described as follows:

$$\mathbf{y}(x_k, u_k) = OCV + V_{RC1} + V_{RC2} + R_0 \cdot i_k \quad (4.5)$$

Consider that the parameters  $OCV$ ,  $R_1$ ,  $R_2$ ,  $C_1$ , and  $C_2$  are modeled as a function of SOC for the 1D discrete model, and also the temperature in the 2D discrete model. It is important to acknowledge that the state transition equation  $f(\cdot)$  exhibits linearity, as demonstrated by decomposing the vector in relation to  $x_k$  and  $u_k$ . The non-linearity of the model lies inside the parameters, such as the SOC–OCV correlation, resistors, and capacitors correlation to SOC.

As previously explained, the discrete model is developed in MATLAB, and the 1D discrete model is the basis of our complete 2D EKF. It is crucial to have an accurate and functioning discrete model prior to developing the EKF, to separate the error proportion of discretization error and the tuning error of EKF as we will discover later. This section will take 25°C dataset as an example to have continuity in explanation and to allow comparison with the continuous model. In general, the idea of the discrete model is the same as the previous continuous model. Apart from the obvious discretization, the only difference is how we model the look-up table. In the Simulink environment, loading the table and breakpoints could be easily and graphically achieved by using a 1D-LUT toolbox. In MATLAB however, such behavior could be replicated by using a *griddedInterpolant* function. Refer to MATLAB documentation for further information.

Inside our model, each of the six parameters is loaded up onto a 1D grid (or table), containing SOC as the sample point ( $x$ ) and the parameter's value profile for a given temperature as the corresponding point ( $v$ ). Figure 4.1 depicts a snapshot of part of the code with the upper part, explaining how the grid interpolants for all parameters are formed. The first input is the SOC array, which is different depending on the parameter referred. As explained in Chapter 4, the RC pairs have only 7 levels of SOC as a means to compare them to other publications, while the  $R_0$  and  $OCV$  both use the complete 14 levels of SOC. The second input shows the array profile of the parameters at a given temperature. The third input imposes the type of interpolation desired, which in this case is linear interpolation. The

extrapolation method is set to be nearest as imposed in the fourth input for the griddedInterpolant function.

```

% Making griddedinterpolants
R0_grid = griddedInterpolant(SOC_array, unique_R0, "linear", "nearest");
R1_grid = griddedInterpolant(SOC_array1, R1_array, "linear", "nearest");
C1_grid = griddedInterpolant(SOC_array1, C1_array, "linear", "nearest");
R2_grid = griddedInterpolant(SOC_array1, R2_array, "linear", "nearest");
C2_grid = griddedInterpolant(SOC_array1, C2_array, "linear", "nearest");
OCV_grid = griddedInterpolant(SOC_array, OCV, "linear", "nearest");

for i = 1:length(time_d)
    % Call the griddedinterpolant to insert the query value
    R0_d = R0_grid(SOC_d(i));
    R1_d = R1_grid(SOC_d(i));
    C1_d = C1_grid(SOC_d(i));
    R2_d = R2_grid(SOC_d(i));
    C2_d = C2_grid(SOC_d(i));
    OCV_d = OCV_grid(SOC_d(i));
end

```

**Figure 4.1:** Gridded Interpolant approach as a look-up table for DP ECM parameters in a given SOC

The discretization approach requires a *for* loop in MATLAB to be deployed on the code to move the time from one instance to the next one. Subsequently, on every given time step, the grid interpolants of all parameters are called with the current  $SOC(k)$  as the input.

The results of the discrete model for 25°C on various tests are to be compared with the Simulink continuous model and the measurement data. As a means to measure the discrepancy between them, a quantitative metric root mean squared error (RMSE) is used.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (4.6)$$

where  $y_k$  represents the actual measurement voltage from the dataset at a given timestep  $k$  and  $\hat{y}_k$  represents the simulated voltage from our model. Figure 4.2 depicts the voltage output comparison between discrete model, continuous model, and the measurement value on the HPPC test. As seen, there a very little difference between those three. A further analysis was conducted to see the voltage discrepancy between the discrete and continuous model (Figure 4.3 above), and discrete model and measurement data (Figure 4.3 below). Looking at the scale of the above plot, the discrete model could follow the continuous model very closely, with a negligible discrepancy of under 2.5 mV or 0.0025 V. By increasing in the sampling frequency, the discrepancy could only be minimized. The RMSE value between discrete and continuous models is 0.0734 mV. We could conclude that the discrete

and continuous model have no significant difference.

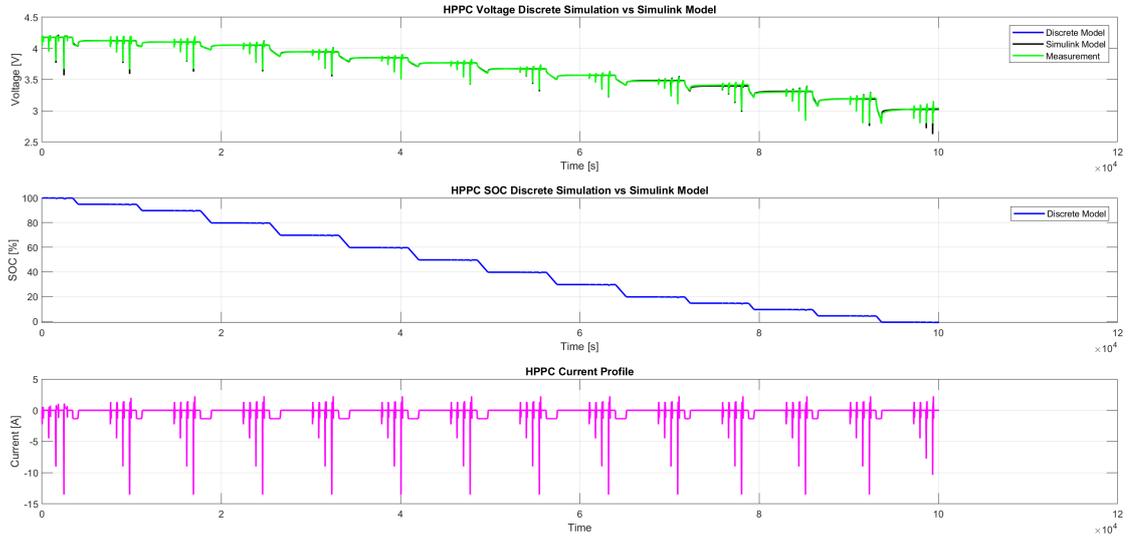


Figure 4.2: Discrete model voltage output comparison

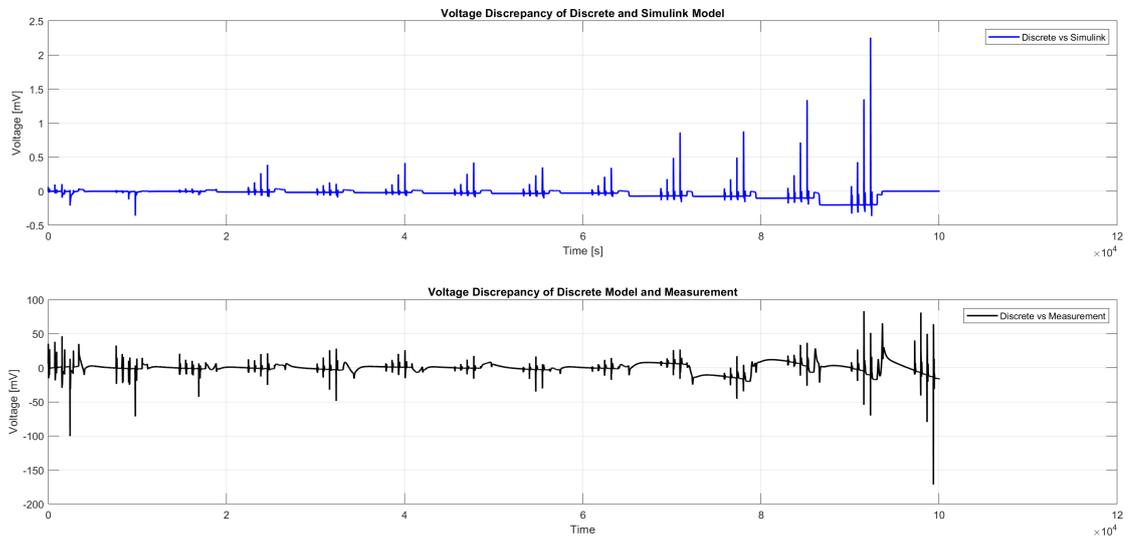


Figure 4.3: Voltage discrepancy between discrete model and continuous model (above) and discrete model and measurement data (below)

## 4.2 One-state Extended Kalman Filter

The one-dimensional discrete DP ECM model introduced earlier provides a physics and mathematical foundation between the input, states, and output of our Kalman Filter model. Early in this chapter, it has deliberately explained the importance of the Kalman filter as an observer algorithm to obtain a good estimate of the hidden states of the system, such as SOC, by taking the input and output measurement of the model. It implies that an accurate discrete model would contribute the the less measurement error of the Kalman Filter and hence, increase its estimation accuracy.

As the basis of our Extended Kalman Filter development, Plett [5] introduced the extended Kalman filter as a modified general standard Kalman filter that linearizes the non-linear system at each instance. Before starting the development of Extended Kalman filter, some notations are important to be understood as it will be needed to indicate an extra-information about the variables' status.

- A superscript “-” indicates a predicted quantity based only on past measurements.
- A superscript “+” indicates an estimated quantity based on both past and present measurements. It implies that estimated quantity is different from a predicted quantity.
- The hat annotation “^” on top of a variable indicates either a predicted or an estimated value of that variable. For example,  $\hat{x}_k^-$  indicates the predicted value of  $x_k$  and  $\hat{x}_k^+$  indicates the estimated value of  $x_k$ , both at time step  $k$
- The tilde decoration “~” on a variable means an error between true and predicted or estimated quantity. For example,  $\tilde{x}_k = x_k - \hat{x}_k^-$  and  $\tilde{x}_k^+ = x_k - \hat{x}_k^+$ .
- The symbol “ $P$ ” indicates the correlation between the two components in its subscript.

$$P_{xy} = \mathbb{E}[xy^T] \quad \text{and} \quad P_x = \mathbb{E}[xx^T].$$

### 4.2.1 Extended Kalman Filter development: six-step method

The development of Extended Kalman Filter is broken down into 6 steps as follows. This section is presented as a pragmatic and practical approach of Extended Kalman Filter without having to dig into the derivation of equations:

**Phase zero: Initialization** Initialization of the first time instance is an important process since it lays the iterative foundation for the next timestep. This initialization is part of the tuning process and will be further explained in the following subsection.

**First phase: Prediction**

In this phase, our EKF calculates the predicted quantities based on the past and present measurements and states. This phase is composed of three prediction: states, error-covariance, and the system output.

**Step 1a: State-prediction time update**

Based on the knowledge of the previous timestep (or from information obtained prior to the measurement being made at time  $k$ ) and the system model of our discrete model, we try to compute an updated prediction of the current value of  $x_k$  at each time step. In this step, we introduce the state-space matrices A and B:

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \exp\left(-\frac{dt}{R_1 C_1}\right) & 0 \\ 0 & 0 & \exp\left(-\frac{dt}{R_2 C_2}\right) \end{bmatrix}$$

$$B = \begin{bmatrix} \frac{dt}{Q_{\text{batt}} \cdot 3600} \\ -\left(1 - \exp\left(-\frac{dt}{R_1 C_1}\right)\right) R_1 \\ -\left(1 - \exp\left(-\frac{dt}{R_2 C_2}\right)\right) R_2 \end{bmatrix}$$

The state prediction is based on the past state value  $\hat{x}_{k-1}^+$  and matrices  $(A_{k-1} B_{k-1})$  which is described by the equation below.

$$\hat{x}_k^- = A_{k-1} \hat{x}_{k-1}^+ + B_{k-1} u_{k-1} \tag{4.7}$$

where  $u_{k-1}$  is the input of previous iteration.

**Step 1b: Error-covariance time update**

To compute the prediction-error covariance, the previous timestep corrected-error covariance and the estimated A matrix from the previous step are needed, along with the process noise covariance  $Q_{x,k-1}$ . The relations between those parameters are as follows:

$$P_{x,k}^- = \hat{A}_{k-1} P_{x,k-1}^+ \hat{A}_{k-1}^\top + Q_{x,k-1} \tag{4.8}$$

**Step 1c: Prediction of system output  $y_k$**  The voltage output is predicted using Equation 4.5. One important step that differentiate Extended Kalman Filter and the rests is the linearization process. In this step, the OCV voltage is linearized. The means of linearization in the MATLAB environment of OCV and other parameters are exactly the same as in Discrete Model by using a *griddedInterpolant* function. In the equation below, the RC voltages are modified as the second and third state estimates of our model as follows:

$$\hat{y}_k = OCV_k + \hat{x}_k^-(2) + \hat{x}_k^-(3) + R_{0,k} \cdot i_k \quad (4.9)$$

**Second phase: Correction**

We correct the parameters for the advanced time instance based on our predicted and estimated quantities.

**Step 2a: Defining the estimator gain matrix  $L_x$**

Before computing the estimator gain matrix, we first define the output prediction as

$$y_k = y_k - \hat{y}_k = g(x_k, u_k, v_k) - g(\hat{x}_k^-, u_k, \bar{v}_k)$$

In the Extended Kalman Filter assumption, the error is approximated using a truncated Taylor-series expansion of  $y_k$  around the setpoint  $q_k = \{\hat{x}_k^-, u_k, \bar{v}_k\}$ , hence, the previous equation could be written in linearized Jacobian matrices

$$y_k \approx \hat{C}_k \tilde{x}_k^- + \hat{D}_k \tilde{v}_k$$

Based on this equation and the covariance matrices resulted from this derivation, we could define the estimator gain matrix as

$$L_{x,k} = \frac{\hat{P}_{x,k} C_k^T}{C_k \cdot \hat{P}_k \cdot C_k^T + R_{x,k}} \quad (4.10)$$

**Step 2b: State-estimate measurement update** The state for next step is corrected by the Kalman gain following this equation:

$$\tilde{x}_k^+ = \hat{x}_k^- + L_{x,k}(y_k - \hat{y}_k) \quad (4.11)$$

where  $y_k$  is the measured voltage at that instant  $k$ . It is important to limit the SOC state. We allow 0.5% percent of overcharging and undercharging the battery.

$$\hat{x}_k(1) = \begin{cases} -0.005 & \text{if } \hat{x}_k(1) < -0.005 \\ 1.005 & \text{if } \hat{x}_k(1) > 1.005 \\ \hat{x}_k(1) & \text{otherwise} \end{cases}$$

**Step 2c: Error-covariance measurement update** At last, the error-covariance is corrected according to this equation:

$$P_{\tilde{x},k}^+ = P_{\tilde{x},k}^- - L_k P_{\tilde{y},k} L_k^T \quad (4.12)$$

## 4.2.2 Parameters Initialization and Tuning

As mentioned above, the parameter initialization is a major part of EKF design. There are mainly 4 items to be initialized: the initial state vector, the corresponding initial state covariance matrix  $P_x$ , and the process  $Q_x$  and measurement  $R_x$  noise covariance matrices. Regarding the state vector, each test always starts from 100% SOC after the battery stabilizes. It implies that the second and third states of the voltage across RC pairs are almost certainly zero. Subsequently, there must be uncertainty about this prediction that will be accounted for in the initial state covariance matrix, the higher the value, the higher the confidence level of our initial prediction. Schneider and Georgakis [32] extensively described the systematic approach to determine the covariance matrices. Haykin [33] also gives a detailed methodology for finding the order of magnitude of these matrices. Most of the writers, though, agree that the order of magnitude of the measurement noise  $R_x$  covariance matrix (or in our case, a single value) could be easily determined by knowing the square root of measurement error. Nonetheless, the measurement error is not given in the data set and thus remains undetermined. Plett [8] on the other hand, suggested that due to inconsistent performance, where some methods tend to perform well in some applications but not as effectively in others, a trial-and-error approach is advisable to get the tuned and working EKF. This approach is also selected to determine ours.

As a base reference, we referred to Khanum's work [13] on the order of magnitude and started the tuning procedure by changing the three parameters. The objective of the tuning is to obtain the parameters that with the best SOC prediction, quantified by the least RMSE across different tests (HPPC and driving cycles) under multiple different state initialization and offsets. The details of these tests are explained in the following subsection. After several trials, we obtained the value of  $P_x$ , the process noise covariance  $Q_x$  and measurement noise covariance  $R_x$  as:

$$P_x = \text{diag}(0.025, 0.01, 0.01)$$

$$Q_x = \begin{bmatrix} 1 \times 10^{-5} & 0 & 0 \\ 0 & 20 \times 10^{-5} & 0 \\ 0 & 0 & 20 \times 10^{-5} \end{bmatrix}$$

$$R_x = 0.05$$

## 4.2.3 One-state EKF Model Validation on the HPPC and Driving Cycle Tests

In validating the model, there are mainly three different steps to be done:

1. Simulation on the normal test with the correct initialization of the SOC (starting from 100%) on the corresponding temperature, i.e HPPC test, RE-ORDERED1 driving cycle, and HWGRADE1 driving cycle.
2. Simulation with the wrong initialization of the SOC: 80% and 0%.
3. Simulation with the offset value of current input  $I_{batt}$ .

In each case, we are going to quantify the RMSE between the estimated SOC from our EKF model and the measured SOC from the dataset.

### **SOC estimation on the HPPC Test using 1D EKF**

The result of HPPC is satisfying with the SOC predicted using the EKF could closely predict the SOC as shown in Figure 4.4. The coulomb counting annotation acts as the reference of SOC, which array was calculated and extracted from the Arbin measurement tool. The  $1-\sigma$  uncertainty bound is also plotted in the figure based on the following equation for 0-100% scale of SOC:

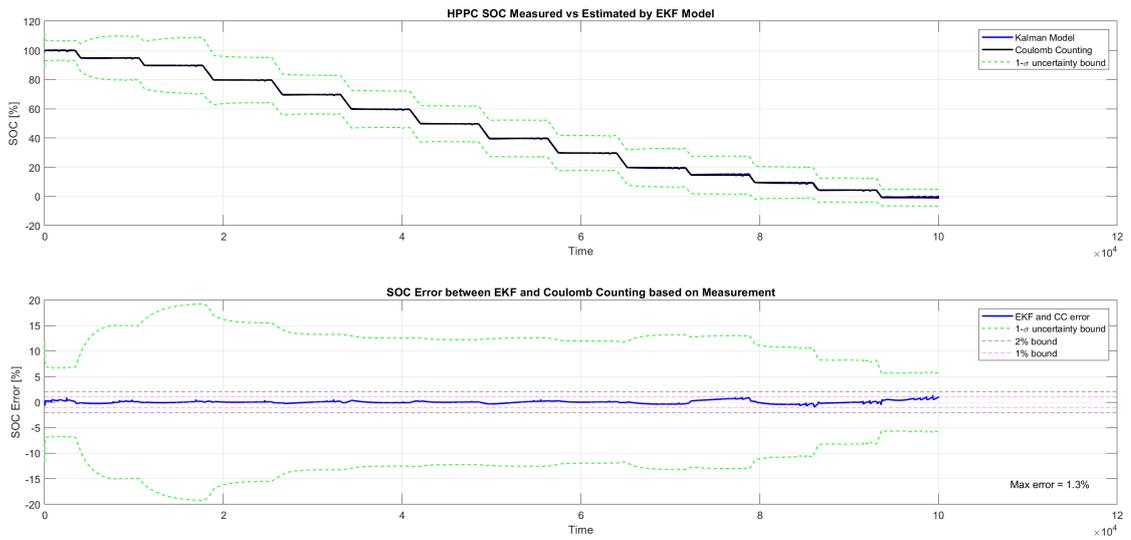
$$1-\sigma = \pm\sqrt{P_x(1)} \times 100$$

with  $P_x(1)$  being the state covariance of SOC state. This  $1-\sigma$  bounds covers around 68% of the potentially true values in a normal distribution. On the second plot of Figure 4.4, the error between EKF and the reference is shown. Notice how the error are minimal, with SOC RMSE of 0.281%, meaning that our EKF model functions properly for this particular temperature, HPPC test, and with correct initialization.

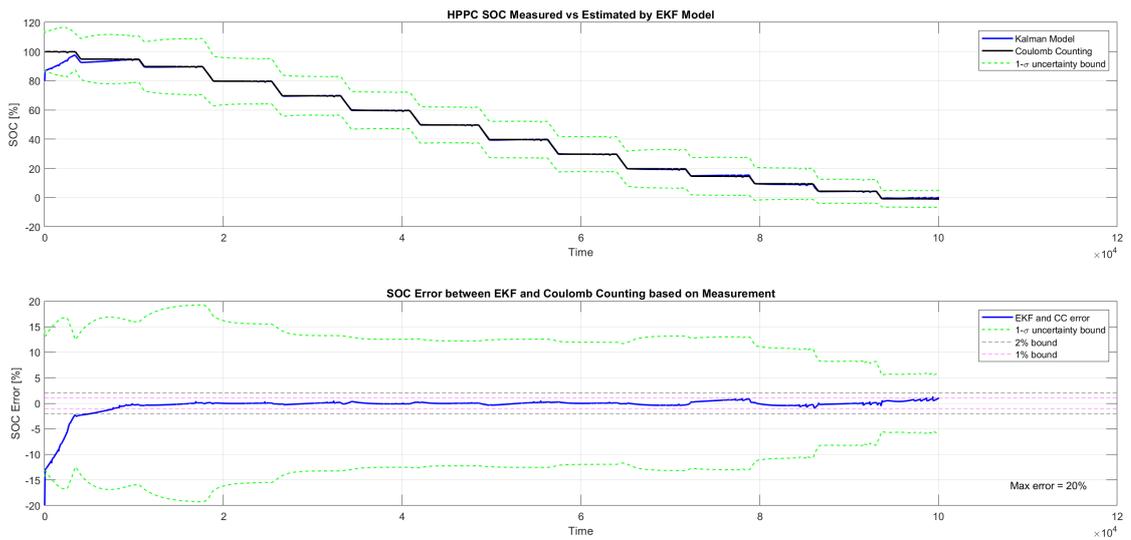
The next step as mentioned, is to test the robustness of the EKF model. Not in every case we know the initial value of the SOC like in a controlled test, especially when the algorithm is about to be deployed in a real driving environment where the initial SOC state is uncertainly known and where it's very likely that the sensor's current reading is being interfered. These next two tests: wrong initialization and current offset act as our robustness test.

### **EKF wrong initialization test on HPPC**

Instead of using 100% SOC, we are going to impose the initial SOC as 80% and 50% and check whether the EKF is adaptive. The simulation output is plotted in Figure 4.5 and 4.6 for 80% and 50% initial SOC, respectively. Both plots show an adaptive EKF indicated by the SOC returning to the initial estimate to the value of our reference. Looking at the errors, after the huge discrepancy at the start, the SOC returns exponentially to under 1% of the absolute error bound and remains quasi-constant until the end of the simulation. This proves that our tuning is robust enough to handle wrong initialization on the HPPC test. The



**Figure 4.4:** SOC comparison between EKF model and measured data on HPPC with correct initialization



**Figure 4.5:** EKF wrong initialization (80%) test on HPPC

SOC RMSE for 80% and 50% initial SOC are 1.69% and 4.10%. respectively, of which the major part of the error comes from the initial phase.

### EKF offset current test on HPPC

The purpose of this test is to replicate the faulty condition of current sensor reading and check how far the EKF could still give an accurate prediction. There are two

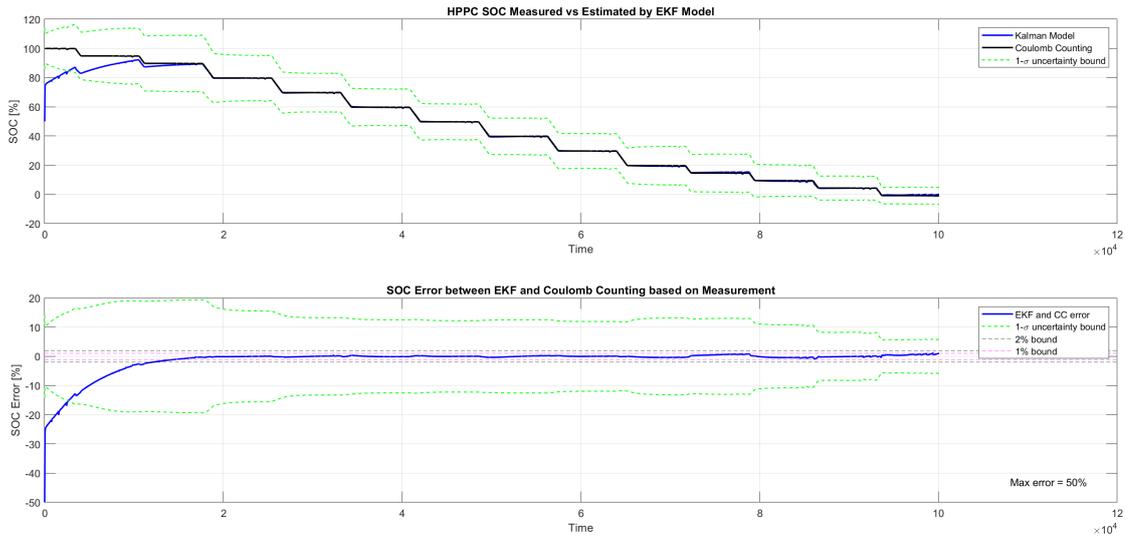


Figure 4.6: EKF wrong initialization (50%) test on HPPC

sub-case of the current test, with the offset being 0.1 A and 0.5 A. The snapshots of HPPC offset current profiles  $I_{batt}$  are illustrated in Figure 4.7. The latter is considered to be a severe faulty condition, meaning the test is at a high level of disturbance. In each plot, there are three signals to be compared: SOC of our EKF model considering the offset current, the reference SOC without any offset, and the reference SOC coulomb counting if the offset is considered. The overall RMSE between true reference SOC and our EKF model is 0.50% which is still in the tolerable range.

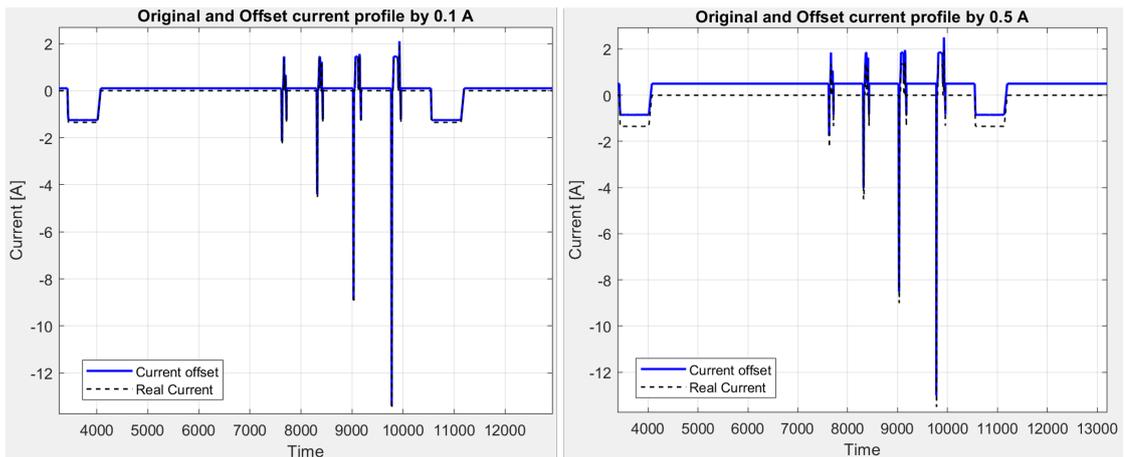


Figure 4.7: Current offset test for HPPC with 0.1 and 0.5 A offset

As shown in Figure 4.8, the coulomb counting that considers the current offset has a higher SOC in every instance since the battery is always charged more by 0.1 A throughout the test. What is unique is that our EKF follows the true reference, which is the dataset's original SOC even though the input for our EKF has 0.1 A offset. This result is fascinating and it proves that our tuning allows the EKF model to auto-align itself based on the voltage-measured quantity and the tuned covariance matrices.

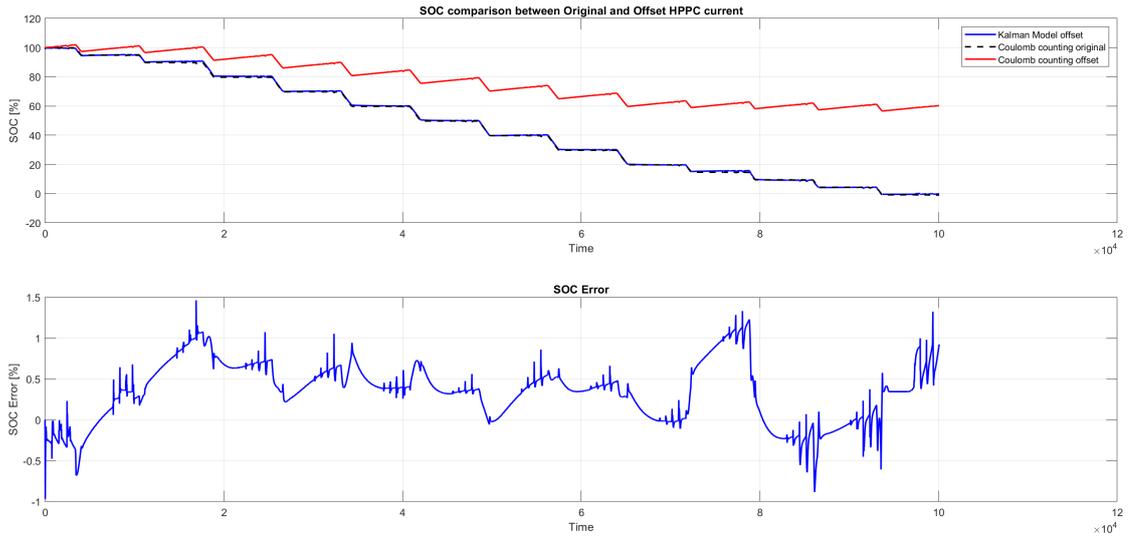


Figure 4.8: Adaptive behaviour of EKF model under 0.1 A offset current test

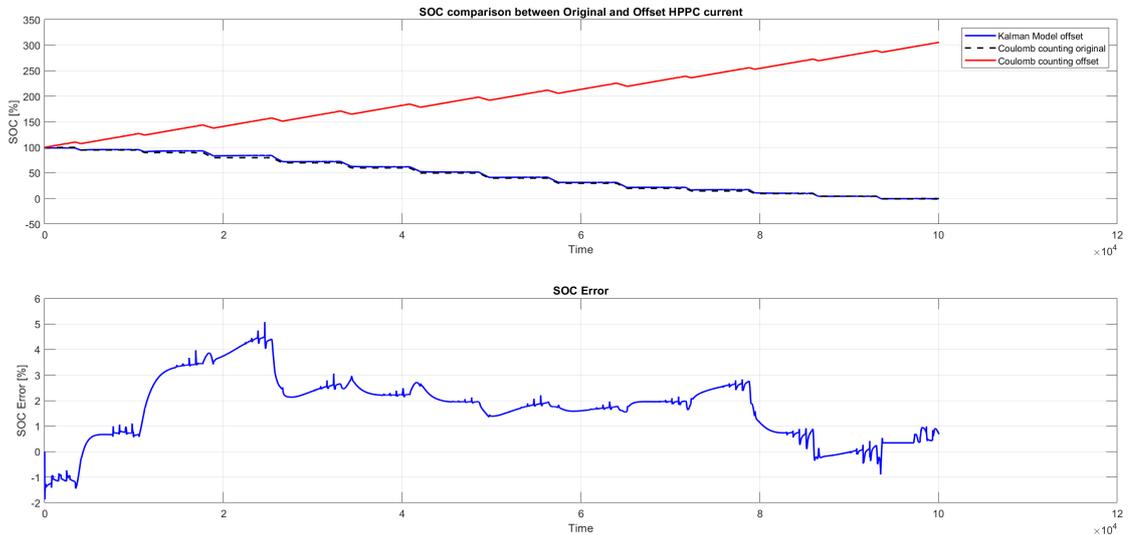


Figure 4.9: Adaptive behaviour of EKF model under 0.5 A offset current test

Similarly, Figure 4.9 shows alike trendline for the 0.5 A offset on EKF. The coulomb counting SOC based on our offset current shows an overall charging condition, affirming our previous statement on how severe this test case is. Our SOC prediction from EKF model could still follow the true reference value, indicating an excellent performance even under high disturbance.

### SOC estimation on the REORDERED1 Driving Cycle using 1D EKF

With similar manner, the model is validated on REORDERED1 Driving cycle. Figure 4.10 depicts the SOC prediction result and its corresponding error to the measurement reference. We could clearly see the value discrepancy kept on increasing between the predicted SOC and the measured one. In order to understand whether or not the EKF tuning is not suitable for this particular cycle, we will take a look at the voltage error between the 1D discrete model and the measured value. Figure 4.11 shows that the discrete model simulated voltage is drifting apart from the measured voltage from the dataset, particularly towards the end of the simulation. This proves that the EKF is doing its job just fine by following the simulated voltage coming from the discrete model. ECM model is known to have a poor estimates around the low SOC, especially when the hysteresis is high, and this might be the case that explains what we encountered. In any case, the drift is simply minimal, with the maximum error being 4% at the end of the simulation. The RMSE throughout the whole cycle is around 1.72% which is still highly acceptable. We will see later that by incorporating temperature in our discrete model, the accuracy of it will increase, and thus reducing the SOC discrepancy coming from the EKF model.

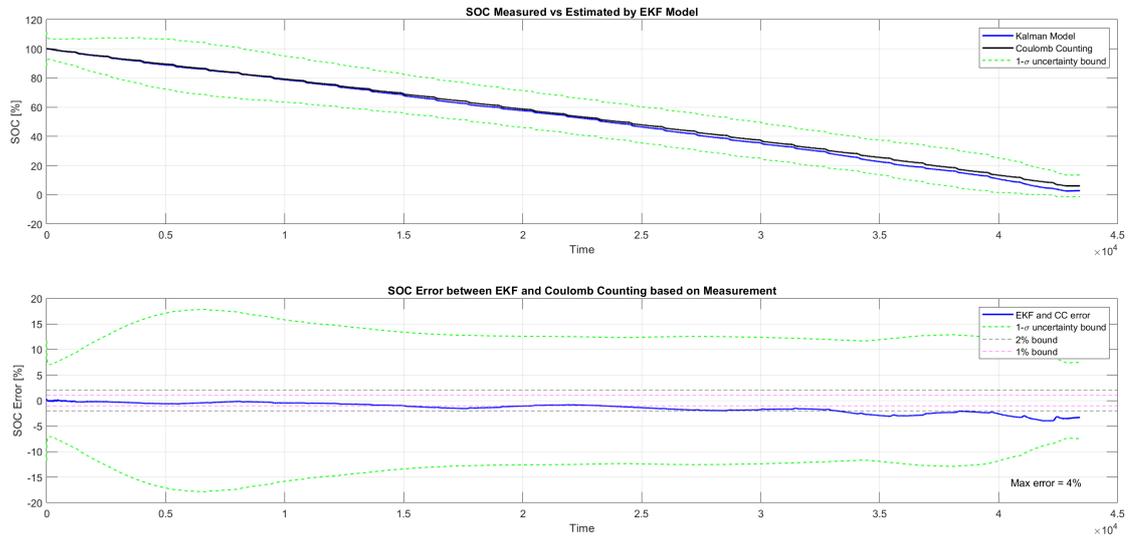
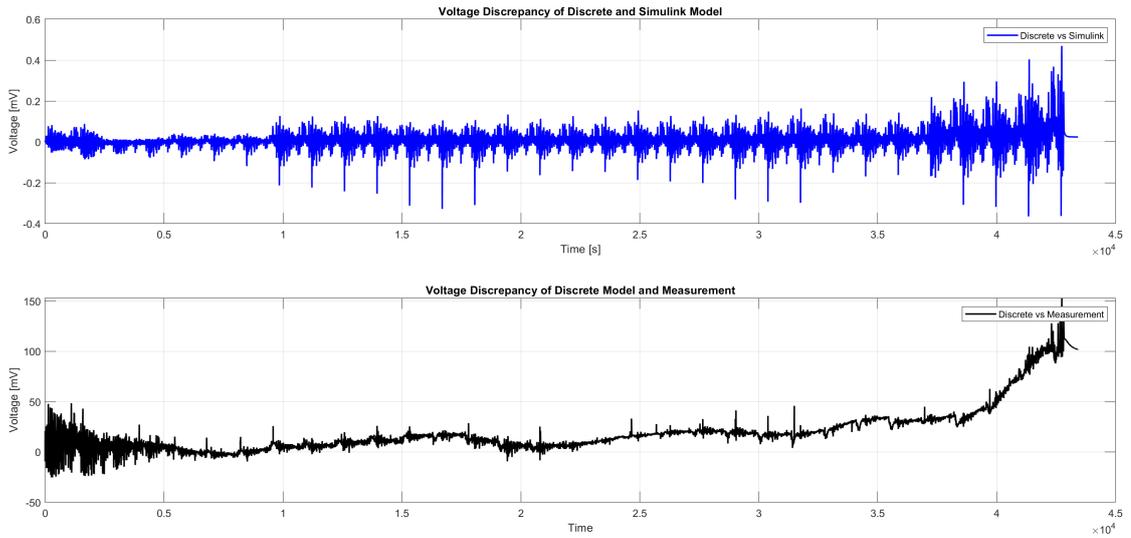
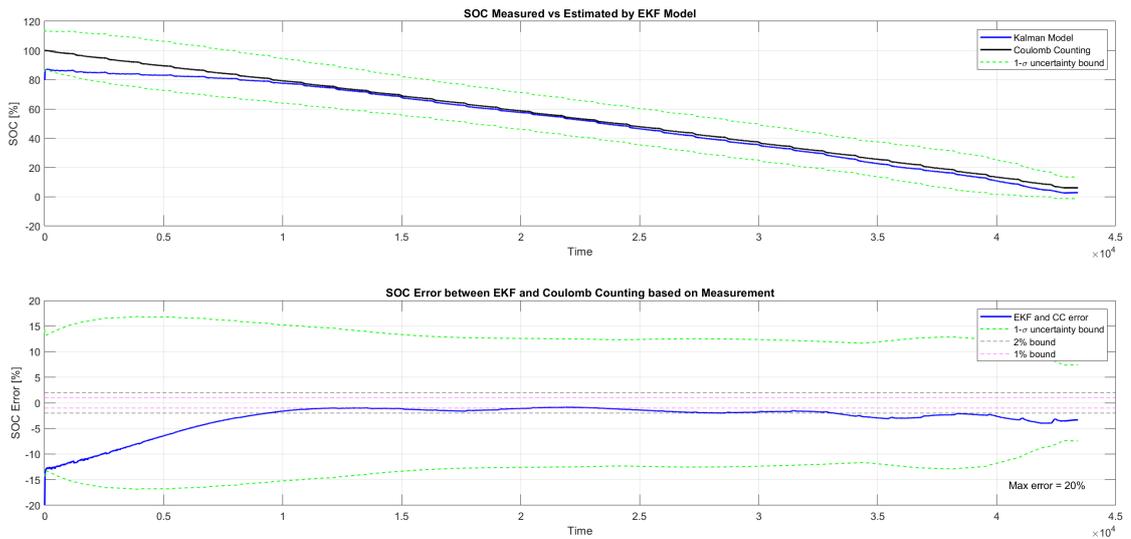


Figure 4.10: EKF performance under REORDERED 1 Driving cycle



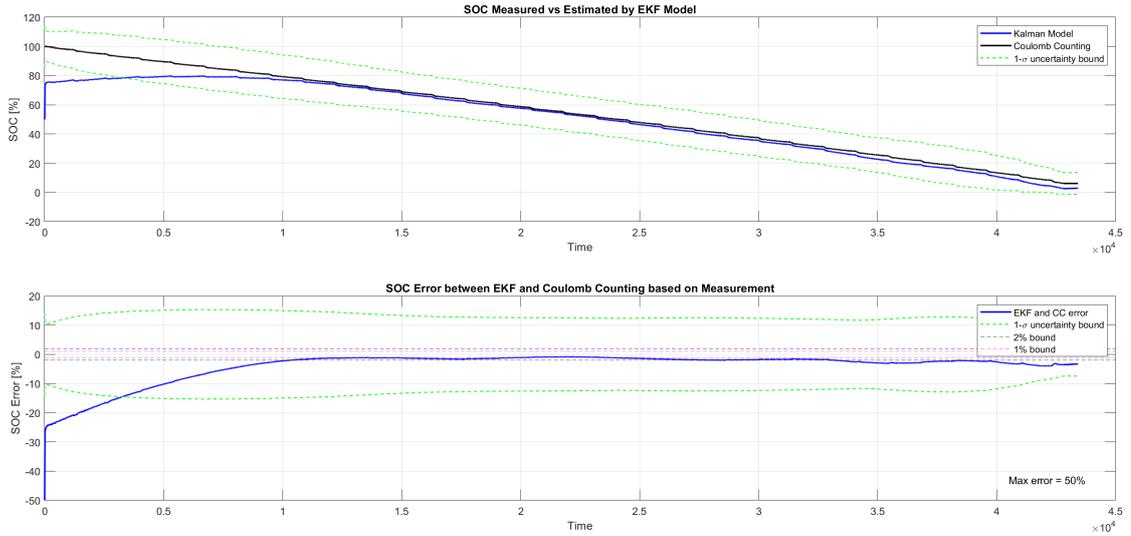
**Figure 4.11:** Voltage discrepancy between the 1D discrete model and the Simulink (above) and the measured value (below). The drift between the discrete model and the measured value results in poor estimation for the EKF



**Figure 4.12:** EKF wrong initialization (80%) test on REORDERED1

### EKF wrong initialization test on REORDERED 1

As before, the EKF model SOC is mistakenly initialized which result can be observed in Figure 4.12 for 80% initial SOC and 4.13 for 50% initial SOC, Besides the gradually increasing error explained earlier, in both cases, the EKF model is adaptive and capable of closing the gap to the true reference SOC value quite



**Figure 4.13:** EKF wrong initialization (50%) test on REORDERED1

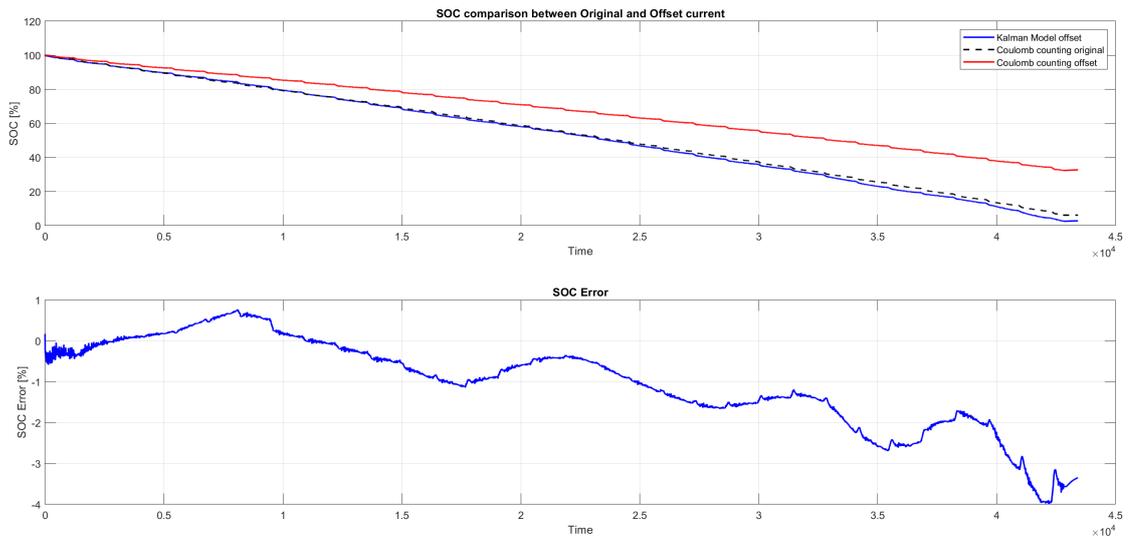
promptly, showcasing the robustness and adaptability of our EKF model. The RMSE for these 80% initial SOC and 50% initial SOC are 4.05% and 6.56%, respectively.

### EKF offset current test on REORDERED 1

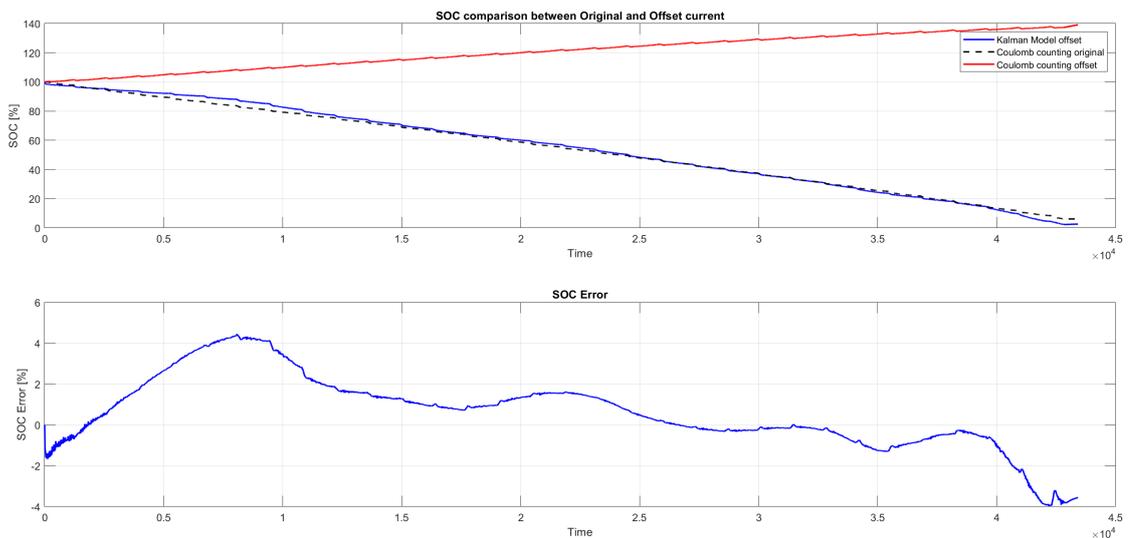
Under the current offset test, the EKF model for REORDERED1 is performing even exceptionally for both offset cases. In both cases, the EKF follows the true reference value as shown in Figure 4.14 and 4.15. The SOC errors between the measured value and the simulated value are corrected in both plots (lower figures) before the voltage error of the discrete model becomes prominent. What is fascinating about these results is the RMSE of these tests. The 0.1 A offset test resulted in 1.46% of SOC error while the 0.5 A offset augmented the RMSE only to 1.88%. In other words, the disturbance coming from the faulty current measurement has the same order of magnitude of RMS error as if there is no disturbance.

### SOC estimation on the HWGRADE1 Driving cycle

HWGRADE is one of the most severe cycles, indicated by intense and long C-rate with several charging phases, presumably from regenerative braking. It is interesting to see how the EKF model will perform under such cycles and different test cases as an integral part of our understanding of our model characteristics. Similar to previous cycles, we will conduct normal test, wrong initialization test, and current offset test.

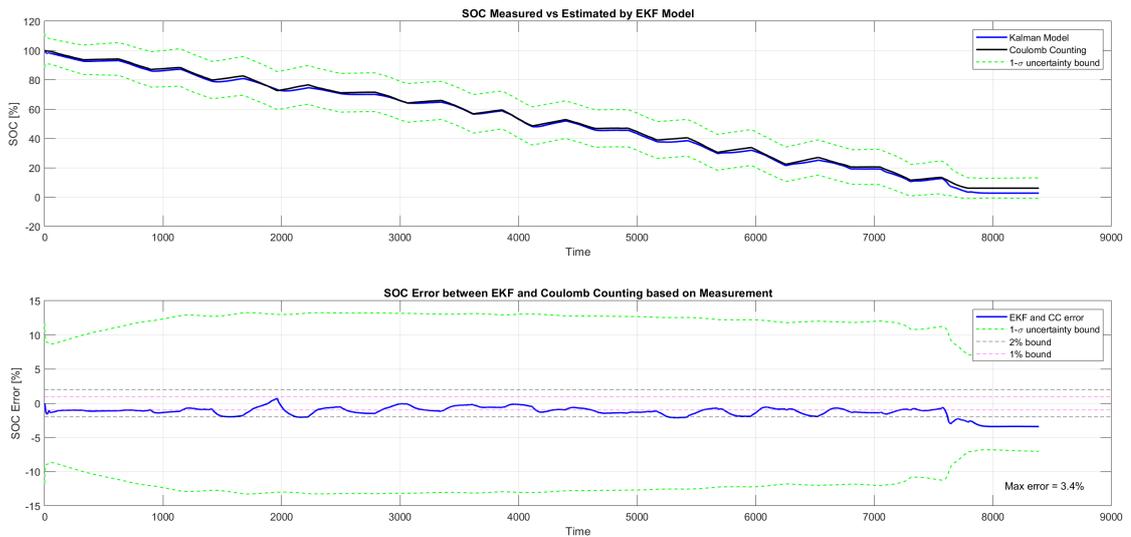


**Figure 4.14:** EKF result on REORDERED 1 with 0.1 A current offset

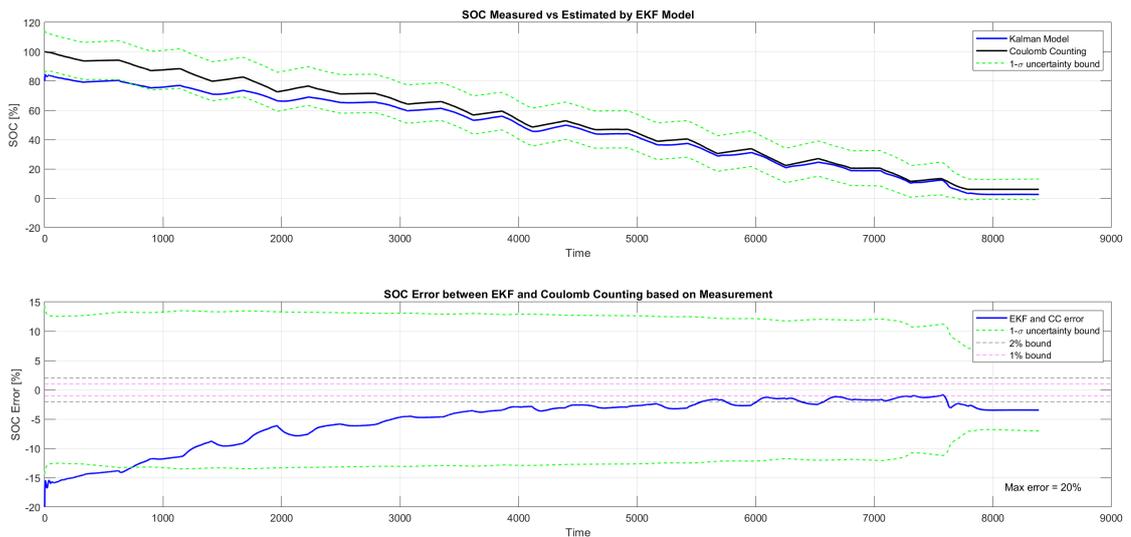


**Figure 4.15:** EKF result on REORDERED 1 with 0.5 A current offset

Figure 4.16 again shows promising EKF performance with 1.46% of SOC RMSE. The SOC error is maintained under  $\pm 2\%$  bound throughout the cycle until the simulation approaches the end relaxation phase, where the error grows slightly to 3.4%. This last SOC error is trivial and is not caused by the incapability of EKF but due to inaccuracy of the discrete model (ECM model in general) under low SOC condition.



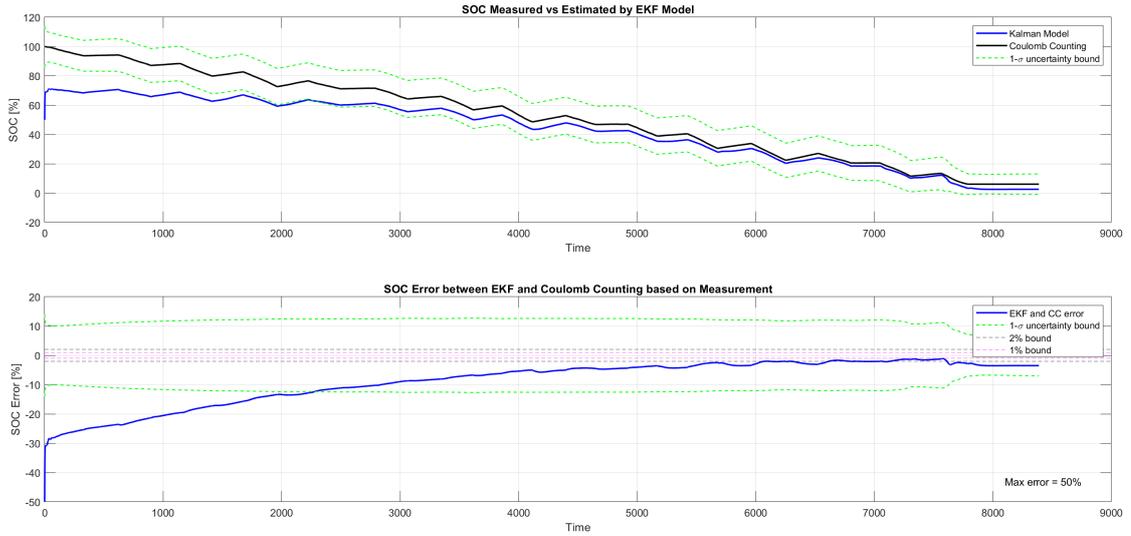
**Figure 4.16:** SOC comparison between EKF model and measured data on HWGRADE 1 with correct initialization



**Figure 4.17:** SOC comparison between EKF model and measured data on HWGRADE 1 with 80% initial SOC

### EKF wrong initialization test on HWGRADE 1

Under HWGRADE1 with the wrong SOC initialization, the results are not quite satisfying (Fig 4.17 for 80% SOC initial and 4.18 for 50% SOC initial). As we know,

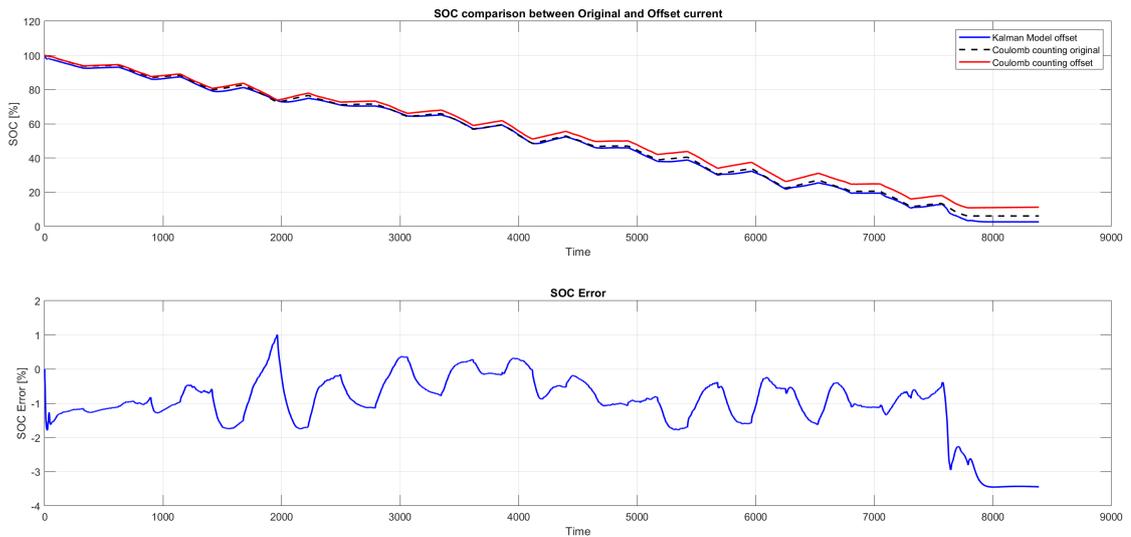


**Figure 4.18:** SOC comparison between EKF model and measured data on HWGRADE 1 with 50% initial SOC

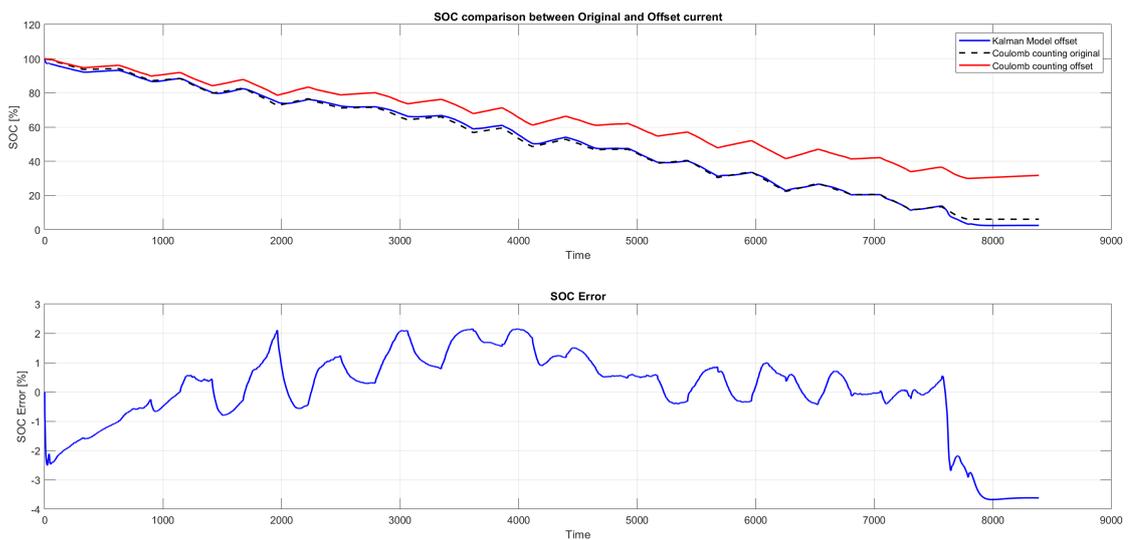
HWGRADE1 has an intense and dynamic profile, and by looking at the figures for both cases, the EKF model doesn't seem to be agile enough for this. Recall that in REORDERED1, the results for the wrong initialization were on par with the correct SOC initialization, meaning the only variable that changes here is the driving cycle. The RMSE reported for 80% and 50% initial SOC case are 6.62% and 11.60% which are considerably high. One thing that we could do is to tune the value of our process noise covariance  $Q_x$  to have a lower value. Lowering the  $Q_x$  matrix would reduce the response time, making our model more agile. Nonetheless, it comes with the tradeoff of reducing the accuracy of our model to follow the true reference value. This decision will lead to the depletion of our model's accuracy, especially on test cases with the correct SOC initialization (and also on other driving cycles and HPPC tests). Subsequently, it will increase the RMSE of those tests for far more. In any case, we considered that 50% offset of SOC initialization is quite a severe test case and we opted not to change the tuning but to accept this tradeoff as part of the design choice.

### EKF current offset test on HWGRADE 1

The EKF under HWGRADE1 performs distinctively compared to the other test. As shown in the lower plot of Figure 4.19 for 0.1 A offset and 4.20 for 0.1 A offset, the SOC error is bounded to 1% and circa 2%, respectively, ignoring the last part where the error is enlarging due to the discrete model's voltage error. Throughout the cycle, we obtain 1.41% and 1.32% of SOC RMSE for 0.1 A and 0.5 A offset in that order.



**Figure 4.19:** EKF result on HWGRADE 1 with 0.1 A current offset



**Figure 4.20:** EKF result on HWGRADE 1 with 0.5 A current offset

Concluding the section, we could say our tuned EKF model performs highly on HPPC and driving cycles, and under different test cases for the 25°C dataset. Apart from that, we also noticed how closely related the voltage error of the discrete model to the EKF performance in predicting SOC, since the discrete model’s voltage output acts as the foundation for the EKF to estimate the SOC. Considering the discrete model’s inaccuracy as noise, it is obvious that increasing the discrete model’s performance will surely also improve our EKF model due to minimized

system noise. What remains is to replicate the analysis on different cases and incorporate battery temperature to achieve better discrete model's performance across multiple temperature dataset.

## 4.3 Two-state Extended Kalman Filter

Following our workflow, the two-state Extended Kalman Filter is due to develop after completing the single-state counterpart validated over the driving cycles. As in the previous one-state EKF, we will start by developing the discrete model. Nevertheless, we will not be describing all the details, only the changes with respect to the one-state model.

### 4.3.1 Model Description

Instead of using a 1D gridded interpolant, we needed to use the *griddedInterpolant* function for 2 dimensions. The 2D griddedInterpolant input requires to create a grid of the breakpoints: SOC array and average temperature array from the HPPC average temperature; and subsequently casting it into griddedInterpolant as presented in the first two lines in Figure 4.21. As shown, two grids are needed since we have two different SOC levels, with 7 and 13 SOC levels indicated with `SOC_array1` and `SOC_array`, respectively. Figure 4.22 shows the code to call the parameters' value based on the SOC `SOC` and the battery temperature `T_batt` at a given instance.

```
% Create grid arrays using ndgrid
[SOC_grid, temp_grid] = ndgrid(SOC_array./100, avg_temp_list);
[SOC_grid1, temp_grid1] = ndgrid(SOC_array1./100, avg_temp_list);

% Making griddedtemp_gridInterpolants
R0_gridint = griddedInterpolant(SOC_grid, temp_grid, R0_grid,"linear","nearest");
R1_gridint = griddedInterpolant(SOC_grid1, temp_grid1, R1_grid,"linear","nearest");
C1_gridint = griddedInterpolant(SOC_grid1, temp_grid1, C1_grid,"linear","nearest");
R2_gridint = griddedInterpolant(SOC_grid1, temp_grid1, R2_grid,"linear","nearest");
C2_gridint = griddedInterpolant(SOC_grid1, temp_grid1, C2_grid,"linear","nearest");
OCV_gridint = griddedInterpolant(SOC_grid, temp_grid, OCV_grid,"linear","nearest");
```

**Figure 4.21:** Two-state *griddedInterpolant* on MATLAB

As soon as the model is set, the previous EKF parameters' tuning obtained from the 25°C needs to be validated over the rest of the temperature datasets. Unfortunately, even without going into the robustness tests, the previous tuning couldn't perform for the whole dataset, particularly for the lower temperatures dataset of the HWGRADE1 driving cycle. This means that the matrices ( $P_x$ ,  $Q_x$ , and  $R_x$ )

```

%% Main iteration loop of Kalman Filter
for i = 1:length(time_d)

    SOC = x_hat(1); %in fractional

    T_batt = T_batt_d(i);

    R0 = R0_gridint(SOC,T_batt);
    R1 = R1_gridint(SOC,T_batt);
    C1 = C1_gridint(SOC,T_batt);
    R2 = R2_gridint(SOC,T_batt);
    C2 = C2_gridint(SOC,T_batt);
    V_OCV = OCV_gridint(SOC,T_batt);

```

Figure 4.22: Calling the 2D griddedInterpolant

need to be redefined so that the EKF model works for all driving cycles on all of the temperature datasets. The main cause is due to the low ECM accuracy on the two lowest temperature datasets (-20°C and -10°C) as explained in Chapter 4. Figure 4.23 gives us an illustration of the previous tuning on -10°C HWGRADE1. As shown, the errors were accumulated and the EKF failed to follow the true reference indicated by 'Coulomb Counting' legend. Similar results occurred also on the -20°C HWGRADE1. The mission is to get the EKF working on the lowest two temperature datasets while maintaining the robustness and agility of the model.

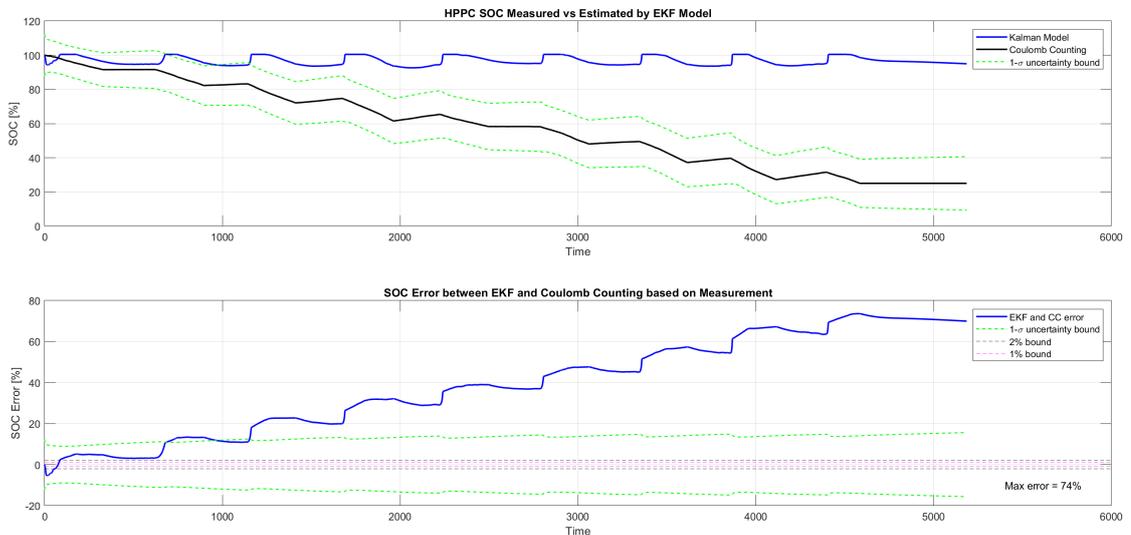


Figure 4.23: Old tuning 2D EKF result on -10°C HWGRADE1 driving cycle

The new tuned parameters are presented below:

$$P_x = \text{diag}(0.025, 0.01, 0.01)$$

$$Q_x = \begin{bmatrix} 1 \times 10^{-6} & 0 & 0 \\ 0 & 20 \times 10^{-5} & 0 \\ 0 & 0 & 20 \times 10^{-5} \end{bmatrix}$$

$$R_x = 0.05$$

The only parameter being re-tuned was the  $Q_x$  as we changed the order of magnitude of the first element of the process noise covariance, which corresponds to the SOC noise covariance, from  $10^{-5}$  to  $10^{-6}$ . What does lowering the  $Q_x$  significantly mean? Reducing it by a factor of 10 suggests that we rely more on the process model rather than on frequent measurement updates compared to the previous tuning. Most importantly, lowering the  $Q_x$  results in smoother estimates since the EKF is less responsive to fluctuations. As we will see in the results, the new tuning parameters will perform relatively poorer compared to the old ones on the robustness test, as it becomes less agile to interferences and noises. Despite that trade-off, this new tuning offers us a better performance at steady-state conditions and less sensitivity to the noises, which we commonly encounter in the temperatures and hysteresis voltages, especially on low-temperature datasets.

### 4.3.2 Two-state EKF Model Validation on Driving Cycles

In the validation process, similar test cases to the one-state are applied. Each of the following test cases are repeated on driving cycles (REORDERED1 and HWGRADE1 driving cycle) at each temperature datasets: [-20, -10, 0, 10, 25, 40] °C.

1. Simulation on the normal configuration with the correct initialization of the SOC (starting from 100%).
2. Simulation with the wrong initialization of the SOC: 80% and 0%.
3. Simulation with the offset value of current input  $I_{batt}$ .

For the sake of conciseness, the results are tabulated in Table 4.1 for the first two test cases and Table 4.2 for the third robustness test case, while only the plots of important scenarios are presented and discussed. Under the normal test case, we could clearly notice some great results for both REORDERED1 and HWGRADE1. As mentioned several times, the ECM accuracy will impact the EKF performance, hence explaining the reason of degrading EKF performance on both REORDERED1 and HWGRADE1 for the two lowest temperatures. The SOC RMSEs follow closely the RMSE of ECM displayed in Figure 3.34 and 3.33. Understanding the limit of the EKF performance on the two lowest temperatures, the EKF seems to present an exceptional RMSE on the other temperature datasets, none of which reach the

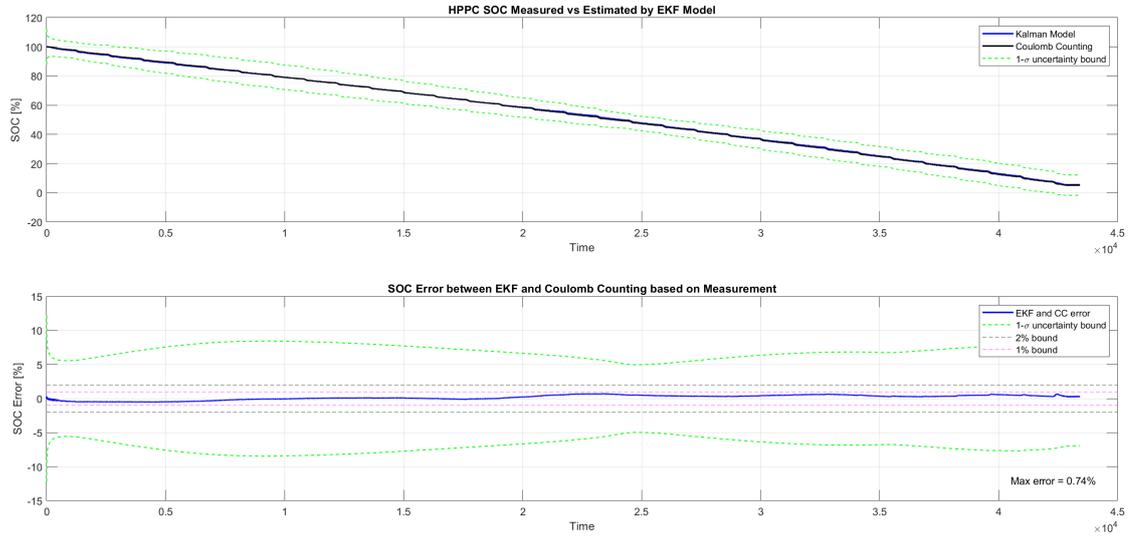
limit of 2% SOC. Khanum [13] tested their EKF solution on LA92 drive cycle under normal configuration and obtained the RMSE of 9.89%, 1.78%, 1.01%, 2.90%, and 1.94% for -10, 0, 10, 25, 40 °C. LA92 is one of the supplementary Federal Test Procedure (FTP) standard, which is one of the four driving cycles constituting the reordered driving cycles. This indicates that REORDERED1 is the closest comparison to LA92, in which our results show a much higher accuracy. The increase in SOC RMSE around the low temperature is also reported in Khanum’s work, affirming our previous finding.

**Table 4.1:** Temperature RMSE for REORDERED1 and HWGRADE in normal and wrong initialization test cases

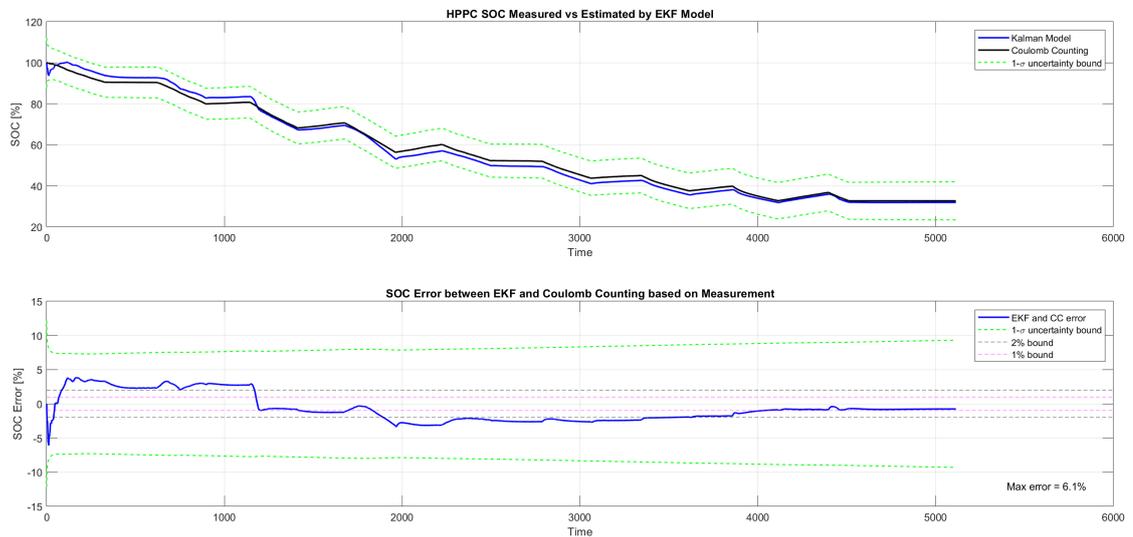
Temperature [°C]	REORDERED1 RMSE [%]	HWGRADE RMSE [%]
<b>Normal Configuration</b>		
-20	5.93	2.10
-10	4.01	3.54
0	1.84	0.70
10	1.33	0.96
25	0.38	0.89
40	0.78	1.68
<b>Wrong Initialization SOC = 80%</b>		
-20	11.03	3.67
-10	13.46	14.17
0	10.11	12.92
10	7.54	10.26
25	3.78	6.43
40	2.84	3.68
<b>Wrong Initialization SOC = 50%</b>		
-20	34.37 (F)	32.34 (F)
-10	24.64 (F)	29.40 (F)
0	16.16	22.01
10	12.19	16.47
25	7.11	11.94
40	5.68	6.73

Wassilidis [11] tuned a single-state dual-extended Kalman Filter (DEKF) under a logged real dynamics driving cycle profile with 20°C test temperature. Even though the temperature effect is not covered in their analysis, under 100% SOH, the DEKF resulted in 0.5% of RMSE, comparable to our results. Figure 4.24 presents the best-achieving scenario for our two-state EKF model (under REORDERED1

25°C) with RMSE of 0.38% and maximum absolute SOC error of 0.74%. Notice the accurate RMSE is a product of good tuning and most importantly, an accurate ECM model as we achieved only 8.2 mV RMSE under this scenario (Figure 3.33). These comparisons further assert our positive outcome of high performing EKF model.

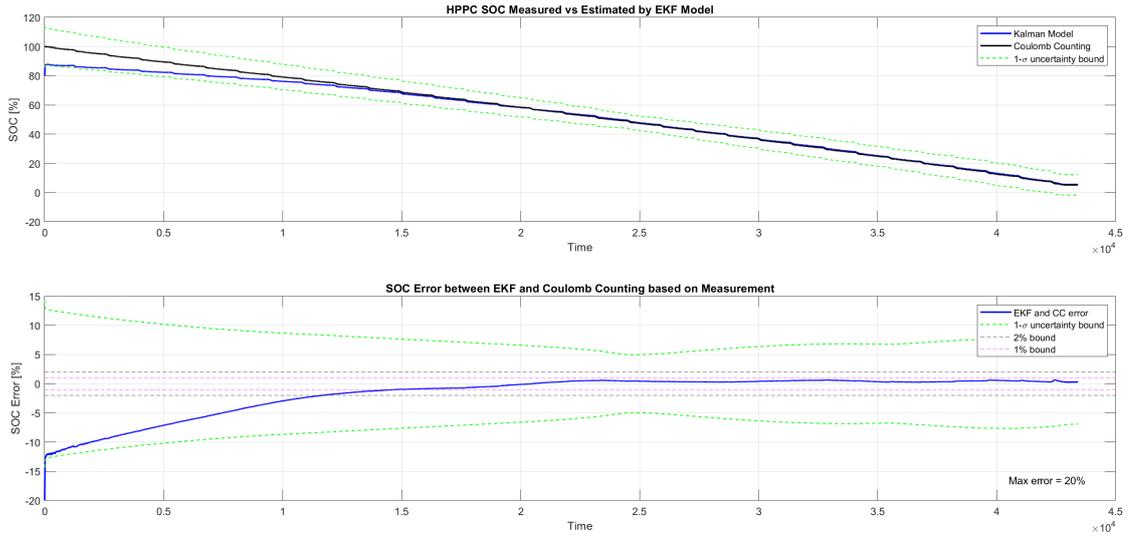


**Figure 4.24:** Best scenario for EKF model under REORDERED1 25°C with 0.38% RMSE



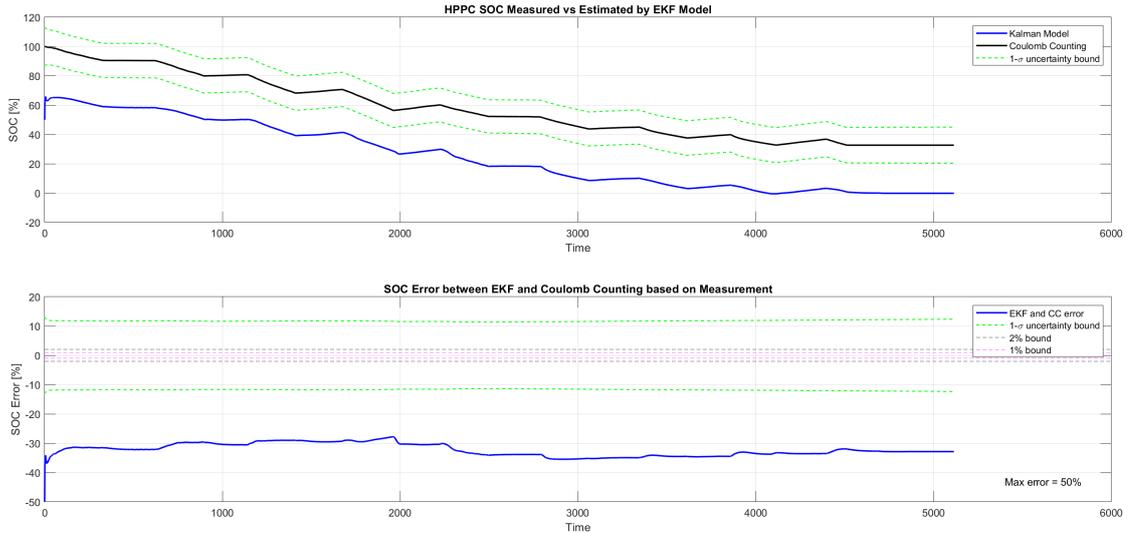
**Figure 4.25:** Challenging scenario for EKF model under HWGRADE -20°C due to low ECM accuracy. SOC RMSE = 2.10%

On the contrary, Figure 4.25 depicts the struggle of our EKF model for  $-20^{\circ}\text{C}$  HWGRADE scenario given the ECM couldn't provide a good fit to the measurement voltage reference. Nonetheless, we could highlight an important property of EKF. Notice that the EKF tries to stabilize the error shown by the plateauing RMSE to zero. It takes a significantly longer time for the EKF due to the higher process (or modeling) error from the ECM.



**Figure 4.26:** Agile EKF performance under  $25^{\circ}\text{C}$  REORDERED1 with 80% initial SOC

The two-state EKF model was tested also with some wrong SOC initialization. In general, the model is capable of handling the 80% SOC initialization in both driving cycles and all temperature datasets. Figure 4.26 depicts an agile behavior of our EKF as it closes down the initial SOC error from 20% to under 1% SOC for REORDERED1  $25^{\circ}\text{C}$  case. In the 50% initialization, however, the EKF fails at the low temperatures for both cycles. Failing EKF is indicated by the incapability of the model to close down the initial error during the cycle as illustrated in Figure 4.27 where the SOC error remains constant throughout the driving cycle. Furthermore, affirming our previous finding, the failing scenarios and the highest RMSE, in general, could be found in lower temperature scenarios, corresponding to the increase in the RMSE of ECM. We understand that accuracy and robustness are always part of design trade-off. Considering that 50% error of SOC initialization is a massive number, we could accept the shortcoming for a fraction of the test temperatures scenarios.



**Figure 4.27:** Failing EKF under HWGRADE1 -20°C dataset with 50% initial SOC

### 4.3.3 Two-state EKF Model Robustness Test on Driving Cycles

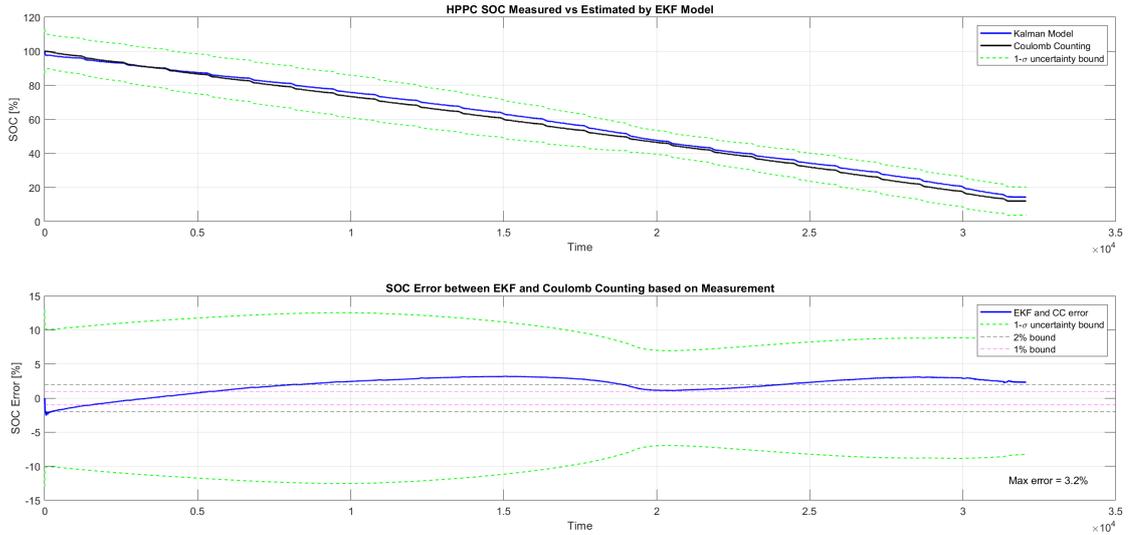
During its lifetime, the EKF might undergo some unintended use cases related to the credibility of the model’s input and the reliability of the model itself. In these critical yet reasonable cases, the model shall still perform at an acceptable level. One of the common methods to test the robustness of the model is by simulating an input current offset, as reported by Khanum et. al. [13] and Kollmeyer et. al. [14], similar to what we did in the one-state model validation. Table 4.2 summarized the robustness test performed using 0.1 and 0.5 A current input offset to our EKF model for both driving cycles and multiple temperatures.

Under 0.1 A current offset, all scenarios resulted in an acceptable RMSE. Figure 4.28 depicts the EKF result on REORDERED1 under 10°C dataset. On the lower plot, one could argue that the maximum error is considerably high (3.2%) and that the EKF is not capable to eliminate completely the error at the end. Nonetheless, the error is bounded. It grows up until the maximum error, shrinks to 1%, grows again to reach the maximum error before going back to 2%. The second error increase is caused by the degrading performance of the discrete ECM at low SOC, as we discussed in the one-state EKF model. The model, however, could still tolerate the sensor error and perform at an acceptable level.

On the other hand, most of the tests for 0.5 A current offset ended up in error, particularly for the REORDERED1. As shown in Figure 4.29, the EKF model follows the coulomb counting offset. In the newly tuned parameters, we assigned

**Table 4.2:** Temperature RMSE for REORDERED1 and HWGRADE under Current Offset Tests

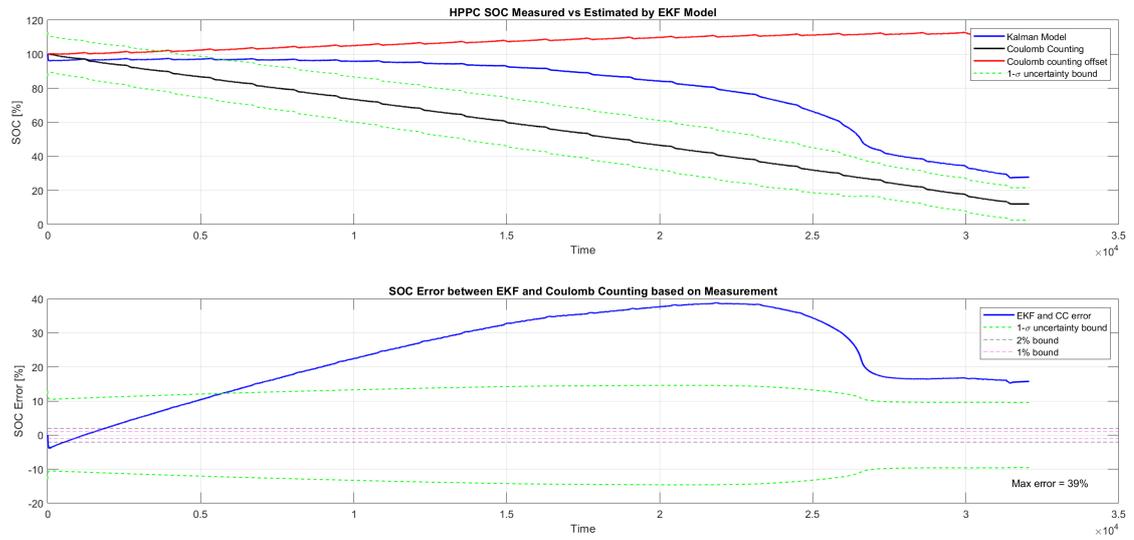
Temperature [°C]	REORDERED1 RMSE [%]	HWGRADE RMSE [%]
<b>Current Offset Test 0.1 A</b>		
-20	3.59	1.87
-10	2.86	5.04
0	2.85	1.26
10	2.26	1.13
25	2.43	1.63
40	2.49	2.47
<b>Current Offset Test 0.5 A</b>		
-20	FAIL	FAIL
-10	FAIL	4.02
0	FAIL	4.42
10	FAIL	4.21
25	11.50	5.16
40	9.16	6.40



**Figure 4.28:** Two-state EKF performance under 10°C REORDERED1 with 0.1 A current offset

quite a low value of SOC noise covariance component of  $Q_x$ , which means that we give more confidence level on the model rather than the measurement feedback update. Consequently, the time needed for the EKF model to realize that it has

gotten the wrong track is significantly long. Around the  $2.5 \times 10^4$  second time mark, the EKF starts to correct itself to follow the true reference. Had the ECM model been more accurate, this problem could easily be solved by reducing the SOC noise covariance and allowing the feedback to promptly correct the model. Nonetheless, such a level of current offset is considerably a severe case (Khanum tested only for 0.1 A). Given the excellent result on 0.1 A, a fair result on half of the 0.5 A, and our ECM model limitation causing this failure, we argue that our model has an acceptable robustness test. The conclusion marks our discussion on two-state Extended Kalman Filter. Further discussion regarding alternative solutions to tackle our hindrance will be discussed in the next chapter.



**Figure 4.29:** Failing two-state EKF robustness test for 10°C REORDERED1 with 0.5 A current offset

## Chapter 5

# Attempts to enhance the ECM performance at low temperatures

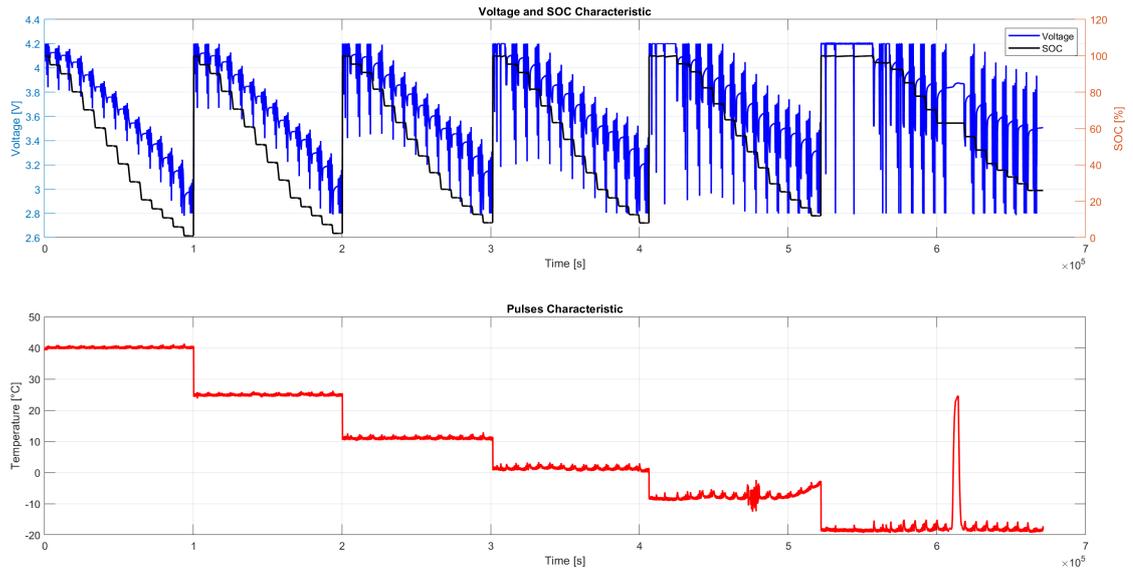
Up until this point, it is somewhat clear that the main issue of degrading two-state EKF performance is mainly caused by its incapability of the equivalent circuit model to perform accurately, particularly at low temperatures. This chapter is designed to present some of the efforts made to tackle the problem, along with the outcome of each attempt. Interested readers could take these trials into consideration to avoid re-inventing the wheel and start immediately with the most significant solution. Mainly, there are four solutions explored:

1. Tuning ECM parameters directly on the two-state model with concatenated HPPC test from all temperatures.
2. Tuning ECM parameters directly on the two-state model with concatenated driving cycles from all temperatures.
3. Casting the OCV-SOC curve directly to the DP ECM without Parameter Estimation.
4. Modeling the ECM hysteresis

### 5.1 Tuning ECM parameters on concatenated HPPC test

In our previous workflow presented in Figure 2.3, the parameter estimation processes using Simulink were conducted on one-state ECM on each temperature, before grids

were made to combine those parameters. The idea of this trial is to use the two-state model, which takes into account the real-time battery temperature, as the base of our parameter estimation. This implies that the HPPC test must be concatenated throughout the temperatures, allowing the Simulink Parameter Estimation to configure the most optimum parameter value within the imposed range. First, using a simple MATLAB code, a single HPPC test containing all test temperature datasets in a random order was concatenated. Figure 5.1 depicts the concatenated HPPC data. In total, 4 quantities are combined from all six temperature datasets: measured SOC, voltage, current, and battery temperatures.



**Figure 5.1:** Concatenated HPPC test

With an equivalent procedure to the single-state ECM model, the optimized parameters from Simulink Parameter Estimation are obtained in grid forms. Figure 5.2 and 5.3 depicts the  $OCV - R_0$  and RC pairs optimized profiles, respectively. The  $OCV$ ,  $R_0$ , and capacitors are similar to the Figure 3.21 to 3.24. Notice that due to the concatenated dataset, the missing SOC values present in some of the datasets are compensated by another dataset, hence, explaining some differences at the low SOC for these parameters. This finding concludes that the single-temperature HPPC datasets are representative enough (concatenating them is not necessary) to the whole characteristic of the battery model. On the other hand, resistors in the RC pairs are quite different compared to the old parameters, particularly we can no longer appreciate the trend of increasing resistor value with decreasing temperature.

These parameters are validated on HWGRADE1 and REORDERED1 as in the previous chapters. The voltage RMSE results are summarized in Figure 5.4 and

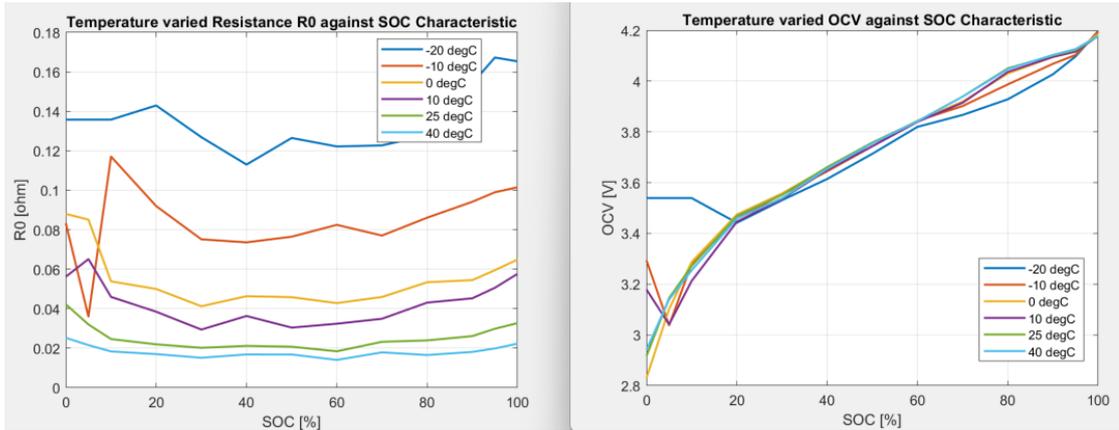


Figure 5.2: Optimized OCV and  $R_0$  from HPPC tests concatenated

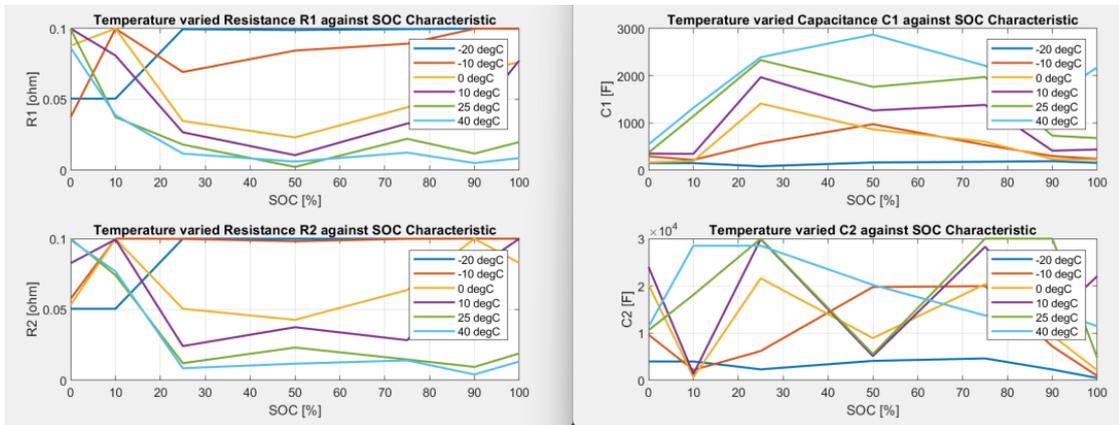


Figure 5.3: Optimized RC pairs parameters from HPPC tests concatenated

5.5 and are compared to the previous result as the benchmark. We could clearly appreciate that this brute-force approach didn't enhance the model's RMSE. We knew the main problem was in the hysteresis around the low temperatures. In this approach, we thought of compensating the hysteresis found in the low-temperature dataset with another dataset with the same temperature. Parameter estimation will optimize the parameter's value of the same SOC point and same temperature (since we also cast the temperature into the model) from the concatenated HPPC tests. Nonetheless, the temperatures in HPPC are mostly constant, which might be the reason for the degrading performance since the hysteresis is still uncaptured by the parameter estimation. It is the temperature noises and disturbances (which are not valid) are the ones being captured and thus estimated, explaining the worsening result we got. That being said, the next idea is to use a test dataset in which the temperatures fluctuate significantly, which is found in the driving cycles.

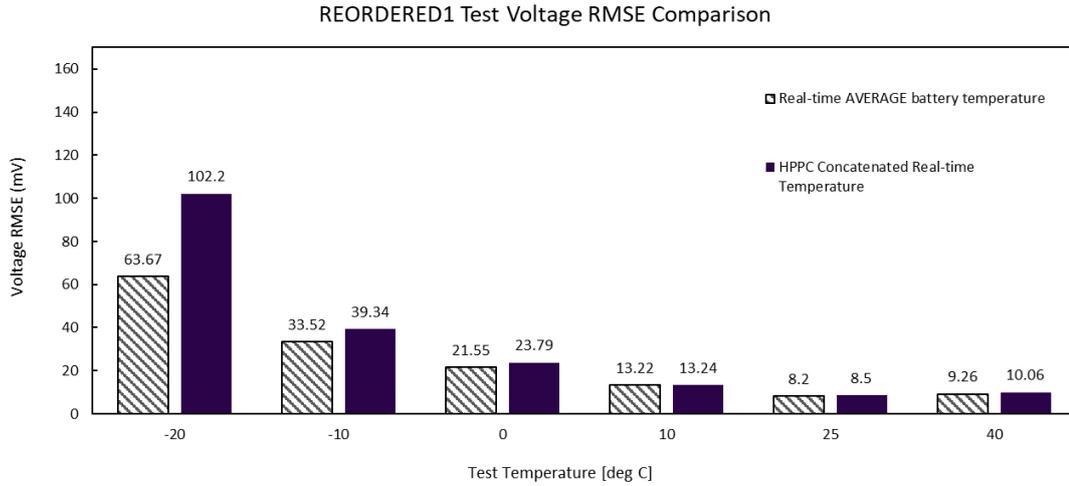


Figure 5.4: Voltage RMSE for concatenated HPPC tuning on REORDERED1

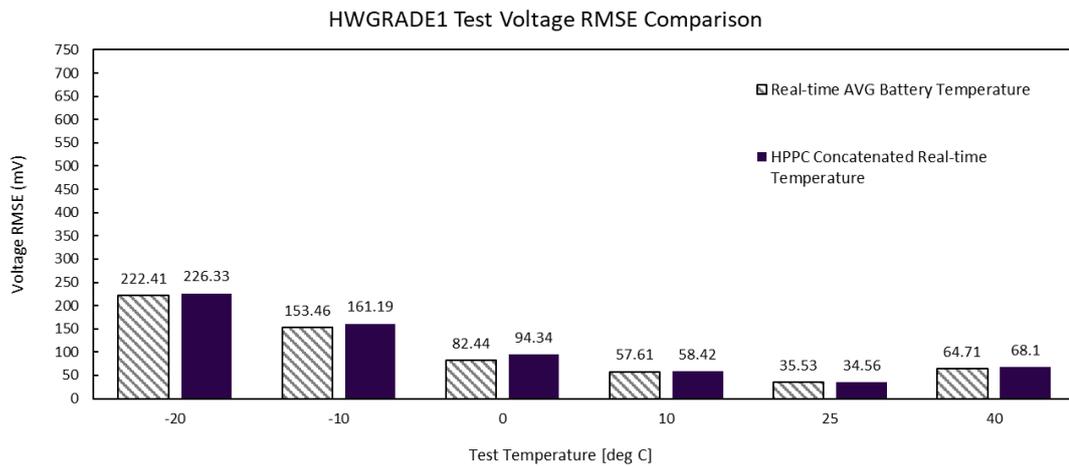
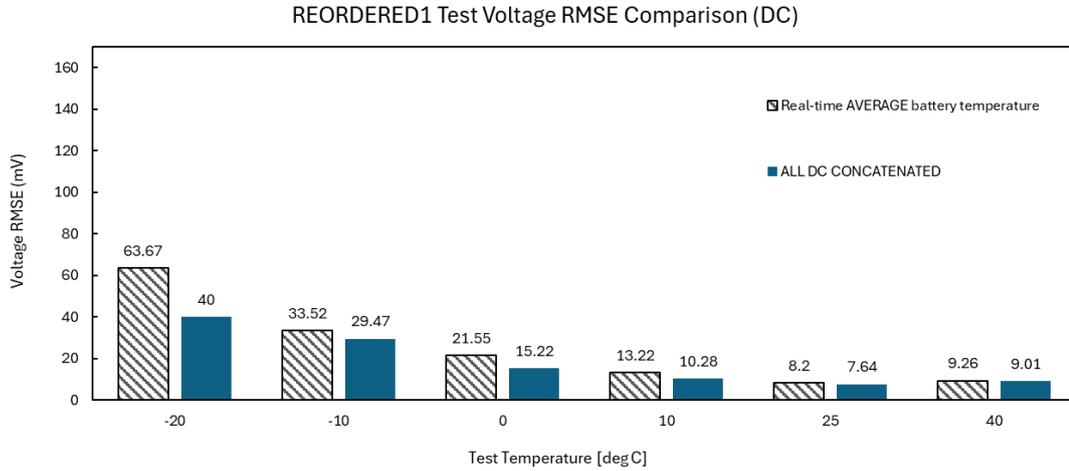


Figure 5.5: Voltage RMSE for concatenated HPPC tuning on HWGRADE1

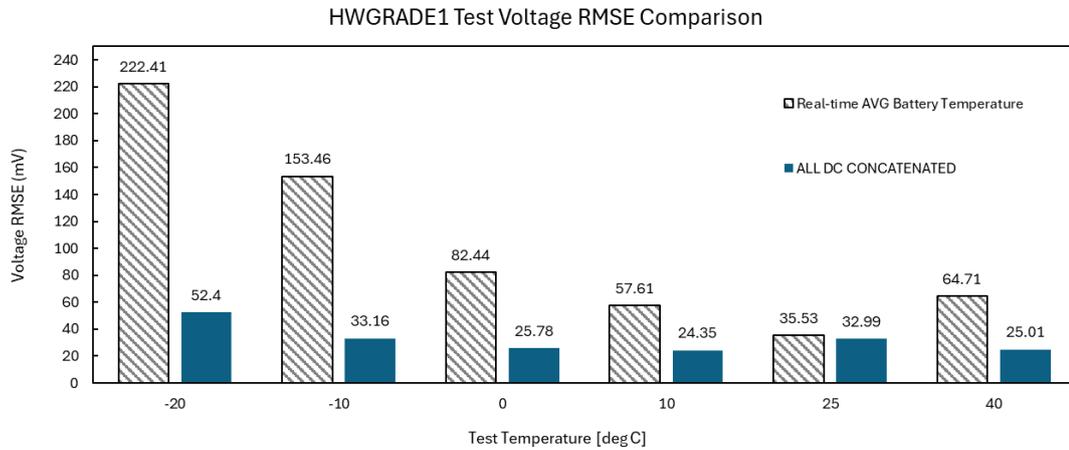
## 5.2 Tuning ECM parameters on concatenated Driving Cycle test

Similar to the previous section, the REORDERED1 and HWGRADE1 from all temperatures are concatenated into one, and out of this data, the parameter estimation was performed. The method is considered as not a scientific approach since the parameter estimation is commonly conducted on the characterization test such as the HPPC test. Whatever our result is, it won't be a global solution.

Nevertheless, we argue that this attempt is still useful in providing us with the best possible value if parameter estimation and model validation are both performed on the driving cycle.



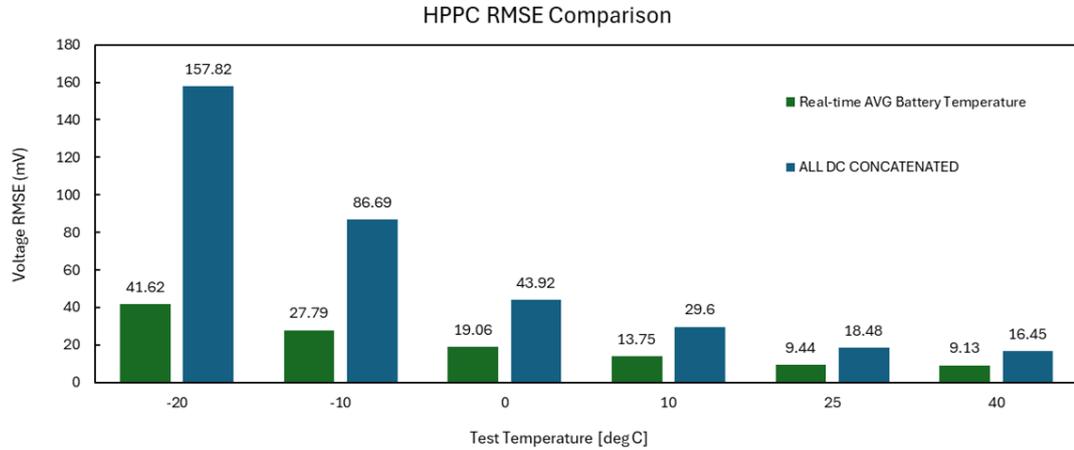
**Figure 5.6:** Voltage RMSE for concatenated driving cycles tuning on REORDERED1



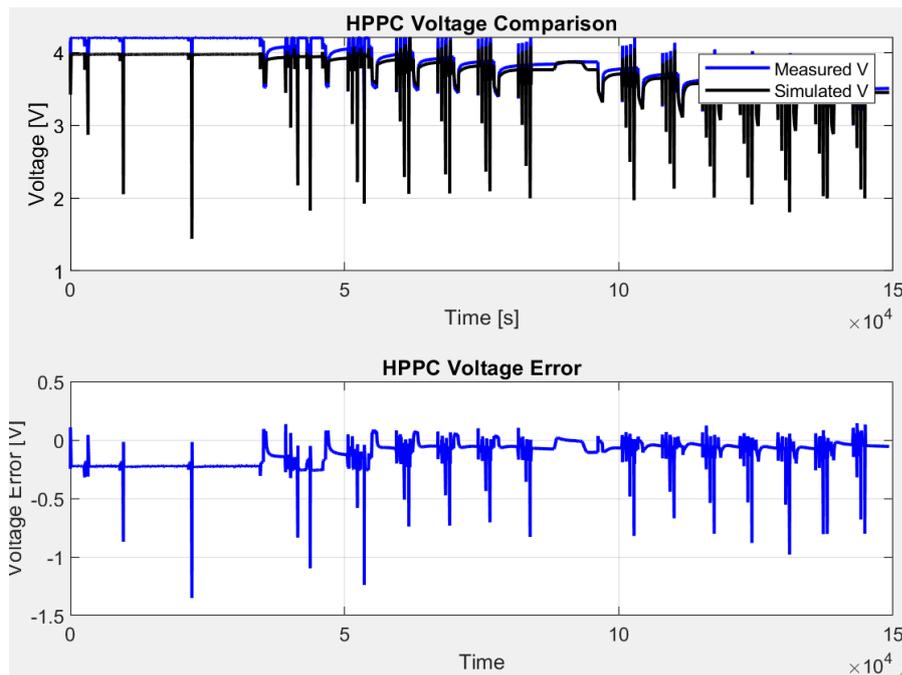
**Figure 5.7:** Voltage RMSE for concatenated driving cycles tuning on HWGRADE1

Without going into the details (due to similar procedures), we could now analyze the results on REORDERED1 and HWGRADE1 presented in Figure 5.6 and 5.7. It is evident that the RMSE would be lower on the REORDERED1 and HWGRADE1 since the parameters are extracted out of these driving cycles, so this finding is expected. The HWGRADE1 seemed to have benefited significantly

from this approach as we cut down the RMSE to only 52.4 mV, which gives us an idea of the expected order of magnitude of a good HWGRADE1 RMSE.



**Figure 5.8:** Voltage RMSE for concatenated driving cycles tuning on HPPC



**Figure 5.9:** Huge voltage offset on HPPC test with model using the parameters extracted from concatenated driving cycles

The true validation occurs when we compare the result between the HPPC test

for this approach and the old parameters' results which are extracted from HPPC. Figure 5.8 presents the RMSE summary on HPPC. We notice immediately the degradation of accuracy on the HPPC. The hysteresis around the low temperature again is showing in the graph as the RMSE is way worse on this. By analyzing each waveform as depicted in Figure 5.9, we found another problem correlated to the hysteresis. The driving cycles seem to have a lower OCV, which is proven by the underestimation of simulated voltage compared to the measured voltage during zero current (early stabilization phase in Figure 5.9). This could be another sign of hysteresis, or it could simply be a sign of aging since the same battery cell is tested in series starting from the HPPC test followed by the driving cycles. To be certain of a definitive cause, the hysteresis or the state-of-health (SOH) should be modeled. Any solution that could significantly improve or even eliminate this discrepancy will be our answer. Considering that such work is out of the scope of this thesis, the author invites future collaborators to contribute to developing the existing model.

### 5.3 Direct use of the OCV-SOC curve

When the battery is discharged or charged very slowly, the contribution of diffusion voltage and the ohmic internal losses (caused by  $R_0$ ) are very small [5]. This test is equal to the discharge capacitance test described in Chapter 4 as the initial estimation of the OCV profile before casting it into Simulink Parameter Estimation. Meanwhile, during the parameter estimation, the solver will adjust this profile to fit the HPPC voltage profile, which has a considerably higher current demand. We hypothesized that the parameter estimation process causes the offset we saw earlier in Figure 5.9 (the discussion in the previous sections was concluded after all of the trial ended, hence, during this effort of using direct OCV-SOC, we didn't know the exact cause of voltage offset). In this attempt, the OCV-SOC relationship profile was not cast into the Simulink Parameter Estimation (SPE). The rest of the parameters were undergoing the same procedure for SPE similar to the one described in Chapter 4. Instead, the OCV-SOC profile is calculated by:

$$OCV(SOC) = V_{C/40}(SOC) - R_0(SOC) * C/40 \quad (5.1)$$

with  $V_{C/40}(SOC)$  is the OCV profile derived from capacitance test,  $R_0(SOC)$  is the internal resistance profile, and  $C/40$  is the C-rate. The result of this calculation is presented in Figure 5.10. One can immediately notice the higher value of OCV throughout all the SOC ranges although the trendline still follows the original parameter (from SPE). The rest of the parameters are indifferent to the original parameter, hence, will not be presented again in this section.

For succinctness purposes, we will only analyze REORDERED1 and HW-GRADE1 under  $-20^\circ\text{C}$  as this temperature represents the worst dataset, as depicted

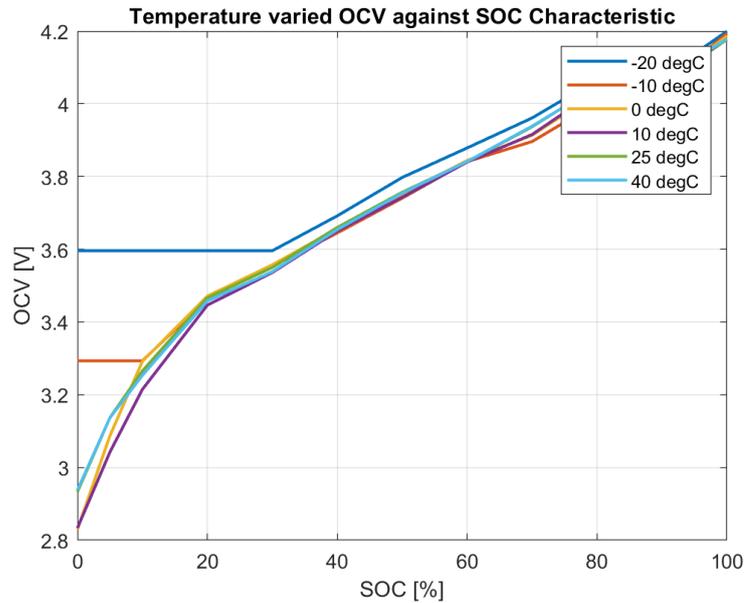


Figure 5.10: OCV-SOC profile calculated using Equation 5.1

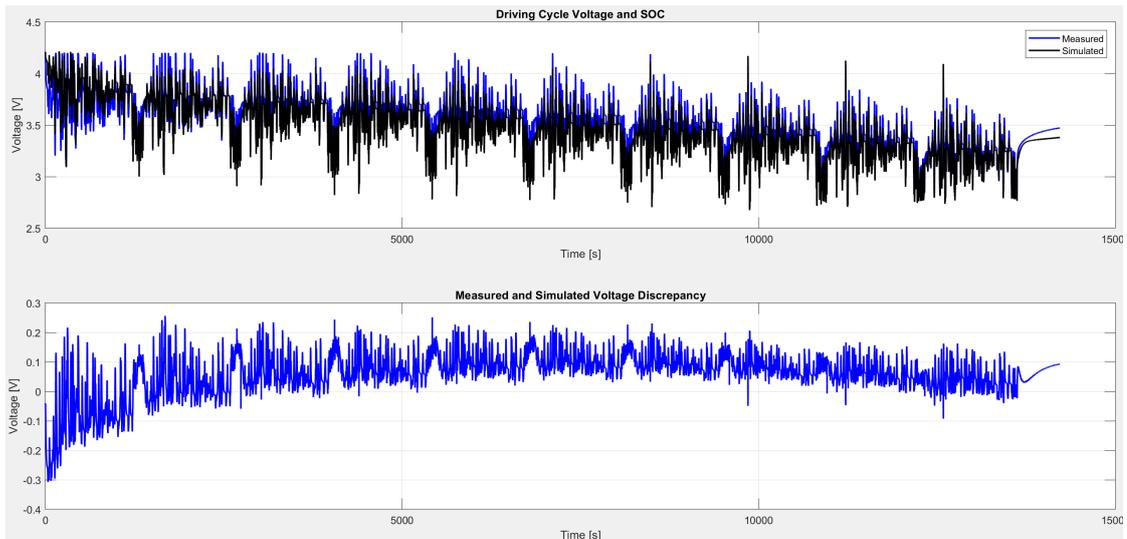
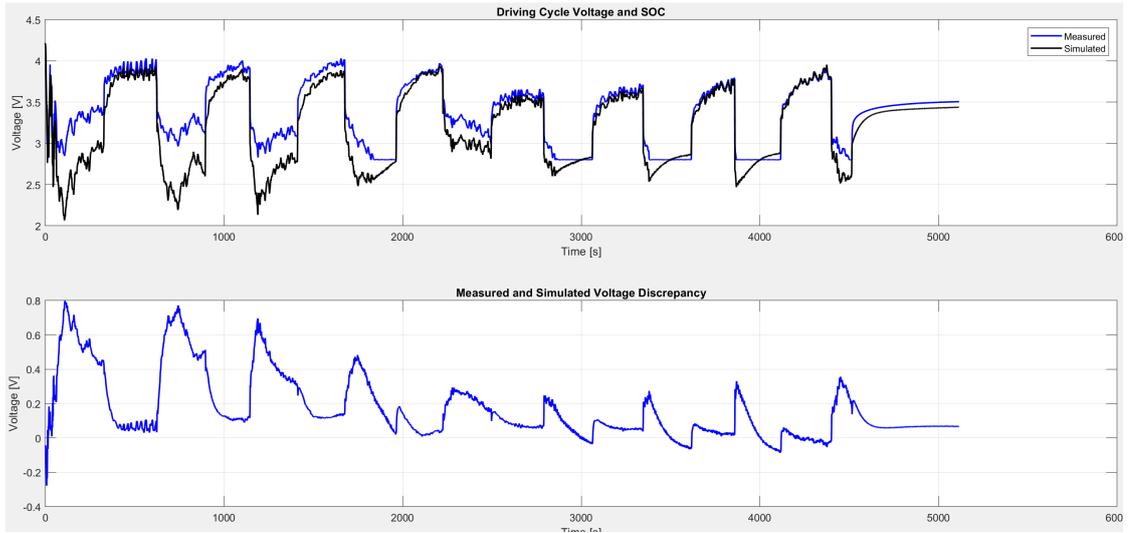


Figure 5.11: REORDERED1 -20°C result for direct OCV usage attempt

in Figure 5.11 and 5.12, respectively. In both of the attempts, the RMSE worsened by 33 mV to 91.4 mV for the REORDERED1 and by 28 mV for the HWGRADE1. Looking at the REORDERED1, we could understand why. The errors are no longer fluctuating around 0 as was the case with our normal parameters (recall



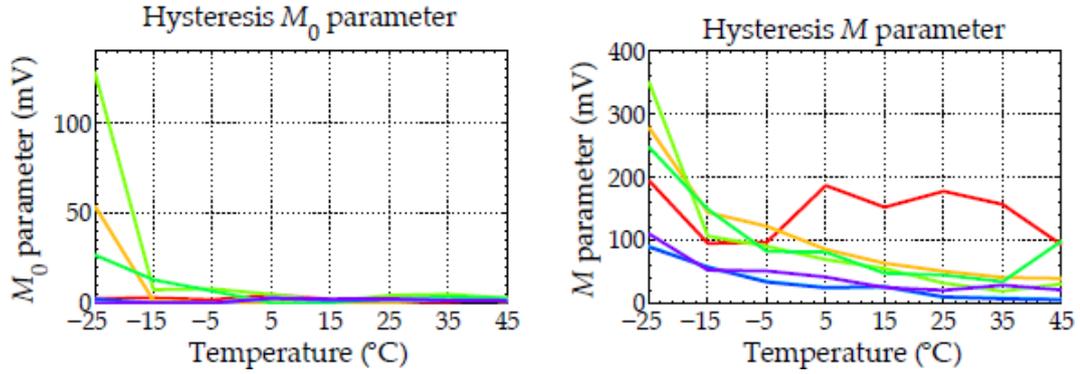
**Figure 5.12:** HWGRADE1 -20°C result for direct OCV usage attempt

that our original model has fluctuations due to the uncaptured dynamics nature, while in a steady state, the result is decent with minimized error). Looks like the calculated OCV profile is underestimating the measured voltage reference despite the additional  $R_0(SOC) * C/40$  component already accounted for inside the calculation, meaning there is another factor contributing to this difference between the capacitance test and the driving cycle. This effort is thus abandoned due to insufficient and unpromising results.

## 5.4 Hysteresis Voltage Modeling

The conclusion of the thesis ultimately comes to the need for hysteresis voltage modeling to improve the ECM model accuracy at low temperatures, thus allowing the EKF to perform accurately and robustly under multiple circumstances and use cases. The modeling of hysteresis in DP ECM, however, is out of the scope of this work. This section will only lay the foundation of how the hysteresis model might be implementable in real-time scenarios for the ECM and most importantly, the Extended Kalman Filter. hopefully sparking the interest of the reader to continue improving this existing model. Plett [5] and Bodnar [31] mentioned that voltage hysteresis speeds up at lower temperatures and decreases in magnitude as temperature increases as illustrated in Figure 5.13. Readers are referred to Plett's and Rzepka's [7] work for the methodology to retrieve these hysteresis parameters and its implementation on the Simulink.

Bodnar's work revolves around the enhancement model's accuracy at lower



**Figure 5.13:** SOC-averaged hysteresis as a function of temperature. Retrieved from Plett [5]

temperatures by modeling temperature temperature-dependent hysteresis model, which is identical to what we needed. It is reported in the work a promising result as the calculated RMSE on INR18650 batteries for five multiple temperatures as summarized in the following table.

**Table 5.1:** RMSE Comparison of Base Model and Model with Hysteresis by Bodnar [31]

T [°C]	RMSE of base model [mV]	RMSE of model with hysteresis [mV]	Improvement [%]
40	0.0165	0.0081	50.91
30	0.0198	0.0111	43.94
20	0.0264	0.0095	64.02
10	0.0342	0.0096	71.93
0	0.0468	0.0107	77.14

The improvements are significant throughout the temperature dataset by over 50%. Moreover, the amount of benefit is bigger as the temperature is lower, showing a huge advantage of low-temperature usage. This finding becomes an interesting topic to be explored and sets a new development direction for the existing model.

## Chapter 6

# Conclusion and Suggestion

The work could be categorized into two major milestones: the development of two-state dual-polarization equivalent circuit model and on top of it, the Extended Kalman Filter. Both ECM and EKF models are validated on a randomized standard driving cycle (REORDERED1) and highway-mountainous driving cycles (HWGRADE1) with multiple temperature datasets: [-20, -10, 0, 10, 25, 40] °C. In the first milestone, model parameters were extracted from characterization tests such as HPPC and capacitance tests. The Simulink Parameter Estimation tool is then used to optimize these parameters to fit the measured data. These processes were iterated throughout all temperature datasets, forming grid profiles for all parameters. Enhancing a dual-polarization RC equivalent circuit model by incorporating the effect of temperature (two-state model) demonstrates a significant increase in ECM accuracy (up to 66.3% on the HWGRADE1 driving cycle). During the validation phase, our model showed promising results, particularly in predicting the output voltage on higher temperatures, with the best-case scenario RMSE to the measured value of 8.2 mV and 35.33 mV for REORDERED1 and HWGRADE1, respectively, of which both were found on the 25°C dataset. On the other hand, the results for low temperatures (-20 and -10 °C) have significant inaccuracy, indicated by the surging RMSE of 63.67 mV for REORDERED1 and 222.41 mV for HWGRADE1. We figured out that the lack of accuracy in lower temperatures is hypothetically caused by the prominent voltage hysteresis around low temperatures as reported by multiple journals [5, 7, 31].

In the second milestone, a temperature- and SOC-dependent Extended Kalman Filter was established on the basis of the equivalent circuit model. The aim is to develop an accurate and robust EKF, capable of operating in multiple temperatures. The proposed EKF-based SOC estimation achieves excellent accuracy and robustness at higher temperatures with an acceptable limited error band. The EKF, however, couldn't perform adequately at low temperatures (-20°C and -10°C) due to the increasing hysteresis phenomenon which is uncaptured by the ECM

model. Several solutions are proposed to tackle the problem, leaving the work open for further improvement.

The author strongly suggests exploring the possibility of incorporating the hysteresis model into EKF to improve its accuracy around low temperatures. Such a solution is possible by adding one more state to our EKF model accounting of this hysteresis parameter and subsequently, testing the model on the same driving cycle. Another possibility is by having a different EKF tuning for multiple temperatures, which allows our model to adapt its covariance matrices depending on the accuracy of our model. By doing so, we can assign a higher trust in the model for datasets with higher ECM accuracy and vice versa.

# Bibliography

- [1] IEA. *Trends in electric vehicle batteries – Global EV Outlook 2024 – Analysis*. Paris: IEA, 2024. URL: <https://www.iea.org/reports/global-ev-outlook-2024/trends-in-electric-vehicle-batteries> (cit. on p. 1).
- [2] Automotive Cells Co. *Analyzing the Cost of an Electric Vehicle Battery Cell*. Mar. 2024. URL: <https://www.acc-emotion.com/stories/analyzing-cost-electric-vehicle-battery-cell> (cit. on p. 1).
- [3] Zhenjie Cui, Weihao Hu, Guozhou Zhang, Zhenyuan Zhang, and Zhe Chen. «An extended Kalman filter based SOC estimation method for Li-ion battery». In: *Energy Reports* 8 (2022). ICPE 2021 - The 2nd International Conference on Power Engineering, pp. 81–87. ISSN: 2352-4847. DOI: <https://doi.org/10.1016/j.egyр.2022.02.116> (cit. on p. 1).
- [4] Shuo Pang, Jay Farrell, Jie Du, and Matthew Barth. «Battery state-of-charge estimation». In: vol. 2. Feb. 2001, 1644–1649 vol.2. DOI: [10.1109/ACC.2001.945964](https://doi.org/10.1109/ACC.2001.945964) (cit. on p. 2).
- [5] G.L. Plett. *Battery Management Systems, Volume I: Battery Modeling*. Artech House power engineering series. Artech House, 2015. ISBN: 9781630810245 (cit. on pp. 2, 12, 14, 18, 19, 21, 30, 33, 34, 45, 46, 51, 81, 83–85).
- [6] 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». In: *Applied Sciences* 8.5 (2018). ISSN: 2076-3417. DOI: [10.3390/app8050659](https://doi.org/10.3390/app8050659) (cit. on p. 2).
- [7] Benedikt Rzepka, Simon Bischof, and Thomas Blank. «Implementing an Extended Kalman Filter for SoC Estimation of a Li-Ion Battery with Hysteresis: A Step-by-Step Guide». In: *Energies* 14.13 (2021). ISSN: 1996-1073. DOI: [10.3390/en14133733](https://doi.org/10.3390/en14133733) (cit. on pp. 2, 83, 85).
- [8] G.L. Plett. *Battery Management Systems, Volume II: Equivalent-Circuit Methods*. Artech House power engineering and power electronics. Artech House, 2015. ISBN: 9781630810283. URL: <https://books.google.it/books?id=1kSPCwAAQBAJ> (cit. on pp. 2, 54).

- [9] Zheng Chen, Yuhong Fu, and Chunting Chris Mi. «State of Charge Estimation of Lithium-Ion Batteries in Electric Drive Vehicles Using Extended Kalman Filtering». In: *IEEE Transactions on Vehicular Technology* 62.3 (2013), pp. 1020–1030. DOI: 10.1109/TVT.2012.2235474 (cit. on p. 2).
- [10] H.S. Ramadan, M. Becherif, and F. Claude. «Extended kalman filter for accurate state of charge estimation of lithium-based batteries: a comparative analysis». In: *International Journal of Hydrogen Energy* 42.48 (2017), pp. 29033–29046. ISSN: 0360-3199. DOI: <https://doi.org/10.1016/j.ijhydene.2017.07.219>. URL: <https://www.sciencedirect.com/science/article/pii/S0360319917331312> (cit. on p. 2).
- [11] Nikolaos Wassiliadis, Jörn Adermann, Alexander Frericks, Mikhail Pak, Christoph Reiter, Boris Lohmann, and Markus Lienkamp. «Revisiting the dual extended Kalman filter for battery state-of-charge and state-of-health estimation: A use-case life cycle analysis». In: *Journal of Energy Storage* 19 (2018), pp. 73–87. ISSN: 2352-152X. DOI: <https://doi.org/10.1016/j.est.2018.07.006>. URL: <https://www.sciencedirect.com/science/article/pii/S2352152X18301786> (cit. on pp. 2, 18, 21, 34, 69).
- [12] 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)*. 2014, pp. 1–8. DOI: 10.1109/IEVC.2014.7056126 (cit. on p. 2).
- [13] Fauzia Khanum, Eduardo Louback, Federico Duperly, Colleen Jenkins, Phillip J. Kollmeyer, and Ali Emadi. «A Kalman Filter Based Battery State of Charge Estimation MATLAB Function». In: *2021 IEEE Transportation Electrification Conference and Expo (ITEC)*. 2021, pp. 484–489. DOI: 10.1109/ITEC51675.2021.9490163 (cit. on pp. 2, 54, 69, 72).
- [14] Phillip J. Kollmeyer, Mina Naguib, Fauzia Khanum, and Ali Emadi. «A Blind Modeling Tool for Standardized Evaluation of Battery State of Charge Estimation Algorithms». In: *2022 IEEE Transportation Electrification Conference and Expo (ITEC)*. 2022, pp. 243–248. DOI: 10.1109/ITEC53557.2022.9813996 (cit. on pp. 2, 4, 7, 72).
- [15] Matthieu Dubarry and George Baure. «Perspective on Commercial Li-ion Battery Testing, Best Practices for Simple and Effective Protocols». In: *Electronics* 9 (Jan. 2020), p. 1. DOI: 10.3390/electronics9010152 (cit. on p. 5).
- [16] Hongwen He, Xiaowei Zhang, Rui Xiong, Yongli Xu, and Hongqiang Guo. «Online model-based estimation of state-of-charge and open-circuit voltage of lithium-ion batteries in electric vehicles». In: *Energy* 39.1 (2012). Sustainable

- Energy and Environmental Protection 2010, pp. 310–318. ISSN: 0360-5442. DOI: <https://doi.org/10.1016/j.energy.2012.01.009> (cit. on p. 11).
- [17] Cheng Zhang, Kang Li, Lei Pei, and Chunbo Zhu. «An integrated approach for real-time model-based state-of-charge estimation of lithium-ion batteries». In: *Journal of Power Sources* 283 (2015), pp. 24–36. ISSN: 0378-7753. DOI: <https://doi.org/10.1016/j.jpowsour.2015.02.099>. URL: <https://www.sciencedirect.com/science/article/pii/S0378775315003444> (cit. on p. 11).
- [18] Davide Cittanti, Alessandro Ferraris, Andrea Airale, Sabina Fiorot, Santo Scavuzzo, and Massimiliana Carello. «Modeling Li-ion batteries for automotive application: A trade-off between accuracy and complexity». In: *2017 International Conference of Electrical and Electronic Technologies for Automotive*. 2017, pp. 1–8. DOI: 10.23919/EETA.2017.7993213 (cit. on pp. 11–13, 18, 21).
- [19] Long Lam, P. Bauer, and Erik Kelder. «A practical circuit-based model for Li-ion battery cells in electric vehicle applications». In: *Telecommunications Energy Conference (INTELEC), 2011 IEEE 33rd International* (Oct. 2011), pp. 1–9. DOI: 10.1109/INTLEC.2011.6099803 (cit. on p. 12).
- [20] Gianluca Aurilio, Daniele Gallo, Carmine Landi, Mario Luiso, Aniello Rosano, Marco Landi, and Vincenzo Paciello. «A battery equivalent-circuit model and an advanced technique for parameter estimation». In: *Conference Record - IEEE Instrumentation and Measurement Technology Conference 2015* (July 2015), pp. 1705–1710. DOI: 10.1109/I2MTC.2015.7151537 (cit. on p. 12).
- [21] Sara Ha, Gabriele Pozzato, and Simona Onori. «Electrochemical characterization tools for lithium-ion batteries». In: *Journal of Solid State Electrochemistry* 28.3 (2024), pp. 1131–1157 (cit. on pp. 14, 16, 18).
- [22] Anup Barai, Kotub Uddin, W.D. Widanage, Andrew McGordon, and Paul Jennings. «A study of the influence of measurement timescale on internal resistance characterisation methodologies for lithium-ion cells». In: *Scientific Reports* 8 (Jan. 2018). DOI: 10.1038/s41598-017-18424-5 (cit. on p. 18).
- [23] P. van Overschee and B.L. de Moor. *Subspace Identification for Linear Systems: Theory — Implementation — Applications*. Springer US, 2012. ISBN: 9781461304654. URL: <https://books.google.it/books?id=39ziBwAAQBAJ> (cit. on p. 21).
- [24] Javier M Cabello, Eric Bru, Xavier Roboam, Fabien Lacressonnière, and Sergio Junco. «BATTERY DYNAMIC MODEL IMPROVEMENT WITH PARAMETERS ESTIMATION AND EXPERIMENTAL VALIDATION». In: *8th International Conference on Integrated Modeling and Analysis in Applied*

- Control and Automation, IMAACA 2015*. Berggegi, Italy, Sept. 2015. URL: <https://ut3-toulouseinp.hal.science/hal-03286755> (cit. on p. 28).
- [25] Minggao Ouyang, Guangming Liu, Languang Lu, Jianqiu Li, and Xuebing Han. «Enhancing the estimation accuracy in low state-of-charge area: A novel onboard battery model through surface state of charge determination». In: *Journal of Power Sources* 270 (2014), pp. 221–237. ISSN: 0378-7753. DOI: <https://doi.org/10.1016/j.jpowsour.2014.07.090> (cit. on p. 28).
- [26] Jiahao Li, Joaquin Klee Barillas, Clemens Guenther, and Michael A. Danzer. «A comparative study of state of charge estimation algorithms for LiFePO<sub>4</sub> batteries used in electric vehicles». In: *Journal of Power Sources* 230 (2013), pp. 244–250. ISSN: 0378-7753. DOI: <https://doi.org/10.1016/j.jpowsour.2012.12.057> (cit. on p. 28).
- [27] D. Andre, M. Meiler, K. Steiner, H. Walz, T. Soczka-Guth, and D.U. Sauer. «Characterization of high-power lithium-ion batteries by electrochemical impedance spectroscopy. II: Modelling». In: *Journal of Power Sources* 196.12 (2011). Selected papers presented at the 12th Ulm ElectroChemical Talks (UECT):2015 Technologies on Batteries and Fuel Cells, pp. 5349–5356. ISSN: 0378-7753. DOI: <https://doi.org/10.1016/j.jpowsour.2010.07.071> (cit. on p. 28).
- [28] Alexander Farmann and Dirk Uwe Sauer. «A study on the dependency of the open-circuit voltage on temperature and actual aging state of lithium-ion batteries». In: *Journal of Power Sources* 347 (2017), pp. 1–13. ISSN: 0378-7753. DOI: <https://doi.org/10.1016/j.jpowsour.2017.01.098> (cit. on p. 33).
- [29] YuTao Huo, Wen Hu, Zhe Li, and Zhonghao Rao. «Research on parameter identification and state of charge estimation of improved equivalent circuit model of Li-ion battery based on temperature effects for battery thermal management». In: *International Journal of Energy Research* 44.14 (2020), pp. 11583–11596. DOI: <https://doi.org/10.1002/er.5784> (cit. on pp. 33, 34).
- [30] James L. Lee, Lukas L. Aldrich, Kirk D. Stetzel, and Gregory L. Plett. «Extended operating range for reduced-order model of lithium-ion cells». In: *Journal of Power Sources* 255 (2014), pp. 85–100. ISSN: 0378-7753. DOI: <https://doi.org/10.1016/j.jpowsour.2013.12.134>. URL: <https://www.sciencedirect.com/science/article/pii/S0378775314000081> (cit. on p. 34).
- [31] Daniel Marcin Dávid Bodnár and František Ďurovský. «Temperature-dependent hysteresis model for Li-ion batteries». In: *Automatika* 65.3 (2024), pp. 1315–1324. DOI: 10.1080/00051144.2024.2368365 (cit. on pp. 45, 83–85).

- [32] René Schneider and Christos Georgakis. «How To NOT Make the Extended Kalman Filter Fail». In: *Industrial and Engineering Chemistry Research* 52.9 (2013), pp. 3354–3362. DOI: 10.1021/ie300415d (cit. on p. 54).
- [33] Simon Haykin. «Kalman Filters». In: *Kalman Filtering and Neural Networks*. John Wiley and Sons, Ltd, 2001. Chap. 1, pp. 1–21. ISBN: 9780471221548. DOI: <https://doi.org/10.1002/0471221546.ch1>. eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/0471221546.ch1>. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1002/0471221546.ch1> (cit. on p. 54).