



ÉCOLE POLYTECHNIQUE  
FÉDÉRALE DE LAUSANNE



**Engineering college**  
**Master in Electrical Engineering**  
**Distributed Electrical System Laboratory**

Experimental validation of a model predictive  
control based on energy management strategy  
for electro-chemical storage

Academic year 2016/17

*Professors:*

Gianfranco Chicco

Mario Paolone

*Supervisors:*

Fabrizio Sossan

Emil Namor

*Student:*

Antonello Schiraldi

*To my family for their  
irreplaceable support and  
encouragement throughout  
my life.*

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Experimental Setup</b>	<b>3</b>
2.1	Test bench features . . . . .	3
2.2	New measurement setup . . . . .	6
<b>3</b>	<b>Identification of a battery cell dynamic equivalent circuit models</b>	<b>10</b>
3.1	State of charge estimation . . . . .	11
3.2	Grey-box model . . . . .	14
3.2.1	Models formulation . . . . .	17
3.2.2	Identification of models . . . . .	18
3.2.3	Model validation . . . . .	27
<b>4</b>	<b>An Energy management strategy for the battery cell</b>	<b>40</b>
4.1	Real-time operation . . . . .	40
4.2	Control objective formulation . . . . .	43
4.3	Model predictive control . . . . .	45
4.3.1	Derivation of the transition matrices for MPC . . . . .	47
4.3.2	Formulation and implementation . . . . .	49
4.3.3	Optimization problem . . . . .	50
4.4	State estimation . . . . .	53

4.5 Feedback control and control actions . . . . . 55

**5 Experimental results 57**

5.1 Experiment 1 . . . . . 59

5.2 Experiment 2 . . . . . 63

**6 Conclusion 69**

**Bibliography 71**

# Chapter 1

## Introduction

Energy is the base for social and economic development. Over the last 50 years, the demand for energy has increased to support the agricultural, industrial and domestic activities. This has meant rapid growth in the level of greenhouse gas emissions and the increase in fuel prices, which are the main driving forces behind efforts to utilize renewable energy sources more effectively.

Although the advantages of renewable energy, it is often characterized by discontinuity of generation, since most of the renewable resources energy resources depend on the climate.

A way to face the discontinuity of renewable resources is to integrate a storage system, in order to ensure continuity of service when the renewable resource can not supply energy. Furthermore, predictive control strategies coupled with forecasts of stochastic consumption and generation applied to distributed energy resources is considered with increased interest to tackle the challenges related to power systems operation with high proportion of production from renewable sources.

In this thesis, we focus our attention on energy management of a battery cell, which receive different power set-points depending on which the battery cell operation changes. In particular, the cell operation depends on sign of

set-point, i.e. when it is positive the battery is charged by power source, instead if it is negative the battery is discharged and it provides energy to the load.

The goal of this work is to compare two energy management strategies, that have to ensure the respect of voltage and current bounds during battery cell operation in order to adjust the battery cell power injection such that the average power consumption at the end of a 5-minutes period matches the received set-point. The two control strategy, model predictive control and feedback control, are compared in terms of tracking error, namely the difference, at end of 5-minutes interval, between the power set-point received and the realization of battery cell. Furthermore, we also observe which of two strategy is more suitable to guarantee respect of voltage and current bounds.

Firstly, we improve the measurement set-up in order to obtain a more accurate current measure.

Secondly, we perform an identification of battery cell dynamic equivalent circuit models through grey-box modelling, which combines a partial theoretical structure with measured data to complete the model. It leads to parameters estimation of equivalent circuit for different ranges of state of charge. The equivalent circuit adopted is a TTC (Two Time Constant), that consist is two branches made of a parallel between resistor and a capacitor, one series resistor and a controlled voltage source.

The models, are necessary to develop the model predictive control which relies on dynamic model of the system thank to it is able to optimise the current time slot keeping future time slot into account. We describe in detail the implementation of the real-time operation, the control objective and the control actions for both control type, with particular emphasis on model predictive control case. Finally, we perform some experiment to show which of two strategy is more suitable for energy management of battery cell.

# Chapter 2

## Experimental Setup

In this section we present the experimental set-up describing how it is composed and how it works. Furthermore, in the second part of this section we describe the improvement of measurement set-up introducing a shunt resistor with the relative circuit of data acquisition.

### 2.1 Test bench features

The test bench is composed by the following elements:

- a climatic chamber hosting the targeted cells and keeping a constant temperature at  $25^{\circ}\text{C}$ , in steady state conditions the temperature ripple is of  $0.2^{\circ}\text{C}$ ;
- a power source working in the following V/I ranges  $0 - 80\text{ V}$ ,  $0 - 120\text{ A}$ ;
- an electronic load working in the following V/I ranges  $0 - 80\text{ V}$ ,  $0 - 200\text{ A}$ ;
- control PC where a suitable realized software developed using the *LabVIEW* programming environment is able to perform the monitoring and control of the whole system (shown in figure 2.1).

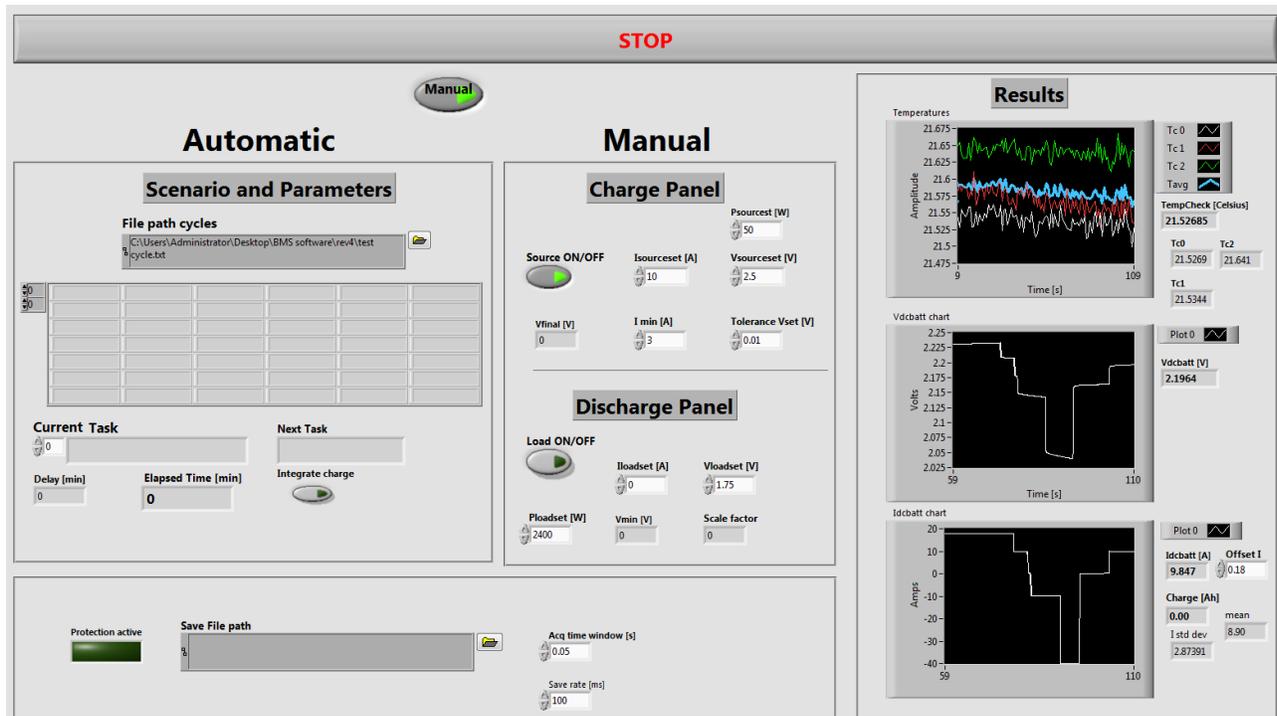


Figure 2.1: Front panel that controls the test bench.

The cell voltage is directly sampled using an analog-to-digital converter operating a 16 bits with a maximum sampling frequency of  $100\text{kHz}$  and characterized by a bandwidth from DC to 100 kHz (3 dB) with an overall accuracy of 1,5 mV.

The cell current is measured using a dedicated hall-effect sensor characterized by a bandwidth from DC to 100 kHz (3 dB) and an overall accuracy of less than  $\pm 0.5\%$  of the measured current.

Our goal is to improve the current measurement when the current setted by control panel is near to zero or zero, because the hall-effect sensor used presents an offset for that value of current, represented in figure 2.2.

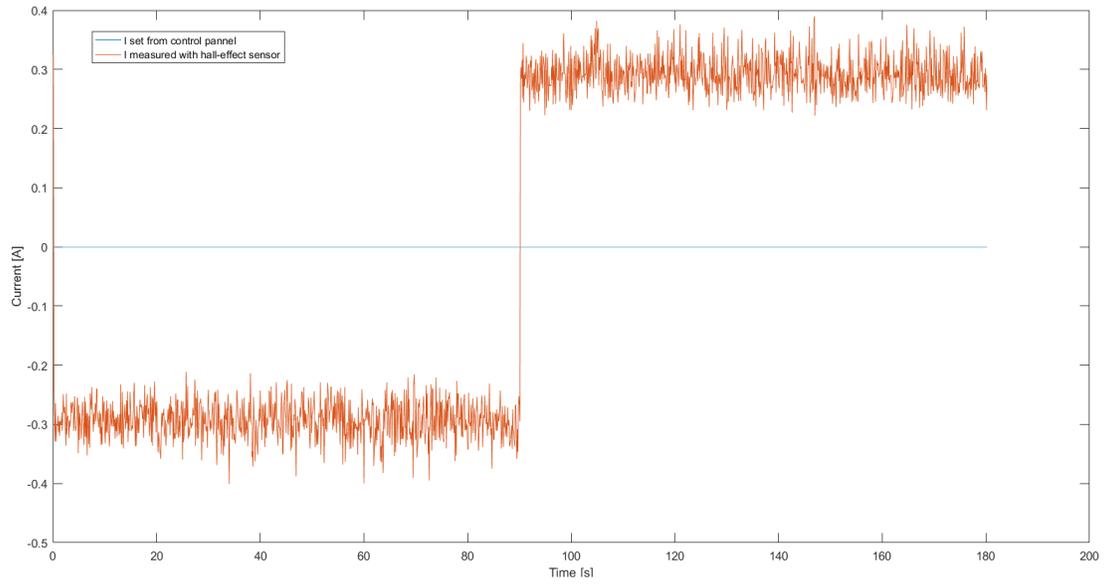


Figure 2.2: Comparison between the measured current by hall-effect sensor (in orange) and the current setted by control panel (in blue).

## 2.2 New measurement setup

We add to measurement setup a shunt resistor and an analog circuit to amplify and to filter the signal from shunt in order to obtain a better accuracy of measurement.

The development of filter starts with computation of the cut-off frequency. These parameters are fixed by the *NI 9215*, which is an analog input module that we use with *NI CompactDAQ*, through a *LabVIEW VI* for the acquisition of data. The *NI 9215* includes four simultaneously sampled analog input channels and successive approximation register (SAR) 16-bit analog-to-digital converters (ADCs) [1].

We have to consider the Nyquist sampling theorem to set the maximum value of cut-off frequency at 5 Hz, because the sampling frequency setted on *NI 9215* is 10 Hz. Instead, to set the desired gain we have to rate the resolution of *NI 9215* that is 16 bit and the shunt resistor's features (60 mV @60 A) . Exploiting the mentioned parameter, and considering that we want a step of at least 50 mA we compute the minimum gain which is 6.1 V/V=7.853 dB. But we decide to used a bigger gain in order to obtain a better accuracy of measurement.

To amplify and filter the signal we use two different operational amplifiers, namely we use *AD620* to amplify the signal and the *LTC 1050* to filter it. The circuit implemented is shown in figure 2.3 and it has the following features

$$f_{cut-off} = \frac{1}{2\pi C_3 R_2} = 4.7747 Hz \quad (2.1)$$

$$Gain = G = 1 + \frac{49.4k\Omega}{R_1} = 198.6 \quad (2.2)$$

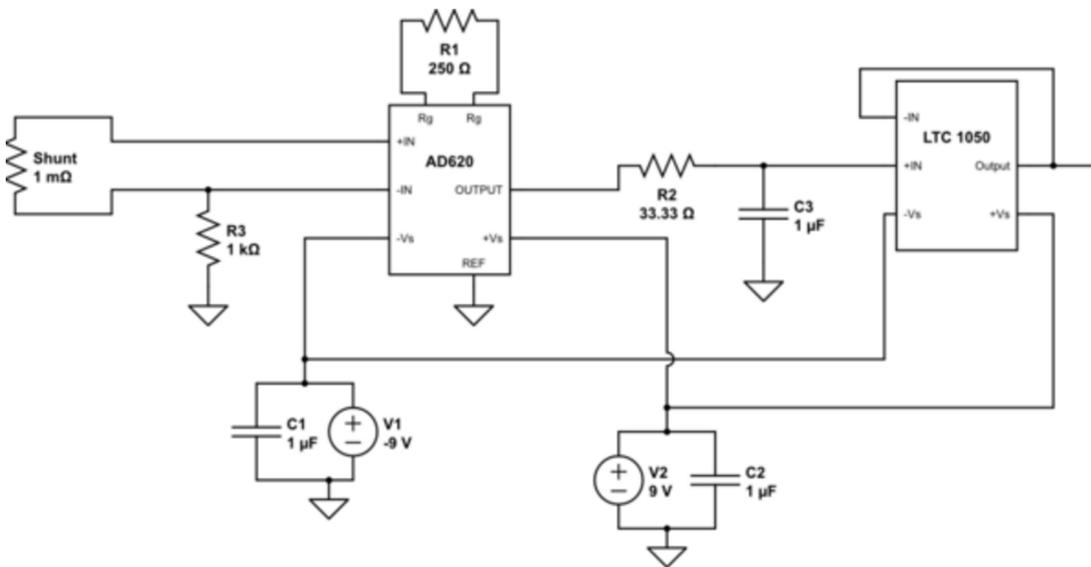


Figure 2.3: Implemented circuit.

Hence, we test the new measurement setup for different values of current in order to compare the performance of shunt current monitoring system with that of hall-effect sensor. The comparison of results is shown in figure 2.4, where it emerges that the hall-effect sensor has better measurement capability for current of value far from 0 A respect to shunt resistor.

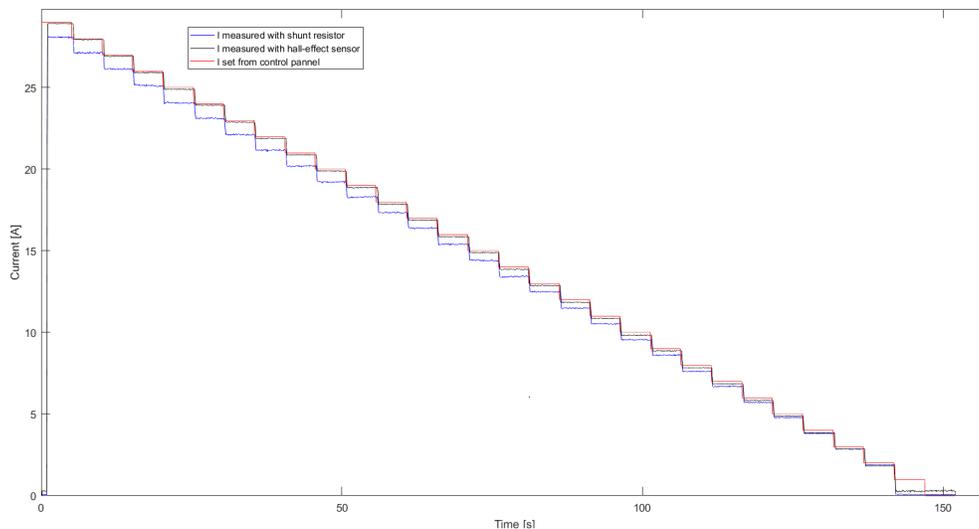


Figure 2.4: Current measurement with hall-effect sensor and shunt resistor.

Instead, when the setted current is near to zero, the shunt resistor gives us a better measurement of the current, as reported in figure 2.5.

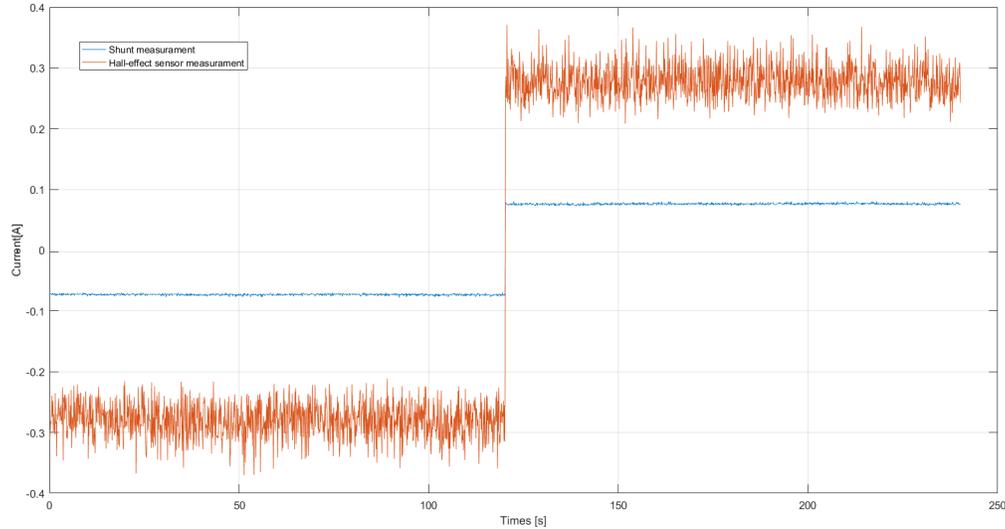


Figure 2.5: Comparison of measurement between hall-effect sensor and shunt resistor when the current is setted to 0.078 A.

It is important to note that the current is assumed positive when the battery is charging and negative when it is discharging.

Furthermore, in figure 2.4, it is possible to note that the trend of current measured by shunt resistor does not follow with constant difference the current setted, but its trend changes. We observed with other experiments, that this phenomenon is due to non-linearity of op-amp’s gain for small signal(few mV).

For this reason we decide to use a “ hybrid” system to measure the current. We configure the acquisition of data in the following way:

- when the setted current is near to zero ( $i \in [-0.3, 0.3]$  A) the current measurement comes from the shunt resistor;
- when the setted current is not included in the range  $[-0.3, 0.3]$  A the current measurement comes from hall-effect sensor.

Then, we configure the DAQ (data acquisition system) in order to perform

the current measure. It is worth noting that it could be possible to obtain an accurate measurement with shunt resistor for all value of current but this is out of our purpose.

## Chapter 3

# Identification of a battery cell dynamic equivalent circuit models

The main topic of this section is identification of the system by grey-box modelling. The aim of identification of models is to obtain circuit models which are able to describe in the best way the system dynamics in order to use these models in the MPC formulation.

Furthermore, we illustrate the procedure to obtain the discharge characteristic of battery cell, which is necessary to study the dynamics of state of charge

Dynamic model identification is carried out by applying grey-box modelling which is a framework to identify and validate a mathematical model of a system incorporating its physical knowledge together with measurements from a real device. Since the values of model parameters strongly depend on the cell state of charge (SOC), we perform a number of PRBS sessions where the cell is kept in a specific SOC interval (0-20%, 20-40%, 40-60%, 60-80%, 80-100%), and, for each interval, a model is estimated.

Measurements of current and voltage at cell terminals are acquired through the *LabVIEW* VI presented in previous section, and they were acquired at 0.1 s.

### 3.1 State of charge estimation

Since we need to capture the dynamics of the battery cell for different value of state of charge, we have to know which is the level of SOC at the beginning of experiment and how it evolves while the battery cell receives a power set-point. Since in the VI of acquisition data there is not the SOC computation we can estimate the state of charge exploiting current and voltage measurements of the battery cell.

What we want to obtain is the so-called “ discharge characteristic”, which allows to associate a voltage value to a specific value of state of charge.

The battery cell adopted is a Li-ion cell with the following features:

Parameter	Unit	Value
Cell minimum voltage	V	1.7
Cell nominal voltage	V	2.3
Cell maximum voltage	V	2.7
Cell nominal capacity	Ah	30

Table 3.1: Cell features.

To obtain the discharge characteristic of cell we follow the next steps:

- First of all we discharge completely the cell and after we have to wait enough time to allow the cell to reach the open circuit voltage.
- Once the cell is completely discharged and the open circuit voltage is approximately constant we charge the cell employing a square wave of current that assumes values 0 and 10 A, as shown in figure 3.1.

We use the square wave because in this way we can alternate a charge period with a steady state period (both period are long 3 minutes). During the charge period the cell is charged with constant current of 10 A, instead the steady state (current equal to zero) period is necessary to allow the cell to reach the open circuit voltage.

- At the end of cycle the cell is fully charged, and we exploit the current and voltage measurements to estimate the relationship between the open circuit voltage ( $V_{oc}$ ) and the state of charge (SOC).

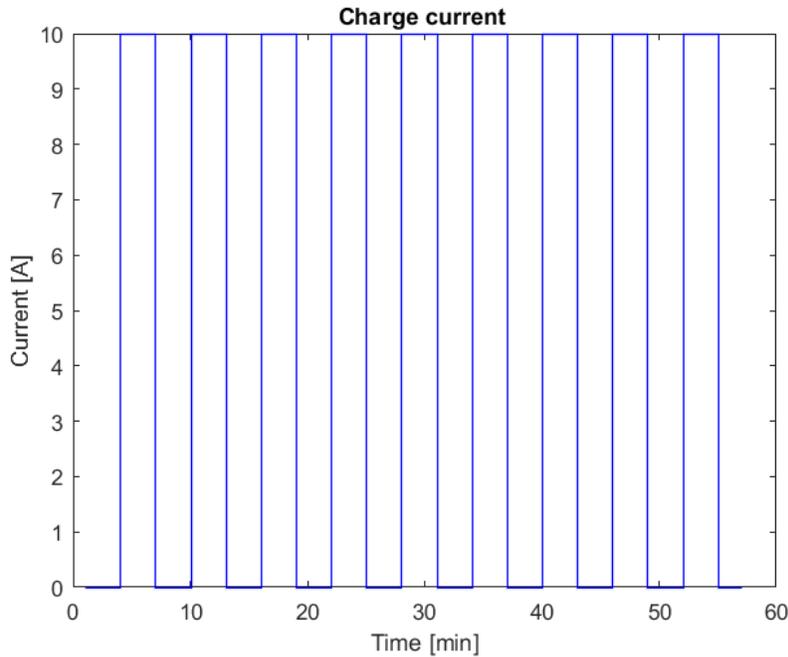


Figure 3.1: First 60 minutes of waveform of current setted used to charge the cell.

- Therefore, we take as  $V_{oc}$  all the value of measured voltage at the end of all steady state period, just an instant before of the beginning of charge period. Instead, we compute the charge of cell from the current measurement according to the Eq. 3.1

$$Q = \int_0^t i(t)dt \quad (3.1)$$

and the state of charge in accordance with Eq.3.2

$$SOC = \frac{Q(t)}{Q_n} \quad (3.2)$$

where  $Q_n$  is the cell nominal capacity.

From the described operation we obtain the discharge characteristic reported in figure 3.2.

As shown in figure 3.2, the minimum open circuit voltage measured is different from the minimum voltage reported in table 3.1 because the latter is a value that can only be reached during a transitory period. Furthermore, it is interesting to note that the value of SOC exceeds 100%, that because the battery is charge with a current smaller than 30 A and it makes possible to store more than the nominal charge, in this case 2.8322 Ah more that corresponds to 9.44% of nominal capacity.

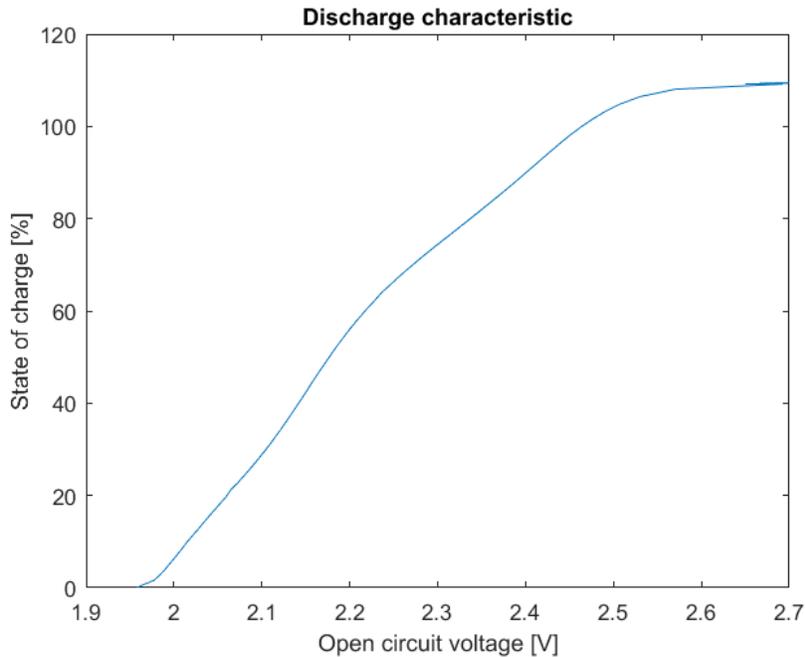


Figure 3.2: Computed discharge characteristic.

It is worth noting that the discharge characteristic is strictly dependent on temperature but in this case, as mentioned before, the temperature effect is negligible because the battery cell is installed in a climatic chamber which keeps the temperature constant at 25° C.

## 3.2 Grey-box model

Grey-box methodology is a modelling technique which models a dynamic system by incorporating the physical knowledge of the process with evidence obtained from experimental data.

The grey-box procedure consist in the following steps [2]:

1. **Experiments design** Given a device to model, suitable physical quantities should be selected for measurements. The process should be excited in all the frequency operation range in order to explore all the dynamics of the system. The signal used to excite the system is the PRBS (Pseudo Binary Random Signal), that is a signal that assumes two states and whose duty cycle is randomly chosen from a uniform distribution. Ideally, two datasets should be available, one for parameters estimation and a second one for validation of the model.
2. **Data acquisition and measurements post-processing** The physical quantities of interest have to be measured using appropriate sensors, discretized with convenient sampling time, sampled at opportune resolution and stored. Then, we should perform a data post-processing to remove from the measurements information not inherent to the physical process to model.
3. **Model formulation** This phase consists in identifying a set of suitable physical relationships that describes the nature and the working principles of the process to model.
4. **Parameters estimation** The aim of this phase is to find the most suitable set of values for the model's parameters. It is based on maximizing the likelihood function of the observation (MLE, Maximum Likelihood Estimation), i.e. determining the set of parameters that maximizes the probability of observing the events described by the measurements.

5. **Model validation** This phase consists in verifying that the model with the freshly identified parameters is actually able to describe the physical phenomena it was intended for. For example, by evaluating any correlations in the model 1-step ahead prediction errors (or residuals) it is possible to infer if the model can capture all system dynamics or not.
6. **Model expansion** If the validation process is not satisfactory, an expansion of the model should be considered, for example by adding a new state, new parameters or an alternative representation of the physical process. Each time a new model is defined, the parameters identification and validation procedures should be repeated (as discussed in the previous two paragraphs). Once the new model and the values of its parameters are available, statistical tests (e.g. likelihood ratio test) should be performed in order to verify if the model extension that has been introduced is meaningful. This allows to avoid over-fitting due to an excess of parameters.

As cited before, we want to estimate one model for each of following interval of SOC (0-20%, 20-40%, 40-60%, 60-80%, 80-100%), thus we have to apply a PRBS at each of these intervals.

Starting with the battery completely discharged we charge it until to reach the  $V_{oc}$  that corresponds to 10% of SOC and then we apply it the PRBS.

The PRBS applied is a current signal with mean value approximately near to zero in order to have at the end of cycle roughly the same value of SOC of the beginning of experiment. We repeat the same process for the other ranges, considering as starting point of PRBS the middle value of each interval (i.e. 30%, 50%, 70%, 90%).

A set of mathematical relationships to describe the physical process to model is used.

Battery voltage models normally adopted for control application consists in

electric equivalent circuits, which models the electrochemical reactions for increased tractability. In this case, we adopt the so-called two time constant (TTC), that is an equivalent circuit model for batteries often adopted in which consists in two series RC branches, where the values of model parameters normally depend on the battery SOC, cells temperature, and C-rate. The dependency between parameters and SOC is captured by performing parameters estimation for different SOC ranges [3]. Temperature is not considered since the system is installed in a climatic chamber and the third is neglected because the cell is operated with a C-rate very close to nominal one.

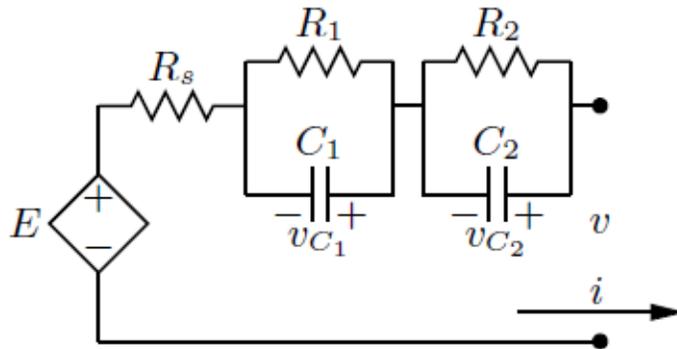


Figure 3.3: Structure of the equivalent circuit model [3].

The TTC model structure is shown in figure 3.3, where

$$E = \alpha + \beta SOC \quad (3.3)$$

( $\alpha$  and  $\beta$  are parameters to identify) is a controlled voltage source which describes the open circuit voltage as an affine function of the cell SOC. Model parameters  $R_1$ ,  $C_1$ ,  $R_2$ ,  $C_2$ ,  $R_s$  are normally a function of the cell SOC. In order to capture this relationship, they are estimated for different SOC ranges (0-20%, 20-40%, 40-60%, 60-80%, 80-100%).

### 3.2.1 Models formulation

Models are formulated by adopting the stochastic continuous-time state-space representation:

$$dx = \mathcal{A}(\theta)xdt + \mathcal{B}(\theta)u(t)dt + \mathcal{K}_c(\theta)d\omega \quad (3.4)$$

$$v_k = \mathcal{C}x_k + \mathcal{D}(\theta)u_k \quad (3.5)$$

where  $v_k$  is the model output and cell terminal voltage,  $x \in \mathbb{R}^n$  system state vector,  $n$  model order,  $\mathcal{A}$  system matrix,  $\mathcal{B}$  input matrix,  $\mathcal{K}_c$  input disturbance matrix,  $\mathcal{C}$  output matrix,  $\mathcal{D}$  feedforward matrix,  $u$  input vector,  $\omega$  a  $n$ -dimension standard Wiener process, and  $\theta$  is the set of model parameters to estimate.

The model matrices are composed as shown below:

$$\mathcal{A} = \begin{bmatrix} -\frac{1}{\tau_1} & 0 \\ 0 & -\frac{1}{\tau_2} \end{bmatrix} \quad \mathcal{B} = \begin{bmatrix} \frac{1}{C_1} & 0 & 0 \\ \frac{1}{C_2} & 0 & 0 \end{bmatrix} \quad \mathcal{C} = \begin{bmatrix} -1 & -1 \end{bmatrix}$$

$$\mathcal{D} = \begin{bmatrix} R_s & \alpha & \beta \end{bmatrix} \quad \mathcal{K}_c = \begin{bmatrix} k_1 \\ k_2 \end{bmatrix}$$

where  $R_1, C_1, R_2, C_2, R_s, \tau_1 = R_1C_1, \tau_2 = R_2C_2, \alpha, \beta$  (values of the circuit components),  $k_1, k_2$  (components of the system noise matrix) are the parameters to be estimated.

Instead, the state vector and system input matrix are:

$$x = \begin{bmatrix} v_{C_1} & v_{C_2} \end{bmatrix} \quad u_{t_k} = \begin{bmatrix} i_{t_k} \\ 1 \\ SOC_k \end{bmatrix}$$

### 3.2.2 Identification of models

Once the model is formulated, specifying the characteristic parameters, and data have been collected, we have to evaluate its goodness of fit, i.e. how good it is able to fit the observed data. Goodness of fit is assessed by finding parameter values of a model that best fits the data procedure called parameter estimation. Parameters of the model are estimated by applying maximum likelihood estimation (MLE) on the model one-step-ahead prediction error.

MLE is one of the most widely used method for estimating the parameters. The MLE selects the set of values of the model parameters which maximizes the likelihood function. Practically, this maximizes the “correspondence” of the selected model with the observed data [4].

When data have been collected and the likelihood function of a model given the data is determined, it is possible to carry out statistical inferences about the population, which is, the probability distribution that underlies the data. Given that different parameter values indicates different probability distributions, we are interested in finding the parameter value that corresponds to the desired probability distribution.

MLE affirms that the desired probability distribution is the one that makes the observed data “most likely”, namely that one must look for the value of the parameter vector what maximizes the likelihood function  $L(w|y)$ .

The resulting parameter vector, which is sought by searching the multi-dimensional parameter space, is called the MLE estimate, and is denoted by  $w_{MLE} = (w_{1_{MLE}}, w_{2_{MLE}}, \dots, w_{k_{MLE}})$  [5].

In short, MLE is one of most adopted method to seek the probability distribution which makes the observed data most likely.

We apply the principle of maximum likelihood estimation through the *MATLAB* function *greyest* setting the appropriate options. It is worth noting that it is necessary to set a bound for the estimation of parameters, namely

the estimated parameters can not be negative because they represent resistors and capacitors.

The estimated values of the model parameters are reported in the table 3.2.

<b>SOC</b>	<b>0–20%</b>	<b>20–40%</b>	<b>40–60%</b>	<b>60–80%</b>	<b>80–100%</b>
$R_1$	$9.2398e - 04$	$5.1545e - 04$	$5.0961e - 04$	$4.4545e - 04$	$6.0889e - 04$
$C_1$	$5.2442e + 04$	$1.0896e + 05$	$9.6127e + 04$	$3.5281e + 04$	$9.2099e + 04$
$R_2$	$2.4515e - 04$	$2.4773e - 04$	$2.0527e - 04$	0.0022	$2.3196e - 04$
$C_2$	284.220	$3.4592e + 04$	$3.4701e + 04$	$2.5077e + 05$	$2.2895e + 04$
$R_s$	0.0053	0.0030	0.0027	0.0027	0.0027
$\alpha$	1.915	1.9699	1.9299	1.8147	1.7310
$\beta$	0.00952	0.0045	0.0050	0.0067	0.0074
$k_1$	-0.1252	-0.1379	-0.1351	0.08819	-0.1878
$k_2$	1.429	0.0416	0.08801	-0.1597	0.1421

Table 3.2: Estimated parameters as a function of the cell SOC.

The following figures show the comparison of the system response with the measured data for each interval of state of charge.

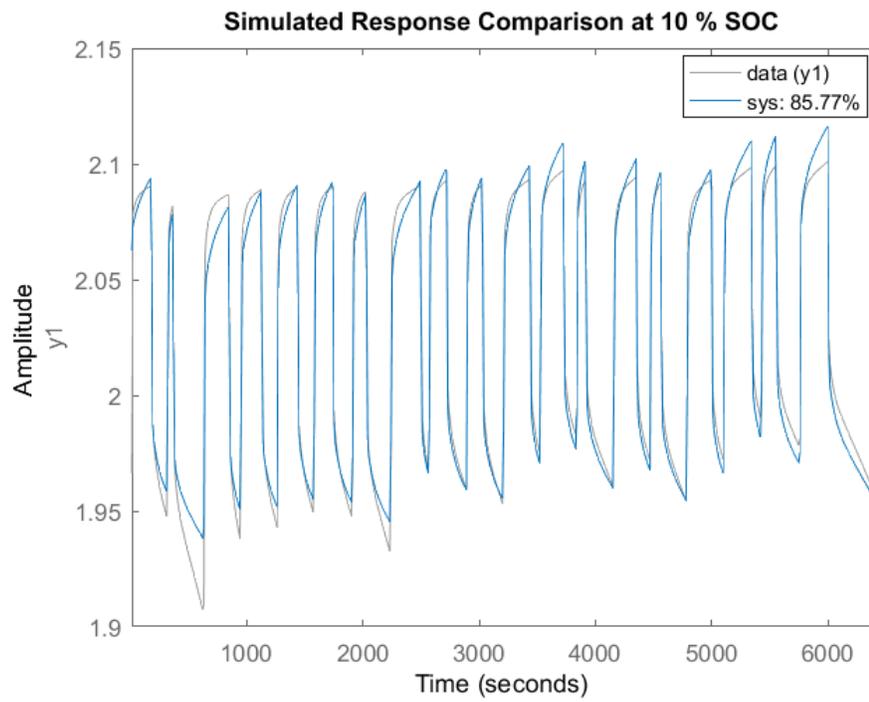


Figure 3.4: Comparison of the system response with the measured data at 10% SOC.

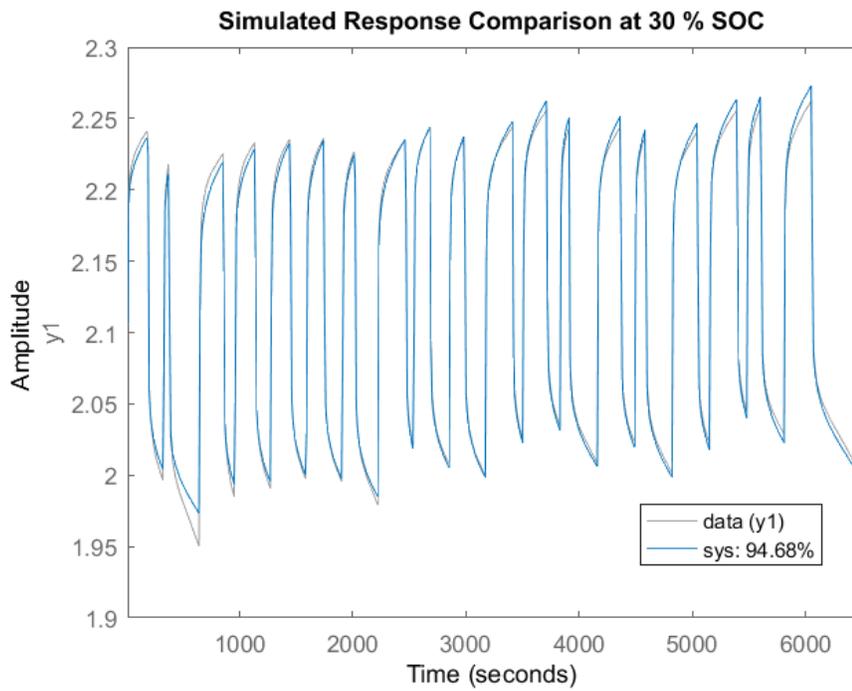


Figure 3.5: Comparison of the system response with the measured data at 30% SOC.

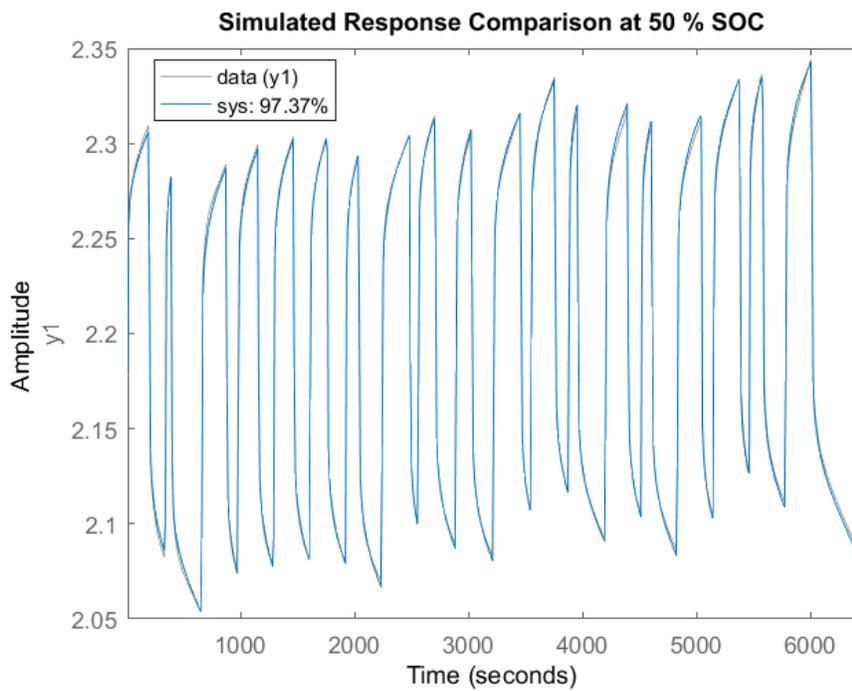


Figure 3.6: Comparison of the system response with the measured data at 50% SOC.

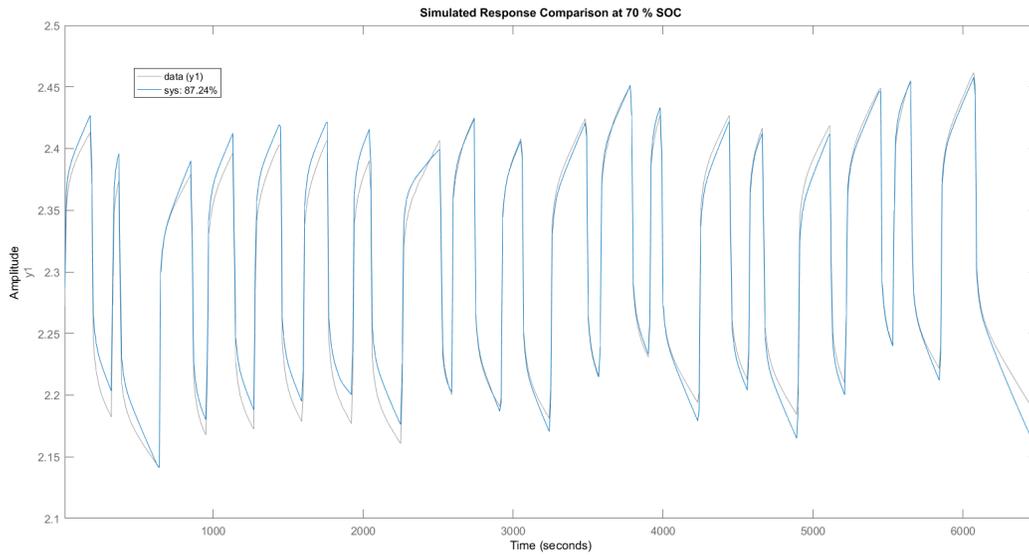


Figure 3.7: Comparison of the system response with the measured data at 70% SOC

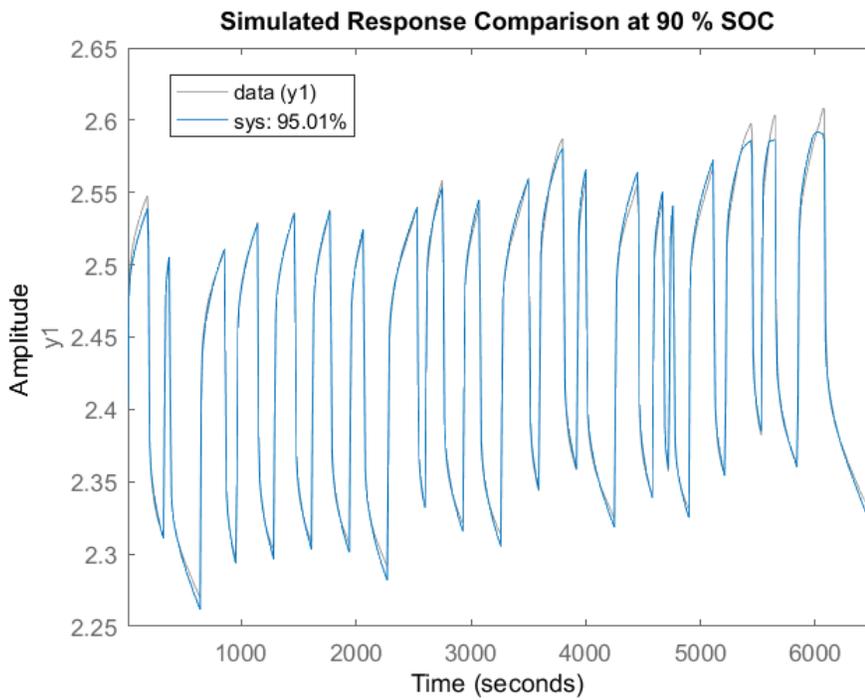


Figure 3.8: Comparison of the system response with the measured data at 90% SOC.

The shown results are obtained using the *MATLAB* function *compare*

which plots the simulated response of a dynamic system model, superimposed over validation data, for comparison. Furthermore, the plots also displays the normalized root-mean-square (NRMSE) measure of the goodness of the fit. It is to note that the result obtained are satisfactory, especially for the ranges 20-40%, 40-60%, 80-90% of SOC (figure 3.5, 3.6, 3.8) where the NRMSE has respectively the following values 94,68%, 97.37% and 95.01%. Instead, in the other two cases, the NRMSE is slightly lower with values of 85.77% for 0-20% SOC and 87.24% for 60-80% SOC (figure 3.4, 3.7).

We test the prediction performance of the identified models on different forecasting horizons (1-3600 seconds) and we compare it against a persistent predictor, namely where the point prediction for a certain time step is the last available observation. The benchmarking metric is the percentage root-mean-square of the voltage prediction error:

$$percentageRMSE(\%) = \sqrt{\frac{1}{N} \sum_{i=1}^N (V_i - \hat{V}_i)^2} \quad (3.6)$$

where  $V_i$  is the voltage measurements at time step  $i = 1; \dots; N$ , and  $\hat{V}_i$  the voltage prediction. The percentage RMSE is evaluated for both the two-time-constant and persistent models. At each time step, the correct set of parameters is chosen according to the level of SOC. The model is discretized and used to calculate the prediction as a function of the previous conditions and current value. The predictions of the TTC and persistent models are shown in the next figures.

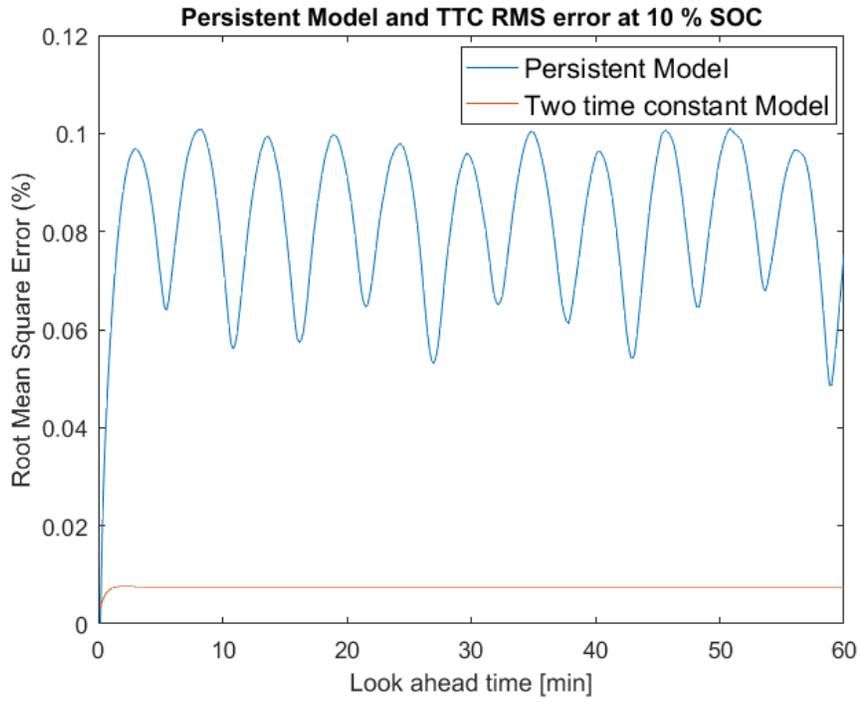


Figure 3.9: RMSE of TTC and persistent models for 10% SOC.

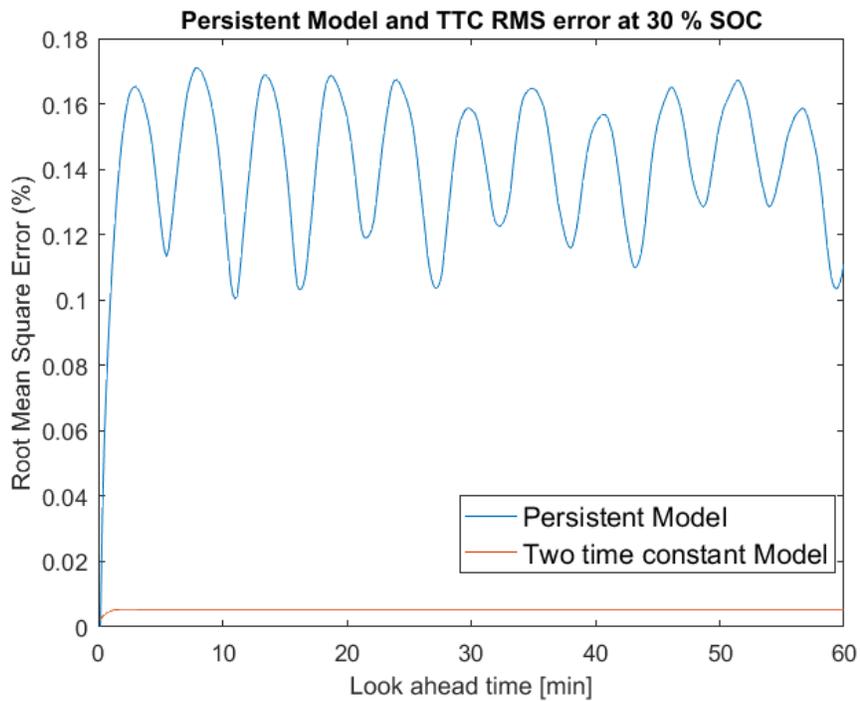


Figure 3.10: RMSE of TTC and persistent models for 30% SOC.

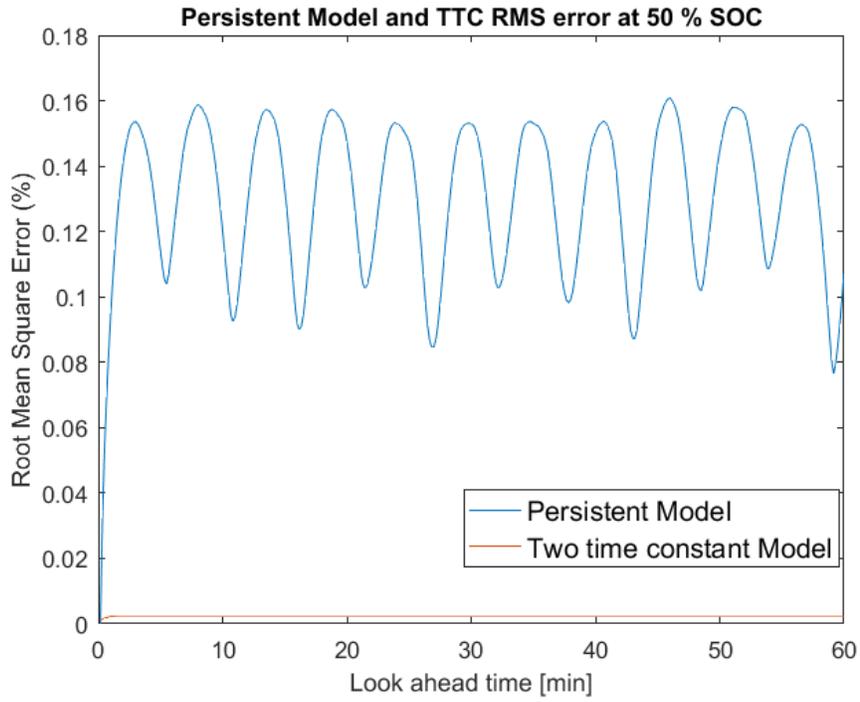


Figure 3.11: RMSE of TTC and persistent models for 50% SOC.

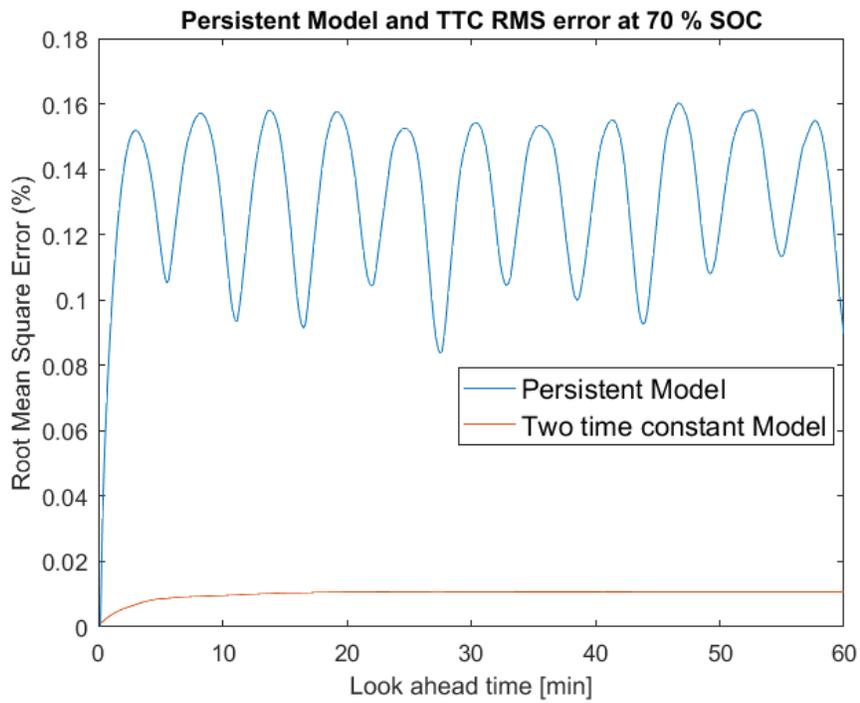


Figure 3.12: RMSE of TTC and persistent models for 70% SOC.

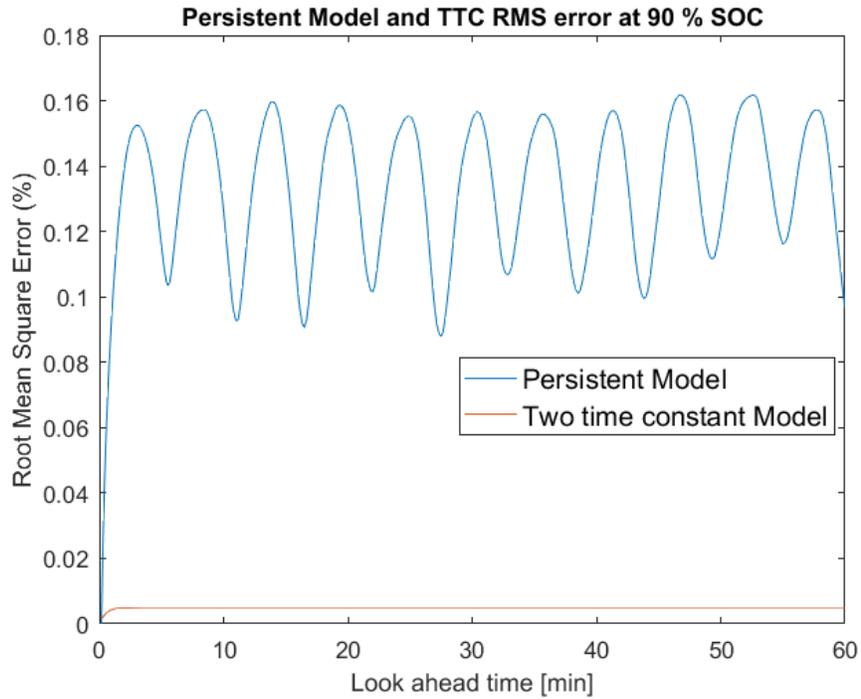


Figure 3.13: RMSE of TTC and persistent models for 90% SOC.

It can be noted that the percentage RMS error of the TTC model is sharply smaller than the one of persistent model. As it emerges from the results, the percentage RMS error has a kind of periodicity due to the PRBS application.

### 3.2.3 Model validation

The model validation consists in evaluating if the models found are able to capture all time dynamics contained in the training data set.

This is thanks to evaluating residual autocorrelation in time of the model one-step-ahead prediction errors, which in the ideal case should not contain any predictable structure and behave as an independent identically distributed (iid) random process. That means, we have to analyse the autocorrelation of prediction error for identified model. The prediction error is determined by subtracting the K-step ahead predicted response from the measured output and it is computed exploiting the *MATLAB* function *pe*, instead we used the function *autocorr* for the autocorrelation. The next figures represent the result of autocorrelation for each model identified.

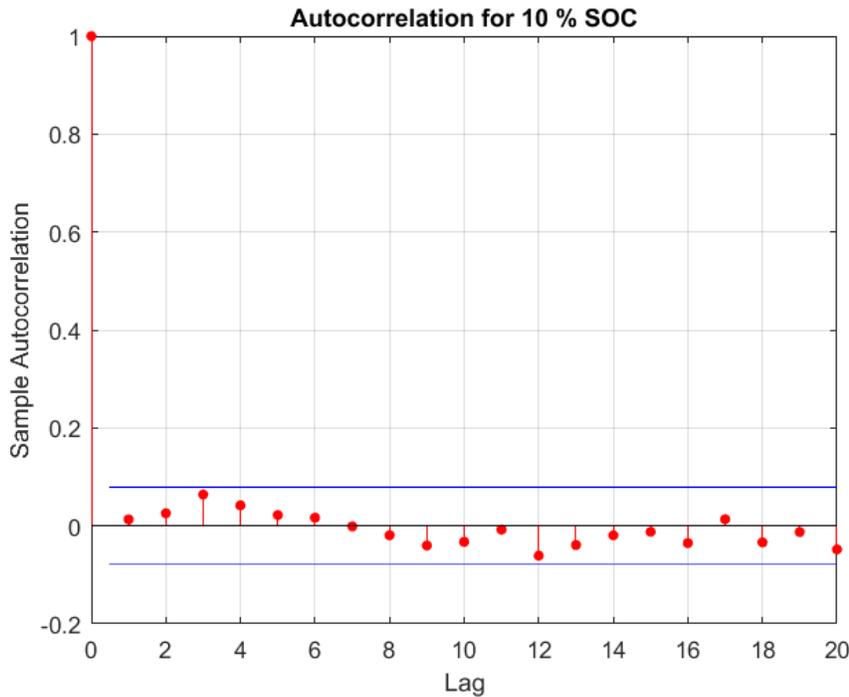


Figure 3.14: Residual autocorrelation function of predicted error for model at 10% SOC.

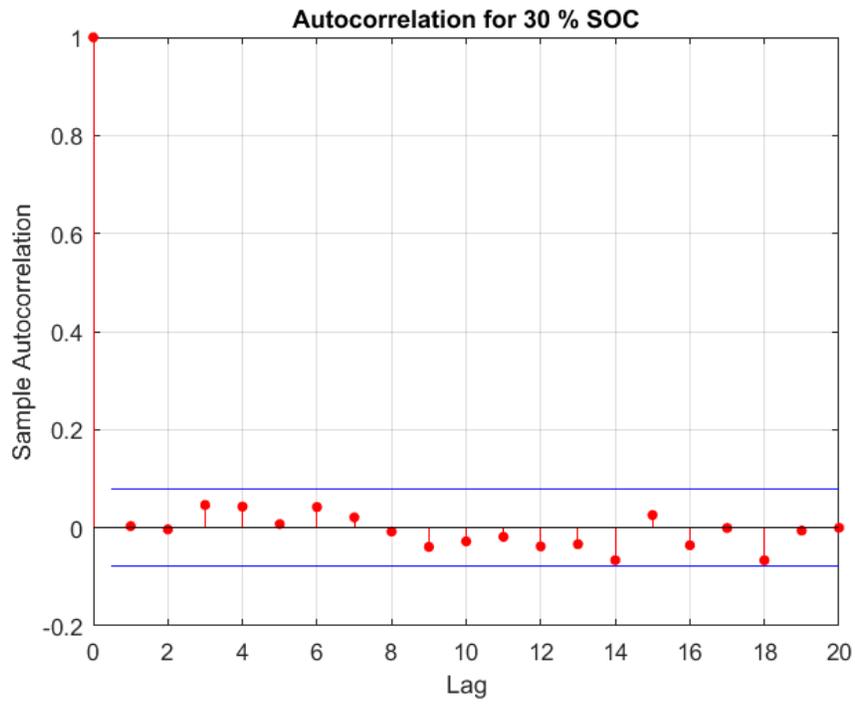


Figure 3.15: Residual autocorrelation function of predicted error for model at 30% SOC.

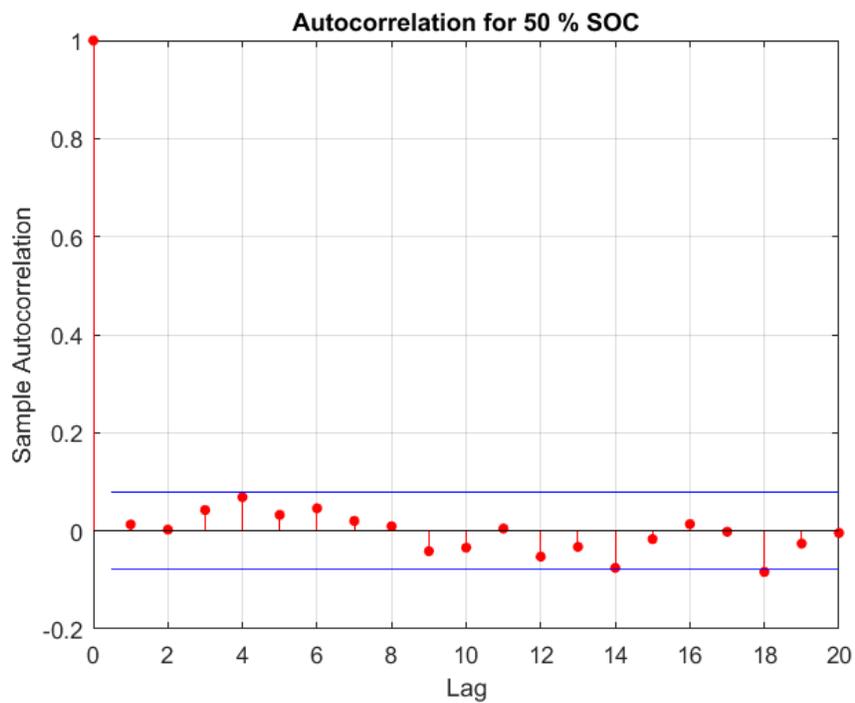


Figure 3.16: Residual autocorrelation function of predicted error for model at 50% SOC.

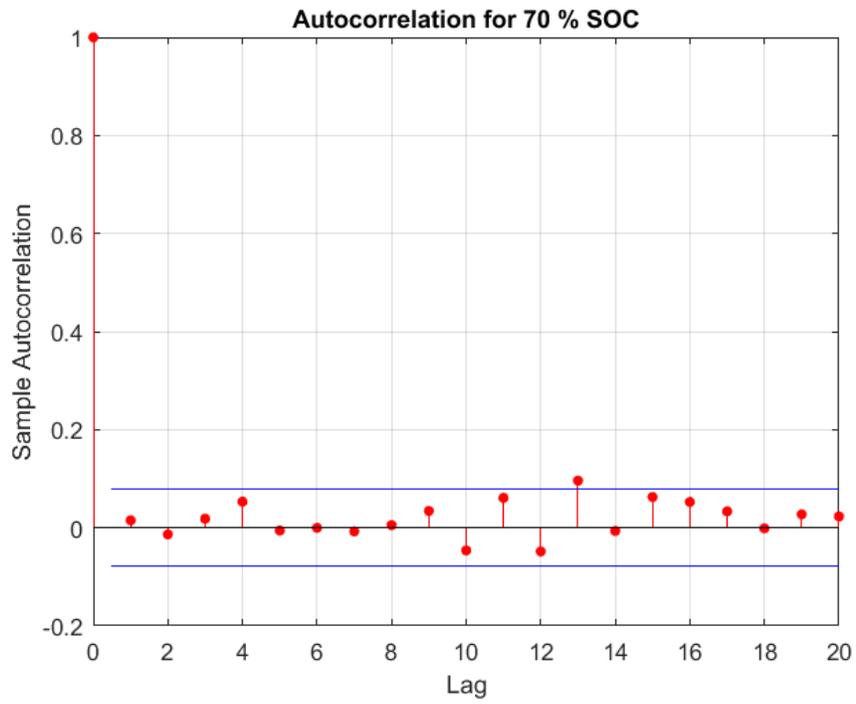


Figure 3.17: Residual autocorrelation function of predicted error for model at 70% SOC.

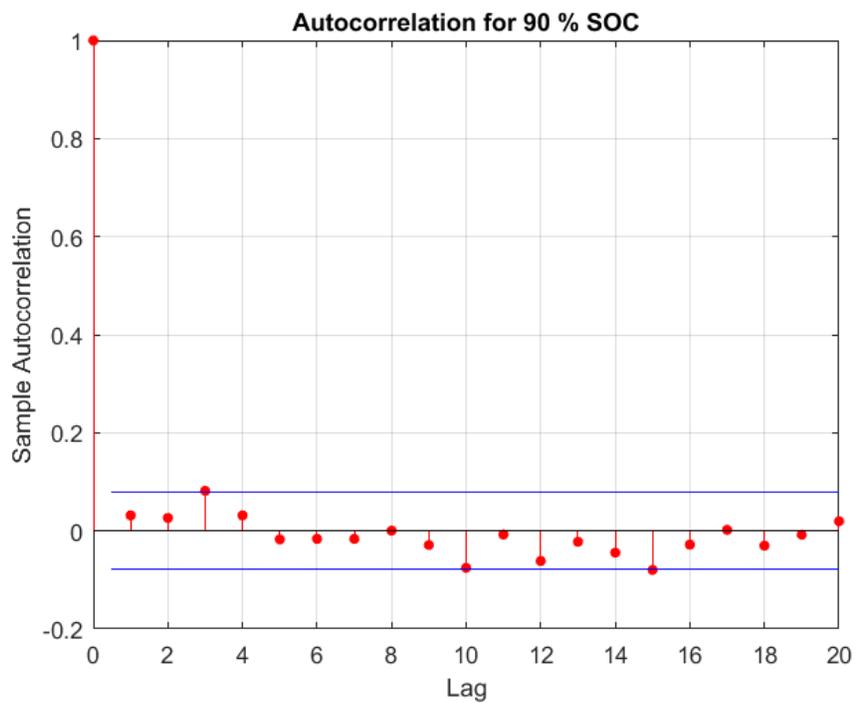


Figure 3.18: Residual autocorrelation function of predicted error for model at 90% SOC.

From the results of autocorrelation function it is possible to observe that the model for each range of SOC is adequate because it is able to absorb all the possible dynamics of system.

The results shown are relative to training data, namely the data used to identify the model.

Once identified models, we do some experiments to check the performance models. For example, we charge the battery from empty to full charge. Then, we divided the measurement data depending on SOC level and for each range of level we associate the relative found model. After that, we compare the system response with the measured data, the persistent model with the TTC model.

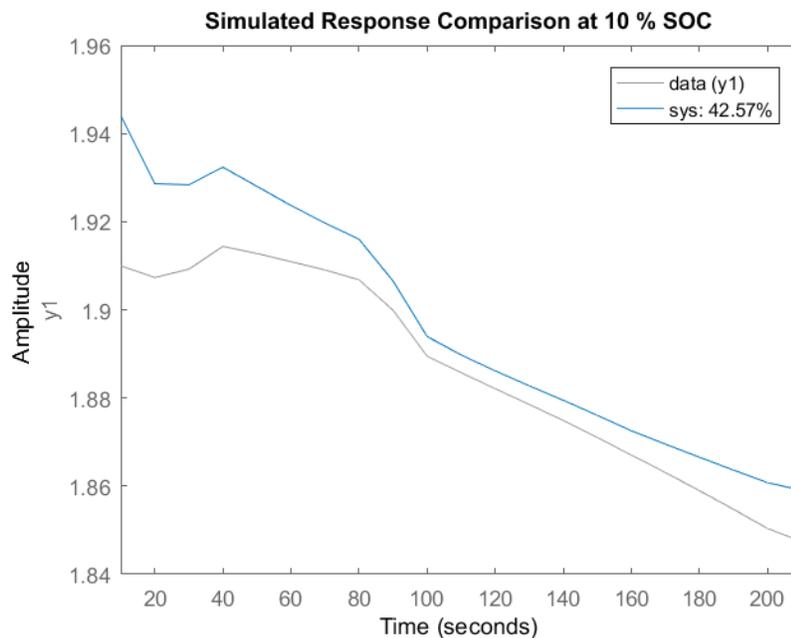


Figure 3.19: Comparison of the system response with the measured data at 10% SOC for validation data.

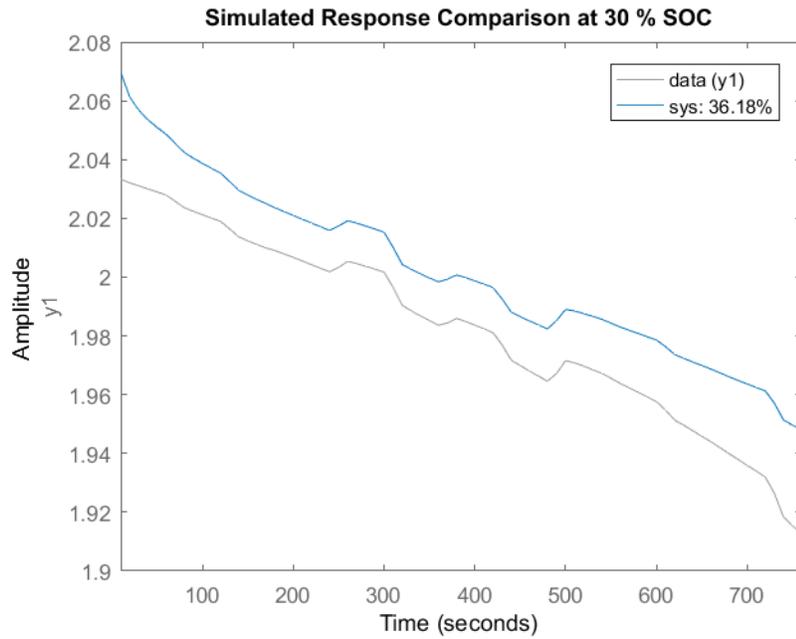


Figure 3.20: Comparison of the system response with the measured data at 30% SOC for validation data.

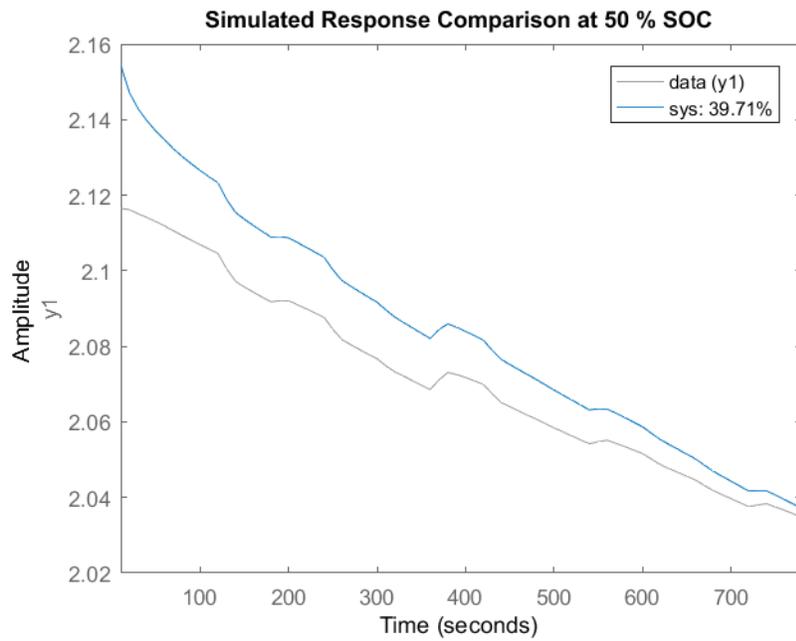


Figure 3.21: Comparison of the system response with the measured data at 50% SOC for validation data.

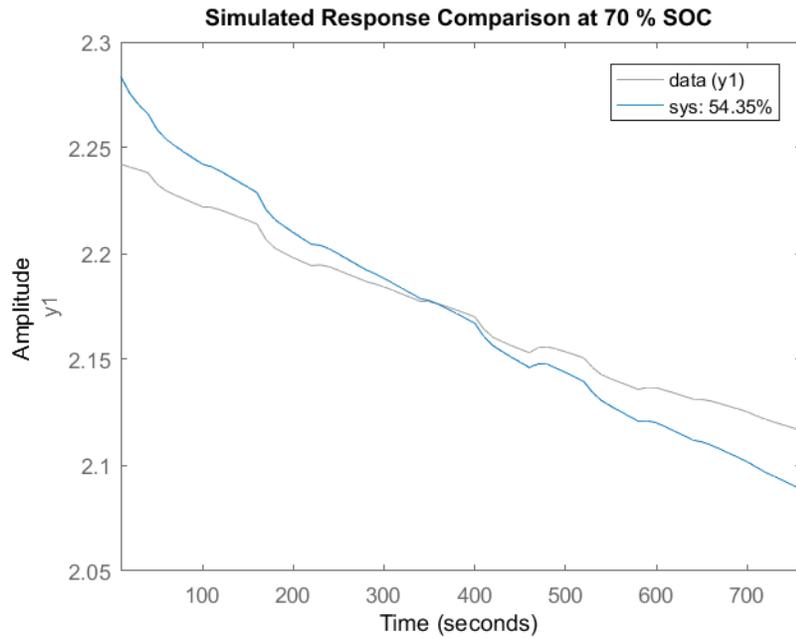


Figure 3.22: Comparison of the system response with the measured data at 70% SOC for validation data.

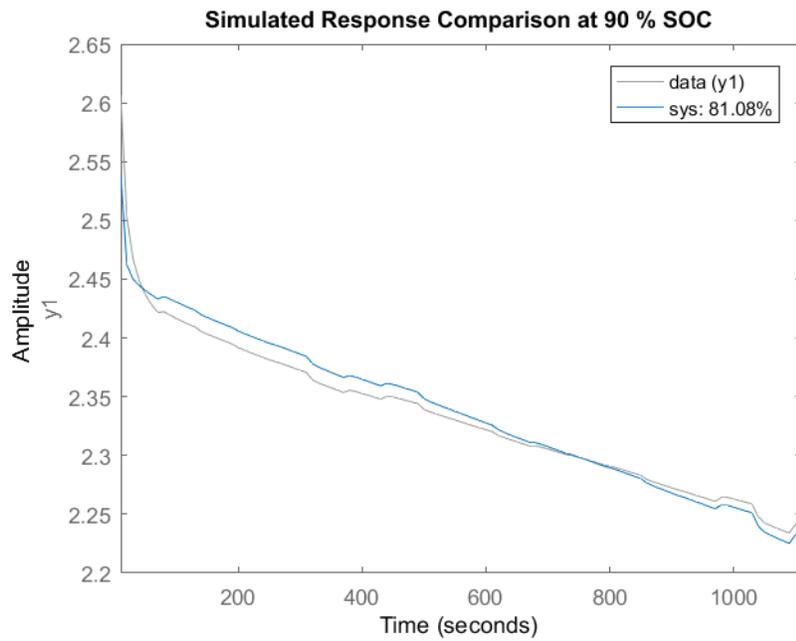


Figure 3.23: Comparison of the system response with the measured data at 90% SOC for validation data.

Comparing the results obtained using training data with the results shown in the previous figures, it emerges that the system response has worsened, but at the same time we can observe in the following figures how the TTC models still has better performance than persistent model for all ranges of SOC.

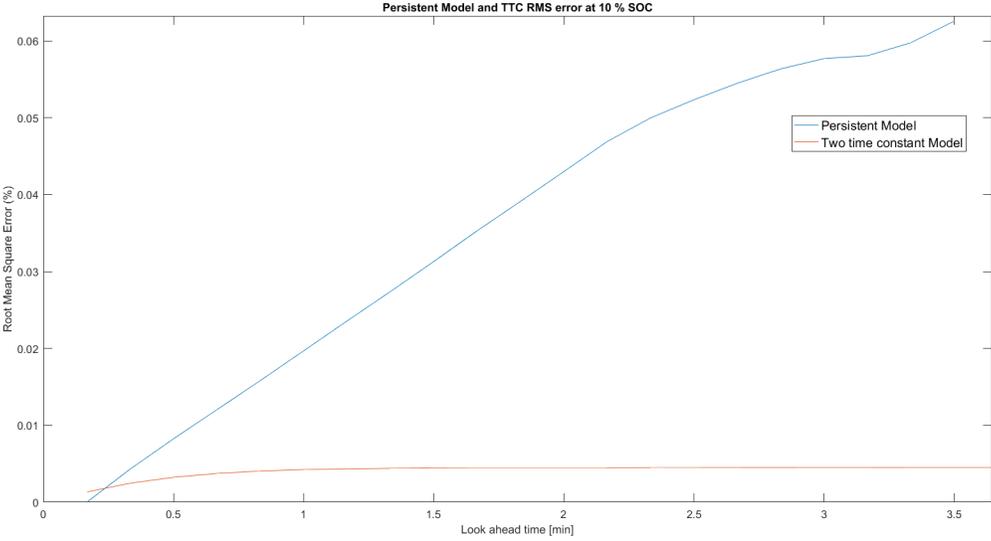


Figure 3.24: MSE of TTC and persistent models for 10% SOC for validation data.

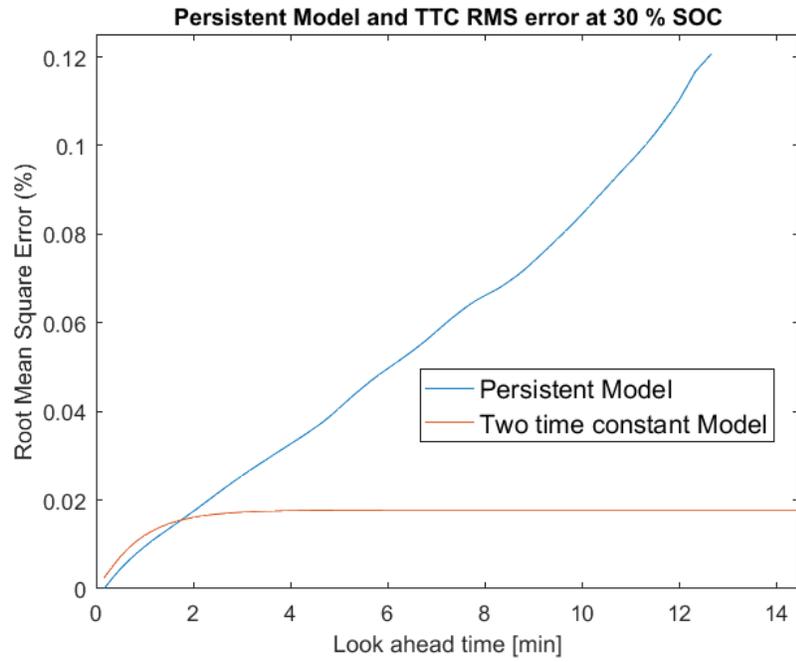


Figure 3.25: MSE of TTC and persistent models for 30% SOC for validation data.

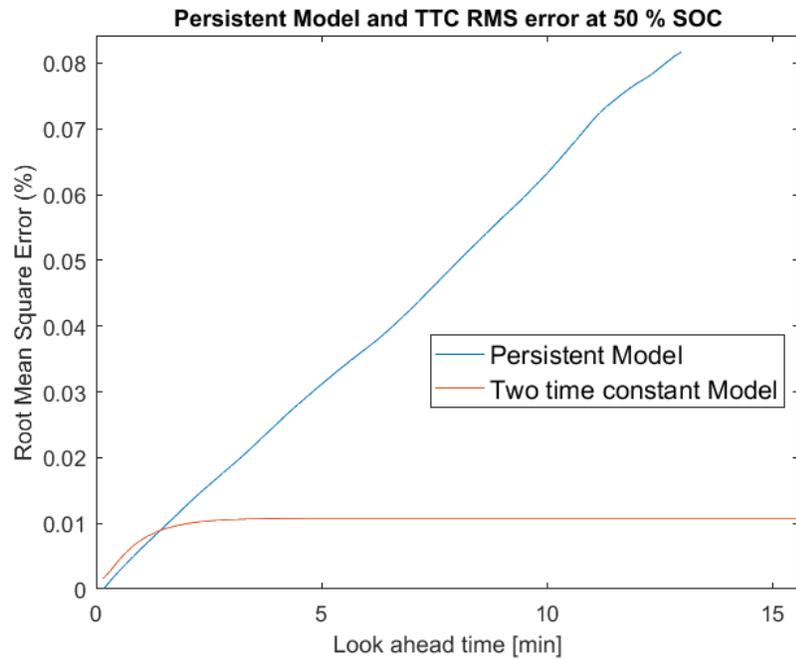


Figure 3.26: MSE of TTC and persistent models for 50% SOC for validation data.

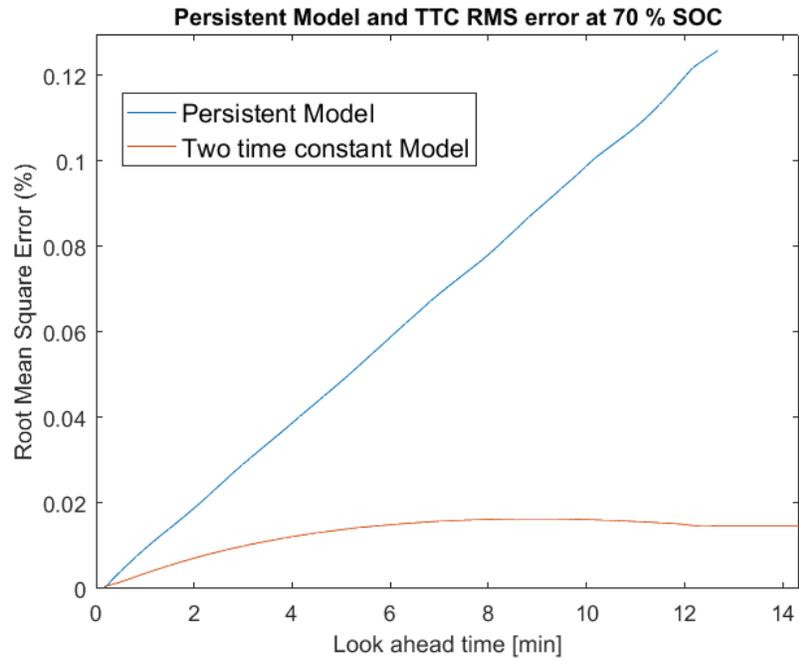


Figure 3.27: MSE of TTC and persistent models for 70% SOC for validation data.

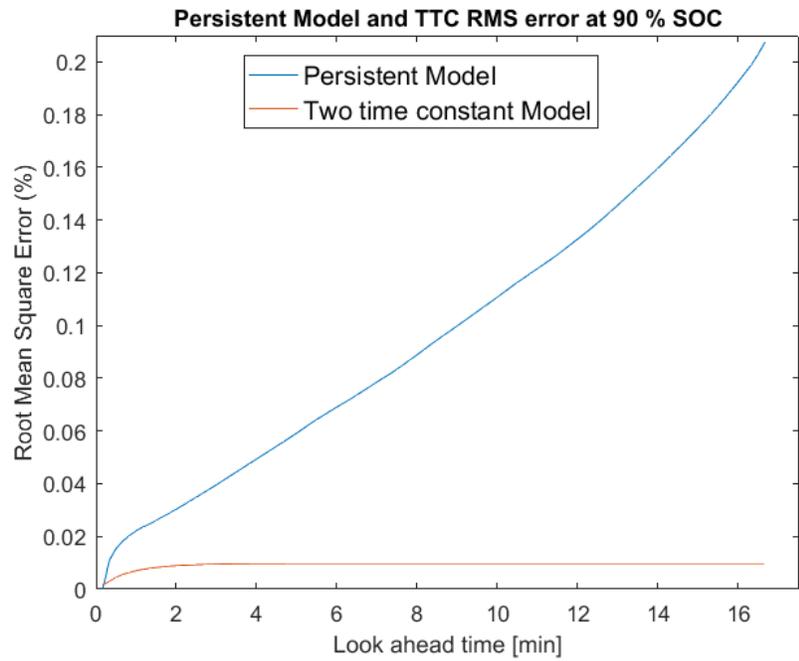


Figure 3.28: MSE of TTC and persistent models for 90% SOC for validation data.

It is also important to note that in this case models found are not able to

absorb the complete dynamics of systems, as it is possible to observe from the plots of autocorrelation function reported in the next figures.

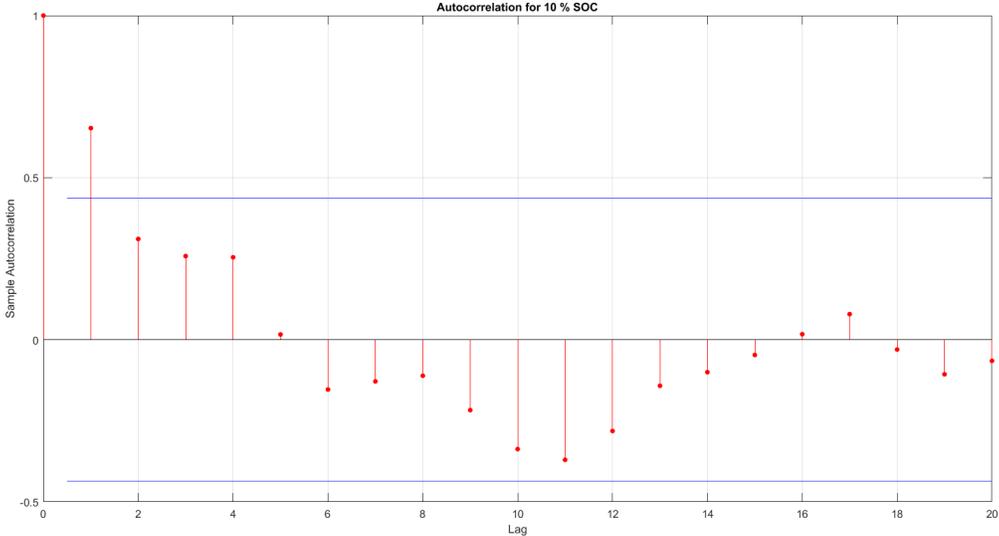


Figure 3.29: Residual autocorrelation function of predicted error for validation data at 10% SOC.

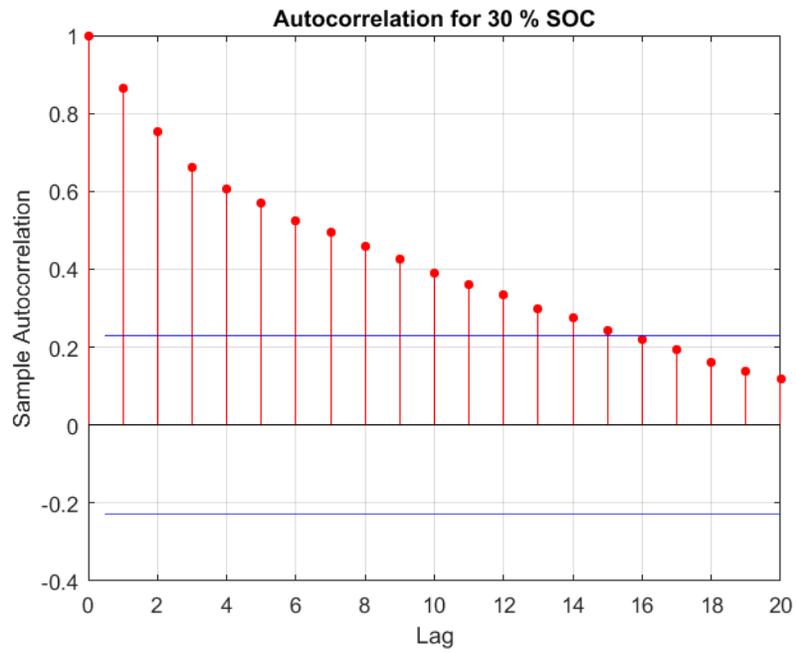


Figure 3.30: Residual autocorrelation function of predicted error for validation data at 30% SOC.

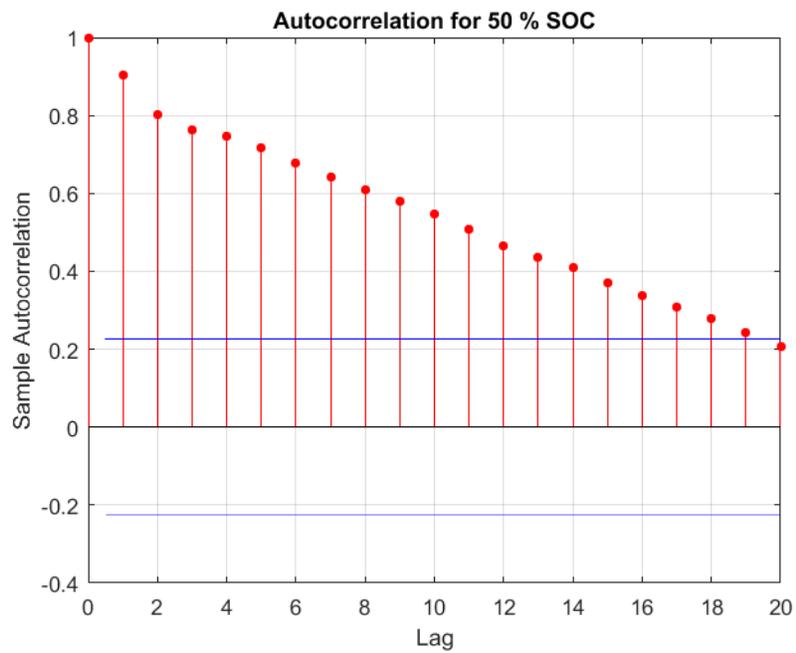


Figure 3.31: Residual autocorrelation function of predicted error for validation data at 50% SOC.

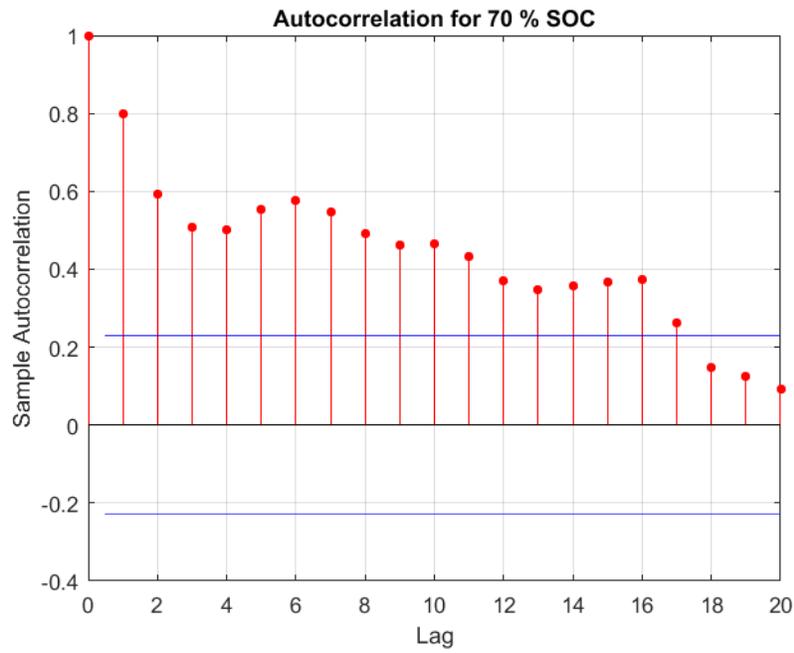


Figure 3.32: Residual autocorrelation function of predicted error for validation data at 70% SOC.

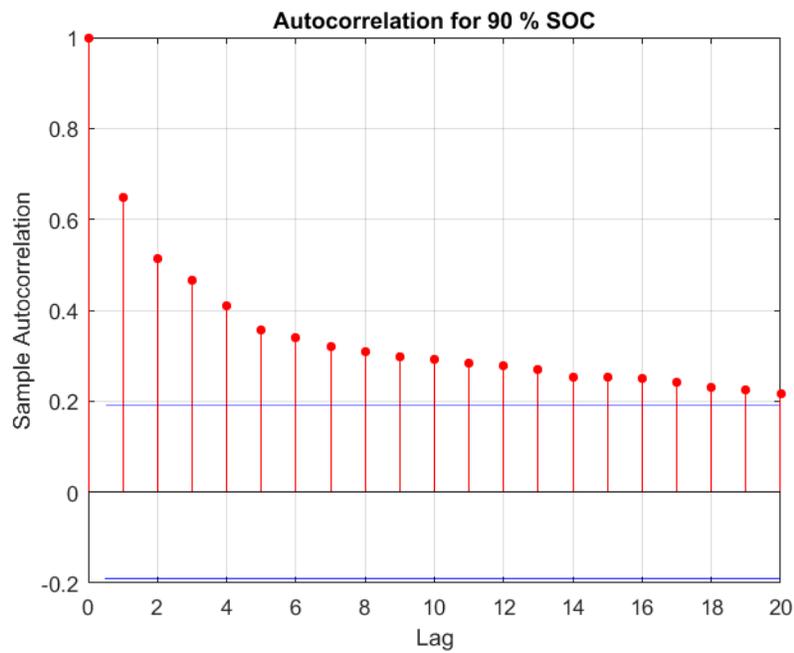


Figure 3.33: Residual autocorrelation function of predicted error for validation data at 90% SOC.

It is worth noting that, performing others experiments using a different value of C-rate respect to the one used during identification of model we found that performances of models found worse further. This means that to obtain a more accurate representation of dynamic of system it is necessary to consider the variation of C-rate in the model identification. However, for our application we can still consider valid the hypothesis done (i.e. the cell is operated with a similar C-rate as used in the real operation).

# Chapter 4

## An Energy management strategy for the battery cell

In this section, first, we provide a detail description of real time operation describing the variables involved for the control strategy. Secondly, the control objective is carefully presented, highlighting the difference for MPC and feedback control. Then, we give a detailed account of model predictive control illustrating its formulation and implementation.

Subsequently, the Kalman filtering is introduced to estimate the states and the state of charge.

Lastly, we describe the control operation and the difference between feedback control and MPC.

### 4.1 Real-time operation

The target of the real-time operation is to adjust the cell power injection such that the average power consumption at the end of each 5-minute period matches the respective set-point from the scenario.

The scenario consists in a set of positive and negative value of power, in particular a positive value of power corresponds to charge the cell, instead a negative one to discharge.

Prior to describe the control strategy, it is necessary to introduce the following notations which are valid both for MPC and for feedback control [6]:

- the control strategy is actuated with a sample time of 10 seconds in order to capture early time dynamics of cell and assure good control performance taking into account the computing times.
- The index  $k$  indicates the rolling 10 seconds time interval of operation.
- At the beginning of each interval  $k$ , the real power flow, the cell flow and the disturb for the previous interval ( $k - 1$ ) become known thanks to measurements. They are respectively denoted by  $P_{k-1}$ ,  $B_{k-1}$  and  $L_{k-1}$ .

Where

$$P_k = B_k + L_k \quad (4.1)$$

The disturb  $L$  is represented as an autoregressive model of third order (AR(3)) which is a representation of a random process; as such, it is used to describe certain time-varying processes that in our case could consist of both demand and generation of power.

The autoregressive model specifies that the output variable depends linearly on its own previous values and on a stochastic term (an imperfectly predictable term); thus the model is in the form of a stochastic difference equation.

The general form of AR(3) model is:

$$y_t = c + \delta_1 y_{t-1} + \delta_2 y_{t-2} + \delta_3 y_{t-3} + \epsilon_t \quad (4.2)$$

where  $\delta_1$ ,  $\delta_2$ ,  $\delta_3$  are the parameters of the model,  $c$  is a constant, and  $\epsilon_t$  is white noise [7].

- The value of the set-point to match, denoted by  $P_k^*$ , is retrieved from the scenario  $\hat{P}_0, \hat{P}_1, \hat{P}_2, \dots, \hat{P}_{N-1}$  as:

$$P^* = \hat{P}_{\lfloor \frac{k}{30} \rfloor} \quad (4.3)$$

where  $\lfloor \cdot \rfloor$  indicates the nearest lower integer of the argument and 30 is the number of 10-seconds intervals in a 5-minutes slot [6].

- The  $k$ -index of the first 10-seconds interval is named  $\underline{k}$  and is defined as:

$$\underline{k} = \left\lfloor \frac{k}{30} \right\rfloor \times 30 \quad (4.4)$$

Instead, the  $k$ -index of the last 10-seconds interval for the current 5-minute slot is [6]:

$$\bar{k} = \underline{k} + 30 - 1 \quad (4.5)$$

Figure 4.1 illustrates in a clearer way the nomenclature, it shows the situation at the beginning of the time interval  $k = 2$ , that means the second 10-seconds period of 5-minutes slot, where the cell power set-points  $B_0^0$  and  $B_1^0$  were actuated already in the previous two intervals,  $B_2^0$  has been just determined using recent information and the average presumption set-point to achieve in the 5-minutes interval is given by the first value of scenario  $\hat{P}_0$  [6].

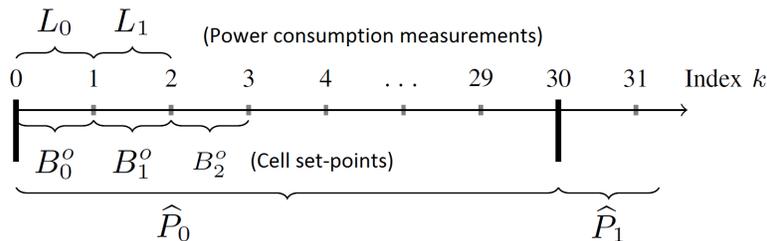


Figure 4.1: Representation of first 31 10-seconds period of operation, it is sketched the situation at the beginning of time interval corresponding to  $k = 2$  [6].

## 4.2 Control objective formulation

The control objective that we are going to describe in this paragraph is valid for both MPC and feedback control, with some differences described in detail later.

At the beginning of each time interval  $k$ , the average composite power flow for the current 5-minutes slot is given by averaging the available information until  $k$ . If  $k$  corresponds to the beginning of a 5-minute period, no information is available yet, and we say that the average composite consumption is zero [6]. Namely, we define

$$P_k = \begin{cases} 0 & k = \underline{k} \\ \frac{1}{k-\underline{k}} \sum_{j=\underline{k}}^{k-1} (L_j + Bj) & k > \underline{k} \end{cases} \quad (4.6)$$

In model predictive control implementation we introduce two kinds of prediction of disturb  $\hat{L}_{k|k}$ ,  $\hat{L}_{k+1|k}$ ,  $\hat{L}_{k+2|k}$ ,  $\dots$ ,  $\hat{L}_{\bar{k}|k}$  that allow to calculate the expected average composite consumption for whole duration of the current 5-minutes slot.

In particular, we use:

1. a persistent predictor

$$\hat{L}_{j|k} = L_{k-1} \quad (4.7)$$

with  $j = k, \dots, \bar{k}$ ;

2. a predictor defined according to the definition of AR(3) in Eq. 4.2 without the stochastic term  $\epsilon$ , namely

$$\hat{L}_k = c + \delta_1 L_{k-1} + \delta_2 L_{k-2} + \delta_3 L_{k-3} \quad (4.8)$$

Hence, the expected average consumption accounting for the short-term prediction is

$$P_k^+ = \frac{1}{30} \cdot \left( (k - \underline{k})P_k + \sum_{j=\underline{k}}^{\bar{k}} \hat{L}_{j|k} \right) \quad (4.9)$$

We define also the energy error, expressed in Wh, that is the difference between the set-point and the realizations in the current 5-minute slot:

$$e_k = \frac{300}{3600} \cdot (P_k^* - P_k^+) \quad (4.10)$$

where 300 is the number of seconds in a 5 minutes interval, while 3600 are the seconds in an hour interval.

### 4.3 Model predictive control

In this paragraph we describe the MPC (Model Predictive Control) implementation, which cares to adjust the power injections of the cell to compensate the mismatch between the power set-point (scenario) and real-time realization.

MPC is an advanced method of process control consists in determining the control action for a given system by solving at each time step an optimization problem with updated information, where the system constraints are enforced by implementing prediction models in the optimization problem [6].

MPC rely on dynamic models of the process, obtained according the process described in the previous chapter. The strength of MPC is that it allows to optimize the current timeslot keeping future timeslots in account. This is achieved by optimizing a finite time-horizon, but only implementing the current timeslot. This means that MPC is able to anticipate future events and can take control actions accordingly [8].

Therefore, MPC is more suitable to solve an energy management problem. For example, considering that the short-term predictions indicate that there will be a mismatch between realization and scenario in the second half of a 5-minute time slot, the MPC could react in advance while respecting battery cell operational constraints thanks to enforcing them explicitly in the formulation. With MPC algorithms it is possible to control large scale systems with many control variables, and, most importantly, MPC provides a systematic method of dealing with constraints on inputs and states, even though it is simple to design and implement [9].

In our case MPC determines the current in order to determine the cell battery power injection to achieve zero tracking error by the end of each 5-minutes slot while respecting the DC voltage and current operational limits.

In the MPC optimization problem we adopt as decision variable the DC

current of battery cell which assumes positive values when the battery is charging and negative when it is discharging.

We use the current because it admits a convex equivalent formulation of the optimization problem [6].

### 4.3.1 Derivation of the transition matrices for MPC

The voltage of battery cell is modelled by using a linear electrical circuit, as shown in figure 3.3, its dynamic evolution can be expressed as a linear function of the battery current and SOC. By applying the transition matrices, which are developed starting from the voltage discrete state-space model representation, the battery voltage can be expressed by:

$$\mathbf{v}_{\bar{k}|k} = \phi^v x_k + \psi_i^v \mathbf{i}_{\bar{k}|k} + \psi_1^v \mathbf{1} + \psi_{SOC}^v \mathbf{SOC}_{\bar{k}|k} \quad (4.11)$$

where  $x_k$  is the state vector of the voltage model and it is known thanks to Kalman filtering (better described later).

In general, the transition matrices are computed considering the following discrete state-space representation:

$$x_{k+1} = Ax_k + Bu_k \quad (4.12)$$

$$y_k = Cx_k + Du_k \quad (4.13)$$

where  $x_k \in \mathbb{R}^n$  is the state vector at discrete time interval  $k$ ,  $u_k \in \mathbb{R}$  is the input,  $y \in \mathbb{R}$  is the system output,  $A$  is the  $n \times n$  system matrix,  $B$  is the  $n \times 1$  input matrix,  $C$  is the  $1 \times n$  output matrix, and the scalar  $D$  is the feed-forward matrix [6].

For simplicity, we take in consideration the case with only one input signal, and we will complete after for the case with multiple inputs.

According to Eq. 4.12, the evolution of the state vector  $x$  from a known initial state  $x_0$  as a function of a given input sequence  $u_0, u_1, \dots, u_N$  is:

$$x_1 = Ax_0 + Bu_0 \quad (4.14)$$

$$x_2 = Ax_1 + Bu_1 = A(Ax_0 + Bu_0) + Bu_1 = A^2x_0 + ABu_0 + Bu_1 \quad (4.15)$$

iterating until  $k = N$

$$x_N = A^N x_0 + A^{N-1} Bu_0 + \dots + A^0 Bu_{N-1} \quad (4.16)$$

Then, we apply the output equation in order to find:

$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^N \end{bmatrix} x_0 + \begin{bmatrix} D & 0 & \cdots & 0 & 0 \\ CA^0B & D & \cdots & 0 & 0 \\ CA^1B & CA^0B & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ CA^{N-1}B & CA^{N-2}B & \cdots & CA^0B & D \end{bmatrix} \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{N-1} \\ u_N \end{bmatrix} \quad (4.17)$$

which can be written in compact form as:

$$\mathbf{y} = \phi x_0 + \psi_u \mathbf{u} \quad (4.18)$$

where  $\mathbf{y} = y_0, y_1, \dots, y_N$  and  $\mathbf{u} = u_0, u_1, \dots, u_N$ .

In the case there are multiple inputs, we add an input  $\mathbf{p} = p_0, \dots, p_N$  to the space model:

$$x_{k+1} = Ax_k + B_u u_k + B_p p_k \quad (4.19)$$

$$y_k = Cx_k + D_u u_k + D_p p_k \quad (4.20)$$

Then, the system output is written by applying the transformation  $\psi_p$  to  $\mathbf{p}$ :

$$\mathbf{y} = \phi x_0 + \psi_u \mathbf{u} + \psi_p \mathbf{p} \quad (4.21)$$

For our system we have four transition matrices, namely  $\phi^v$ ,  $\psi_i^v$ ,  $\psi_1^v$  and  $\psi_{SOC}^v$ .

### 4.3.2 Formulation and implementation

Before to introduce the formulation of the MPC, it is necessary define the battery cell energy throughput (in Wh) in the discretized time period from  $k$  to  $\bar{k}$ :

$$E_{\bar{k}|k}(v_k, \dots, v_{\bar{k}}, i_k, \dots, i_{\bar{k}}) = \alpha \sum_{j=k}^{\bar{k}} v_j i_j \quad (4.22)$$

where  $v_k$  and  $i_k$  are the battery voltage and current (positive when charging and vice-versa), respectively, and the scale factor  $\alpha = 10/3600$  is to convert from power (in W) in the discretized 10 seconds time interval to energy (in Wh).

The equation 4.22 can be expressed as a matrix product:

$$E_{\bar{k}|k}(\cdot) = \alpha \mathbf{v}_{k|\bar{k}}^T \mathbf{i}_{\bar{k}|k} \quad (4.23)$$

where the bold notation denotes sequences obtained by stacking in column vectors the realizations in time of the referenced variables.

Replacing Eq. 4.11 into Eq. 4.23 we find:

$$\begin{aligned} E_{\bar{k}|k}(\mathbf{i}_{\bar{k}|k}) &= \alpha \cdot \left( \phi^v x_k + \psi_i^v \mathbf{i}_{\bar{k}|k} + \psi_1^v \mathbf{1} + \psi_{SOC}^v \mathbf{SOC}_{\bar{k}|k} \right)^T \mathbf{i}_{\bar{k}|k} = \\ &\alpha \cdot \left( x_k^T \phi^{vT} \mathbf{i}_{\bar{k}|k} + \mathbf{i}_{\bar{k}|k}^T \psi_i^{vT} \mathbf{i}_{\bar{k}|k} + \mathbf{1} \psi_1^{vT} \mathbf{i}_{\bar{k}|k} + \psi_{SOC}^v \mathbf{SOC}_{\bar{k}|k} \mathbf{i}_{\bar{k}|k} \right) \end{aligned} \quad (4.24)$$

where  $\mathbf{1}$  indicates the all ones vector.

We want that  $E_{\bar{k}|k}$  is a convex function of  $\mathbf{i}_{\bar{k}|k}$ , then we have to check if it satisfies the proprieties of convex function.

Since the non-negative sum between functions preserves convexity, to ensure the convexity of Eq. 4.24 we need that all its four addends are convex. In particular, we have that the first, the third and the fourth addend are linear in  $\mathbf{i}_{\bar{k}|k}$ , therefore convex. Instead, the second term is a quadratic form of  $\mathbf{i}_{\bar{k}|k}$ , the necessary and sufficient condition for its convexity is given by  $\psi_i^v$  being

semi-definite positive. This hypothesis has been verified numerically for all the battery cell identified voltage models, consequently,  $E_{\bar{k}|k}$  is convex in  $\mathbf{i}_{\bar{k}|k}$ .

### 4.3.3 Optimization problem

As mentioned so far, the energy tracking problem consist in achieving a zero tracking error at the end of each 5-minutes slot. It is formulated by minimizing the squared deviation between  $E_{\bar{k}|k}(\mathbf{i}_{\bar{k}|k})$  and  $e_k$ , namely

$$\left(E_{\bar{k}|k}(\mathbf{i}_{\bar{k}|k}) - e_k\right)^2 \quad (4.25)$$

The energy tracking problem can be formulated as convex optimization problem, which is the combination of a linear cost function with an inequality constraint in the form  $f(x) \leq 0$ , where  $f$  is convex in  $x$ .

As known from the function's composition rules Eq. 4.23, the convexity of  $p(x) = q(r(x))$  when  $r(x)$  is convex requires  $q$  convex non decreasing, which is not this case because the squared function of the difference is convex but not non-decreasing on all its domain. Therefore, we reformulate the objective and achieve a convex equivalent formulation of the original problem: it consists in maximizing the cell DC current while imposing that the convex cell energy throughput Eq. 4.24 is smaller than or equal to the energy target Eq. 4.10; this achieves the energy throughput to hit the upper bound of the inequality, thus achieving the same value as the target energy.

In order this equivalent formulation to hold, the cell energy throughput must be a monotonically increasing function of the current [6].

We formulate the MPC optimization problem by augmenting the just described formulation with:

- constraints on the cell current;
- open open-loop predictive constraints on cell voltage.

It is worth noting that the main goal of control problem is determining a control decision which is respectful of cell operation constraints. Precisely, the decision problem is

$$\mathbf{i}_{\bar{k}|k}^0 = \arg \max_{\mathbf{i} \in \mathbb{R}^{(k-\bar{k}+1)}} \{\mathbf{1}^T \mathbf{i}_{\bar{k}|k}\} \quad (4.26)$$

subject to

$$\alpha \left( x_{\bar{k}|k}^T \phi^{vT} \mathbf{i}_{\bar{k}|k} + \mathbf{i}_{N|t}^T \psi_r^{vT} \mathbf{i}_{\bar{k}|k} + \mathbf{1}^T \psi_r^{vT} \mathbf{i}_{\bar{k}|k} + \psi_{SOC}^v \mathbf{SOC}_{\bar{k}|k} \mathbf{i}_{\bar{k}|k} \right) \leq e_k \quad (4.27)$$

$$\mathbf{1} \cdot i_{min} \preceq \mathbf{i}_{\bar{k}|k} \preceq \mathbf{1} \cdot i_{max} \quad (4.28)$$

$$\mathbf{v}_{\bar{k}|k} = \phi^v v_k + \psi_i^v \mathbf{i}_{\bar{k}|k} + \psi_1^v \mathbf{1} + \psi_{SOC}^v \mathbf{SOC}_{\bar{k}|k} \quad (4.29)$$

$$\mathbf{1} \cdot v_{min} \preceq \mathbf{v}_{\bar{k}|k} \preceq \mathbf{1} \cdot v_{max} \quad (4.30)$$

where  $\mathbf{i}_{\bar{k}|k}^0 \in \mathbb{R}^{(k-\bar{k}+1)}$  is the computed control action trajectory,  $\mathbf{1}$  denotes the all-ones column vector, the multiplication  $\mathbf{1} \cdot \gamma$  denotes the all- $\gamma$  column vector, and the symbol  $\preceq$  is the component-wise inequality.

The cost function in Eq. 4.26 consists in maximizing the sum of the equally weighted current values over the shrinking horizon from  $k$  to  $\bar{k}$ . This, in combination with the inequality of Eq. 4.27, achieves the cell energy throughput to be as close as possible to  $e_k$  [6].

The inequalities of Eq. 4.28 enforces minimum and maximum magnitude for cell current, where  $i_{min}$  and  $i_{max}$  are respectively the lower and the upper limits. Instead, the equality of Eq. 4.29 is the electrical equivalent circuit model of the cell according to the notation previously discusses for Eq. 4.11, while Eq. 4.30 imposes cell voltage limits, which are denoted by the couple  $v_{min}$  and  $v_{max}$ .

The optimization problem, adopted for the formulation of MPC, is convex since the cost function is linear and all the inequality constraints are convex in  $\mathbf{i}_{\bar{k}|k}$ .

Worthy of note is that we should adopt the real power injection as the optimization variable, but in this case the problem would not have been convex

because the cell voltage evolution is non-linear in the power and thus the constraints in Eq. 4.11 would have been nonconvex [6].

The optimization problem is solved at each time step  $k$  (with updated information) on a shrinking horizon from the index  $k$  to  $\bar{k}$ , namely from current time until the end of the current 5-minute slot. At each  $k$ , the control trajectory for the whole residual horizon is available, however only the first component of the current control law is considered for actuation, which we indicate by  $i_k^0$  [6]. Since the cell power flow is controlled by using a real power reference signal, it is required to transform from  $i_k^0$  to the power set-point  $B_k^0$ . By using the same model applied in Eq. 4.22, it is:

$$B_k^0 = v_k \cdot i_k^0 \tag{4.31}$$

## 4.4 State estimation

As shown in section 3.2.1 the system states are the voltage drops on capacitors  $C_1$  and  $C_2$ , i.e. the voltage drops on two  $RC$  parallel branches of next figure.

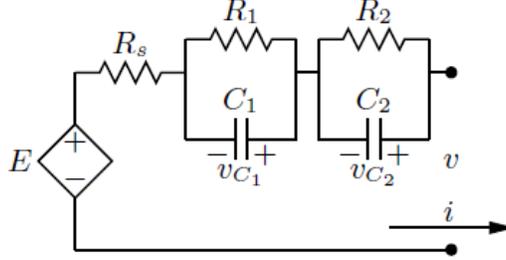


Figure 4.2: Equivalent circuit of battery cell [3].

Since the circuit is an abstract model, the states cannot be measured, and we have to estimate them in order to compute the cell voltage predictions. Therefore, the states are estimated from measurements of the battery DC voltage by applying Kalman filtering.

The estimation consists in a two-stage procedure, repeated at each discrete time interval: a prediction step to determine the system evolution (state expected value and covariance matrix  $P$ ) solely on the basis of the knowledge on the system [10]

$$x_{k|k-1} = \mathcal{A}x_{k-1|k-1} + \mathcal{B}u_{k-1} \quad (4.32)$$

$$P_{k|k-1} = \mathcal{A}P_{k-1|k-1}\mathcal{A}^T + \mathcal{K}\mathcal{K}^T \quad (4.33)$$

and an update stage, where the predicted state is corrected accounting for the last measurement  $v_k$

$$x_{k|k} = x_{k|k-1} + G(y_k - Cx_{k|k-1}) \quad (4.34)$$

$$P_{k|k} = (P_{k|k-1}^{-1} + C^T\sigma_g^{-1}C)^{-1} \quad (4.35)$$

where  $G$  is the Kalman gain:

$$G = P_{K|k-1}C^T(CP_{K|k-1}C^T + \sigma_g^2)^{-1} \quad (4.36)$$

and  $\sigma$  is the measurement noise (known from the parameters' estimation). Kalman filtering requires full system observability, that in our case is enforced by construction since the model is estimated from measurements [6].

Besides, we exploit the estimate states to estimate the state of charge of battery cell, exploiting the discharge characteristic in figure 3.2 in section 3.1. Considering the circuit in figure 4.2, we can estimate the open-circuit voltage  $v_{oc}$  in order to obtain the value of SOC through the  $v_{oc}$ -SOC curve. In particular, once are known the states (i.e. the voltages  $v_{C1}$  and  $v_{C2}$ ), the measured current  $i$  and voltage  $v$ , and the value of  $R_s$  we can estimate  $E$  (i.e.  $v_{oc}$ ) according to Kirchhoff's voltage law. To estimate the state of charge, we create a Matlab function which takes as input the measured voltage and current, the state of charge at previous step, the different values of  $R_s$  and the discharge characteristic and it gives as output the estimated SOC. It worth noting that, at the first step the open circuit voltage is measured since the system is in steady state condition.

## 4.5 Feedback control and control actions

Another control strategy for energy management is implemented, namely a feedback control loop strategy for which the control action depends only on current and past values and it does not allow to schedule the whole power trajectory within the targeted time horizon as for MPC [6].

The real time operation and the control objective in detail described respectively in section 4.1 and 4.2, are also valid for feedback control loop with the difference that for feedback control it is not used a prediction of disturb. Moreover, feedback control does not need a model identification and it computed the value of current in a simpler way without solve an optimization problem.

Below, the control action are reported, specifying the difference between MPC and feedback control. The control strategy consists in the following action:

1. at the beginning of experiment the battery cell is in steady state, namely the DC current is zero and the measured voltage  $v_m$  correspond to the open circuit voltage  $v_{oc}$ ;
2. once measured  $v_{oc}$ , the SOC is estimated by means of discharge characteristic;
3. known the initial condition,  $P^*$  is computed according Eq. 4.3 from the scenario  $\hat{P}$ , then the values of  $\bar{k}$  and  $\underline{k}$  become known;
4. thus,  $P_k^+$  and  $e_k$  are computed according respectively to Eq. 4.9 and Eq. 4.10;
5. hence, the current set-point to send is computed and it is sent to the battery cell. For the MPC, the current is computed solving the optimization problem presented before, instead for the feedback control we compute,

first the error  $B_i$

$$B_i = P^*(k - \underline{k}) - \sum_{j=\underline{k}}^k P_j \quad (4.37)$$

where  $B_i$  is the difference between the set-point and the realization, then we compute the current as  $i = B_i/v_m$ . The computed current enters in a saturation block with bounds setted at  $\pm 30A$ , then the measured voltage  $v_m$  is compared with the imposed voltage bounds and if it is bigger than one of this the current is forced to zero to avoid burning the battery cell.

6. the current computed is sent to battery cell;
7. the SOC value is updated thanks to voltage and current information from battery cell;
8. finally, the value of  $P_k$  (Eq. 4.1) is updated considering the value of  $B_m = i_m \times v_m$ , where  $i_m$  and  $v_m$  are the voltage and the current measured after the application of set point, and the disturb  $L_i$ .

Since the control decision is re-evaluated every 10 seconds, errors on the voltage predictions which arise in the current actuation period are absorbed in the next cycle, where updated measurements are used [6]. As mentioned above, the battery cell is commanded through the *LabVIEW* VI shown in figure 2.1, while the code of control strategy was written using Matlab.

To create a communication between the two software we exploited a TCP connection allowing Matlab to receive the measurements from the data acquisition system, to execute the control strategy and to send to *LabVIEW* the computed current set-point.

# Chapter 5

## Experimental results

The main reason whereby we implemented two kinds of control is to point out that MPC is more suitable for energy management than feedback control.

To compare the performance of two control strategy we analyse the following cases:

1. as mentioned previously, one of the goals of the control is to assure that voltage and current bounds are not violated. For this reason we created a scenario with only positive value of power, that means to charge constantly the battery cell till to the cell is charged and the control should limit the current in order to respect the bounds imposed, namely
  - (a) maximum current:  $i_{max} = \pm 30$  A
  - (b) maximum voltage:  $v_{max} = 2.55$  V
  - (c) minimum voltage:  $v_{min} = 1.80$  V

It is worth noting that the maximum voltage for the battery cell is 2.7 V according to data sheet, but we choose a value a little lower for security reason, in order to not ruin the cell battery in the event that control action fails. The same is done for the lower bound. Instead, the maximum current is chosen equal to 30 A in order to work with a C-rate equal to or lower than 1.

2. we compare the performance of two control in terms of tracking error, namely comparing power set-point with the value of power reach at the end of 5-minutes slot.

To perform the experiment it is necessary to establish the initial condition, which must be the same for both cases. Furthermore, the scenario must also be the same in order that the two controls work under the same conditions.

## 5.1 Experiment 1

The first experiment starts with the battery cell in steady-state and with a SOC equal to 81%. We apply to the cell a constant positive set-points, namely we want to charge the cell in order to force the MPC to work near the upper voltage bound.

In figure 5.1 there are four charts reporting the results experiment. In the first and in the second chart, the dotted red lines represent respectively the upper voltage bound ( $2.55V$ ) and the current bounds ( $\pm 30A$ ). For the same charts, the blue line represents the voltage trend while the green one is the current trend during the experiment.

In the same figure, the third chart is the evolution of error, namely the difference between the set-point imposed and the realization, and the fourth chart report the set-point (blue) and the realization (red).

As we can see from results, the MPC is able to follow the trajectory of power setted until the cell voltage reaches the value of imposed bound. Once the bound is reached, the tracking error starts to increase because the MPC limits the value of output current in order to respect the imposed voltage bound.

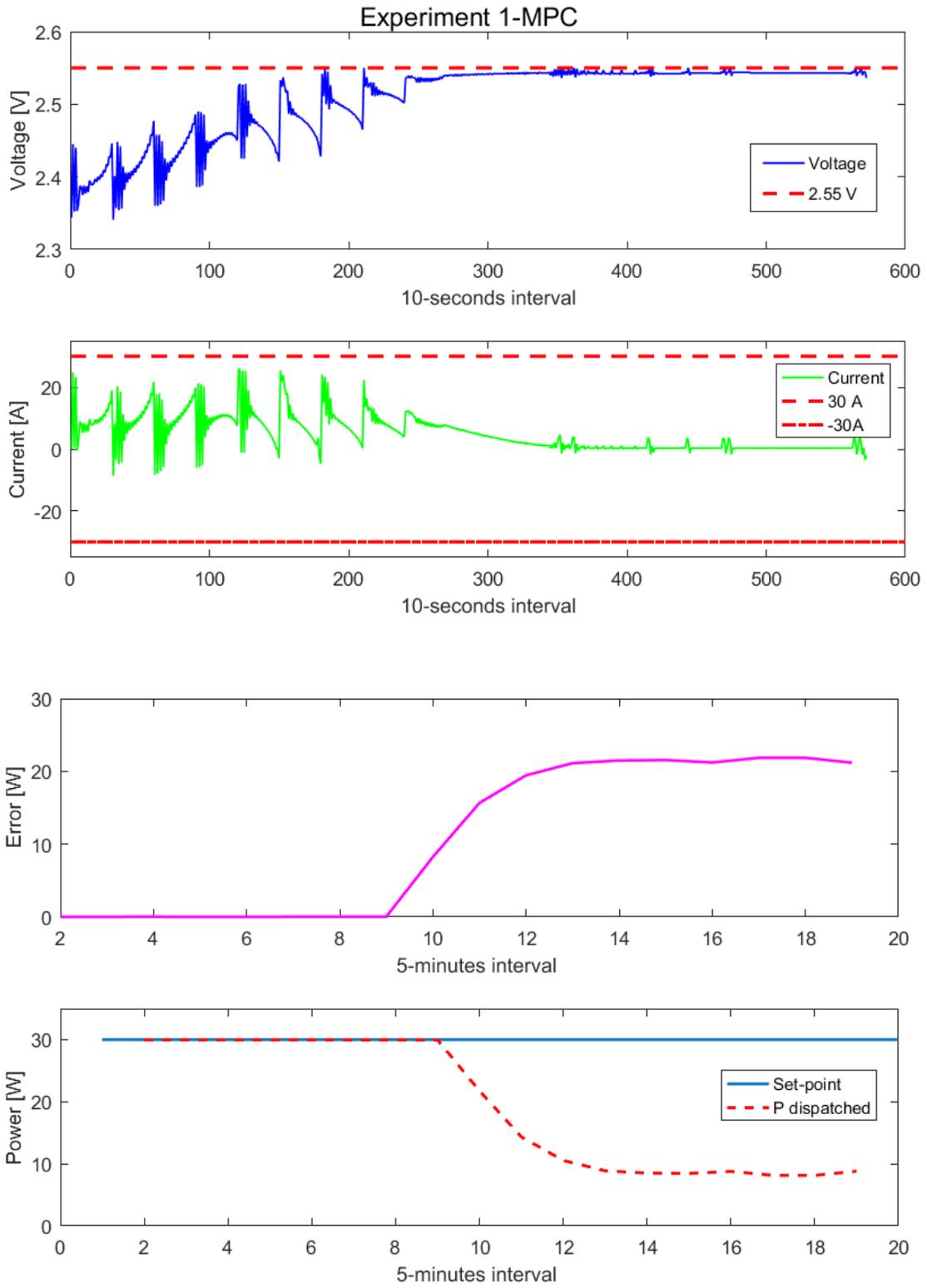
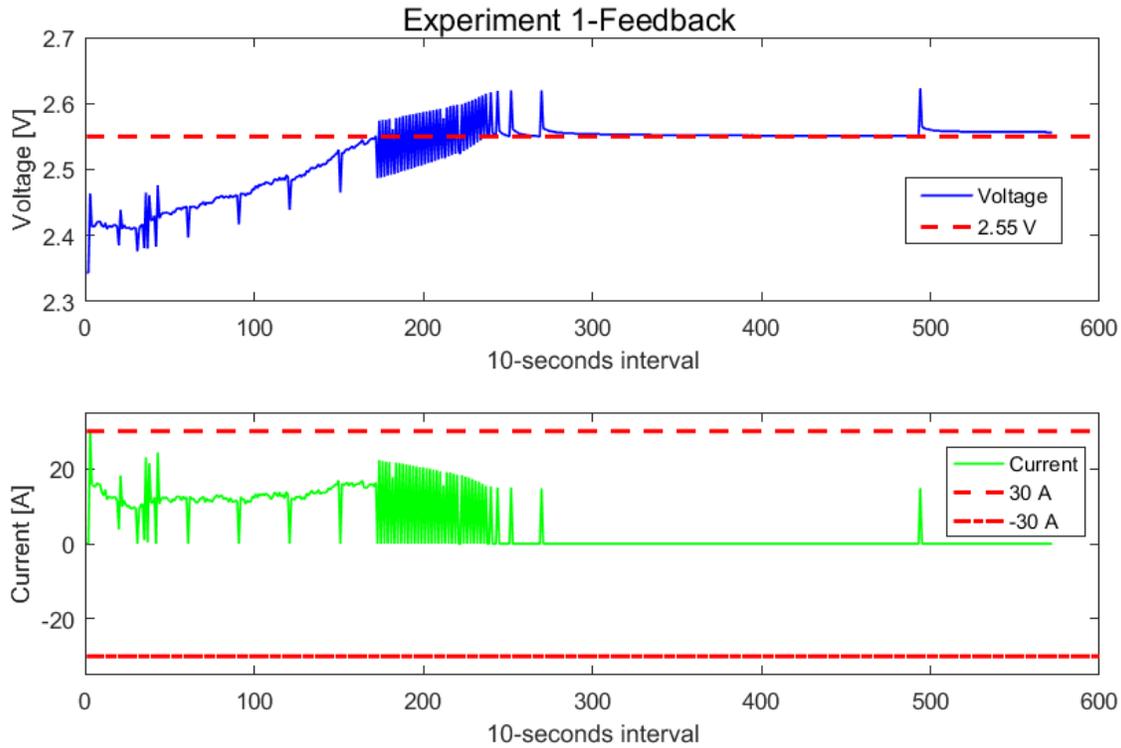


Figure 5.1: Experiment 1 - MPC results.

The same experiment, with the same initial condition and the same set-points, was done using a feedback control instead of MPC.



As the next and the figures above show, the feedback control is not able to respect the imposed voltage bound, overcoming it for a few moments. Moreover, it worth to note that the tracking error starts to increase earlier and it reaches higher values compared to the MPC case.

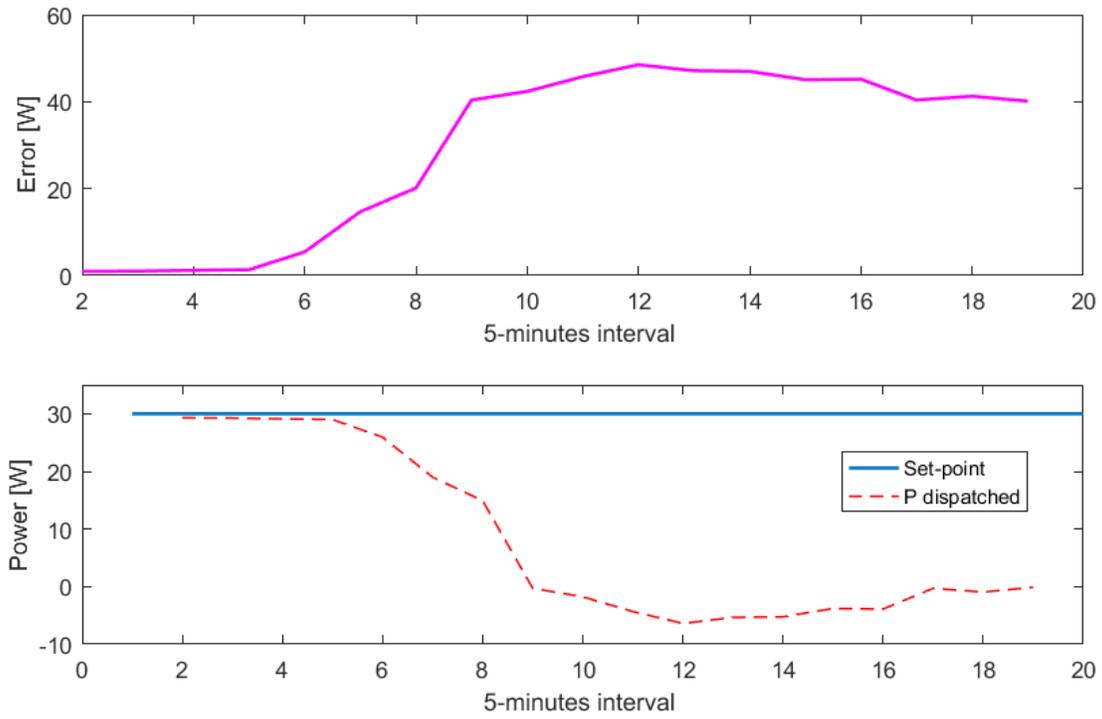


Figure 5.2: Experiment 1 - Feedback results.

This two experiments demonstrate that model predictive control has better performance, in terms of tracking error and respectability of bounds, than the feedback control loop when we force the control to work near the bounds.

## 5.2 Experiment 2

We did other experiments to compare the performance of two kinds of control in terms of tracking error at the end of each 5 minute interval.

The experiment starts with the battery in a steady state with a SOC of 51%. We applied at the battery a scenario composed by positive and negative values of power.

The experiment was done once using once the feedback control and once using MPC. In this case, for the MPC we show two cases: one using a persistent predictor and one using the predictor based on AR(3) process framework.

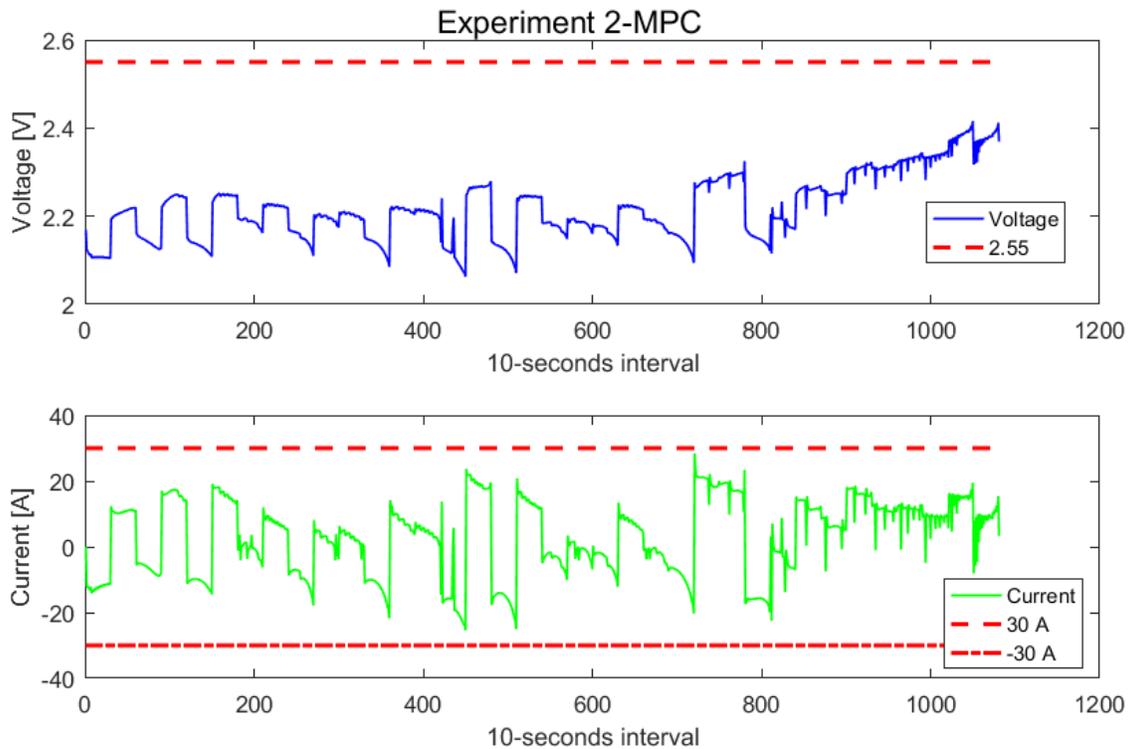


Figure 5.3: Experiment 2- MPC with AR(3) predictor, voltage and current trends.

Figure 5.3 shows the voltage and current trend during the experiment. As it is possible to see, both the voltage and current limits are respected for the duration of experiment. Whilst, what is most significant for this experiment

is the value of tracking error.

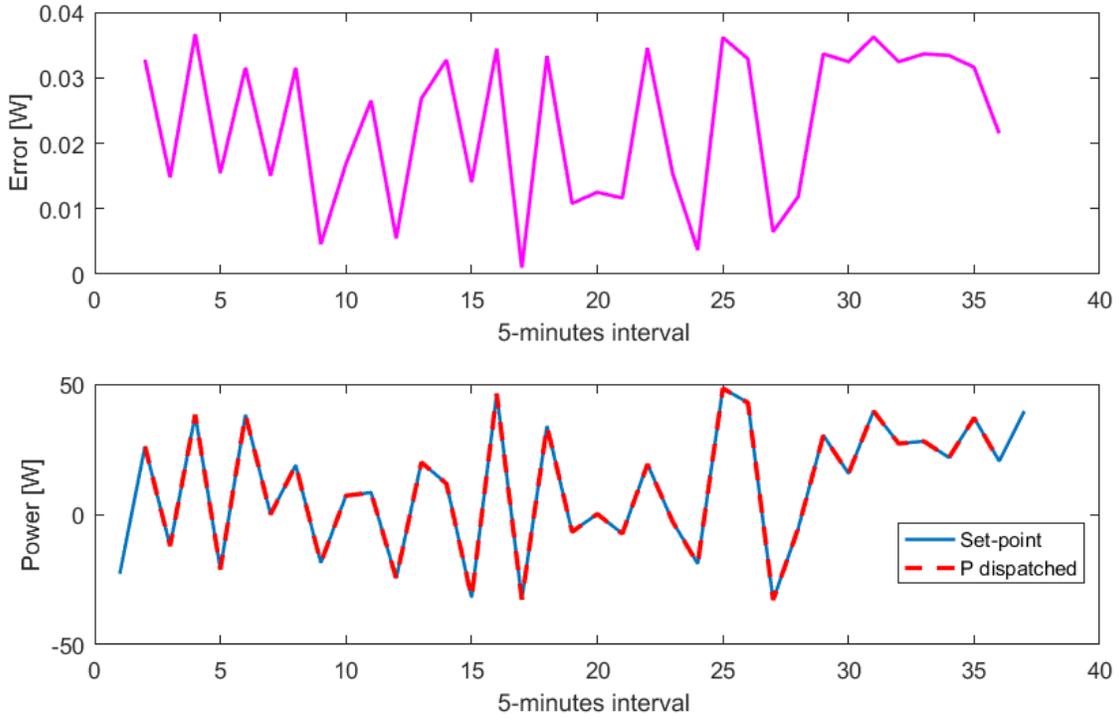


Figure 5.4: Experiment 2- MPC with AR(3) predictor, error and power trends.

In particular, as figure 5.4 shows, the error value is included between  $9,777 \times 10^{-4}$  W and 0.036 W with a mean value equal to 0.023 W.

At the end of experiment described, the battery cell was discharge until to reach again a SOC value equal to 51. Then, we wait until it reaches the steady state and the experiment start again with feedback control strategy. Figure 5.5 reports the results obtained by feedback control. It is worth noting that in this case the error is higher than former, indeed the error is included in the range  $[-3.279, 4.0262]$  W, with a mean value of 0.499 W

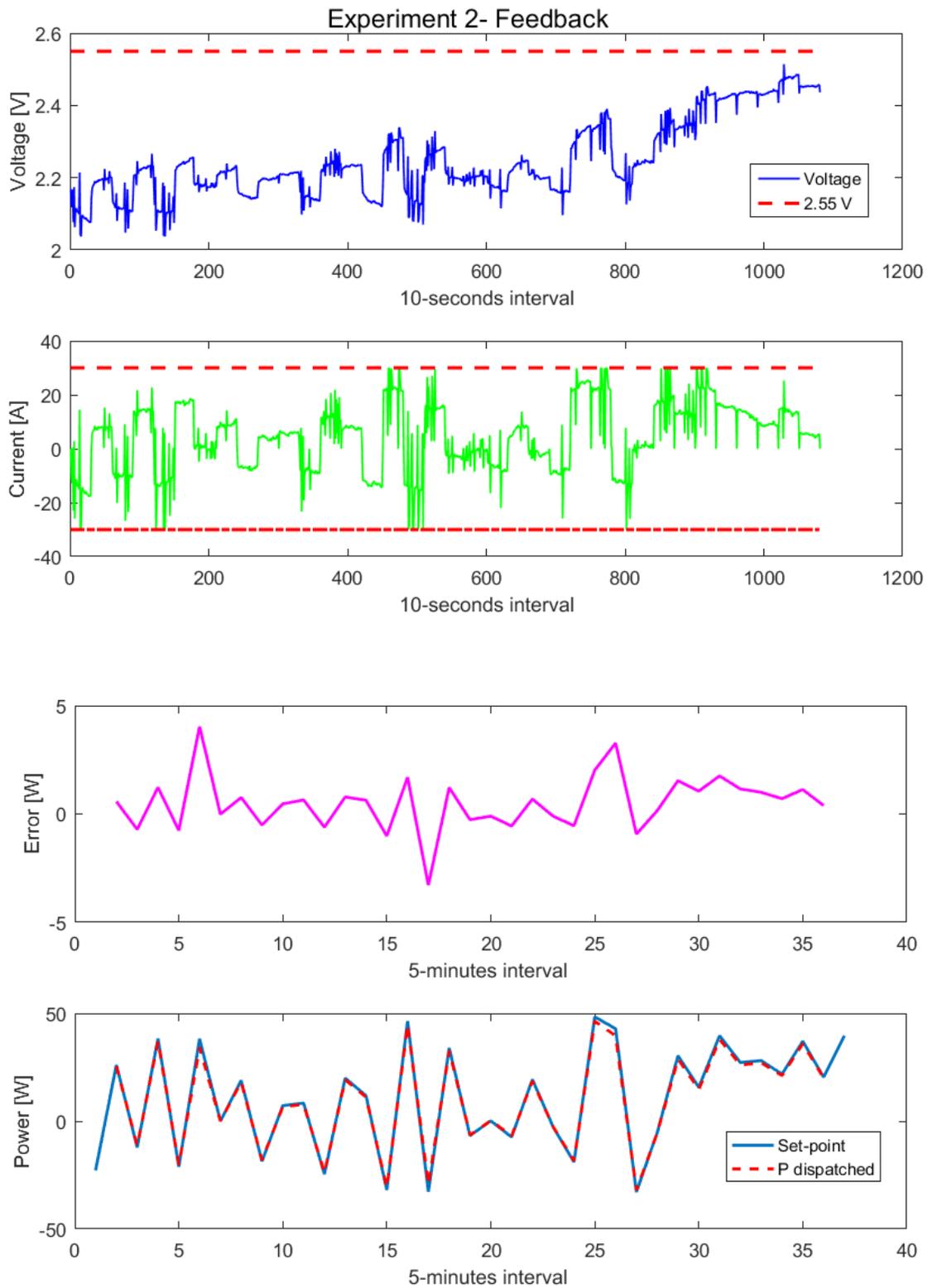
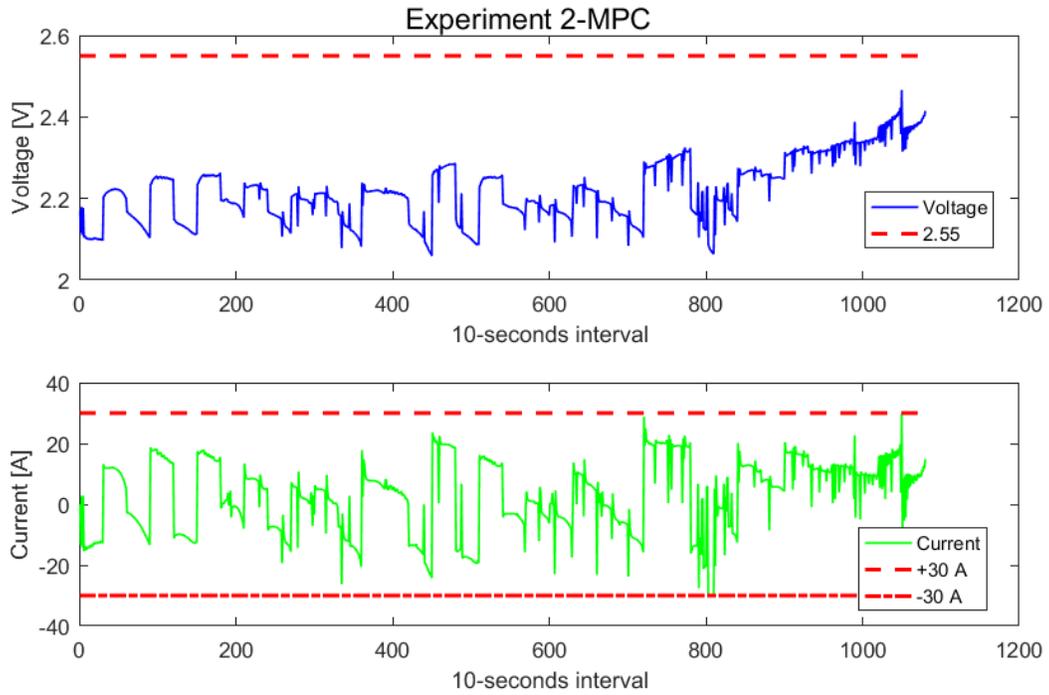


Figure 5.5: Experiment 2- Feedback results.

Finally, we repeat the experiment (same initial condition and same scenario) using MPC with persistent predictor.



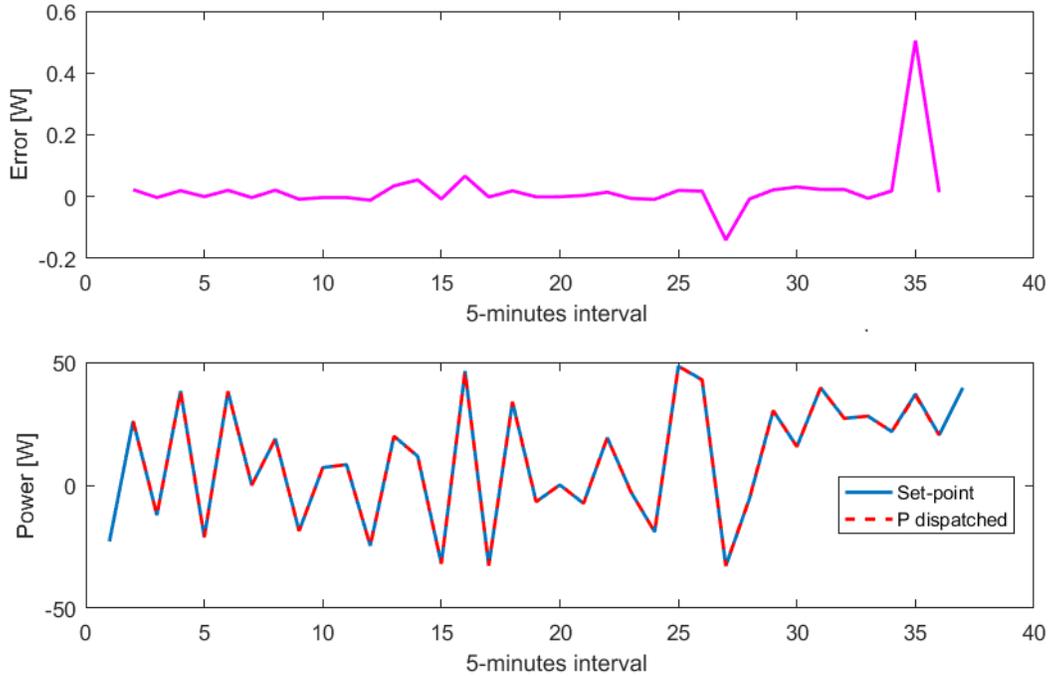


Figure 5.6: Experiment 2- MPC with persistent predictor results.

In this case, the current and voltage bounds are still respected, but figure 5.6 shows the performance of control decrease using the persistent predictor instead of AR(3) one.

We can summarize the results found in terms of tracking error in the following table.

Error	MPC persistent predictor	MPC AR(3) predictor	Feedback	Unit of measure
MAX	$5,05 \times 10^{-1}$	$3.67 \times 10^{-2}$	4.0262	[W]
MIN	$-1.402 \times 10^{-1}$	$9.77 \times 10^{-4}$	-3.279	[W]
MEAN	$2,3 \times 10^{-2}$	$2.12 \times 10^{-2}$	$4.99 \times 10^{-1}$	[W]

Table 5.1: Experiment 2 - results.

Comparing the results of the first and second column, it emerges that the MPC with AR(3) predictor has a better performance than the MPC with persistent predictor, namely the max value of error is smaller of an order of magnitude, while the min value is smaller of three order of magnitude. This

means that the performance of control depend on accuracy of prediction of disturb, i.e. a better prediction model of disturb gives better performance.

Besides, feedback control has significantly worse performance, in fact the error assumes a max value around  $4 W$  and a minimum around  $3 W$ , respectively two and four order of magnitude higher than the MPC case with AR(3) predictor.

Moreover, as table reports, MPC with persistent predictor has also better performance of feedback control.

# Chapter 6

## Conclusion

We have discussed two control frameworks for the energy management of a battery cell, in particular we adopted a Li-ion cell with a nominal capacity of 30 Ah.

The control strategy is used to compensate the mismatch between a dispatched power trajectory based on forecast and a stochastic realization and scenario. We implemented a feedback control and a model predictive control to solve the energy management decision problem, comparing the performance of two kinds of control.

We improved the measurement set-up in order to have a more accurate current measure introducing a shunt resistor, since the previous set-up had an offset problem. Prediction models that are applied in the problem are identified from measurements applying grey-box modelling; we obtained five models, one for each range of state of charge, i.e. 0 – 20%, 20 – 40%, 40 – 60%, 60 – 80%, 80 – 100%.

Then, we adapted from the existing literature the convex optimization problems which were developed as a part of the MPC control framework. In particular, the MPC allows formulating the battery cell energy throughput in the objective function while retaining the linearity of the expressions of the DC voltage and current constraints.

Hence, we developed the feedback control loop with the same control objective as the MPC, but without implementing the predictive layer in the cost function and constraints.

The two control frameworks are validated through various experiments, that are repeated both for MPC and for feedback control, applying the same scenario and starting the experiment under the same initial conditions.

Through the experiments, we demonstrate that MPC is more suitable for energy management than feedback control since MPC is able to optimize the current timeslot, while keeping future timeslots in account. This allows for energy management to follow the scenario respecting voltage and current bounds with a good tracking performance.

Future works concern the improvement of predictive models, considering the effect of temperature and of C-rate. Moreover, it could be possible to implement the control strategy on LabVIEW or with some other tool in order to not use the TCP connection to exchange the data allowing to reduce the computation time therefore the control performance.

# Bibliography

- [1] Datasheet of NI 9215 [http://www.ni.com/pdf/manuals/373779a\\_02.pdf](http://www.ni.com/pdf/manuals/373779a_02.pdf)
- [2] Fabrizio Sossan, Bindner Henrik W., Nørgård Per Bromand  
*Indirect control of flexible demand for power system applications.*2014
- [3] Fabrizio Sossan, Emil Namor, Scolari Enrica, Mario Paolone  
*Experimental Assessment of the Prediction Performance of Dynamic Equivalent Circuit Models of Grid-connected Battery Energy Storage Systems,*  
Distributed Electrical Systems Laboratory (DESL), EPFL Lausanne, Switzerland
- [4] Christophe Hurlin  
*Maximum Likelihood Estimation,* Advanced Econometrics - HEC Lausanne, University of Orleans, December 9, 2013
- [5] In Jae Myung  
*Tutorial on maximum likelihood estimation,* Journal of Mathematical Psychology 47 (2003) 90–100
- [6] Fabrizio Sossan, Emil Namor, Rachid Cherkaoui, Mario Paolone  
*Achieving the Dispatchability of Distribution Feeders through Prosumers Data Driven Forecasting and Model Predictive Control of Electrochemical Storage*  
IEEE TRANSACTIONS ON SUSTAINABLE ENERGY, VOL. 7, NO. 4, OCTOBER 2016

- [7] Sudhakar Pandit *Time Series and System Analysis with Applications* 5 maggio 1983
- [8] Michèle Arnold, Göran Andersson  
*Model Predictive Control for energy storage including uncertain forecasts*,  
ETH , Zurich, Switzerland
- [9] Mark Cannon  
*C21 Model Predictive Control* Lecture notes, 2016, University of Oxford
- [10] R. Stengel  
*Optimal Control and Estimation*. Dover Publications, 1994.