modelling

MDPI

Article

A Model-Based Strategy for Active Balancing and SoC and SoH Estimations of an Automotive Battery Management System

Lorenzo Breglio [1], Arcangelo Fiordellisi [1], Giovanni Gasperini [1], Giulio Iodice [1], Denise Palermo [1], Manuela Tufo [1,2,*], Fabio Ursumando [1] and Agostino Mele [1,3]

[1] Kineton S.r.l.-Società Benefit, 80146 Napoli, Italy; lorenzo.breglio@kineton.it (L.B.); arcangelo.fiordellisi@kineton.it (A.F.); giovanni.gasperini@kineton.it (G.G.); giulio.iodice@kineton.it (G.I.); denise.palermo@kineton.it (D.P.); fabio.ursumando@kineton.it (F.U.); agostino.mele@kineton.it (A.M.)
[2] Department of Engineering, University of Sannio, 82100 Benevento, Italy
[3] Department of Engineering, University of Campania Luigi Vanvitelli, 81031 Aversa, Italy
* Correspondence: manuela.tufo@unisannio.it

Abstract: This paper presents a novel integrated control architecture for automotive battery management systems (BMSs). The primary focus is on estimating the state of charge (SoC) and the state of health (SoH) of a battery pack made of sixteen parallel-connected modules (PCMs), while actively balancing the system. A key challenge in this architecture lies in the interdependence of the three algorithms, where the output of one influences the others. To address this control problem and obtain a solution suitable for embedded applications, the proposed algorithms rely on an equivalent circuit model. Specifically, the SoCs of each module are computed by a bank of extended Kalman filters (EKFs); with respect to the SoH functionality, the internal resistances of the modules are estimated via a linear filtering approach, while the capacities are computed through a total least squares algorithm. Finally, a model predictive control (MPC) was employed for the active balancing. The proposed controller was calibrated with Samsung INR18650-20R lithium-ion cells data. The control system was validated in a simulation environment through typical automotive dynamic scenarios, in the presence of measurement noise, modeling uncertainties, and battery degradation.

Keywords: lithium-ion batteries (LIBs); battery management system (BMS); state of charge (SoC); state of health (SoH); active balancing; extended Kalman filter (EKF); model predictive control (MPC); battery degradation

Citation: Breglio, L.; Fiordellisi, A.; Gasperini, G.; Iodice, G.; Palermo, D.; Tufo, M.; Ursumando, F.; Mele, A. A Model-Based Strategy for Active Balancing and SoC and SoH Estimations of an Automotive Battery Management System. *Modelling* **2024**, *5*, 911–935. https://doi.org/10.3390/modelling5030048

Received: 8 May 2024
Revised: 26 June 2024
Accepted: 4 August 2024
Published: 7 August 2024

1. Introduction

Lithium-ion batteries (LIBs) are an excellent energy storage device which play a pivotal role in supporting the transition from fossil-fueled vehicles to more environmentally friendly alternatives [1,2].

These types of batteries are integral components in various types of vehicles commonly referred to as "xEVs". Among xEVs, LIBs find applications in both battery electric vehicles (BEVs), where the battery is the sole source of power, and hybrid applications, where the battery cooperates with an alternative source of power, as in fuel cell electric vehicles (FCEVs), hybrid electric vehicles (HEVs), and plug-in hybrid electric vehicles (PHEVs). Therefore, given LIBs widespread importance in all xEVs, the development of efficient and reliable battery control and monitoring strategies is critical [3].

The battery management system (BMS) is the central control unit responsible for ensuring the optimal operation of the battery while protecting it against critical faults, including overcharging, overdischarging, overcurrent, cell short-circuits, and extreme temperatures. Furthermore, the BMS offers essential diagnostic capabilities, such as monitoring the battery state of health (SoH), and it plays a crucial role in performance management, which involves estimating the state of charge (SoC) of each battery cell, calculating the

available energy and power, ensuring charges are balanced among the cells, and several more features [4].

Thus, from a control system and machine learning perspective, a BMS needs to address several challenges. Among these, researchers have identified as critical problems the estimation of the battery SoC and SoH [5], and the cell balancing [6].

Various approaches have been explored to achieve satisfactory SoC estimation, as reviewed in [7] and more recently in [8]. These approaches range from simple methods, like look-up tables and ampere-hour integration, to more advanced techniques, such as model-based and data-driven strategies. When a model-based algorithm is employed to reconstruct the SoC, a proper estimation framework is required. Typical choices are Kalman-based filters, such as the extended Kalman filter (EKF), sigma point Kalman filter (SPKF), unscented Kalman filter (UKF) and dual Kalman filter (DKF), or particle filters (PFs). Finally, neural networks (NNs) are a popular alternative that involve training a black-box model with a dataset of battery measurements to predict the relationship between input parameters (such as current, voltage, and temperature) and the target state of charge (SOC) [9]. Physics-based modeling offers a well-balanced solution between the overly simplistic rule-based approaches (e.g., Coulomb counting and voltage methods) and the more complex NN algorithms. While NN algorithms provide satisfactory results, they require large amounts of data and offer poor physical interpretability of the estimated parameters [10]. One drawback of the physics-based approach combined with Kalman filtering is its demanding computational burden for real-time applications. However, with recent advances in hardware and estimation frameworks, several studies have shown promising results from a computational perspective as well [11–13].

However, accurately estimating SoC remains an ongoing challenge, particularly for low SoC values and cell chemistries that exhibit a weak correlation between the open circuit voltage (OCV) and the SoC [14].

The SoH estimation problem can be tackled using similar estimation frameworks. Lithium-ion battery degradation is an inevitable outcome of cycling, involving both mechanical and chemical degradation processes [15]. The degradation of lithium-ion batteries is primarily evident through a reduction in the maximum remaining capacity and an increase in the internal ohmic resistance. Therefore, two failure thresholds typically define the end of life (EOL) of the battery for automotive applications: (i) a 100% increase in internal resistance and (ii) a 20% decrease in capacity [16]. Thus, two different SoH values are defined: the first to account for the increase in the internal resistance R_0, and the second one for the drop in the capacity Q.

$$\begin{aligned} SoH_{R_0} &= \frac{R_{EOL} - R_0}{R_{EOL} - R_{BOL}} \\ SoH_Q &= \frac{Q}{Q_{rated}} \end{aligned} \tag{1}$$

where R_{BOL} and R_{EOL} are the internal resistances at the beginning and end of life, while Q_{rated} is the nominal capacity value at the beginning of life (BOL). Assuming R_{BOL}, R_{EOL}, and Q_{rated} are already known, the SoH problem resolves into estimating both R_0 and Q.

Finally, with respect to the balancing challenge, a passive balancing of the cell is the most widespread approach in current xEVs. However, implementing appropriate active balancing, which employs extra switching circuits to equalize the imbalances in the charge levels within the pack, has shown significant benefits in terms of system efficiency and lifespan extension [17].

While the coupled problem of estimating the SoC and SoH has received significant attention in the literature (e.g., [12,18–23]), as well as the integration of SoC estimation and active balancing (e.g., [24–29]), only a limited number of papers have presented results on integrating these three BMS functionalities within a comprehensive framework despite their close interdependence.

The authors in [30] emphasized the effectiveness of adopting an SoH-conscious balancing strategy over traditional approaches, resulting in a notable 6% improvement in the cell aging rate. In [31], the integration of active balancing with SoC and SoH algorithms is discussed. However, the employed balancing logic used is a simplistic rule-based approach, and the SoH estimation requires costly and faulty data for calibrating the coefficients of the proposed fading model. In [32], a Tenergy ICR 18650-2600 lithium-ion battery is utilized to validate an integrated algorithm that considers SoC, SoH, and active balancing. This solution leverages battery impedance computation and a neural network model. The approach offers several advantages, including increased operation time, improved energy utilization efficiency, and prevention of over-discharging. However, data-driven algorithms require numerous training data, and retrieving online the impedance information might not be practical.

In this paper, an integrated framework which deals with the reconstruction of the SoC and SoH of each module of an automotive battery pack is presented, together with their active balancing. The algorithms are model-based and they have been designed in the spirit of being deployable on a real electronic control unit (ECU) and having a satisfactory balance between development cost and performance. The SoC estimation is conducted with an EKF, while the resistance R_0 and the capacity Q are computed by means of a linear filtering and a proportional total least square (PTLS) approach, respectively; finally, the cell balancing is performed via a model predictive control (MPC). Temperature dependency is not accounted for in this paper and ideal thermal management is considered (constant temperature hypothesis). Nevertheless, the proposed BMS control architecture remains valid in the case of variable temperature conditions. This can be achieved by incorporating the temperature dependency into the BMS internal predictive models.

The key contributions of this paper are the following:

1. a novel model-based control system which encompasses SoC and SoH estimations and the active cell balancing of an automotive system is presented, whereas these control problems are typically addressed separately;
2. the proposed development framework is presented in a step-by-step fashion, covering the entire process from data acquisition to modeling, calibration, and algorithm testing;
3. the experimental data obtained during this study are shared as Supplementary Materials.

The research methodology illustrated in Figure 1 outlines the steps followed in this paper to develop the proposed model-based BMS algorithms and a testing environment to validate their functionalities and produce the relevant results. First, an appropriate control-oriented model structure was selected from the literature to serve as the foundation for the BMS algorithms. Simultaneously, an extensive set of measurements were gathered from a real battery cell to ensure that its internal state was sufficiently excited to calibrate and validate the selected model. The same data were used to design and tune the plant model for the implementation of the closed-loop testing environment. Next, the proposed BMS algorithms were designed to satisfy the challenging control and estimation problem requirements (i.e., high SoC and SoH accuracy, satisfactory controller dynamic behavior, and robustness against modeling uncertainties and sensor noise). Then, a closed-loop testing environment was implemented to evaluate the BMS algorithms using realistic driving cycle profiles and including the capability to inject faults into the plant model at runtime, allowing for the assessment of the BMS behavior in the presence of degradation. The testing environment served as a prototyping platform to refine the algorithm design, ultimately converging on the solution which is presented in this article.

The rest of the paper is organized as follows: In Section 2, the battery module model adopted by the control algorithms and the testing setup used for collecting the relevant experimental data are reported. Section 3 describes the formulation of the implemented control algorithms and provides an overview of the simulation environment which was developed for testing purposes. In Section 4, the results are presented and discussed. Finally, some conclusions and remarks complete the paper in Section 5.

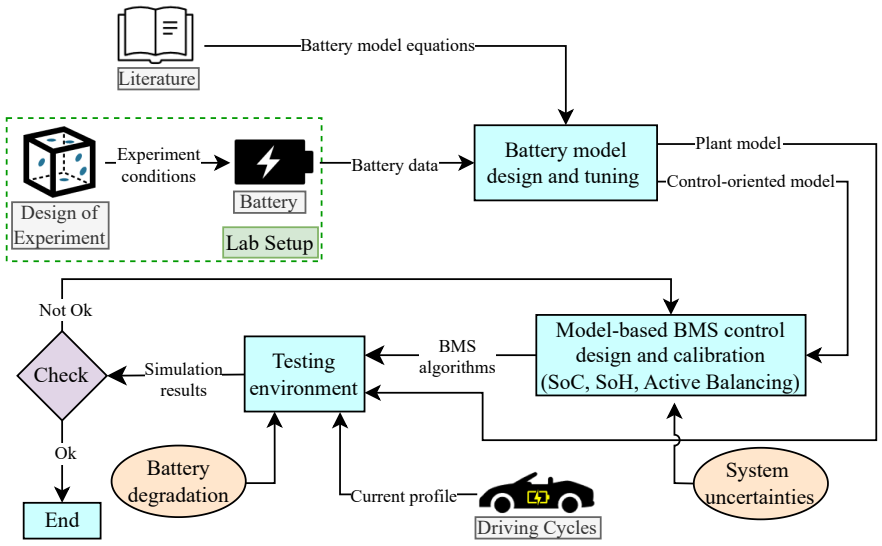

Figure 1. Research methodology.

2. Battery Modeling

The proposed BMS solution integrates three model-based algorithms; therefore, it strongly relies on the design of a control-oriented model which strikes a satisfactory balance between model complexity and accuracy. This model will be utilized by the BMS algorithm to perform the control and estimation tasks.

For this reason, an equivalent circuit model (ECM) representation, which is an established choice in the literature for BMS applications [33], was selected. ECMs leverage the electrical-circuit analogs to define a phenomenological approximation of the voltage response to various current stimuli.

In this paper, a special topology of ECM, named a Thevenin model [34], was adopted. A Thevenin model consists of resistors and capacitors only, as shown in Figure 2. A key advantage of Thevenin models is that they can easily be framed in a state-space representation, which enables the adoption of different mathematical methodologies and control theory approaches; therefore, they naturally fit with SoC and SoH estimation problems, cell balancing, and many more BMS functionalities.

The Thevenin model is represented by means of Equation (2).

$$\begin{cases} \dot{SoC} & = -\frac{I}{3600\,Q} \\ \dot{V}_1 & = -\frac{V_1}{R_1 C_1} + \frac{I}{C_1} \\ & \cdots \\ \dot{V}_n & = -\frac{V_n}{R_n C_n} + \frac{I}{C_n} \\ V & = OCV(SoC) - \sum_{i=1}^{n} V_i - R_0 I \end{cases} \tag{2}$$

where SoC is the state of charge; I is the current; Q is the capacity; V is the voltage; OCV is the open circuit voltage; R_0 is the ohmic internal resistance; R_i, C_i, and V_i are the polarization resistance, the polarization capacitance, and the voltage drop in the i-th RC pair, respectively; and n is the number of RC pairs. All the reported quantities are in S.I. units, except the capacity Q, which is expressed in Ah.

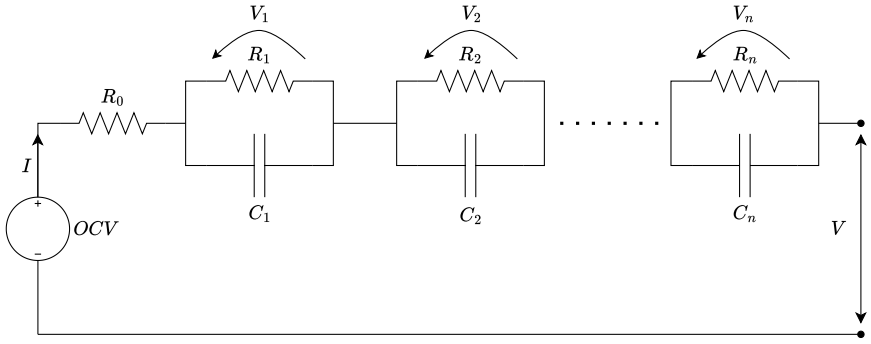

Figure 2. Thevenin model.

In general, each of the parameters OCV, $R_{0,\ldots,n}$, $C_{1,\ldots,n}$ is a function of the operating temperature and the actual SoC. For the sake of simplicity, we neglect the influence of the temperature, and we assume that only the OCV depends on the SoC, while the remaining parameters are constant. In particular, a polynomial correlation between the OCV and the SoC is assumed, as in [35].

$$OCV(SoC) = a_0 + \sum_{i=1}^{p} a_i \cdot SoC^i \tag{3}$$

Equations (2) and (3) can be compacted in the state-space representation reported in Equation (4), which is convenient for the modeling workflow and the control algorithms addressed in this paper

$$\begin{cases} \dot{x} = f(x, u \mid \theta) \\ x = [SoC; V_1; \ldots; V_n] \\ u = [I] \\ y = g(x, u \mid \theta) = [V] \end{cases} \tag{4}$$

where θ is the parameters vector, defined as $\theta = [a_0; \ldots; a_p; R_0; \ldots; R_n; C_1; \ldots; C_n]$.

Therefore, the model structure is fully defined once both the number of RC parallels (i.e., n) and the order of the OCV polynomial (i.e., p) are set. Finally, the model design was completed by calibration of the parameters vector θ.

The ECM model structure presented in Equation (4) is versatile, as the parameters vector θ can be appropriately scaled to represent a wide spectrum of battery circuitry complexities, ranging from a single cell to an entire battery pack. In this study, Equation (4) refers to a battery module known as a parallel-connected module (PCM), consisting of many parallel circuit branches, each accommodating different cells in series. The scalability property allows for designing and calibrating the ECM model using cell-level test data first, and then scaling up the parameters from cell level to module level to represent the characteristic of the PCM.

2.1. Lab Setup and Cell Data Acquisition Process

A real-time autonomous discharging and monitoring experimental bench, consisting of a host computer, a controllable electronic load, and a lithium-ion cell, were set up.

The host computer controls the entire testing procedure and enables data acquisition and storage. The selected programmable DC electronic load is a Korad KEL103, with a power capacity of 300 W, and a resolution of 0.1 mV and 0.1 mA. The adopted cell is an INR18650-20R lithium-ion cell, manufactured by Samsung, whose characteristics are outlined in Table 1. This cell was selected due to its suitability for automotive applications and its widespread utilization within the literature [36].

Table 1. Samsung INR 18650-20R characteristics.

Cell Parameters	Specification (Values)
Capacity rating (mAh)	2000
Nominal voltage (V)	3.6
Upper/lower cut-off voltage (V)	4.20/2.50
Cell chemistry	INR
Weight (g)	45.0
Diameter (mm)	18.33 ± 0.07
Length (mm)	64.85 ± 0.15

Throughout the testing procedure, the cell was secured within a holder and linked to the Korad KEL 103 device via a 14 AWG wire connection whose resistance was neglected in this study. The test bench was used to perform two different tests, namely, the Multiple Step Test (MST) and the Dynamic Stress Test (DST). Both experimental tests were conducted under ambient temperature conditions. The MST was adopted to capture the OCV cell characteristic. On the other hand, the DST was employed to identify the dynamic response of the system.

The MST entails the following procedure [37]. The fully charged cell is discharged using a current train pulse. Each pulse ranges between 0 A and 2 A. The pulse width is 360 s, which is set to achieve a 10% drop in the SoC. Conversely, the relaxation time (i.e., when the current is null) lasts for 900 s allowing the cell to reach an equilibrium condition.

The DST, developed by the US Advanced Battery Consortium (USABC), emulates a dynamic discharge scenario that can be adjusted to meet the desired maximum demand in accordance with the specified performance requirements of the test samples [38]. For the DST, the cell starts with an SoC value equal to 80%. Since the DST incorporates various current steps with different amplitudes and durations, which extensively excite the cell, it is a natural choice for parameters identification.

The current profiles for both tests are depicted in Figure 3.

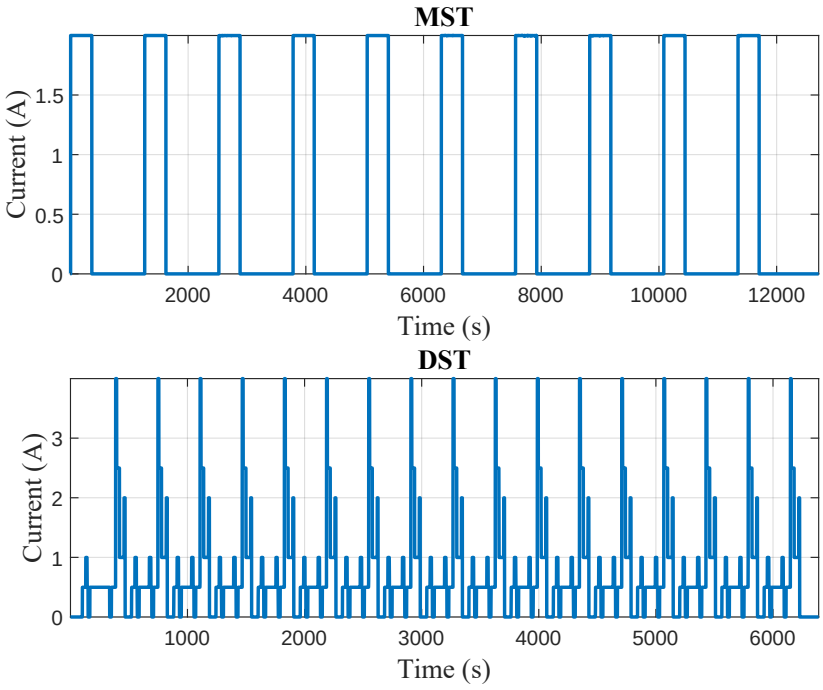

Figure 3. Current profiles for the MST and DST tests.

2.2. Model Design and Calibration

The model design and calibration process is illustrated in Figure 4.

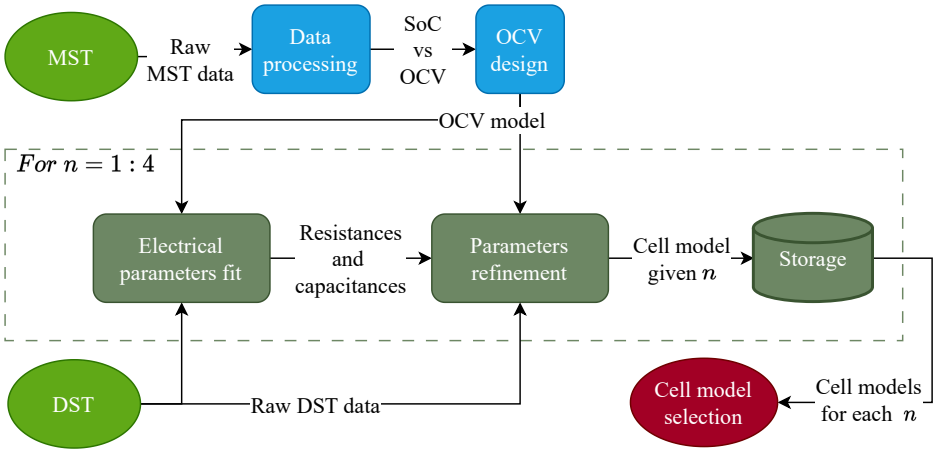

Figure 4. Model design and calibration.

The procedure exploits the MST and DST results gathered through the test bench to obtain a cell model which achieves a satisfactory trade-off between accuracy and simplicity. To obtain a more representative model, three different cells were tested, and the recorded voltages were averaged. Consequently, for both tests, henceforth, we will refer to the test results as the average value derived from the three individual test results, which are shared in the Supplementary Materials of this paper.

Firstly, the MST data were processed to model the OCV characteristic. Each data point right before a new pulse was extracted assuming that the measured voltage coincided with the OCV. This assumption holds since the ohmic effects are negligible given that the current is null and no transient effects are present. Furthermore, as discussed above, the shape of the MST input current was designed to induce a 10% drop in the SoC for each pulse. This allows for obtaining the OCV-SoC curve based on the experimental data. To determine the order p of the OCV polynomial model, a series of statistical F-tests were performed with significance level α equal to 0.05 [39]. Specifically, the approach begins by conducting an F-test between simple polynomials of order two ($p = 2$) and three ($p = 3$). If the null hypothesis is rejected, indicating that the higher-order polynomial significantly improves the model fit, a new F-test is conducted between polynomials of order three and four. This process continues iteratively until the null hypothesis is not rejected, which means that further increases in the polynomial order do not lead to significant improvements in the model. Following this procedure, the value of p was set equal to 4. The OCV design phase is now complete since the OCV model is defined, which means that the order of the OCV polynomial p is set, and a first guess of the related polynomial coefficients is determined.

The DST data were first employed for the identification of the electrical parameters and then to refine the entire model calibration through an optimization approach aimed at minimizing the root mean squared error (RMSE) between the measured voltage and the simulated one. In fact, a first optimization aimed at optimizing only the electrical parameters was run while keeping the OCV fixed. Afterward, a second optimization on the entire θ vector was run, using as the initial condition the outcome of the previous optimization. This procedure was conducted for different values of the RC circuits (i.e., n), from 1 to 4. Eventually, the value of n was chosen empirically and set to 1 since, among the evaluated ECMs, it struck the best compromise between RMSE performance and model complexity.

The full model calibration results are reported in Table 2.

Table 2. Control-oriented cell model parameters in S.I. units.

OCV					
	a_0	a_1	a_2	a_3	a_4
value	3.4211	1.1649	−3.0180	4.5692	−1.9155
Electrical Parameters					
	R_0	R_1	C_1		
value	0.0889	0.0337	3013.5		

2.3. Plant Model

A higher fidelity plant model was developed to simulate the battery system to be monitored and controlled. This plant model was used solely for testing purposes, enabling validation of the proposed BMS functionalities in a closed-loop simulation environment, considering some modeling mismatch between the system under test and the control-oriented model. In general, a plant model is more detailed and accurate compared to a control-oriented one as it can accommodate higher complexity without being limited by constraints related to embedded applications.

For the sake of simplicity, an ECM representation was employed for the plant model as well. The plant is characterized by two RC parallels, and both the OCV and R_0 are defined as polynomial functions of the actual SoC, while the remaining parameters (i.e., R_1, C_1, R_2, C_2) are constant. The order of the polynomial correlation between the OCV and the SoC, defined in Equation (3), was set equal to 6; while the order of the polynomial correlation between the R_0 and the SoC, defined in Equation (5), was set to 3.

$$R_0(SoC) = b_0 + \sum_{i=1}^{3} b_i \cdot SoC^i \tag{5}$$

The plant model parameters obtained through an optimization procedure, which utilized the same MST and DST data adopted for the control-oriented model calibration, are outlined in Table 3.

Table 3. Plant cell model parameters in S.I. units.

OCV							
	a_0	a_1	a_2	a_3	a_4	a_5	a_6
value	3.4228	0.4064	6.4432	−36.3188	77.2681	−70.5189	23.5222
Electrical Parameters							
	b_0	b_1		b_2		b_3	
value	0.1170	−0.2019		0.3601		−0.1874	
	R_1	C_1		R_2		C_2	
value	0.0253	4264.0		0.0095		1127.8	

The fitting results for both the plant and the control-oriented model are shown in Figure 5.

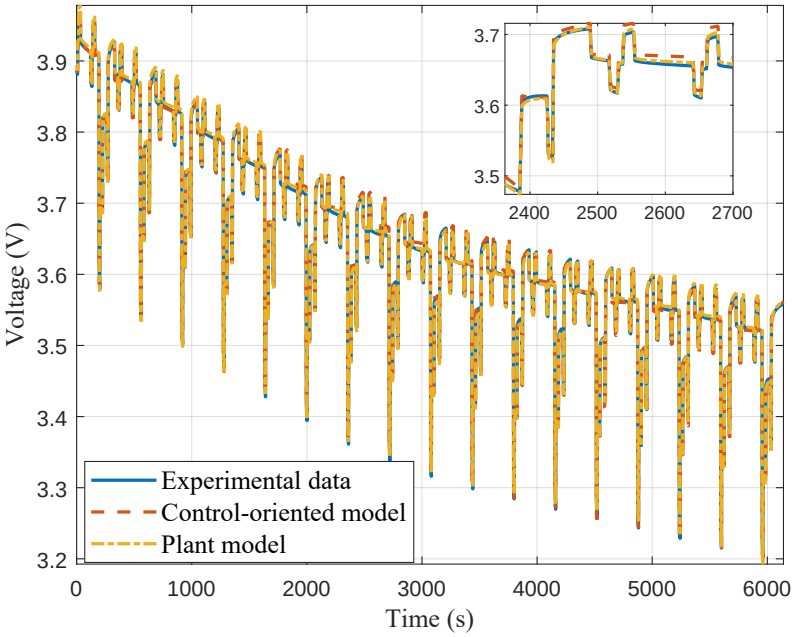

Figure 5. DST voltage data and models response.

2.4. From Cell Model to PCM Model Parameters Scaling

The cell-level parameters detailed in Tables 2 and 3 were scaled to represent a realistic automotive PCM configuration, composed of 74 parallel branches, each hosting 6 cells arranged in series. In this paper, the following approach was adopted, which neglects any loss due to the connections among the cells [40]. This operation solely scales up the physical quantities involved in the problem (e.g., currents and voltages) to an automotive-level application, and it does not affect the essential properties and results of the proposed algorithms.

The OCV correlation was multiplied by the number of parallels. This implies that the parameters $[a_0, \ldots, a_4]$ for the control-oriented model and $[a_0, \ldots, a_6]$ for the plant were all multiplied by 74. The resistances were multiplied by the number of series and divided by the number of parallels. Therefore, the parameters $[R_0, R_1]$ for the control-oriented model, and $[b_0, \ldots, b_3, R_1, R_2]$ for the plant, were multiplied by $6/74$. Conversely, the capacitances were divided by the number of series and multiplied by the number of parallels. Thus, the parameter C_1 for the control-oriented model, and $[C_1, C_2]$ for the plant, were multiplied by $74/6$. Finally, the PCM total capacity was obtained by multiplying the cell capacity Q by the number of parallels (i.e., 74). With a slight refinement of notation throughout the subsequent sections of the paper, the same symbols adopted to denote the cell-level parameters were employed to represent the PCM-level parameters.

3. Simulation Environment

The control-oriented model, discussed in Section 2, was leveraged to develop the three interconnected BMS functionalities presented in this paper (i.e., SoC and SoH estimators, and SoC active balancing). The BMS algorithms were tested in a simulation environment conforming to a typical BEV battery model, named the battery pack plant, consisting of a series connection of 16 PCM modules of 148 Ah, each monitored and controlled.

Hence, considering that the battery model consists of several modules, it is beneficial to introduce the following notation: When a generic quantity ϕ relates to the i-th module of the pack, the superscript i is applied (i.e., ϕ^i). Conversely, the subscript m applied to the generic quantity ϕ represents a vector that encompasses the specified quantity ϕ for all the modules. Therefore, ϕ_m is defined as $[\phi^1; \phi^2; \ldots; \phi^{16}]$.

The simulation environment is illustrated in Figure 6 and its functional components are detailed in the remainder of this section.

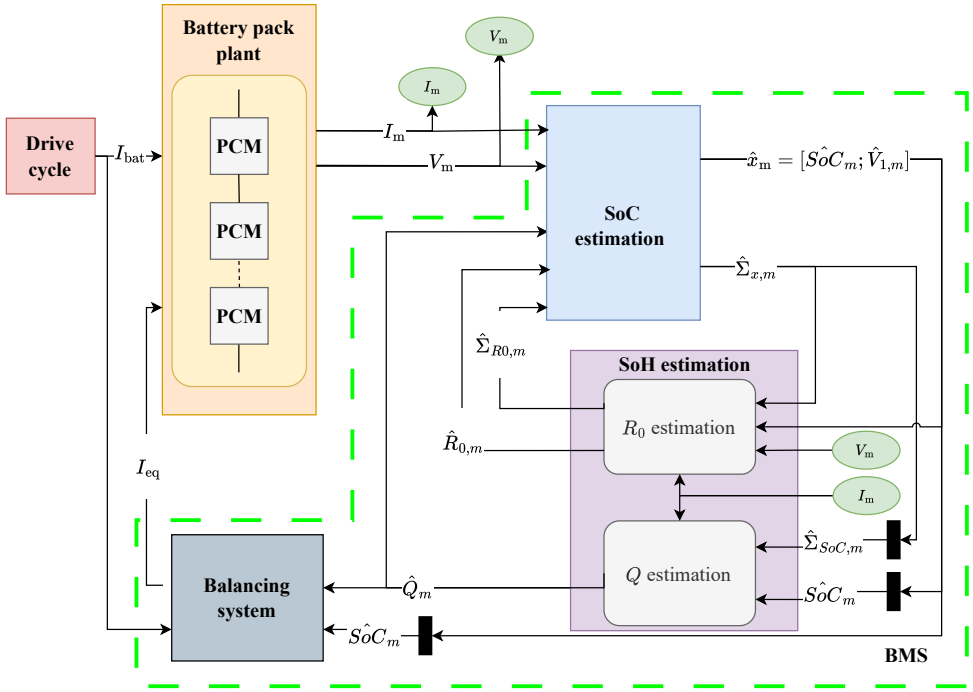

Figure 6. BMS simulation environment block diagram.

From a high-level perspective, the BMS algorithm receives as inputs the voltage and current measurements of all PCMs in the battery pack (i.e., respectively, the vectors V_m and I_m) and produces, for all PMCs in the battery pack, the following outputs:

- SoC estimations (i.e., $S\hat{o}C_m$);
- SoH estimations, in the form of internal resistances and capacity estimations (i.e., $\hat{R}_{0,m}$ and $\hat{Q}_m$, respectively);
- equalizing currents, $I_{eq,m}$, that are injected into each PCM by the balancing system to perform SoC balancing.

To accomplish this, the algorithms mentioned above are executed, at each discrete time instant k, in the following sequence:

1. the SoC estimation algorithm takes as inputs the present values of the voltage and current measurements (i.e., $V_m(k)$ and $I_m(k)$), the previous values of the internal resistance and capacity estimations (i.e., $\hat{R}_{0,m}(k-1)$ and $\hat{Q}_m(k-1)$) and of the internal resistance estimation covariance (i.e., $\hat{\Sigma}_{R0,m}(k-1)$), and produces as outputs the PCMs' internal state estimation (i.e., $\hat{x}_m(k) = [S\hat{o}C_m(k); \hat{V}_{1,m}(k)]$) and its estimation error covariance (i.e., $\hat{\Sigma}_{x,m}(k)$). The SoC estimation adopts an extended Kalman filtering (EKF) technique, as detailed in Section 3.1.
2. the SoH estimation (Section 3.3) algorithm consists of two algorithms running in parallel:
   - the internal resistance estimation algorithm takes as inputs the present values of the voltage and current measurements (i.e., $V_m(k)$ and $I_m(k)$), the present value of the PCM's internal state estimation (i.e., $\hat{x}_m(k) = [S\hat{o}C_m(k); \hat{V}_{1,m}(k)]$) and its estimation error covariance (i.e., $\hat{\Sigma}_{x,m}(k)$), and produces as outputs the internal resistance estimation (i.e., $\hat{R}_{0,m}(k)$) and its estimation error covariance (i.e., $\hat{\Sigma}_{R0,m}(k)$). Akin to SoC estimation, the internal resistance estimation algorithm also utilizes the extended Kalman filtering (EKF) technique, as detailed in Section 3.3.1.
   - the capacity estimation algorithm takes as inputs the present values of the voltage and current measurements (i.e., $V_m(k)$ and $I_m(k)$), the present value of the SoC estimation (i.e., $S\hat{o}C_m(k)$) and its estimation error covariance (i.e., $\hat{\Sigma}_{SoC,m}(k)$), and produces as outputs the capacity estimation (i.e., $\hat{Q}_m(k)$) and its estimation

error covariance (i.e., $\hat{\Sigma}_{Q,m}(k)$). The capacity estimation is performed via the proportional total least squares (PLTS) technique, as detailed in Section 3.3.2.

3. the SoC active balancing algorithm takes as inputs the total battery pack current measurement (i.e., $I_{bat}(k)$), the SoC and capacity estimation (i.e., $S\hat{o}C_m(k)$ and $\hat{Q}_m(k)$, respectively), and produces as output the vector of the equalizing currents (i.e., $I_{eq,m}(k)$). It utilizes the model predictive control (MPC) technique, as detailed in Section 3.2.

3.1. SoC Estimation

The SoC estimation function consists of a bank of 16 parallel EKFs, one per PCM. Each *i*-th EKF receives as input the voltage and current measurements (i.e., V^i and I^i), the internal resistance mean and covariance estimations (i.e., $\hat{R}_0^i$ and $\hat{\Sigma}_{R0}^i$), and the capacity estimation $\hat{Q}^i$ of the respective PCM. Based on this information, the EKF reconstructs the mean and covariance of the *i*-th module hidden state, namely, $\hat{x}^i$ and $\hat{\Sigma}_x^i$.

The EKF prediction and measurement functions are reported in Equation (6), which is a variant of the PCM state-space model reported in Equation (4). All the quantities in Equation (6) refer to the generic i-th PCM; however, the superscript *i* has been omitted for the sake of a simpler notation.

$$\begin{cases} x(k+1) = Ax(x) + \begin{bmatrix} -u(1)t_{EKF}/[3600u(3)] \\ u(1)t_{EKF}/C_1 \end{bmatrix} + w \\ x = [SoC; V_1] \\ A = \begin{bmatrix} 1 & 0 \\ 0 & exp(-t_{EKF}/(R_1C_1)) \end{bmatrix} \\ u \sim \mathcal{N}\left(\left[I; \hat{R}_0; \hat{Q}\right], \Sigma_u\right) \\ y = V = OCV(x(1)) - V_1 - u(1)u(2) + v \end{cases} \tag{6}$$

t_{EKF} is the SoC function sample time. u is the 3×1 input vector, which is assumed to be a normally distributed random variable. The u mean contains the actual current I, and the estimations provided by the SoH function of the actual internal resistance, R_0, and capacity, Q, which are periodically updated. On the other hand, the diagonal of the covariance matrix Σ_u contains the current measurement variance Σ_I, to account for an imperfect acquisition system, the estimated covariance $\hat{\Sigma}_{R0}$, provided by the SoH function, and Σ_Q, which is set to a constant. w and v are white Gaussian noises with zero mean and diagonal covariance matrices equal to Σ_w and Σ_v, respectively. w was adopted to consider the modeling uncertainty; while the v white noise accounts for the voltage measurement error, and the Σ_v value coincides with the voltage measurement variance.

Given the prediction model reported in Equation (6), a well-established EKF algorithm [41] was implemented for each observer of the bank.

The EKF prediction step is reported in Equation (7), where the superscript "$-$" indicates the predicted estimates at time k by utilizing the information of the measurements up to time $k-1$.

$$\begin{aligned} \hat{x}^-(k) &= A\hat{x}(k-1) + Bu \\ \hat{\Sigma}_x^-(k) &= A\hat{\Sigma}_x(k-1)A^T + B\Sigma_u(k)B^T + \Sigma_w \end{aligned} \tag{7}$$

where $B = \begin{bmatrix} -t_{EKF}/[3600u(3)] & 0 & u(1)t_{EKF}/[3600u(3)^2] \\ t_{EKF}/C_1 & 0 & 0 \end{bmatrix}$ is the Jacobian of the generating function with respect to the input vector.

Equation (8) is the update step, where both the mean and the covariance of the hidden state estimate are improved by leveraging the upcoming measurement $y(k)$.

$$\begin{aligned} L(k) &= \hat{\Sigma}_x^-(k)C^T(k)\Big[C(k)\hat{\Sigma}_x^-(k)C^T(k) + D(k)\Sigma_u(k)D^T(k) + \Sigma_v\Big]^{-1} \\ \hat{x}(k) &= \hat{x}^-(k) + L(k)\big[y(k) - \hat{y}^-(k)\big] \\ \hat{\Sigma}_x(k) &= \hat{\Sigma}_x^-(k) - \hat{\Sigma}_x^-(k)C^T(k)L^T(k) \end{aligned} \tag{8}$$

where $C(k) = [\frac{dOCV}{dSoC}, -1]$ and $D(k) = [-u(2), -u(1), 0]$ are the Jacobians of the measurement function with respect to the state and the input vectors.

This procedure allows for recursively obtaining an estimate of the state of each PCM.

3.2. SoC Active Balancing

The SoC active balancing system equalizes the charge levels of the individual modules within the battery pack. In fact, the SoC of one or more modules might diverge too widely as a result of inconsistencies due to different operating conditions, manufacturing discrepancies, or faults among the PCMs. The PCMs are selectively discharged or charged by controlling the individual cell equalizers (ICEs) of the balancing circuits, which can be arranged in a variety of topologies. The ICEs contain transistors or other commutable elements, which allow the actuation of the reference equalizing current $I_{eq,m}$. Therefore, the balancing control problem typically consists of a higher-level controller, which computes the $I_{eq,m}$ vector, and a lower-level controller for the ICEs actuation. However, in this paper, the ICEs are assumed to be ideal and to instantly follow the reference $I_{eq,m}$.

Similarly to other works (e.g., [42]), the control task of selecting at each time step the optimal $I_{eq,m}$ vector is accomplished via a linear model predictive control (MPC) technique [43].

The MPC receives as input the estimates $\hat{SoC}_m$ and $\hat{Q}_m$ from the SoC and the SoH functions to account for both charge and health inhomogeneities among the modules to obtain the optimal control strategy. At each time step, the MPC resolves the following quadratic optimization problem:

$$\begin{aligned} \underset{U}{\text{minimize}}\ J(U|SoC_m(0)) = &\sum_{k=0}^{N_p}\Big(SoC_m(k) - SoC^{Sp}(k)\Big)^T W\Big(SoC_m(k) - SoC^{Sp}(k)\Big) + \\ &\sum_{k=0}^{N_c} u_m(k)^T R u_m(k) + \sum_{k=N_c+1}^{N_p} u_m(k)^T R u_m(k) \\ \text{subject to: } &SoC_m(k+1) = SoC_m(k) + t_{MPC}\hat{\mathbf{Q}}_m^{-1}\big(\boldsymbol{I}_{bat} - T_\eta \boldsymbol{I}_{max} u_m(k)\big); \\ &0 \le SoC_m(k) \le 1 \quad k = 0, \ldots, N_p. \\ &-1 \le u_m(k) \le 1 \quad k = 0, \ldots, N_c. \end{aligned} \tag{9}$$

where:

- W and R are symmetric positive definite weight matrices;
- SoC^{Sp} is the setpoint vector over the prediction horizon, which is equal to the expected value of the predicted state of charges when $k = 0$;
- N_p and N_c are the prediction and the control horizons, respectively;
- t_{MPC} is the sampling time of the MPC controller;
- $\hat{\boldsymbol{Q}}_m$ is a diagonal matrix collecting the estimations of the total capacities of the PCMs;
- $\boldsymbol{I}_{bat}$ is a vector of 16 identical components equal to the instantaneous battery current demand;
- T_η is the topology efficiency matrix, which accounts for the circuitry topology and the coulombic efficiency of the PCMs. For this application, a bidirectional adjacent topology was considered and, consistent with the models presented in Section 2, a unitary coulombic efficiency was adopted;

- $\boldsymbol{I}_{max}$ is a diagonal matrix containing the maximum admissible values of the currents in each ICE;
- $u_m(k)$ is the MPC control vector at time k, which comprises the normalized equalization currents with respect to their maximum admissible values through the ICEs. Each element of u_m can assume both positive and negative values since the currents can flow through the ICEs both ways.

At each control step, the control trajectory U_m^* is computed by solving the optimization problem of Equation (9) in MATLAB, utilizing its built-in quadratic programming solver. However, following the MPC paradigm, only the first element of the sequence is applied (i.e., u_m^*) and the rest is discarded; then, at the next time step, measurements are updated and the same procedure is followed again using the old control trajectory as the initial condition. It is important to highlight that the normalized equalization current u_m^* is rescaled by multiplying it with the maximum allowable current to produce the actual $I_{eq,m}$ that is eventually applied to the battery pack plant.

3.3. SoH Estimation

The SoH estimation function is responsible for outputting the total capacity and resistance estimates $\hat{Q}_m$ and $\hat{R}_{0,m}$ for all PCMs. These two estimation objectives were treated separately, and different algorithms were utilized to address them.

3.3.1. Internal Resistance Estimation

The internal resistance of each individual PCM was computed by a bank of independent linear filters or observers. Each i-th filter needs to provide an estimate of the expected value of the related PCM internal resistance $\hat{R}_0^i$ and of its variance $\hat{\Sigma}_{R0}^i$. The i-th filter receives the following as input: the measurements of the current and the voltage for the respective module, namely, I^i and V^i; an estimate of the state of the PCMs (i.e., $\hat{SoC}_m$ and $\hat{V}_{1,m}$), and their variances $\hat{\Sigma}_{SoC}^i$ and $\hat{\Sigma}_{V1}^i$ provided by the SoC estimation function. Each filter leverages the prediction and measurement correlations reported in Equation (10) to update the actual internal resistance estimation. For the sake of a simpler notation, the superscript i is omitted from the filter equations below.

$$\begin{cases} R_0(k+1) = R_0(k) + w_{R0}(k) \\ V = OCV(\hat{SoC}) - \hat{V}_1 - IR_0(k) + v \end{cases} \tag{10}$$

w_{R0} is a normally distributed process noise, while v was introduced in Equation (6) to model the voltage measurement error.

The observer prediction step is

$$\begin{aligned} \hat{R}_0^-(k) &= \hat{R}_0(k-1) \\ \hat{\Sigma}_{R0}^-(k) &= \hat{\Sigma}_{R0}(k-1) + \Sigma_{w_{R0}}. \end{aligned} \tag{11}$$

To report the filter update step, it is convenient to introduce the following symbols to indicate the expected value and the variance of the term $OCV(\hat{SoC})$. For the expected value, the symbol $\hat{OCV}$ is adopted; while the symbol $\hat{\Sigma}_{OCV}$ is used for the variance. Both these quantities can be computed analytically, leveraging both the $\hat{SoC}$ normality and the polynomial structure of the OCV correlation. Therefore, the update step is equal to:

$$\begin{aligned} L(k) &= \hat{\Sigma}_{R0}^-(k) I \left[(\hat{\Sigma}_{R_0}^-(k) + \hat{R}_0^-(k)^2) \cdot (\Sigma_I + I^2) - I^2 \hat{R}_0^-(k)^2 + \hat{\Sigma}_{OCV} + \hat{\Sigma}_{V1} + \Sigma_v \right]^{-1} \\ \hat{y}^-(k) &= \hat{OCV} - \hat{V}_1 - I\hat{R}_0^-(k) \\ \hat{x}(k) &= \hat{x}^-(k) + L(k)\left[y(k) - \hat{y}^-(k)\right] \\ \hat{\Sigma}_x(k) &= \hat{\Sigma}_x^-(k) - \hat{\Sigma}_x^-(k) I L^T(k) \end{aligned} \tag{12}$$

where Σ_I is the variance associated with the current measurement error.

3.3.2. Capacity Estimation

The total capacity of each PCM was computed by using a bank of estimators implementing the Proportional Total Least Squares (PTLS) algorithm [44], which is a variant of the Total Least Square (TLS). Every i-th PTLS outputs an estimate of the mean of the respective total capacity, namely, $\hat{Q}^i$.

Least-square techniques consider two physical quantities, ξ and μ, which relate to each other by a linear regression model:

$$\xi = \Theta\mu \tag{13}$$

The objective is to identify the parameter Θ by using the observations of ξ and μ. The ordinary least-square technique assumes perfect knowledge of μ and only ξ to be uncertain and noisy. On the other hand, TLS techniques attempt to account for uncertainty in the observation of both variables. This is carried out by solving the following optimization problem:

$$\underset{\hat{\Theta}}{\text{minimize}}\ \chi(\hat{\Theta}) = \sum_{j=1}^{N} \gamma^{N-j}\Big(\frac{\mu_j - M_j}{\Sigma_{\mu_j}} + \frac{\xi_j - Z_j}{\Sigma_{\xi_j}}\Big). \tag{14}$$

where

- ξ_j and μ_j are elements of an N-sized dataset containing the observations of ξ and μ, respectively;
- Σ_{ξ_j} and Σ_{μ_j} are the variances of the noises affecting the observations ξ_j and μ_j;
- $\hat{\Theta}$ is the optimization problem variable;
- M_j and Z_j are the final optimized mapping of the data (μ_j, ξ_j) to the hyperplane $Z_j = \hat{\Theta} M_j$;
- $\gamma \in]0, 1]$ is the forgetting factor.

The PTLS algorithm peculiarity is to move the further assumption on the optimization problem of Equation (14) by assuming that a constant factor κ exists such that $\Sigma_{\mu_j} = \kappa^2 \Sigma_{\xi_j}$ for all j.

For the capacity estimation problem, the general linear regression expression in Equation (13) was particularized by rearranging the Coulomb counting equation as reported in Equation (15).

$$\underbrace{t_Q \sum_{h=0}^{n_s} I(k+h)}_{=\xi_j} = \underbrace{Q}_{=\Theta}\ \underbrace{[\hat{SoC}(k+n_s) - \hat{SoC}(k)]}_{=\mu_j} \tag{15}$$

where t_Q is the sample time at which measurements are acquired and n_s is a design parameter.

The PTLS is a suitable choice for embedded algorithms since its solution can be found recursively, and it does not require an excessive computational footprint. The solution to the PTLS was derived from [45] and is reported in Appendix A.

In general, the challenge of capacity estimation lies in its sensitivity with respect to biases in the SoC. Unfortunately, in practice, the SoC information is inevitably biased since it is reconstructed by an observer and no direct measurement can be obtained. Therefore, several heuristics can be employed to improve the PTLS robustness [46]. In this paper, the capacity estimation is triggered only when the difference between $\hat{SoC}(k+n_s) - \hat{SoC}(k)$ and $\hat{\Sigma}_{SoC}$ are, respectively, greater and lower than the two calibratable thresholds. Furthermore, the output of the PTLS was appropriately filtered to account for the slow dynamic of the degradation process, and saturated within a reasonable range of values, namely, Q_{min} and Q_{max}. Specifically, leveraging the intrinsic monotonicity of the Q evolution given the degradation, when a reliable estimate is obtained, the Q_{max} value can be replaced by the actual capacity estimation. This strategy ensures that the Q_{max} limit remains consistent

with the actual performance of the battery, and prevents capacity overestimation, which might lead to BMS suboptimal performance.

4. Results and Discussion

The simulation environment described in Section 3 was tested under two different driving scenarios. The first scenario aims at assessing the BMS performance under nominal and healthy battery conditions. On the other hand, the second scenario focuses on the BMS algorithms response during an inhomogeneous degradation of the battery modules from their nominal values (at the beginning of life) up to their failure (end of life).

During both scenarios, the battery current demand I_{bat} follows a sequence of consecutive WLTP (Worldwide Harmonised Light Vehicle Test Procedure) regulatory cycles, where a single WLTP is reported in Figure 7. The calibratable parameters of the BMS algorithms were tuned empirically, and their values are reported in Table 4. The current and voltage measurements for each module were corrupted with some additive white noise whose power density was chosen to obtain a 3% ripple for each signal. Finally, the computational burden of the proposed BMS algorithm shows promising results for embedded applications, exhibiting an execution speed 2.7 times faster than real-time when running on an i5-10210U laptop. This is a positive result, especially considering that computational efficiency is not the primary focus of this paper and no specific optimizations were performed from either a software or hardware perspective.

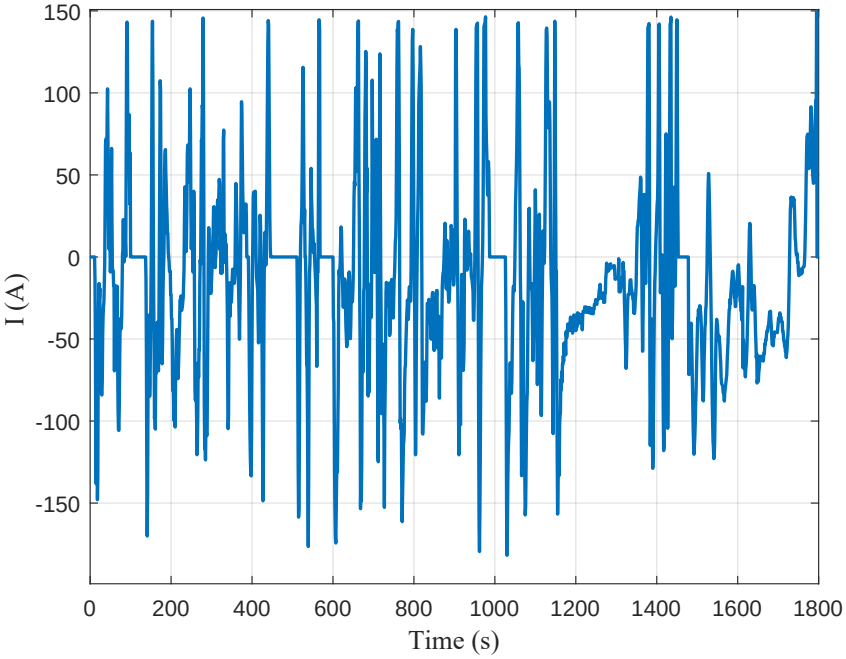

Figure 7. Single WLTP current profile.

Table 4. BMS calibration in S.I. units.

SoC EKF	
Σ_w	2
Σ_Q	10^{-6}
Σ_v	0.1
t_{EKF}	0.1
R_0 Estimator	
$\Sigma_{w_{R0}}$	10^{-12}
Σ_v	0.1

Table 4. *Cont.*

PTLS	
Σ_I	0.1
γ	0.99
t_Q	0.1
MPC	
N_p	30
N_c	10
W	$diag([200, 200, \ldots, 200])$
R	$diag([0.02, 0.02, \ldots, 0.02])$
$I_{eq_{max}}$	148
t_{MPC}	5

4.1. Scenario I

The battery demand profile, I_{bat}, consists of ten identical concatenating WLTP cycles. The actual initial SoC values of each PCM were set between 80% and 90%. Finally, no degradation was considered; therefore, the internal resistances and capacities of the battery plant modules are equal to their nominal values throughout the whole simulation.

The effectiveness of the SoC estimation functionality is reported in the top-left subplot (a) of Figure 8. The SoC estimates, provided by the EKFs, were initialized to 85% for all the modules, while the actual SoCs varied from 80% to 90%, as stated above. The (a) subplot shows the observers rapidly compensate for the discrepancy in the initial conditions and, after 10 s, the errors settle to a value lower than 2%. The estimation error does not converge to 0 due to the presence of noise in the measurements, the system non-linearities, and the modeling mismatch between the EKFs' internal control-oriented model and the plant.

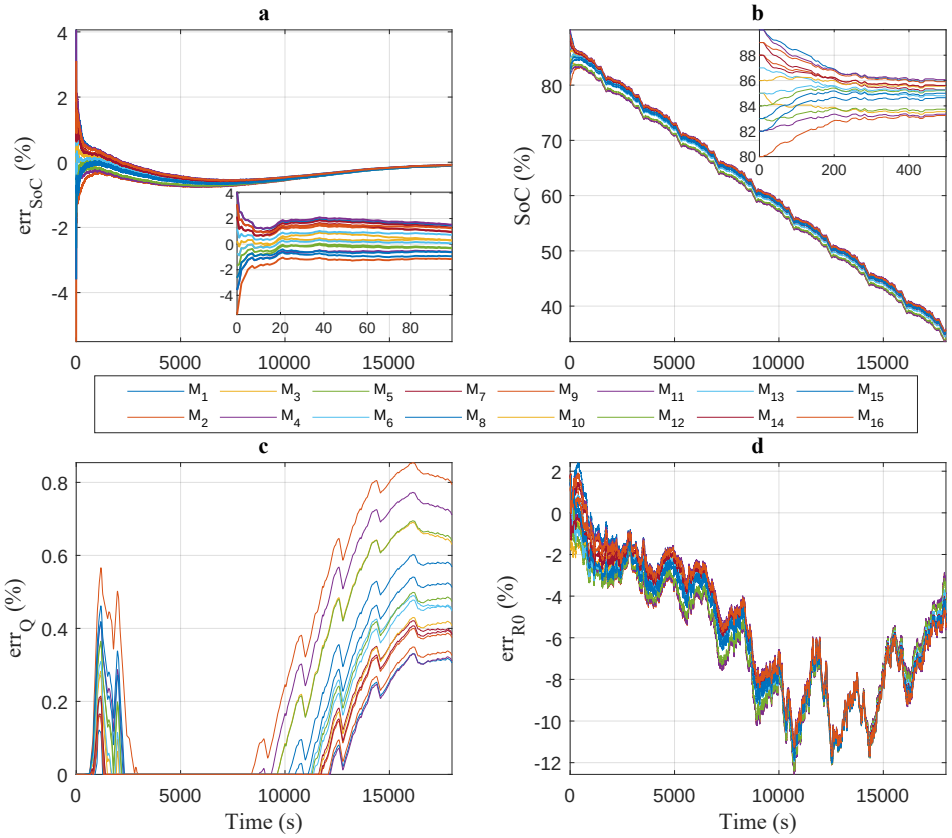

Figure 8. (**a**) State of charges relative errors. (**b**) Actual state of charges. (**c**) Capacities relative errors. (**d**) Internal resistances relative errors.

The top-right subplot (b) of Figure 8 reports the actual SoC values of each PCM. This plot highlights that the active balancing successfully distributes the level of charges among the modules which have been initialized with different SoC values. As a result, the active balancing manages to achieve a low dispersion of the SoCs during the entire scenario. The MPC control action (i.e., the equalization currents) is reported in Figure 9. Figure 9 reports the equalization currents profiles for the first 400 s of simulation, for the first three modules, which were initialized with a high, low, and average level of SoC (i.e., 90%, 80%, and 85%). As expected, the MPC applies a sustained positive and negative control action for the first and second modules since they need to give away and accept charges, respectively. The bottom-right subplot of Figure 9 reports the SoC trajectories of the first three modules and the MPC activation control band. In fact, the MPC is activated only when the difference between the highest and lowest SoC is higher than 2%. It is interesting to observe that the MPC successfully drives the SoCs population on the edge of the acceptance control band and it is activated as soon as the band is violated.

Finally, subplots (c) and (d) of Figure 8 report the response of the SoH function of the proposed BMS system. The (c) subplot shows the capacities relative errors, which do not exceed 1% for all the modules. To achieve this, the estimated capacities are processed using a first-order linear filter with a time constant of 500 s, and their upper limits are saturated to the nominal value of 148 Ah. This filtering approach facilitates the convergence of the SoC estimators while effectively limiting the capacity error that could potentially lead to algorithm divergence. This constraint is particularly crucial to improve the overall system stability and accuracy due to the closed-loop relationship between the SoC and SoH observers.

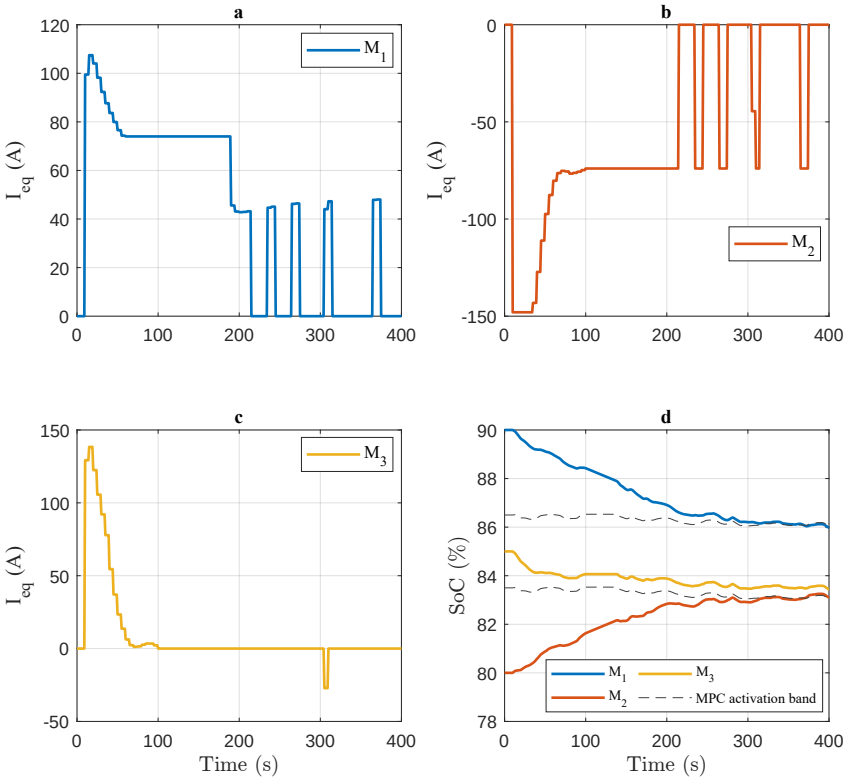

Figure 9. Subplots (**a**–**c**) report the MPC control action for PCM one, two, and three. Subplot (**d**) displays their respective SoC values and the control activation band.

The last subplot (d), displays the evolution of the PCMs internal resistances relative errors, which do not exceed 13%. It is evident that their evolution is a function of the actual SoC of the system. This aligns with expectations, considering that the R_0 observers utilize a control-oriented model which, in contrast to the plant model, does not account for the correlation between the SoC and R_0.

4.2. Scenario II

The second scenario comprises five consecutive driving cycles, each consisting of ten concatenated WLTPs. At the start of each cycle, the SoCs of the PCMs are reset to 85%, and instantaneous degradation is applied to each battery module. The degradation is modeled as a stochastic process, causing variations in both the capacity and internal resistance among the modules. The degradation increments for each cycle follow a uniform distribution with a fixed minimum and maximum, as detailed in Appendix B. The bounds of the uniform distributions were designed so that, on average, both the capacity and internal resistances reached their end-of-life values (i.e., a reduction of 20% in capacity and an increase of 100% in internal resistance) during the fifth (last) cycle. To robustly support the SoH estimation results, the same testing conditions were replicated for additional regulatory cycles (i.e., FTP-72 (UDDS)—Emission Test Cycles and SFTP-US06—Emission Test Cycles). These results are detailed in the Supplementary Materials, while only the results related to the WLTP are presented and discussed in this section.

Figure 10 presents the results of the capacities and internal resistances tracking obtained from the proposed SoH functionality for the whole simulation. The figures report the results for two selected modules for the sake of visualization. On the other hand, Tables 5 and 6 report the SoH estimation error for all the modules.

Figure 10 shows the satisfactory performance of the SoH functionality as it tracks the decreasing capacity and increasing internal resistance trends during degradation. It is important to note that the degradation is imposed instantaneously at the beginning of each cycle, presenting a significant challenge to the SoH algorithm, while in real-world conditions, battery degradation typically occurs as a slow drifting process. The capacity estimation exhibits a slow convergence to its steady-state value, which settles within a 4% error band. Furthermore, once the capacity estimation reaches a quasi-steady value, which means it lays within a calibratable band for a sufficient amount of time, the upper limit of capacity saturation, Q_{max}, is updated with the actual estimation.

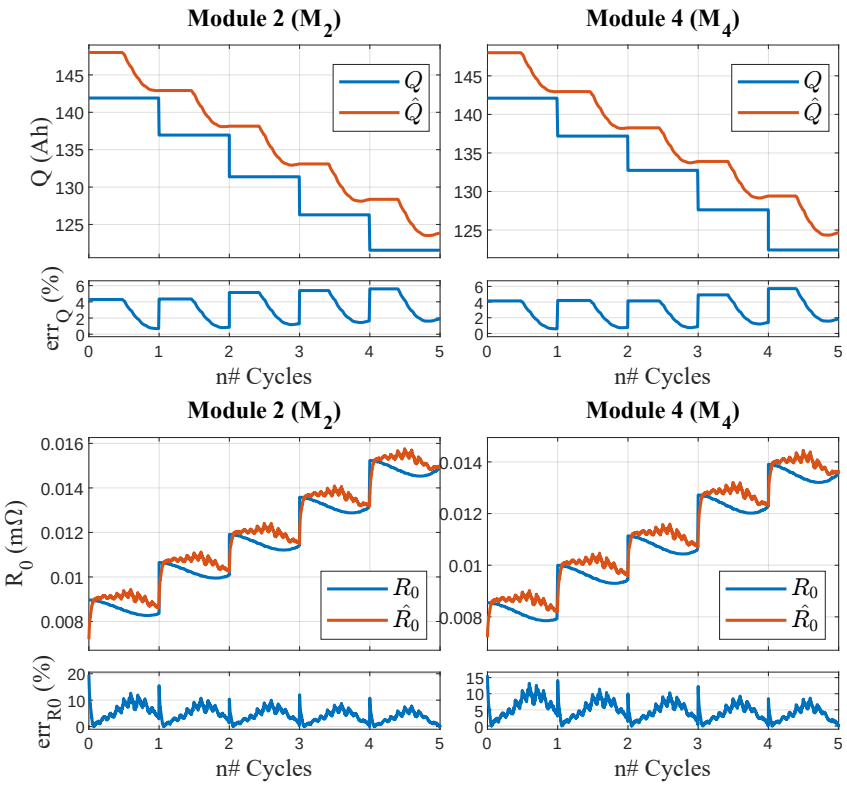

Figure 10. Capacity and internal resistance estimation performance during Scenario II for PCM two and four.

Table 5. Relative capacity estimation error computed when the PTLS reaches its steady-state value.

Q Error [%]								
	M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8
cycle 1	−0.68	−0.70	−0.46	−0.59	−0.28	−0.65	−0.19	−0.37
cycle 2	−1.14	−0.86	−0.68	−0.77	−1.10	−0.88	−0.88	−0.56
cycle 3	−1.40	−1.29	−1.11	−0.86	−1.54	−1.24	−1.39	−0.87
cycle 4	−1.68	−1.61	−1.54	−1.38	−1.79	−1.65	−1.54	−1.73
cycle 5	−1.95	−1.84	−1.82	−1.78	−1.89	−1.90	−1.95	−2.27
	M_9	M_{10}	M_{11}	M_{12}	M_{13}	M_{14}	M_{15}	M_{16}
cycle 1	−0.21	−0.21	−0.75	−0.66	−0.36	−0.79	−0.19	−0.50
cycle 2	−0.56	−1.05	−1.05	−0.68	−0.84	−1.31	−0.98	−0.75
cycle 3	−1.18	−1.29	−1.51	−0.99	−1.54	−1.35	−1.04	−1.38
cycle 4	−1.84	−2.09	−2.10	−1.34	−1.95	−1.82	−1.82	−2.07
cycle 5	−2.43	−2.18	−2.27	−1.68	−2.48	−2.06	−1.81	−2.63

Table 6. Estimated internal resistance root mean square error (RMSE).

R_0 Error [mΩ]								
	M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8
cycle 1	0.5679	0.5755	0.5640	0.5609	0.5493	0.5673	0.5478	0.5579
cycle 2	0.5730	0.5670	0.5536	0.5575	0.5731	0.5593	0.5725	0.5576
cycle 3	0.5781	0.5663	0.5637	0.5437	0.5791	0.5712	0.5797	0.5522
cycle 4	0.5681	0.5732	0.5715	0.5631	0.5729	0.5670	0.5643	0.5769
cycle 5	0.5758	0.5698	0.5637	0.5629	0.5617	0.5689	0.5788	0.5894
	M_9	M_{10}	M_{11}	M_{12}	M_{13}	M_{14}	M_{15}	M_{16}
cycle 1	0.5508	0.5425	0.5763	0.5713	0.5485	0.5702	0.5446	0.5698
cycle 2	0.5484	0.5792	0.5786	0.5594	0.5681	0.5860	0.5672	0.5548
cycle 3	0.5660	0.5671	0.5761	0.5601	0.5822	0.5728	0.5583	0.5711
cycle 4	0.5832	0.6005	0.5870	0.5560	0.5880	0.5722	0.5874	0.5953
cycle 5	0.5940	0.5749	0.5762	0.5673	0.5915	0.5739	0.5666	0.5989

On the other hand, the $\hat{R}_0$ dynamic exhibits a much quicker response as it captures not only the (generally) slow degradation process but also its correlation with the actual SoC. Finally, the SoC observers demonstrate satisfactory behavior, and their estimation errors are reported in Table 7.

Table 7. Estimated state of charge root mean square error (RMSE).

SoC Error [%]								
	M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8
cycle 1	0.96	0.96	0.88	0.94	0.84	0.95	0.81	0.86
cycle 2	1.08	0.97	0.93	0.95	1.08	1.00	0.99	0.88
cycle 3	1.11	1.09	1.03	0.95	1.18	1.06	1.12	0.96
cycle 4	1.17	1.14	1.13	1.07	1.22	1.18	1.14	1.23
cycle 5	1.22	1.19	1.21	1.20	1.22	1.22	1.23	1.38
	M_9	M_{10}	M_{11}	M_{12}	M_{13}	M_{14}	M_{15}	M_{16}
cycle 1	0.81	0.82	0.98	0.95	0.87	1.00	0.81	0.89
cycle 2	0.90	1.04	1.03	0.91	0.97	1.12	1.04	0.96
cycle 3	1.07	1.10	1.17	0.98	1.18	1.08	1.01	1.14
cycle 4	1.25	1.32	1.33	1.07	1.27	1.23	1.23	1.32
cycle 5	1.45	1.34	1.37	1.14	1.46	1.27	1.19	1.52

Figure 11 visualizes a correlation analysis between the actual internal resistance and capacity degradation with respect to the $\hat{SoC}$, $\hat{R}_0$, and $\hat{Q}$ estimations. The figure illustrates how estimation performance decreases as battery degradation increases. The RMSE of the $\hat{SoC}$ estimation increases from 0.89% to 1.3%, while the average steady-state $\hat{Q}$ error

increases in magnitude from 0.47% to 2.0%. On the other hand, the RMSE of $\hat{R}_0$ estimation is not significantly affected by battery degradation as the BMS manages to confine the estimation error within a 0.48 mΩ band under all conditions.

The data reported in the subplots of Figure 11 can be grouped into five clusters of points, each corresponding to a different charge–discharge cycle. For each cluster, both the $\hat{SoC}$ and $\hat{Q}$ errors strongly correlate with the actual capacity degradation; in particular, a higher Q value corresponds to a poorer estimation. Conversely, the actual internal resistance degradation exhibits a much more dispersed and uncorrelated distribution with respect to the estimation errors in the single clusters. Finally, a discussion on alternative SoH estimation frameworks is reported in Appendix C.

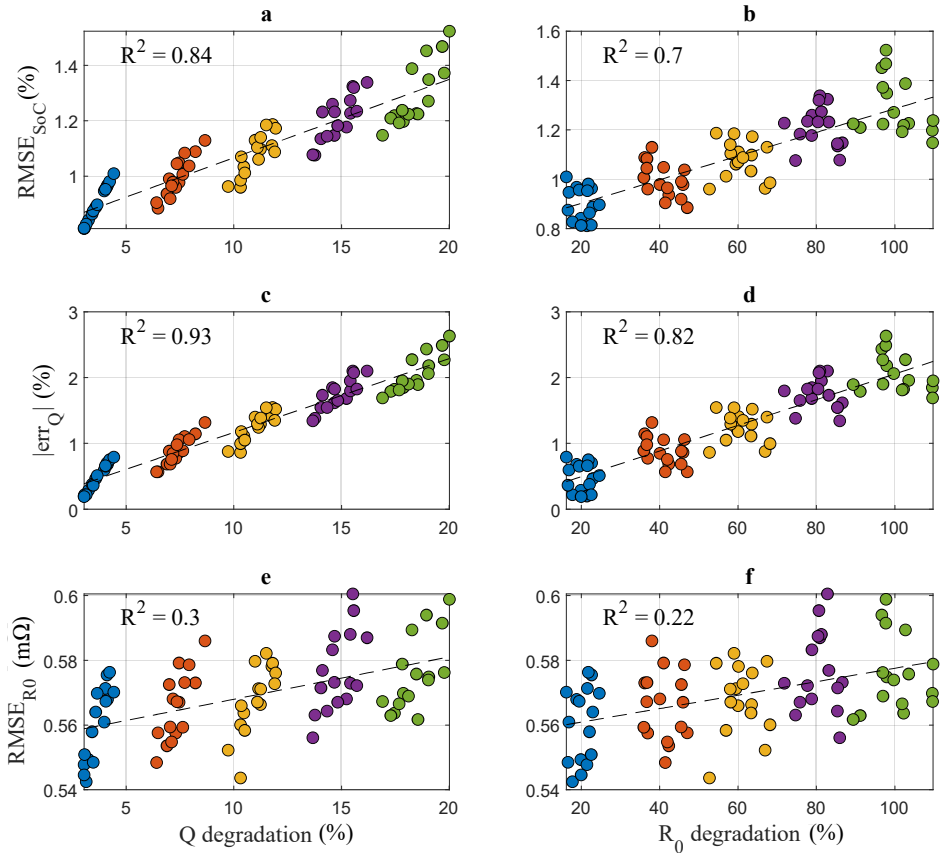

Figure 11. Subplots (**a**,**b**) report the $\hat{SoC}$ RMSE. Subplots (**c**,**d**) report the absolute value of the $\hat{Q}$ error in steady-state. Subplots (**e**,**f**) report the $\hat{R}_0$ RMSE. The different colors cluster the data which belong to the same cycle and R^2 is the coefficient of determination.

5. Conclusions

This paper presents an integrated framework designed to address the development of SoC and SoH estimators, along with an active balancing functionality tailored for BMS in BEV applications. The proposed framework includes three model-based algorithms that are dynamically coupled. Specifically, the active balancing algorithm utilizes real-time information of both the actual SoC and SoH to optimize the distribution of charges through a linear MPC approach. The SoC and SoH are interconnected in a closed-loop configuration, allowing for effective coordination between the two. The state of charge estimation is achieved through an EKF methodology, providing accurate and reliable SoC reconstruction. On the other hand, the state of health functionality responsible for reconstructing the internal resistance adopts a Kalman-based linear filter, while the capacity estimation leverages a PTLS (proportional total least squares) approach.

The algorithms were tuned using laboratory data collected from three Samsung INR18650-20R lithium-ion cells. The BMS functionalities were validated through sim-

ulation by integrating the proposed algorithms with a battery pack model consisting of sixteen modules to be controlled. The strong coupling between the SoC and the capacity observations poses a significant estimation challenge which is exacerbated during transient conditions. This problem was effectively addressed by incorporating in the original PTLS algorithm some fundamental prior information about the degradation process via saturation, and by proper filtering.

The results demonstrate satisfactory and consistent behavior across the various testing scenarios. Notably, the BMS functionalities were validated under both nominal conditions and in the event of non-uniform modules degradation, from the battery's beginning of life to its end of life. The EKF strategy, employed to estimate the modules' SoC, effectively maintains an estimation error below 2% under all conditions. Similarly, the SoH functionality ensures the average estimation errors for R_0 and Q remain below 13% and 3%, respectively. Finally, the MPC responsible for the active balancing task successfully keeps the dispersion of the SoCs population within a desirable range, and it coherently reacts to charge and health imbalances in the battery pack.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/modelling5030048/s1.

Author Contributions: Conceptualization, A.M.; methodology, L.B., G.G., D.P. and F.U.; software, L.B., A.F., G.G. and F.U.; validation, L.B., D.P. and F.U.; formal analysis, A.M.; investigation, A.F., G.I. and F.U.; resources, A.M.; data curation, A.F., G.I. and D.P.; writing—original draft preparation, A.M., D.P. and F.U.; writing—review and editing, L.B., A.M. and D.P.; visualization, L.B. and F.U.; supervision, A.M.; project administration, A.M.; funding acquisition, M.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data presented in this study are available in the Supplementary Materials.

Acknowledgments: The authors would like to acknowledge Gennaro D'Alessio for his technical support in conducting the data acquisition and laboratory setup for this work.

Conflicts of Interest: Authors Lorenzo Breglio, Arcangelo Fiordellisi, Giovanni Gasperini, Giulio Iodice, Denise Palermo, Manuela Tufo, Fabio Ursumando, and Agostino Mele were employed by the company Kineton S.r.l. The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Appendix A. PTLS Recursive Solution

The PTLS algorithm was utilized for estimating the capacity of each PCM. The PTLS solution was formulated using a recursive approach. Specifically, during each j-th time window, the calculation of the total capacity estimate $\hat{Q}_j$ was performed as follows:

$$\begin{aligned} c1_j &= \gamma\, c1_{j-1} + \mu_j^2/\Sigma_{\xi_j} \\ c2_j &= \gamma\, c2_{j-1} + \mu_j\xi_j/\Sigma_{\xi_j} \\ c3_j &= \gamma\, c3_{j-1} + \xi_j^2/\Sigma_{\xi_j} \\ \hat{Q}_j &= \frac{-c1_j + \kappa^2 c3_j + \sqrt{(c1_j - \kappa^2 c3_j)^2 + 4\kappa^2 c2_j^2}}{2\kappa^2 c2_j} \end{aligned} \tag{A1}$$

The quantities $c1_0$, $c2_0$ and $c3_0$ were initialized as reported below:

$$\begin{aligned} c1_0 &= 1/\Sigma_{\xi_0} \\ c2_0 &= \hat{Q}_0/\Sigma_{\xi_0} \\ c3_0 &= \hat{Q}_0^2/\Sigma_{\xi_0} \end{aligned} \tag{A2}$$

where $\hat{Q}_0$ is the total capacity nominal value.

Appendix B. Scenario II—Degradation Paths

In Scenario II, the sixteen PCMs experience non-uniform artificial degradation. The degradation involves increasing the R_0 value and decreasing the Q value for all modules at the beginning of each driving cycle.

The capacities degradation was uniformly sampled for each module and each cycle from the range [−6.9 Ah; −4.8 Ah] independently. Similarly, the internal resistances degradation was independently sampled from the range [1.1 mΩ; 1.8 mΩ].

Tables A1 and A2 report the degradation values of the capacities and internal resistances during Scenario II.

Table A1. Actual capacity fade due to degradation.

	Q Degradation [%]							
	M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8
cycle 1	−4.13	−4.11	−3.58	−3.98	−3.25	−4.05	−3.04	−3.41
cycle 2	−8.21	−7.46	−6.89	−7.30	−7.72	−7.63	−7.00	−6.48
cycle 3	−11.76	−11.23	−10.46	−10.31	−11.79	−11.14	−10.98	−9.73
cycle 4	−15.23	−14.67	−14.04	−13.77	−15.41	−14.82	−14.33	−14.11
cycle 5	−18.47	−17.86	−17.45	−17.29	−18.54	−18.11	−17.82	−18.27
	M_9	M_{10}	M_{11}	M_{12}	M_{13}	M_{14}	M_{15}	M_{16}
cycle 1	−3.06	−3.14	−4.23	−4.04	−3.47	−4.42	−3.05	−3.65
cycle 2	−6.41	−7.45	−7.92	−7.04	−7.17	−8.66	−7.35	−7.10
cycle 3	−10.34	−11.09	−11.94	−10.31	−11.50	−11.90	−10.50	−11.25
cycle 4	−14.58	−15.52	−16.18	−13.66	−15.40	−15.71	−14.67	−15.56
cycle 5	−18.94	−19.04	−19.76	−16.90	−19.67	−19.03	−17.67	−20.00

Table A2. Actual internal resistance increment due to degradation.

	R_0 Degradation [%]							
	M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8
cycle 1	18.81	22.65	22.95	16.86	19.89	19.45	21.46	22.09
cycle 2	36.17	45.97	42.32	36.93	36.75	35.97	45.53	47.02
cycle 3	60.11	63.44	52.70	54.41	59.43	67.37	66.89	60.01
cycle 4	78.79	86.61	85.40	74.67	71.78	75.79	85.32	83.16
cycle 5	103.55	109.54	102.25	91.16	89.37	96.95	109.76	102.69
	M_9	M_{10}	M_{11}	M_{12}	M_{13}	M_{14}	M_{15}	M_{16}
cycle 1	22.54	17.76	21.79	21.55	16.62	16.19	19.98	24.59
cycle 2	41.44	41.02	46.33	45.48	40.03	38.02	36.70	41.97
cycle 3	60.01	58.16	63.62	68.13	58.99	61.24	56.95	58.00
cycle 4	78.86	82.80	80.89	85.96	81.25	77.70	80.40	80.65
cycle 5	96.68	97.92	96.92	109.64	97.71	99.71	101.87	97.73

Appendix C. Discussion on Alternative SoH Estimation Frameworks

This section provides additional considerations on the proposed State of Health (SoH) algorithm, comparing it with an alternative estimation framework.

A key characteristic of the battery management system (BMS) proposed in this paper is its capability to simultaneously estimate the internal state of different modules while actively balancing them. One of the most challenging parameters to continuously update is the PCM's capacity Q. The proposed solution identifies the most likely value of Q that best explains the actual and past measurements from the BMS sensors (i.e., current and voltage), given the model hypothesis.

On the other hand, an alternative methodology presented in the literature (e.g., see [31]) estimates the capacity Q using an "a priori" formulation. This approach estimates Q based

only on the boundary (input) conditions of the module of interest, without accounting for its internal state. For example, the capacity depends on factors such as module aging, driving conditions, temperature, and current. These models, which correlate capacity fading with system conditions, are typically derived empirically, requiring extensive data collection for calibration (e.g., Accelerated Stress Tests (ASTs)). Once calibrated, these models are handy and effective, as they reduce the computational complexity of the estimation framework deployed on the BMS.

However, the main drawback of this "a priori" approach is its poor adaptability to outlier values in the capacity. This means that it would provide a poor estimation of capacity Q if its degradation evolution is significantly faster or slower than the average degradation of the population used to design and train the model.

For instance, in the proposed Scenario II in Section 4, the capacity Q for the fifth cycle (at the end of the battery life) follows a distribution with a mean of 118.7 Ah and a standard deviation of 1.36 Ah, which can be well approximated by a Gaussian distribution. We compare the results from the proposed approach with a similar framework where the Q value is estimated in an open-loop fashion, under the strong, ideal assumption that the comparing algorithm perfectly captures the degradation mean.

The algorithm proposed in this article consistently exhibits an estimation error lower than 2.7% across all testing conditions, including an extensive simulation campaign and accounting for realistic modeling mismatches and uncertainties. In contrast, even if the "a priori" technique perfectly matches the mean Q degradation, the statistical properties of the distribution indicate that using the mean as the estimation would result in an error greater than 2.7% for 2% of the population. This demonstrates that our approach is more robust, providing reliable results even under unlikely battery degradation paths, which might otherwise lead to premature failures or premature replacement.

References

1. Krings, A.; Monissen, C. Review and trends in electric traction motors for battery electric and hybrid vehicles. In Proceedings of the 2020 International Conference on Electrical Machines (ICEM), Gothenburg, Sweden, 23–26 August 2020; Volume 1, pp. 1807–1813.
2. Goli, C.S.; Manjrekar, M.; Essakiappan, S.; Sahu, P.; Shah, N. Landscaping and review of traction motors for electric vehicle applications. In Proceedings of the 2021 IEEE Transportation Electrification Conference & Expo (ITEC), Chicago, IL, USA, 21–25 June 2021; pp. 162–168.
3. Gabbar, H.A.; Othman, A.M.; Abdussami, M.R. Review of battery management systems (BMS) development and industrial standards. *Technologies* **2021**, *9*, 28. [CrossRef]
4. Xing, Y.; Ma, E.W.; Tsui, K.L.; Pecht, M. Battery management systems in electric and hybrid vehicles. *Energies* **2011**, *4*, 1840–1857. [CrossRef]
5. Wang, Z.; Feng, G.; Zhen, D.; Gu, F.; Ball, A. A review on online state of charge and state of health estimation for lithium-ion batteries in electric vehicles. *Energy Rep.* **2021**, *7*, 5141–5161. [CrossRef]
6. Omariba, Z.B.; Zhang, L.; Sun, D. Review of battery cell balancing methodologies for optimizing battery pack performance in electric vehicles. *IEEE Access* **2019**, *7*, 129335–129352. [CrossRef]
7. Xiong, R.; Cao, J.; Yu, Q.; He, H.; Sun, F. Critical review on the battery state of charge estimation methods for electric vehicles. *IEEE Access* **2017**, *6*, 1832–1843. [CrossRef]
8. Zhou, W.; Zheng, Y.; Pan, Z.; Lu, Q. Review on the battery model and SOC estimation method. *Processes* **2021**, *9*, 1685. [CrossRef]
9. Guo, Y.; Zhao, Z.; Huang, L. SoC estimation of Lithium battery based on improved BP neural network. *Energy Procedia* **2017**, *105*, 4153–4158. [CrossRef]
10. Dini, P.; Colicelli, A.; Saponara, S. Review on Modeling and SOC/SOH Estimation of Batteries for Automotive Applications. *Batteries* **2024**, *10*, 34. [CrossRef]
11. He, H.; Xiong, R.; Peng, J. Real-time estimation of battery state-of-charge with unscented Kalman filter and RTOS μCOS-II platform. *Appl. Energy* **2016**, *162*, 1410–1418. [CrossRef]
12. Hossain, M.; Haque, M.; Arif, M.T. Kalman filtering techniques for the online model parameters and state of charge estimation of the Li-ion batteries: A comparative analysis. *J. Energy Storage* **2022**, *51*, 104174. [CrossRef]
13. Guo, L.; Li, J.; Fu, Z. Lithium-Ion Battery SOC Estimation and Hardware-in-the-Loop Simulation Based on EKF. *Energy Procedia* **2019**, *158*, 2599–2604. [CrossRef]
14. Hannan, M.A.; Lipu, M.H.; Hussain, A.; Mohamed, A. A review of lithium-ion battery state of charge estimation and management system in electric vehicle applications: Challenges and recommendations. *Renew. Sustain. Energy Rev.* **2017**, *78*, 834–854. [CrossRef]

15. Luo, G.; Zhang, Y.; Tang, A. Capacity Degradation and Aging Mechanisms Evolution of Lithium-Ion Batteries under Different Operation Conditions. *Energies* **2023**, *16*, 4232. [CrossRef]
16. Hu, X.; Feng, F.; Liu, K.; Zhang, L.; Xie, J.; Liu, B. State estimation for advanced battery management: Key challenges and future trends. *Renew. Sustain. Energy Rev.* **2019**, *114*, 109334. [CrossRef]
17. Daowd, M.; Omar, N.; Van Den Bossche, P.; Van Mierlo, J. Passive and active battery balancing comparison based on MATLAB simulation. In Proceedings of the 2011 IEEE Vehicle Power and Propulsion Conference, Chicago, IL, USA, 6–9 September 2011; pp. 1–7.
18. Tang, X.; Gao, F.; Liu, K.; Liu, Q.; Foley, A.M. A balancing current ratio based state-of-health estimation solution for lithium-ion battery pack. *IEEE Trans. Ind. Electron.* **2021**, *69*, 8055–8065. [CrossRef]
19. Plett, G.L. Sigma-point Kalman filtering for battery management systems of LiPB-based HEV battery packs: Part 1: Introduction and state estimation. *J. Power Sources* **2006**, *161*, 1356–1368. [CrossRef]
20. Andre, D.; Appel, C.; Soczka-Guth, T.; Sauer, D.U. Advanced mathematical methods of SOC and SOH estimation for lithium-ion batteries. *J. Power Sources* **2013**, *224*, 20–27. [CrossRef]
21. Hua, Y.; Cordoba-Arenas, A.; Warner, N.; Rizzoni, G. A multi time-scale state-of-charge and state-of-health estimation framework using nonlinear predictive filter for lithium-ion battery pack with passive balance control. *J. Power Sources* **2015**, *280*, 293–312. [CrossRef]
22. Park, J.; Lee, M.; Kim, G.; Park, S.; Kim, J. Integrated approach based on dual extended Kalman filter and multivariate autoregressive model for predicting battery capacity using health indicator and SOC/SOH. *Energies* **2020**, *13*, 2138. [CrossRef]
23. Azis, N.A.; Joelianto, E.; Widyotriatmo, A. State of charge (SoC) and state of health (SoH) estimation of lithium-ion battery using dual extended kalman filter based on polynomial battery model. In Proceedings of the 2019 6th International Conference on Instrumentation, Control, and Automation (ICA), Bandung, Indonesia, 31 July–2 August 2019; pp. 88–93.
24. Kamel, M.; Sankaranarayanan, V.; Zane, R.; Maksimović, D. State-of-charge balancing with parallel and series output connected battery power modules. *IEEE Trans. Power Electron.* **2022**, *37*, 6669–6677. [CrossRef]
25. Zhang, Z.; Zhang, L.; Hu, L.; Huang, C. Active cell balancing of lithium-ion battery pack based on average state of charge. *Int. J. Energy Res.* **2020**, *44*, 2535–2548. [CrossRef]
26. Liu, R.; Zhang, C. An active balancing method based on SOC and capacitance for lithium-ion batteries in electric vehicles. *Front. Energy Res.* **2021**, *9*, 773838. [CrossRef]
27. Zhang, Z.; Cheng, X.; Lu, Z.Y.; Gu, D.J. SOC estimation of lithium-ion battery pack considering balancing current. *IEEE Trans. Power Electron.* **2017**, *33*, 2216–2226. [CrossRef]
28. Dong, G.; Yang, F.; Tsui, K.L.; Zou, C. Active balancing of lithium-ion batteries using graph theory and A-star search algorithm. *IEEE Trans. Ind. Inform.* **2020**, *17*, 2587–2599. [CrossRef]
29. Naguib, M.; Kollmeyer, P.; Emadi, A. Lithium-ion battery pack robust state of charge estimation, cell inconsistency, and balancing. *IEEE Access* **2021**, *9*, 50570–50582. [CrossRef]
30. Pröbstl, A.; Park, S.; Narayanaswamy, S.; Steinhorst, S.; Chakraborty, S. SOH-aware active cell balancing strategy for high power battery packs. In Proceedings of the 2018 Design, Automation & Test in Europe Conference & Exhibition (DATE), Dresden, Germany, 19–23 March 2018; pp. 431–436.
31. Ren, H.; Zhao, Y.; Chen, S.; Wang, T. Design and implementation of a battery management system with active charge balance based on the SOC and SOH online estimation. *Energy* **2019**, *166*, 908–917. [CrossRef]
32. Xia, Z.; Qahouq, J.A.A. State-of-charge balancing of lithium-ion batteries with state-of-health awareness capability. *IEEE Trans. Ind. Appl.* **2020**, *57*, 673–684. [CrossRef]
33. Nejad, S.; Gladwin, D.; Stone, D. A systematic review of lumped-parameter equivalent circuit models for real-time estimation of lithium-ion battery states. *J. Power Sources* **2016**, *316*, 183–196. [CrossRef]
34. Huo, Y.; Hu, W.; Li, Z.; Rao, Z. 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. *Int. J. Energy Res.* **2020**, *44*, 11583–11596. [CrossRef]
35. Choi, E.; Chang, S. A temperature-dependent state of charge estimation method including hysteresis for lithium-ion batteries in hybrid electric vehicles. *IEEE Access* **2020**, *8*, 129857–129868. [CrossRef]
36. Elmarghichi, M.; Bouzi, M.; Ettalabi, N. Robust Parameter Estimation of an Electric Vehicle Lithium-Ion Battery Using Adaptive Forgetting Factor Recursive Least Squares. *Int. J. Intell. Eng. Syst.* **2020**, *13*, 74. [CrossRef]
37. Zheng, F.; Xing, Y.; Jiang, J.; Sun, B.; Kim, J.; Pecht, M. Influence of different open circuit voltage tests on state of charge online estimation for lithium-ion batteries. *Appl. Energy* **2016**, *183*, 513–525. [CrossRef]
38. Elmarghichi, M.; Ettalabi, N. Online parameter estimation of a lithium-ion battery based on sunflower optimization algorithm. *Bull. Electr. Eng. Inform.* **2021**, *10*, 1505–1513. [CrossRef]
39. Hastie, T.; Tibshirani, R.; Friedman, J.H.; Friedman, J.H. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*; Springer: New York, NY, USA, 2009; Volume 2.
40. Barai, A.; Ashwin, T.; Iraklis, C.; McGordon, A.; Jennings, P. Scale-up of lithium-ion battery model parameters from cell level to module level–identification of current issues. *Energy Procedia* **2017**, *138*, 223–228. [CrossRef]
41. Simon, D. *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*; John Wiley & Sons: New York, NY, USA, 2006.

42. Wang, Y.X.; Zhong, H.; Li, J.; Zhang, W. Adaptive estimation-based hierarchical model predictive control methodology for battery active equalization topologies: Part I–Balancing strategy. *J. Energy Storage* **2022**, *45*, 103235. [CrossRef]
43. Borrelli, F.; Bemporad, A.; Morari, M. *Predictive Control for Linear and Hybrid Systems*; Cambridge University Press: Cambridge, UK, 2017.
44. Kumar, S.; Bhattacharyya, H.S.; Choudhury, A.B.; Chanda, C.K. Capacity Estimation of Lithium-ion Battery with Least Squares Methods. In Proceedings of the 2022 International Conference on Intelligent Controller and Computing for Smart Power (ICICCSP), Hyderabad, India, 21–23 July 2022; pp. 1–6.
45. Plett, G.L. *Battery Management Systems, Volume II: Equivalent Circuit Methods*; Artech House: Norwood, MA, USA, 2015; Volume 2.
46. Farmann, A.; Waag, W.; Marongiu, A.; Sauer, D.U. Critical review of on-board capacity estimation techniques for lithium-ion batteries in electric and hybrid electric vehicles. *J. Power Sources* **2015**, *281*, 114–130. [CrossRef]