IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

SPECIAL SECTION ON ADVANCED ENERGY STORAGE TECHNOLOGIES AND THEIR APPLICATIONS

Received October 7, 2018, accepted October 22, 2018, date of publication November 9, 2018, date of current version December 3, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2879785

Hardware-in-the-Loop Platform for Assessing Battery State Estimators in Electric Vehicles

ROCCO MORELLO, ROBERTO DI RIENZO, ROBERTO RONCELLA, (Member, IEEE), ROBERTO SALETTI, (Senior Member, IEEE), AND FEDERICO BARONTI, (Senior Member, IEEE),
Dipartimento Ingegneria dell'Informazione, University of Pisa, 56122 Pisa, Italy

Corresponding author: Federico Baronti (federico.baronti@unipi.it)

This work was supported in part by the ECSEL Joint Undertaking under Grant 737469 (AutoDrive) and in part by the University of Pisa (Urban districts with zero environmental and energetic impact) under Grant PRA-2017-33.

ABSTRACT The development of new algorithms for the management and state estimation of lithium-ion batteries requires their verification and performance assessment using different approaches and tools. This paper aims at presenting an advanced hardware in the loop platform that uses an accurate model of the battery to test the functionalities of battery management systems (BMSs) in electric vehicles. The developed platform sends the simulated battery data directly to the BMS under test via a communication link, ensuring the safety of the tests. As a case study, the platform has been used to test two promising battery state estimators, the adaptive mix algorithm and the dual extended Kalman filter, implemented on a field-programmable gate array-based BMS. The results show the importance of the assessment of these algorithms under different load profiles and conditions of the battery, thus highlighting the capabilities of the proposed platform to simulate many different situations in which the estimators will work in the target application.

INDEX TERMS Hardware-in-the-loop simulation, battery management system, battery state estimation, electric vehicles, lithium-ion batteries, system testing and verification.

I. INTRODUCTION

Nowadays, lithium-ion battery technology is widely used for the implementation of the energy storage system (ESS) needed in many applications. This happens because of the high power and energy densities and long lifetime that this technology provides. These advantages support the increase of the plug-in hybrid electric vehicles (PHEVs) and electric vehicles (EVs) markets. A fundamental component of the ESS is the battery management system (BMS) that ensures the safe and reliable behavior of the ESS. To this aim, the BMS executes various algorithms to monitor and control the battery charging and discharging phases, to equalize the charge stored into the cells, to manage the cooling system and to store or send to other units the battery information [1]. These functions require the knowledge of the inner state of all the battery cells, represented by the state of charge (SOC) and the state of health (SOH) variables. The SOC variable estimates the residual charge stored in the battery, whereas SOH describes the battery performance degradation and may account for capacity fading and internal resistance increase [2], [3].

The BMS calculates the value of these variables by using the measurable physical quantities of the battery cells, *i.e.*, terminal voltage, current and temperature. Many algorithms have been proposed for SOC estimation so far. Model-based algorithms, such as the popular extended Kalman filter (EKF) [3], [4] or the Unscented Kalman filter (UKF) [5], the Mix algorithm [6] and the Particle Filter [7], among many others, are very suitable for this task. In fact, they offer a good trade-off between accuracy and complexity, which should be affordable for a hardware real-time computation by the BMS, especially when an equivalent circuit model (ECM) of the cell is used to correct the SOC estimation [8]. The ECM parameters vary in dependence of the cell operating conditions and an effective approach to increase the estimation accuracy of the model is to track the parameter variation online. The scientific literature reports many research efforts on the development of new algorithms that jointly estimate both the state and the model parameters, like adaptive filters, such as the Kalman [3], [9], [10] and the H_∞ (H-infinity) [11] filters, alone or in combination with a least squares technique [12], [13]. These techniques provide accurate results to the detriment of their complexity and the required computational power. To meet with these requirements, some recent works propose to implement these algorithms in hardware, using a field-programmable gate array (FPGA) [14]–[17]. In fact, FPGAs are becoming very attractive solutions to achieve affordable and powerful embedded systems. In [14], an observer for SOC estimation has been implemented in hardware and tested by using a pulsed current profile. Reference [15] presents a basic BMS which uses the Adaptive Mix Algorithm (AMA) hardware estimator. Wang *et al.* [16] show the effectiveness of using an FPGA by implementing a Remaining Useful Life (RUL) estimation algorithm, where the various phases of the algorithm are executed in time division multiplexing, dynamically reconfiguring a portion of the FPGA during run-time. A SOC estimator based on a sliding-mode observer has been implemented in [17].

On the other hand, only a few works focus on the assessment of the algorithms, particularly when realistic operating conditions are considered [11], [18]–[25]. Testing the algorithms behavior in different and realistic situations is a key activity in both the development and the verification phases, since it allows the designer to tune the estimation algorithm and to make it reliable in any case. The concept of hardware-in-the-loop (HiL) simulation is applied in these works. In fact, the real BMS, or just the battery state estimator, is tested with the help of a simulation framework that reproduces the conditions in which the battery will operate.

The scope of this work is to propose and to make available a HiL simulation platform that allows us to test the battery state estimation algorithms in a simulation environment that reproduces its usage in an EV. The proposed platform is able to emulate the vehicle behavior, the battery and the entire acquisition system of the BMS and to send to the device under test the current and voltage samples needed by the battery state estimators. This makes it possible to use the platform in the design process to test the algorithm implemented in MATLAB/Simulink® and during the implementation of the system in order to test its behavior in a hardware device, such as an FPGA or a microcontroller.

To show the platform features, two algorithms for battery state estimation, the AMA and the dual EKF (DEKF), have been used as a case-study. They are promising solutions for SOC and parameters co-estimation [26], [27]. Both estimators have been implemented on an Intel MAX® 10 FPGA, which targets low-cost applications.

The remainder part of the paper is organized as follows. The next Section discusses the state-of-the-art of the HiL platforms and the main contributions of this work. The implemented HiL simulation platform and the blocks used to build this framework, including the battery and the electric traction models, are described in more detail in sections III, IV, V, and VI. The case study used to demonstrate the HiL capabilities is discussed in Section VII. The conclusions are finally drawn in Section VIII.

II. BACKGROUND AND MAIN CONTRIBUTIONS

HiL simulation platforms can generally be categorized in: Power-HiL (PHiL) and communication-based HiL [18]. In particular, Power-HiL (PHiL) platforms allow the BMS hardware to be tested when the power fluxes from/to the battery are real and not simulated. Instead, the BMS control algorithms are tested by substituting the battery power path with simulated battery data (*i.e.*, voltage, current, and temperature of each battery cell), provided via a communication link, in the communication-based HiL platforms.

PHiL testing platforms, as those described in [11], [19]–[22], and [28], include a battery, or an emulator able to reproduce the battery behavior, into the loop. In particular, the cited examples report on the assessment of the performance of the battery state estimators with specific current profiles under which the battery is exercised. In [19], a profile based on the electric power measured on an EV driving the Federal Test Procedure (FTP) has been used, while the current has been generated starting from the combination of the Extra-Urban Driving Cycle (EUDC) and the Economic Commission for Europe urban (ECE) driving schedules in [21]. A current profile typical for the use of a smartphone is employed in [20], whereas a dynamic stress test is applied in [11] and [22]. In these works, the implemented systems include complex instrumentation like cyclers or programmable loads and chargers, in order to apply the selected profiles to the battery, and accurate measurement equipment to acquire the cell quantities. In [28], the HiL system is based on an emulator able to reproduce the battery cell dynamics to validate the BMS functions of voltage monitoring, active and passive cell balancing.

Communication-based platforms are presented in [23]–[25]. They allow less expensive, faster and safer tests than those performed on real batteries, because the battery itself belongs to the simulated part of the testing loop. The obvious disadvantage of the communication-based platforms is the impossibility to test some hardware parts of the BMS, such as the measurement and the balancing circuits. On the contrary, any possible load power profile can easily be applied to analyses the BMS behavior and to validate the battery state estimators in the most demanding situations. Moreover, not using a real battery in the HiL platform also allows saving the battery charging time period, so successive profiles can be applied to the battery without pauses. A simple battery model composed of 96 cells is used in [24] to perform a functional test of the BMS. A 12-cell SOC estimation algorithm implemented on a microcontroller is tested in [23] by using a cell ECM that does not take into account the parameter variation with SOC and temperature. The simple models used in these two works are inadequate when high reliability and accuracy of the BMS algorithms is required. Some of these limitations are overcome in [25], where a platform able to simulate the EV behavior under many driving schedules has been presented and used to check the performance of two battery state estimators.

This work presents a communication HiL platform based on that described in [25]. The main innovations with respect to the previous works are that the modeled EV and its battery parameters are fully configurable and that the temperature effects are also considered by using a simple thermal model

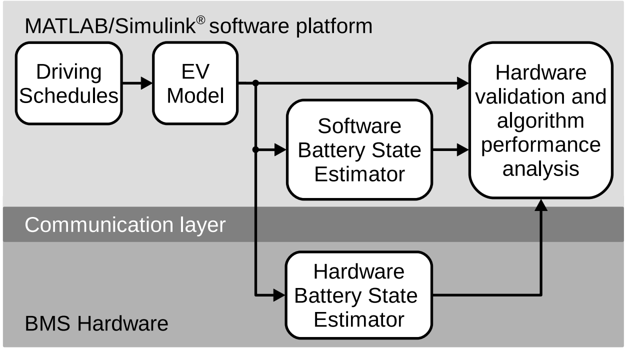

MATLAB/Simulink® software platform
Driving Schedules
EV Model
Software Battery State Estimator
Hardware validation and algorithm performance analysis
Communication layer
BMS Hardware
Hardware Battery State Estimator

FIGURE 1. Block diagram of the developed hardware-in-the-loop simulation environment.

of the battery and temperature-dependent cell model parameters. In this way, the battery behavior is reproduced considering a large set of conditions, such as the imbalance among the battery cells, the parameter variations due to their operating conditions and ageing, and different current loads. In fact, the HiL platform allows us to simulate an EV in many different driving scenarios and thus with different power dynamics. Furthermore, since the communication-based platforms directly send the acquired quantities to the main unit of the BMS, the acquisition chain and the actuators of the BMS are out of the loop and thus neglected in the tests. This problem has been solved by introducing a model of the voltage and current acquisition chain in this platform.

Another important consideration regards the instrumentation required by the platform to execute its tasks. External devices, provided with high computational power and real-time operating systems, are used in [23] and [24] to perform the test in real-time. In our case, the entire system can be built in Simulink in order to obtain a low cost software platform, without using additional devices. However, this does not guarantee the real-time execution of the model and the simulation performances depend on the computational power of the used computer and on the type of digital communication link used to send the simulated data to the hardware estimator under test.

III. HARDWARE-IN-THE-LOOP PLATFORM DESIGN

The HiL platform (FIGURE 1) has entirely been implemented in the MATLAB/Simulink® environment. The driving schedule block allows us to select a standard speed profile, which is the input of the EV model block. This block, described in details in Section V, generates the input signals for the estimation algorithms, which are simultaneously executed by both the BMS hardware and the platform software. In fact, the algorithms under test are also implemented with MATLAB scripts and integrated in the Simulink framework, in order to compare their results with those obtained from the BMS and to validate their hardware implementation.

The communication with the BMS hardware is performed by the communication layer. This part of the HiL platform aims at sending the generated battery quantities to the BMS hardware under test by using a digital link. A large set of communication typologies is supported by the MATLAB environment, such as serial interfaces, CAN-bus, and Ethernet, but also a custom interface available from a third-party can be added.

IV. DRIVING SCHEDULES

The Driving Schedules block provides the speed values relative to the driving cycle selected among the 18 available profiles. Further driving cycles from regulation authorities or custom choices can easily be added to the default portfolio. These driving cycles are used for vehicle emission and fuel economy assessment and are representative of various driving situations. Table 1 shows the main characteristics of each driving cycle. We note a significant variation in the average speed and, thus, in the electric power required from the traction battery. These cycles are defined by various specialized organizations, therefore they can be classified according to the geographical area to which they belong:

TABLE 1. Main characteristics of the driving schedules included in the platform.

Geographical area	Driving schedule	Duration (min)	Distance (km)	Average speed (km/h)
Worldwide	WLTP class 3	30	23.3	46.5
United States	UDDS	23	12.1	31.5
	HWFET	13	16.8	77.5
	FTP	31	17.6	34.1
	IM240	4	3.1	47.1
	SFTP US06	10	12.9	77.2
	SFTP SC03	10	5.8	34.5
	NYCC	10	1.9	11.4
	LA92	24	15.8	39.6
	LA92 short	16	11.1	41.8
European Union	EUDC	7	6.8	58.6
	NEDC	20	8.5	25.4
	ECE R15	3	0.8	16.5
	ArtUrban	17	5.0	17.6
	ArtRoad	18	17.2	57.4
	ArtMw130	18	29.0	96.8
	ArtMw150	18	29.8	99.5
Japan	J1015	15	6.4	25.6

1) WORLDWIDE

The Worldwide harmonized Light vehicles Test Procedures (WLTP) is maintained by the UNECE World Forum for Harmonization of Vehicle Regulations [29]. The class determines the power-to-mass ratio of the vehicles. In particular, class 3 is representative of European and Japanese vehicles.

2) UNITED STATES

The Urban Dynamometer Driving Schedule (UDDS), the Highway Fuel Economy Test (HWFET), the Federal Test Procedure (FTP), the two Supplemental Federal Test Procedures (SFTP US06 and SC03), the New York City Cycle (NYCC) and the Inspection and Maintenance (IM240) driving cycle are defined by the U.S. Environmental Protection Agency [30]. The UDDS has been developed to test heavy-duty vehicles. All the FTP cycles are variants of the UDDS one, used to test light-duty vehicles in urban scenarios. For example, the SFTP US06 represents aggressive, high speed and/or high acceleration driving behaviors, whereas the SFTP SC03 also takes into consideration the engine load due to the use of air conditioning. Another cycle useful to test urban profile is the NYCC, whereas the HWFET is used for simulating highway scenarios. The IM240 is usually employed for emission testing. The LA92 and the LA92 short (which consists of the first 969 s of the LA92) are developed by the California Air Resources Board [31]. They are mostly used for simulating urban driving.

3) EUROPEAN UNION

The New European Driving Cycle (NEDC), the Extra-Urban Driving Cycle (EUDC) and the Economic Commission for Europe urban driving cycle (ECE R15) are maintained by the United Nations Economic Commission for Europe (UNECE) [32]. The Urban cycle (ArtUrban), the Rural road cycle (ArtRoad) and the Motorway cycles (ArtMw130 and ArtMw150, with a maximum speed of 130 and 150 km/h, respectively) are included in the Common Artemis Driving Cycles (CADC), developed within the European Artemis (Assessment and Reliability of Transport Emission Models and Inventory Systems) project.

4) JAPAN

The 10-15 cycle (J1015) had been used in Japan for emission and fuel economy testing of light duty vehicles.

V. ELECTRIC VEHICLE MODEL

The EV model consists of three blocks, as shown in FIGURE 2, and is executed with a 100 ms integration time step. This value is considered sufficient to capture the system dynamics of interest. The battery current I_b is computed as the ratio of the electric power P_e to the battery voltage V_b, given by the sum of the cell voltages $\boldsymbol{V}$ computed by the Battery Model block. P_e is generated by the Electric Traction block starting from the speed values v coming from the Driving Schedules block according to the selected driving cycle. The Battery Model block also provides the vector of the ECM parameters $\boldsymbol{p}$ and the SOC value for each battery cell, which constitute the reference values to be compared against the estimates computed by the BMS under test. Finally, the Sensor Model block generates a noisy version of the battery current and cell voltages, $\hat{I}_b$ and $\hat{\boldsymbol{V}}$, respectively, which form the input of the BMS under test. A detailed description of each block of the implemented EV model is reported below.

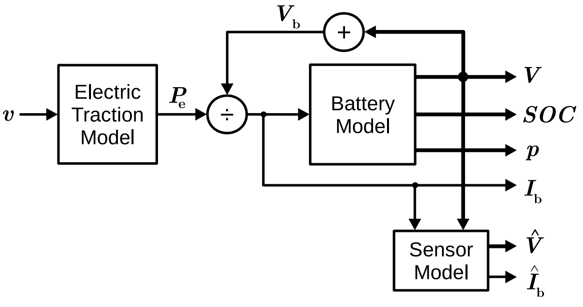


FIGURE 2. Block diagram of the EV Model.

A. ELECTRIC TRACTION MODEL

The electric traction model block simulates an EV travelling on a flat road by using a simple dynamic model, as in [26] and [33]. The mechanical power P_m is computed by (1), where v is the vehicle speed and the two frictional forces are due to the air and rolling resistances. The symbols in (1) are defined in Table 2.

$$P_m = Fv = \left(M\dot{v} + \frac{1}{2}\rho_{air} S C_X v^2 + \alpha_R M g \right) v \tag{1}$$

TABLE 2. Electric traction model parameters for Nissan Leaf.

Symbol	Description	Value
M	Kerb weight	1525 kg
S	Frontal area	2.27 m^2
C_X	Drag coefficient	0.29
α_R	Rolling resistance	0.01
ρ_{air}	Air density	1.2 kg/m^3
g	Gravity acceleration	9.82 m/s^2
η_{wheel}	Efficiency from battery to wheels	0.7
η_{reg}	Efficiency from wheels to battery	0.5

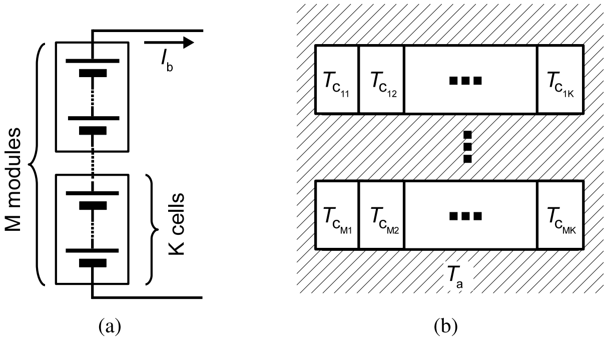
Then, the electric power P_e is obtained from P_m by using (2), in which two different energy efficiency values are taken into consideration, η_{wheel} for the traction and η_{reg} for the regenerative braking.

$$P_e = \left(\frac{1}{\eta_{wheel}} \frac{1 + sgn(P_m)}{2} + \eta_{reg} \frac{1 - sgn(P_m)}{2} \right) P_m \tag{2}$$

The default model parameters used in (1) and (2) are chosen to resemble a commercial electric car (Nissan Leaf) and they are reported in Table 2 [26]. These parameters can easily be changed by the user to fit the model to other vehicles.

B. BATTERY MODEL

The battery model is capable of simulating a battery composed of M modules, each of them consisting of K series-connected cells, for a total of $N = MK$ cells. The only input is the battery current I_b, which is the same for all the series-connected cells, as shown in FIGURE 3(a). The cell



(a) (b)

FIGURE 3. Electrical (a) and thermal (b) organization of the simulated battery.

model is composed of two main parts: the ECM and the cell thermal model (CTM). At each time step, the N instances of the ECM generate the cell voltage and SOC value arrays, as well as the present values of the model parameters. At the same time, the temperature of each cell is determined by the CTM. This allows the simulator to track the temperature distribution inside the battery pack. The modules are assumed to be thermally isolated to each other in the battery pack (see FIGURE 3(b)). Moreover, the K cells contained in a module are placed side by side, so each CTM interacts with the nearest two cells only, in order to simulate the cell-to-cell heat transfer. The inputs of the CTM are the left and right surface temperatures T_{ls} and T_{rs} of the considered cell, the ambient temperature T_{a} and the cell current, terminal voltage and open circuit voltage, provided by the ECM. The surface temperatures of each internal cell are equal to the corresponding surface temperatures of the neighboring cells. The output is the cell core temperature T_{c}. The interaction between the electric and the thermal models is shown in FIGURE 4.

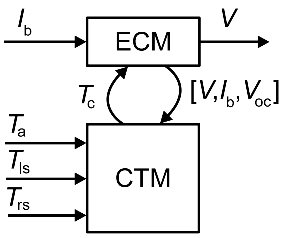
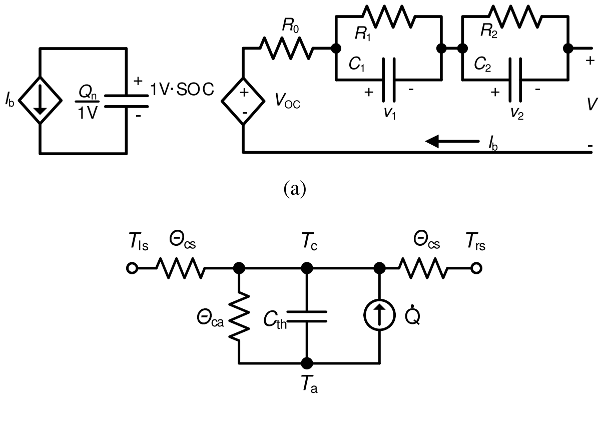


FIGURE 4. Integration of the electric model with the thermal one.

1) CELL MODEL

The battery cell electric model is the ECM shown in FIGURE 5(a), where the cell capacity is modeled by a linear capacitor, whose value is equal to the nominal cell capacity Q_{n} (expressed in Coulomb) divided by 1 V. The terminal voltage V is the sum of the open-circuit voltage V_{OC} and a dynamic term, which incorporates the voltage across the internal ohmic resistance R_0 and the double layer (V_1) and diffusion (V_2) effects occurring in a Li-ion battery during charging and discharging. As the parameters $[R_0, R_1, R_2, C_1, C_2]$ depend on temperature, SOC and current, their values are stored in 3D LUTs. As far as the open-circuit voltage is concerned, the dependency on SOC is only considered, while the capacity is kept constant during the tests.



(a)

(b)

FIGURE 5. Electric circuit model with two RC branches (a) and electrical equivalent thermal model (b) of the battery cells.

The temperature of each cell is determined by the CTM, based on the thermal models developed in [34]–[36]. The CTM is an electrical equivalent model of the thermal system, as shown in FIGURE 5(b). It considers the heat exchange between the cells and the external environment and also the cell-to-cell exchange in the battery module. This rather simple thermal model takes into account the main thermal exchanges while maintaining the computational load affordable, also when the number of cells becomes large. The thermal resistances Θ_{cs} modeling the core-to-surface thermal paths have the same value. The cell-to-air thermal path is modeled by the thermal resistance Θ_{ca}. The cell core temperature T_{c} is represented by the voltage at the top terminal of the thermal capacitance C_{th}. The current generator $\dot{Q}$ models the cell self-heating [37]. It is the sum of a reversible contribution, the entropic heat flow, and an irreversible part, due to the ohmic losses inside the cell, as shown in (3).

$$\dot{Q} = I_{\mathrm{b}} T_{\mathrm{c}} \frac{\partial V_{\mathrm{OC}}}{\partial T_{\mathrm{c}}} + I_{\mathrm{b}}(V_{\mathrm{OC}} - V) \tag{3}$$

2) CELL MODEL DEFAULT PARAMETERS AND VALIDATION

Pulsed current tests (PCTs) were performed at different temperatures and pulse amplitudes on a 1.5 Ah NMC cell to extract the default ECM parameters that fill the LUTs [34]. The measured parameters are scaled to represent a cell from the same technology but with different capacity [26]. The same cell has been used to obtain the default thermal parameters by a thermal characterization with the setup presented in [34] and the procedure described in [38]. However, different types of cells can be simulated by simply using data provided by characterization tests performed on other chemistries. The default cell model has then been validated by using the stepwise current test shown in FIGURE 6. The current measured in the real test is used as input of the model, configured to simulate only one cell. We note in the figure that the predicted cell voltage is in very good agreement with the measured one, resulting in maximum and rms errors of 149 mV and 20.4 mV, respectively. Good results are also achieved for the cell temperature prediction, as the maximum and rms errors are 1.7 °C and 0.48 °C, respectively.

C. SENSOR MODEL

In real applications, the measurements are performed by an acquisition system composed by current and voltage sensors and one or more analog-to-digital converters (ADC) that convert the analog signals into digital values. As this process introduces measurement errors and noise, a model of the

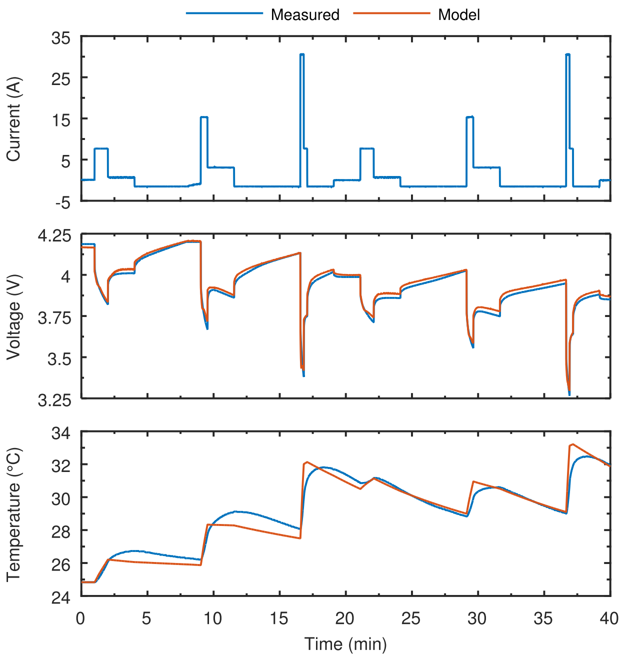


FIGURE 6. Comparison of the model and measured voltages and temperatures.

acquisition system that considers these phenomena has been developed to test the algorithm performance in a real system.

This model takes as input the current I_b and the cell voltages $\boldsymbol{V}$ generated by the battery model and provides as output the acquired current $\hat{I}_b$ and voltages $\hat{\boldsymbol{V}}$, as shown in FIGURE 2. Measurement noise and offset errors are added to the simulated quantities, then converted into digital values by means of an ADC with a given number of bits and an ideal characteristic. The offset and standard deviation of the measurement noise, as well as the ADC bits are independently configurable for the voltage and current acquisition channels. In this way, the real current and voltage sensors can individually be modeled in accordance with their characteristics.

VI. DEVELOPED SIMULINK LIBRARY

The models described in the previous sections have been implemented as configurable Simulink custom blocks and grouped in the library shown in FIGURE 7, which is available online [39]. The Battery Electric model implements the series connection of the ECMs, where the number of cells can be configured by the function block parameters window. The inputs of this block are the battery current and the array of the cell temperatures provided with the Battery Thermal block. The latter is organized as described in FIGURE 3(b), implementing a configurable number of Module Thermal blocks which emulate the temperature exchange in a module composed by a certain number of CTMs.

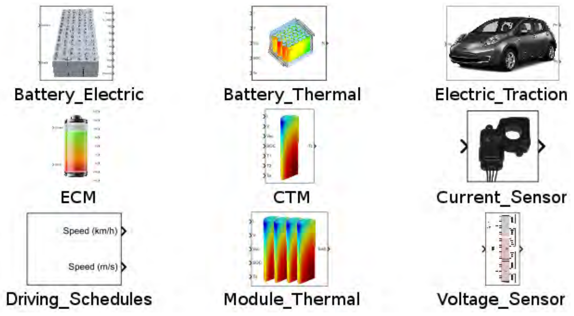


FIGURE 7. Developed Simulink library.

The possibility to connect the blocks in different combinations and to change the configurable parameters allows the user to create a simulation platform specific to the BMS under test. For example, the user can choose to not include the thermal model or the model of the sensors. This allows the reduction of the platform complexity and hence of the simulation time.

VII. HiL PLATFORM FOR VALIDATING HARDWARE ESTIMATORS: A CASE STUDY

A. BMS HARDWARE

The BMS under test is based on an Intel MAX® 10 FPGA (10M50DAF484C6GES device), which is one of the low cost devices among the Intel FPGAs. This BMS has been used to validate the hardware implementation of the AMA and DEKF estimators and to assess their performance without any power path. Besides the estimators, a soft core processor and a JTAG module are hardware programmed into the FPGA to allow the communication with the Simulink framework (FIGURE 8). These peripherals are provided with a memory mapped interface in order to be connected to the other components of the system via the FPGA Avalon Bus.

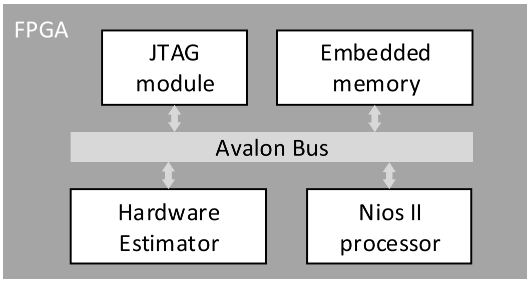


FIGURE 8. Block diagram of the FPGA-based BMS hardware.

Both the AMA and the DEKF algorithms are model-based techniques, in which the ECM of FIGURE 5(a) with only one RC branch is adopted [27]. They implement closed-loop techniques, in which the battery cell voltages are predicted with the help of a model. The estimated model state variables of each cell are corrected to reduce the difference between the predicted cell voltage and the measured one. The model capability to reproduce the cell behavior in a reliable way determines the SOC estimation accuracy. To this aim, the model parameters are continuously updated to follow the variations induced by changes in the battery operating conditions (*i.e.*, SOC and temperature) and ageing, as shown in FIGURE 9. The AMA is based on a combination of the Mix Algorithm for SOC estimation [6] and the Moving Window

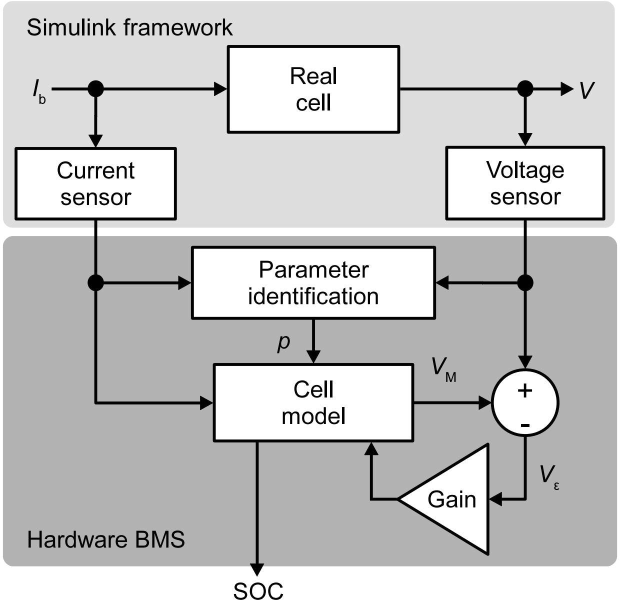


FIGURE 9. Block diagram of the model-based algorithms. In this work, the real cells and the voltage and current sensors are modeled in Simulink.

Least Squares (MWLS) method for identifying online the ECM parameters [R_0, R_1, C_1] [40]. The DEKF technique [4] exploits two cooperating Kalman Filters for non linear systems, one for the state and another for the parameter estimation. Both estimators fit well in the chosen device and the execution time to update both the state and the parameters of one single cell is 34 μs for the AMA and 16.5 μs for the DEKF (measured with a clock frequency of 100 MHz). This very short execution time makes it possible to use the same module in time multiplexing for estimating a large number of cells.

B. HIlL PLATFORM INSTANCE

The user interface of the developed HiL platform is visible in the photograph of FIGURE 10, the bottom of which also shows the FPGA-based BMS board. The library blocks shown in FIGURE 7 have been used to assemble a platform able to simulate the Nissan Leaf traction battery. This battery has the same structure described in Section V-B, and it has been build by using the Battery Electric and the Battery Thermal blocks of the library. The block parameters have been changed in order to emulate 96 series-connected NMC cells with a capacity of 66.2 Ah, obtaining a battery with a nominal voltage of 355.2 V. From a thermal point of view, there are 8 modules of 12 series-connected cells. All the executed simulations consist of the repetition of one of the driving schedules listed in Table 1, until the SOC battery reaches 20 %, starting from 80 %. The simulations have been executed on a computer with an Intel® Core™ i7-4790 processor and 16 GB of RAM. The platform employs about 6 ms per cell to execute a single simulation step. This time includes the computation of the cell state and electrical quantities, the sending of the simulated data to the FPGA, the execution of the algorithm on the FPGA and the receiving of the computed results. Table 3 shows that the limiting factor of the platform performance is the time employed by the communication layer. In this platform instance, the interface between the Simulink framework and the BMS hardware is obtained with a communication layer mapped on the JTAG link. The software that allows the use of this link is contained in the System Console API provided by Intel. This is a set of commands which allows the real-time interaction between the MATLAB environment and the FPGA thanks to the use of a memory mapped interface.

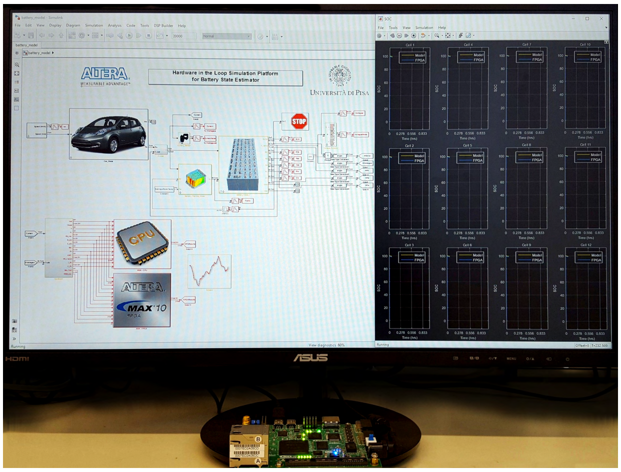

ALTERA
Hardware in the Loop Simulation Platform for Battery State Estimator
UNIVERSITÀ DI PISA
STOP
CPU
ALTERA
MAX 10
ASUS

FIGURE 10. Hardware-in-the-loop platform in action.

TABLE 3. Platform single step execution phases.

Step phase	Execution time per cell
1 - Computation of the battery state and electrical quantities	208 μs
2 - Sending data to the estimator	2.3 ms
3 - Algorithm execution	35 μs
4 - Reading data from the estimator	3.53 ms

The implementation validation of both estimators is carried out by comparing the results coming from the FPGA and those obtained from the software-executed algorithms. Various scenarios have been simulated in order to demonstrate the capability offered by the HiL platform in assessing the algorithm performance.

C. NOISELESS ACQUISITION SYSTEM

The algorithms have first been tested in case of noiseless sensors, in which the acquisition system only introduces the quantization error. Here, the simulations are performed on identical cells that all start from the same SOC. For this reason, all the battery modules behave in exactly the same way. As an example, FIGURE 11 shows the simulation results in SOC estimation for one of the two central cells in a battery module during the UDDS cycle. The estimated SOC is in good agreement with the reference one computed by the HiL

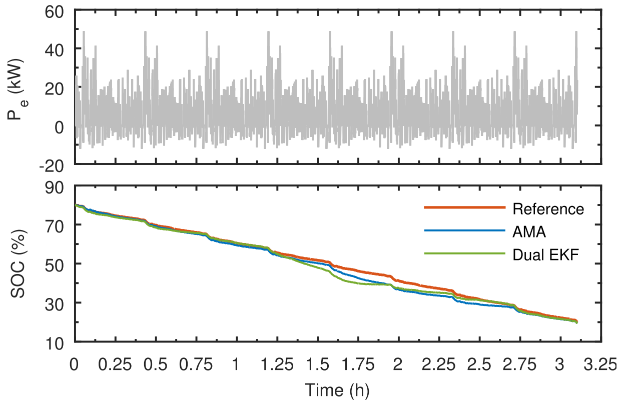


FIGURE 11. **Behavior of SOC for one of the two central cells of the first module, during a UDDS test.**

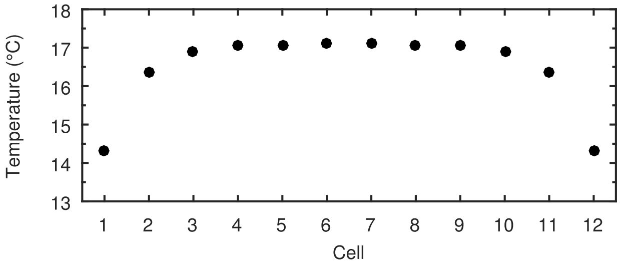


FIGURE 12. **Final temperature of each cell in one module in the UDDS test.**

battery model, except for the values from 50 % down to 30 % where the SOC is poorly observable [27].

The temperature of all the cells has been calculated by the thermal model, starting from a steady-state of 10 °C, which is the value chosen for the ambient temperature. This ambient temperature value emphasizes the cell parameter variations induced by self heating. FIGURE 12 shows the end of test temperature of the 12 cells contained in a module. We note that the external cells show a temperature lower than the others as expected, because of their larger heat exchange with the ambient. Furthermore, the higher heating of the central cells with respect to the external ones allows us to observe the temperature induced variations of the battery parameter values and to test the capability of the algorithms in tracking different parameter changes during the same test. FIGURE 13 shows the temperature and the identified ohmic resistance R_0 for one of the two external cells (cell 1) and one of two central cells (cell 6) in the module. It can be observed that both estimators well identify R_0, also in the last part of the discharge when it changes significantly because of the variation of the electrolyte resistance [41]. This is a remarkable achievement, as R_0 significantly affects the model accuracy and is a good figure of battery ageing.

These results are comparable to those obtained in previous works available in literature [25]. Moreover, the presented platform has been used to test the algorithms implemented on the FPGA by using a wider set of driving cycles in different operating conditions. The maximum and rms SOC errors are reported in FIGURE 14 (empty markers), where the driving schedules are sorted according to their average speed. The rms error is always below 5.1 % and 2.9 % for the AMA and DEKF, respectively, indicating that a good SOC estimation is achieved by both estimators in all the driving cycles.

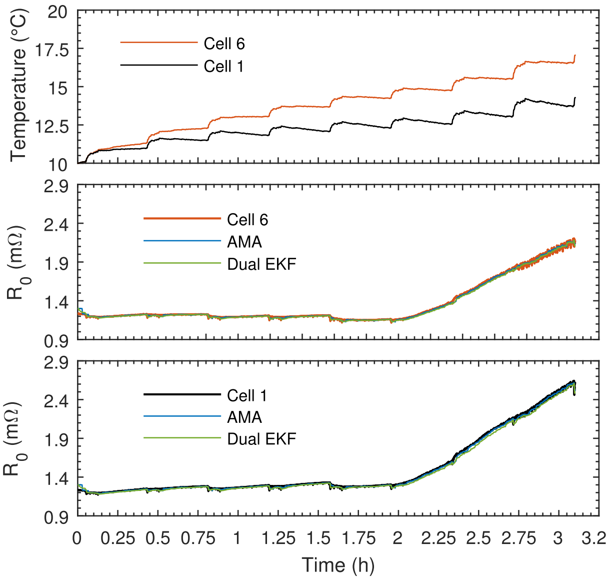


FIGURE 13. **Temperature and series resistance for one of the two central cells and one of the two external cells of the first module, during a UDDS test.**

Finally, we note that the properties of the driving cycle and thus of the corresponding electric power affect the performance of the state estimator. In fact, the AMA has slightly better results on urban cycles, whereas the DEKF is more reliable for motorway driving schedules where the requested electric power is higher.

D. UNBALANCED CELLS

The series connection of the cells, together with the strict operating ranges, is critical as it introduces the well-known issue of cell unbalancing [42]. This phenomena can easily be reproduced by the developed HiL platform by using different SOC initial values for the cells. This feature allows us to analyses the response of an estimation algorithm to uncertainty in the initial SOC value in a much more effective way than using PHiL testing platforms. To this aim, FIGURE 15 shows the SOC estimation computed by the two algorithms when the SOC of two cells is initialized with different values of 85 % (Cell 6) and 75 % (Cell 7) instead of 80 %, as for the other cells in the simulated battery pack. The SOC initialization value of the algorithms are set to 80 %, since the cell unbalancing is an unpredictable state. However, we can note that both algorithms are able to correct wrong SOC initializations and that the DEKF is faster than the AMA [27]. Let us consider the wrong estimation to be fully recovered when the difference between the present value and that obtained with the correct initialization is lower than 1 %. Thus, the DEKF employs 322 s to recover the wrong SOC initialization of the

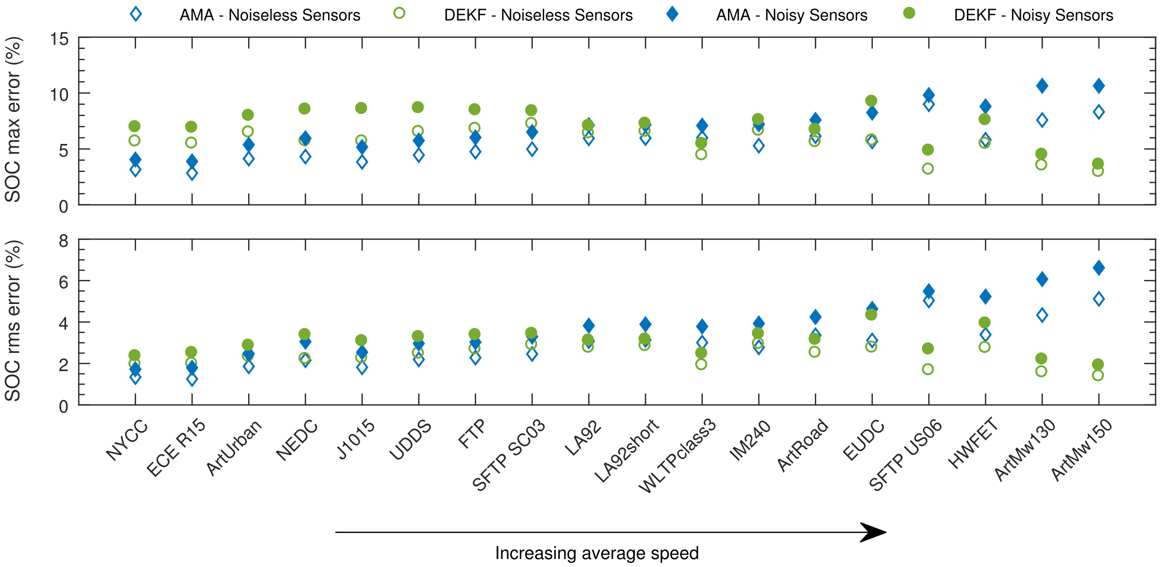


FIGURE 14. SOC estimation errors for the AMA and DEKF algorithms. The empty markers refer to the case of noiseless sensors, whereas the filled markers to the case of noisy sensors (with standard deviation of 2 A and of 5 mV on the current and voltage measurement, respectively).

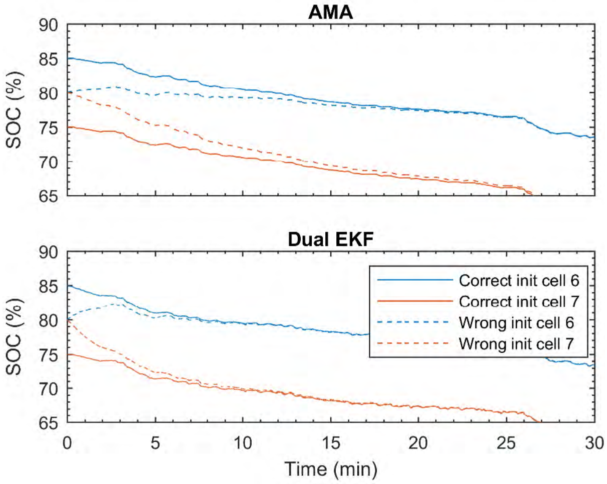


FIGURE 15. SOC correction after a wrong initialization compared to the value estimated when the right initialization value is used.

cell 6, instead of 798 s needed by the AMA for the same cell. In any case, the recovering time is small compared to the overall discharge time, which is longer than 3 h.

E. NOISY ACQUISITION SYSTEM

As the implemented platform is able to simulate non-ideal sensors, it can also be used to analyses the algorithm behavior in presence of measurement noise. In our case study, the platform simulates two ADCs with a resolution of 16 bits and a full scale value $FS_{\mathrm{I_b}}$ of 400 A for the current sensor and FS_{V} of 5 V for the voltage sensor. Furthermore, they introduce no offset and a white Gaussian noise whose standard deviation is chosen as a reasonable percentage of the full scale values. We considered a standard deviation of $\sigma_{\mathrm{I_b}} = 2\,\mathrm{A}$ (0.5 % $FS_{\mathrm{I_b}}$) on the current and of $\sigma_{\mathrm{V}} = 5\,\mathrm{mV}$ (0.1 % FS_{V}) on the voltage measurements.

The estimation errors obtained in these noisy conditions for every driving schedules are reported in FIGURE 14 (filled markers). Both algorithms show a slight degradation of the accuracy in SOC estimation. However, the degradation is not large because the AMA and DEKF methods are capable of rejecting the noise on the current measurement, while they suffer from some sensitivity to errors on the voltage measurement [43].

VIII. CONCLUSIONS

This paper has discussed the development of a HiL platform, implemented in the MATLAB/Simulink® environment, for testing BMS with battery state estimators in EV applications, under realistic operating conditions and without any power path. The HiL platform consists of a dynamic model of an EV, a model of the traction battery, which takes into account the electrical and the thermal behavior of the cells, and a model of the acquisition system. It is able to emulate the battery behavior with high accuracy in many scenarios, which differ in the driving style, ambient temperature, type of vehicle and battery conditions.

To validate the platform and prove its capabilities, a BMS provided with two promising model-based estimators, the AMA and the DEKF algorithm, has been implemented on an Intel MAX® 10 based board. The two hardware implemented algorithms have been tested using the developed HiL platform. Simulation results have shown that both algorithms are well executed on the FPGA-based BMS with good estimation performance, but have also highlighted that the estimator performance may significantly change with the current load profile and the operating conditions of the battery.

Thanks to our HiL platform, both estimators have extensively been assessed in different situations without the need of a real battery and in a completely safe environment. The simulations enabled by the HiL platform has allowed us to show that these estimators are suitable for battery state and ECM parameters estimation in EVs.

REFERENCES

[1] M. A. Hannan, M. M. Hoque, A. Hussain, Y. Yusof, and P. J. Ker, "State-of-the-art and energy management system of lithium-ion batteries in electric vehicle applications: Issues and recommendations," *IEEE Access*, vol. 6, pp. 19362–19378, 2018.

[2] H. Rahimi-Eichi, U. Ojha, F. Baronti, and M.-Y. Chow, "Battery management system: An overview of its application in the smart grid and electric vehicles," *IEEE Ind. Electron. Mag.*, vol. 7, no. 2, pp. 4–16, Jun. 2013.

[3] S. Nejad, D. T. Gladwin, and D. A. Stone, "On-chip implementation of extended Kalman filter for adaptive battery states monitoring," in *Proc. 42nd Annu. Conf. IEEE Ind. Electron. Soc. (IECON)*, Oct. 2016, pp. 5513–5518.

[4] G. L. Plett, "Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs: Part 3. State and parameter estimation," *J. Power Sour.*, vol. 134, no. 2, pp. 277–292, 2004.

[5] W. Wang, X. Wang, C. Xiang, C. Wei, and Y. Zhao, "Unscented Kalman filter-based battery SOC estimation and peak power prediction method for power distribution of hybrid electric vehicles," *IEEE Access*, vol. 6, pp. 35957–35965, 2018.

[6] F. Codeca, S. M. Savaresi, and G. Rizzoni, "On battery state of charge estimation: A new mixed algorithm," in *Proc. IEEE Int. Conf. Control Appl.*, Sep. 2008, pp. 102–107.

[7] D. Zhou, A. Ravey, F. Gao, A. Miraoui, and K. Zhang, "On-line estimation of lithium polymer batteries state-of-charge using particle filter based data fusion with multi-models approach," in *Proc. IEEE Ind. Appl. Soc. Annu. Meeting*, Oct. 2015, pp. 1–8.

[8] C.-S. Huang and M.-Y. Chow, "Accurate Thevenin's circuit-based battery model parameter identification," in *Proc. IEEE 25th Int. Symp. Ind. Electron. (ISIE)*, Jun. 2016, pp. 274–279.

[9] A. Baba and S. Adachi, "Simultaneous state of charge and parameter estimation of lithium-ion battery using log-normalized unscented Kalman filter," in *Proc. Amer. Control Conf. (ACC)*, Jul. 2015, pp. 311–316.

[10] C. Huang, Z. Wang, Z. Zhao, L. Wang, C. S. Lai, and D. Wang, "Robustness evaluation of extended and unscented Kalman filter for battery state of charge estimation," *IEEE Access*, vol. 6, pp. 27617–27628, 2018.

[11] Y. Zhang, R. Xiong, H. He, and W. Shen, "Lithium-ion battery pack state of charge and state of energy estimation algorithms using a hardware-in-the-loop validation," *IEEE Trans. Power Electron.*, vol. 32, no. 6, pp. 4421–4431, Jun. 2017.

[12] H. Dai, T. Xu, L. Zhu, X. Wei, and Z. Sun, "Adaptive model parameter identification for large capacity Li-ion batteries on separated time scales," *Appl. Energy*, vol. 184, pp. 119–131, Dec. 2016.

[13] J. Zhang, Y. Wei, and H. Qi, "State of charge estimation of $LiFePO_4$ batteries based on online parameter identification," *Appl. Math. Model.*, vol. 40, nos. 11–12, pp. 6040–6050, 2016.

[14] N. Otero, H. Rahimi-Eichi, J. J. Rodriguez-Andina, and M.-Y. Chow, "FPGA implementation of an observer for state of charge estimation in lithium-polymer batteries," in *Proc. Int. Conf. Mechatronics Control (ICMC)*, Jul. 2015, pp. 1646–1651.

[15] R. Morello, R. Di Rienzo, F. Baronti, R. Roncella, and R. Saletti, "System on chip battery state estimator: E-bike case study," in *Proc. 42nd Annu. Conf. IEEE Ind. Electron. Soc. (IECON)*, Oct. 2016, pp. 2129–2134.

[16] S. Wang, D. Liu, J. Zhou, B. Zhang, and Y. Peng, "A run-time dynamic reconfigurable computing system for lithium-ion battery prognosis," *Energies*, vol. 9, no. 8, p. 572, Jul. 2016.

[17] G. Vicidomini, G. Petrone, E. Monmasson, and G. Spagnuolo, "FPGA based implementation of a sliding-mode observer for battery state of charge estimation," in *Proc. IEEE Int. Symp. Ind. Electron. (ISIE)*, Jun. 2017, pp. 1268–1273.

[18] J. V. Barreras *et al.*, "An advanced HIL simulation battery model for battery management system testing," *IEEE Trans. Ind. Appl.*, vol. 52, no. 6, pp. 5086–5099, Nov. 2016.

[19] Y. He, W. Liu, and B. J. Koch, "Battery algorithm verification and development using hardware-in-the-loop testing," *J. Power Sources*, vol. 195, no. 9, pp. 2969–2974, 2010.

[20] G. V. Avvari, B. Pattipati, B. Balasingam, K. R. Pattipati, and Y. Bar-Shalom, "Experimental set-up and procedures to test and validate battery fuel gauge algorithms," *Appl. Energy*, vol. 160, pp. 404–418, Dec. 2015.

[21] Y. Ma, B. Li, X. Zhou, and H. Chen, "Battery state of charge estimation hardware-in-loop system design based on xPC target," in *Proc. World Congr. Intell. Control Automat. (WCICA)*, Jun. 2016, pp. 1338–1343.

[22] C. Chen, R. Xiong, and W. Shen, "A lithium-ion battery-in-the-loop approach to test and validate multiscale dual H infinity filters for state-of-charge and capacity estimation," *IEEE Trans. Power Electron.*, vol. 33, no. 1, pp. 332–342, Jan. 2017.

[23] H. Rathmann, C. Weber, W. Benecke, and D. Kähler, "Sophisticated estimation of hardly measurable conditions of lithium-ion batteries," in *Proc. Ind. Electron. Conf. (IECON)*, Nov. 2013, pp. 1862–1866.

[24] R. Subramanian, P. Venhovens, and B. P. Keane, "Accelerated design and optimization of battery management systems using HIL simulation and rapid control prototyping," in *Proc. IEEE Int. Electr. Vehicle Conf. (IEVC)*, Mar. 2012, pp. 1–5.

[25] R. Morello *et al.*, "Hardware-in-the-loop simulation of FPGA-based state estimators for electric vehicle batteries," in *Proc. IEEE 25th Int. Symp. Ind. Electron. (ISIE)*, Jun. 2016, pp. 280–285.

[26] F. Baronti *et al.*, "Parameter identification of Li-Po batteries in electric vehicles: A comparative study," in *Proc. IEEE Int. Symp. Ind. Electron. (ISIE)*, May 2013, pp. 1–7.

[27] R. Morello *et al.*, "Comparison of state and parameter estimators for electric vehicle batteries," in *Proc. 41st Annu. Conf. IEEE Ind. Electron. Soc. (IECON)*, Nov. 2016, pp. 5433–5438.

[28] H. Dai, X. Zhang, X. Wei, Z. Sun, J. Wang, and F. Hu, "Cell-BMS validation with a hardware-in-the-loop simulation of lithium-ion battery cells for electric vehicles," *Int. J. Electr. Power Energy Syst.*, vol. 52, pp. 174–184, Nov. 2013.

[29] *The UNECE World Forum for Harmonization of Vehicle Regulations (WP.29)*. Accessed: Nov. 2, 2018. [Online]. Available: https://www.unece.org/trans/main/wp29/meeting_docs_wp29.html

[30] *United States Environmental Protection Agency (U.S. EPA)*. Accessed: Nov. 2, 2018. [Online]. Available: http://www3.epa.gov/

[31] *California Environmental Protection Agency, Air Resources Board*. Accessed: Nov. 2, 2018. [Online]. Available: http://www.arb.ca.gov/homepage.htm

[32] *United Nations Economic Commission for Europe (UNECE)*. Accessed: Nov. 2, 2018. [Online]. Available: http://www.unece.org/info/ece-homepage.html

[33] J. Xi, M. Li, and M. Xu, "Optimal energy management strategy for battery powered electric vehicles," *Appl. Energy*, vol. 134, pp. 332–341, Dec. 2014.

[34] F. Baronti, G. Fantechi, E. Leonardi, R. Roncella, and R. Saletti, "Enhanced model for Lithium-Polymer cells including temperature effects," in *Proc. 36th Annu. Conf. IEEE Ind. Electron. Soc. (IECON)*, Nov. 2010, pp. 2329–2333.

[35] T. Huria, M. Ceraolo, J. Gazzarri, and R. Jackey, "High fidelity electrical model with thermal dependence for characterization and simulation of high power lithium battery cells," in *Proc. IEEE Int. Electr. Veh. Conf.*, Mar. 2012, pp. 1–8.

[36] A. Thanheiser, T. P. Kohler, C. Bertram, and H.-G. Herzog, "Battery emulation considering thermal behavior," in *Proc. IEEE Vehicle Power Propuls. Conf.*, Sep. 2011, pp. 1–5.

[37] J. Du, Z. Chen, and F. Li, "Multi-objective optimization discharge method for heating lithium-ion battery at low temperatures," *IEEE Access*, vol. 6, pp. 44036–44049, 2018.

[38] K. Makinejad *et al.*, "A lumped electro-thermal model for Li-ion cells in electric vehicle application," in *Proc. 28th Int. Electr. Vehicle Symp. Exhibit. (EVS)*, 2015, pp. 1–13.

[39] University of Pisa. (2018). *EV Battery HiL*. [Online]. Available: https://github.com/batterylabunipi/EV-Battery-HiL

[40] H. Rahimi-Eichi, F. Baronti, and M.-Y. Chow, "Online adaptive parameter identification and state-of-charge coestimation for lithium-polymer battery cells," *IEEE Trans. Ind. Electron.*, vol. 61, no. 4, pp. 2053–2061, Apr. 2014.

[41] S. M. R. Islam, S.-Y. Park, and B. Balasingam, "Circuit parameters extraction algorithm for a lithium-ion battery charging system incorporated with electrochemical impedance spectroscopy," in *Proc. IEEE Appl. Power Electron. Conf. Expo. (APEC)*, Mar. 2018, pp. 3353–3358.

[42] F. Baronti, G. Fantechi, R. Roncella, and R. Saletti, "High-efficiency digitally controlled charge equalizer for series-connected cells based on switching converter and super-capacitor," *IEEE Trans. Ind. Informat.*, vol. 9, no. 2, pp. 1139–1147, May 2013.

[43] T. Dong, J. Li, and H. Dai, "Analysis on the influence of measurement precision of the battery management system on the state of charge estimation," in *Proc. Asia–Pacific Power Energy Eng. Conf. (APPEEC)*, Mar. 2010, pp. 1–5.

ROCCO MORELLO was born in Tricase, Italy, in 1988. He received the bachelor's and master's degrees in electronic engineering from the University of Pisa, Italy, in 2011 and 2015, respectively, where he is currently pursuing the Ph.D. degree.

His research interests are in the field of Li-ion battery management systems, especially on the development of battery state and parameter co-estimation algorithms and their implementation in embedded systems.

ROBERTO DI RIENZO was born in Avellino, Italy, in 1989. He received the M.Sc. degree in electronic engineering from the University of Pisa in 2014 and the Ph.D. degree in information engineering from the University of Pisa, Italy, in 2018.

He is currently a temporary Research Fellow with the Department of Information Engineering, University of Pisa. His research focuses on the Li-ion battery modeling and the design of electronic systems for the management of batteries for high- and medium-power applications.

ROBERTO RONCELLA (M'91) received the M.Sc. degree in electronic engineering and the Ph.D. degree from the University of Pisa, Pisa, Italy, in 1984 and 1989, respectively.

He was an Officer at the Italian Navy with technical functions. In 1990, he became a Researcher with the Department of Information Engineering, University of Pisa, where he is currently an Associate Professor with the Faculty of Engineering.

His main research interests are in the field of very-large-scale-integration integrated circuits and the design of high-performance digital and analog electronic circuits for astrophysics, automotive, and biomedical applications.

ROBERTO SALETTI (M'10–SM'12) received the Dr.Ing. degree (Hons.) in electronic engineering from the University of Pisa, Pisa, Italy, in 1981. From 1983 to 1992, he was with the National Research Council as a Research Scientist. In 1987, he was with the Cornell University, Ithaca, NY, USA, as a Visiting Scientist. In 1992, he became an Associate Professor of digital system electronics with the Faculty of Engineering, University of Pisa. In 2001, he was appointed as a Full Professor of electronics with the University of Pisa, where he is currently working. His main research activities are in the fields of design and test of high-performance electronic systems. The applications are integrated circuits for high-resolution delay-locked delay lines and data acquisition systems for two-wheel vehicles and luxury yachts. His most recent works are related to the electronics for monitoring and management of energy storage systems based on Li-ion batteries and the implementation of battery state estimation algorithms in battery management systems realized on affordable hardware platforms. He is a member of the IEEE Industrial Electronics Society and the Instrumentation and Measurements Society. He was one of the co-founders of the IEEE-IES Technical Committee on Energy Storage Devices and Systems.

FEDERICO BARONTI (M'08–SM'16) was born in Pisa, Italy, in 1975. He received the M.Sc. degree in electronic engineering and the Ph.D. degree from the University of Pisa, Italy, in 2001 and 2005, respectively.

He is currently an Associate Professor with the Dipartimento di Ingegneria dell'Informazione, University of Pisa. He works on the design of innovative systems aiming at improving the performance, safety, and comfort of road vehicles. More recent activities concern Li-ion battery modeling and the development of innovative battery management systems. He has co-authored around 100 publications on international journals and conference proceedings.

Dr. Baronti received the Best Paper Award of the *IEEE Industrial Electronics Magazine* in 2013. He is the past Chair of the IEEE-IES Technical Committee on Energy Storage and an Associate Editor of the IEEE Transactions on Industrial Informatics.