IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 1 July 2024, accepted 21 July 2024, date of publication 29 July 2024, date of current version 15 August 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3434716

RESEARCH ARTICLE

Behavioral Models for Lithium Batteries Based on Genetic Programming

GIULIA DI CAPUA[1], (Senior Member, IEEE), FRANCESCO PORPORA[1,2], (Member, IEEE), FILIPPO MILANO[1], (Member, IEEE), NUNZIO OLIVA[3], AND ANTONIO MAFFUCCI[1], (Senior Member, IEEE)

[1]DIEI, University of Cassino and Southern Lazio, 03043 Cassino, Italy
[2]E-LECTRA SRL, 03043 Cassino, Italy
[3]EXELING SRL, 83100 Avellino, Italy

Corresponding author: Francesco Porpora (francesco.porpora@unicas.it)

This work was supported by the IMPRESS project within the MOST—Sustainable Mobility Center and received funding from the European Union Next-GenerationEU (PIANO NAZIONALE DI RIPRESA E RESILIENZA (PNRR)—MISSIONE 4 COMPONENTE 2, INVESTIMENTO 1.4—D.D. 1033 17/06/2022, CN00000023). This manuscript reflects only the authors' views and opinions, neither the European Union nor the European Commission can be considered responsible for them.

ABSTRACT This paper proposes a novel methodology based on the Genetic Programming (GP) to derive behavioral models describing the transient evolution of the terminal voltage of a battery. These models analytically relate the battery voltage to its state of charge, charge/discharge rate, and temperature. Compared to the popular equivalent circuit-based models, one of the main advantages is the significant reduction of the effort to produce the experimental dataset required to identify the model parameters. The GP generates a family of optimal ''candidate'' analytical models, each associated with suitable metrics that quantify performance indicators like simplicity and accuracy. The methodology is applied to describe the transient discharge phase of a Lithium Iron Phosphate (LiFePO4 or LFP) battery under realistic operating conditions, considering the state-of-charge between 20% and 80%, discharge rates comprised between 0.25C and 1C, and temperature ranging from 5°C to 35°C. The GP provides different solutions that can be chosen by imposing the desired trade-off between accuracy and simplicity. Two models are selected and validated against experimental results. The chosen models guarantee a quite low level of the relative root mean square error (maximum 0.31% and 0.22%, respectively) over the range of analysis.

INDEX TERMS Li-ion batteries, behavioral modeling, genetic programming, multi-objective optimization.

I. INTRODUCTION

Batteries have become a key energy storage technology in many fields, including e-mobility applications [1], [2]. Lithium (Li-ion) batteries are especially characterized by a high investment cost compared to other energy storage technologies (e.g., lead acid or nickel metal hydride batteries). However, their success in the automotive application is related to high energy density, limited self-discharge, negligible hysteresis, long life cycle, and lower weight [3]. However, the battery's performance is highly influenced by current, temperature (T), Charge/discharge rate (C_{rate}), and State-of-Charge (SoC). Regardless of the chosen technology, the development of accurate and reliable models represents a fundamental key in all phases of a battery's life [4], [5], [6] in order to:

- drive and optimize its design;
- manage its nominal operating conditions through the Battery Management System (BMS);
- improve charging/discharging techniques;
- prevent unsafe operating conditions (e.g., overcharging or over-discharging, which can lead to serious damage or danger);
- reliably predict its long-term behavior (e.g., estimating end-of-life and implementing predictive maintenance in second-life battery applications).

Among all the others, the electro-thermal models play a special role since they can describe the behavior of the

The associate editor coordinating the review of this manuscript and approving it for publication was Giambattista Gruosso.

battery voltage versus SoC, C_{rate}, and T. The strict correlation among all battery parameters, such as the voltage, current, temperature, and SoC, leads to rather complex and multi-physical models [7].

Given its importance, battery modeling has been given a large and increasing amount of attention by the scientific community in the past decades, leading to a large variety of battery models available in the literature with different levels of complexity and accuracy. They can be grouped into three main categories, depending on the physical level of description.

On the one hand, the electrochemical models describe in detail the battery in its materials (conductors, electrolytes, etc.) and geometry (electrodes, package shapes, etc.), and simulate the actual chemical and electrical phenomena involved in it, possibly taking into account environmental conditions such as the temperature distribution [8], [9]. All the parameters of such models have a physical meaning as they are related to the geometry, the physical and chemical properties of the materials, and so on. The main drawbacks of these models are the need for deep knowledge of the device (to set the parameters) and the high computational cost required when complex systems have to be analyzed (such as battery packs). On the other hand, the behavioral models are built from the knowledge (usually given by an experimental characterization) of the input-output relations imposed by the battery. In this case, the battery is considered a ''black box'', and thus the model parameters do not have any physical meaning as they are set by the adopted strategy of the model identification, like the coefficients of a fitting function.

Behavioral models are based on data-driven methods [10], and can be analytical [11], stochastic [12], or based on Machine Learning (ML) [13]. Analytical models usually rely on a few equations to describe the overall battery properties, while stochastic models mainly refer to the Markov process, where one can predict the future of the process based on its present state but without knowing its full history. The major drawback of both of these models is the complexity of the approaches used to derive them. ML is a specific application of artificial intelligence that allows computers to learn from data and experience via algorithms and has recently emerged as a promising modeling approach for batteries [14], [15]. As a major drawback, ML-based models deeply depend on the availability of large time-consuming data sets.

In the scenario of the behavioral models, there is also a fourth category of models to which the popular Equivalent Circuit Models (ECMs) belong: the hybrid models. In this case, behavioral models are developed based on prearranged and fixed structures. Indeed, ECMs are associated with a given equivalent circuit, for instance, a cascade of branches of resistors and capacitors, with suitable voltage sources. The single circuit elements can be given a physical meaning, like in the electrochemical models, but their values are identified from an input-output characterization of the battery, like in the behavioral models. These models are very popular because of their capability to describe the electro-thermal behavior of the batteries with satisfactory accuracy while retaining a simple structure [16], [17]. Electrical models are available for all kinds of batteries, from lead-acid to Li-ion ones [18]. Depending on the adopted circuital structures, these models fall into three main classes: Thévenin-, impedance-, and runtime-based models, each one with some pros and cons. The Thévenin-based model is the simplest one: in its most basic form, it uses a series resistor and an RC parallel network to predict the battery response to transient load events at a particular SoC, by assuming the Open-Circuit Voltage VOC (SoC) as a constant [19]. Some enhanced Thévenin-based models assume variable capacitor instead of constant VOC (SoC) to represent nonlinear open-circuit voltage [20], or include additional nonlinear relationships to take into account current, temperature, and time dependency [21], or rely on SPICE-based sources implementing electrochemical equations [21]. An improved ECM model is presented in [23], where the identification of the model parameters is fastened by using a genetic algorithm and a least square optimization. However, all these models have some predicting capability limitations in steady-state and runtime operations.

Impedance-based models realize an AC-equivalent impedance model of the battery in the frequency domain using Electrochemical Impedance Spectroscopy (EIS) [24], [25]. Typically, the resulting ECM parameters are valid for a specific battery SoC and temperature and need to be recalibrated during battery operations. However, EIS measurements can be time-consuming, and recalibration is sometimes not an affordable solution, especially if the ECM includes many elements. Moreover, impedance-based models cannot predict the battery's DC response or runtime [18].

Finally, runtime-based models use a complex circuit network to simulate the battery runtime and the DC voltage response for a constant discharge current in SPICE-compatible simulators [26]. Unfortunately, this choice precludes reliable AC or transient analysis. Whatever its structure, when an ECM model is needed to perform the transient analysis of the battery, its calibration must be done through severely time-consuming techniques (like those summarized in Section II) that require a large number of experimental tests in different operating conditions, e.g., in terms of the amplitude of the charging/discharging current and temperature.

In this paper, we propose a new approach to identify suitable behavioral models that describe the transient evolution of the battery terminal voltage as a function of SoC, C_{rate}, and temperature (T) through analytical relations. The models are derived by using the Genetic Programming (GP) [27], which has been recently proposed in the technical literature to identify behavioral models in other realistic applications, such as for power devices and power conversion systems [28], [29], [30]. To the best of the Authors' knowledge, this kind of modeling approach has never been applied to batteries. The

case study analyzed here refers to Lithium-Iron-Phosphate (LiFePO4 or LFP) batteries and their discharge transient. The procedure, however, is general and can be applied to other kinds of batteries or operating conditions.

The paper is organized as follows. Section II provides details on the experimental characterization and the adopted dataset. Section III explains the main concepts of GP and its application to the said dataset, resulting in several possible behavioral analytical models. Based on their associated metrics, two GP-based models are selected and discussed in detail, as they exhibit opposite characteristics in terms of the ''accuracy'' and ''simplicity'' of the resulting analytical functions. Section IV presents the final applicability and reliability proof of the GP-based models selected in Section III, and offers a notable comparison of the performances achieved by the GP-based modeling approach and some popular ML algorithms. Finally, Section V draws conclusions and provides ideas for future works.

II. EXPERIMENTAL CHARACTERIZATION AND GENERATION OF THE MODELING DATASETS

This Section describes the experimental test procedures suitable to generate the datasets to be used for the identification and confirmation of the proposed models. The case study analyzed here refers to a 90-Ah LFP battery from EVE Energy Co., Ltd [31], and the final goal is to obtain an analytical model able to describe the behavior of its terminal voltage during a discharging phase as a function of SoC, C_{rate}, and T. First, a quick overview is given of the standard characterization procedure that would be requested to identify a zero or higher orders ECM. Then, the procedure to generate the datasets needed to identify the behavioral models by using the GP-based modeling approach is described in detail, along with a discussion of the obtained characterization results.

A. BATTERY CHARACTERIZATION PROCEDURE FOR THE IDENTIFICATION OF AN ECM

According to the IEC standards and the main testing manuals for battery systems [32], [33], [34], static and dynamic tests need to be conducted for a battery to correctly calibrate its ECM and thus reproduce its electro-thermal behavior. As for static testing, the multi-rate capacity characterization is usually adopted for evaluating the actual usable capacity of the battery under test and the related terminal voltage characteristics in different operating conditions of C_{rate} and T. In detail, once these conditions are set, multiple full-discharge tests are performed for each combination of C_{rate} and T starting from a fully charged battery. The multi-rate characterization is also called ''static'' because the discharging current for the battery under test during each complete discharge is fixed and constant throughout the entire test and does not involve any dynamic profile. Note that from a static characterization test, it would only be possible to identify a zero-order ECM, consisting of the battery's series resistance and open circuit voltage, the latter estimated through an additional discharge cycle performed at low C_{rate} values [17]. An example of the application of a single test within the static characterization procedure for the 90-Ah LFP battery under study is provided in Figure 1(a) and Figure 1(d), which show the current profile and the related battery voltage response for a single full discharge at a constant C_{rate} of 1C and a temperature of 25°C.

However, dynamic tests, such as the basic pulsed test or the widely adopted Hybrid Pulse Power Characterization (HPPC) [34], are needed for calibrating first or higher-order ECMs. These procedures are both based on the evaluation of the terminal voltage response during repetitive discharging and rest phases, which are alternatively operated at different SoC intervals. The basic pulsed test applied to the battery under study results in the current and voltage profiles shown in Figure 1(b) and Figure 1 (e), respectively. It starts with a fully charged (discharged) battery and initially discharges (charges) the battery at a fixed C_{rate} (here 1C) and temperature (here 25°C) until an SoC decrease (increase) equal to the desired SoC step (here 10%) is reached. Then an adequate rest time is performed for extinguishing the electrochemical transients in the battery, before starting with the next discharging (charging) phase. The HPPC procedure also starts with a fully charged (discharge) battery but relies on single discharge and charge current pulses of 10 seconds, each one followed by a short relaxation time. The resulting current and voltage profiles shown in Figure 1(c) and Figure 1(f) refer to the same C_{rate}, SoC step and T as in the basic pulsed test. Note that all the testing times for the different procedures shown in Figure 1 are expressed in arbitrary units (a.u.) due to their strong variability with the operating conditions in terms of C_{rate} and relaxation times. Due to the need for relaxation times between two consecutive discharging SoC steps, the dynamic tests require a huge amount of time for a specific operating condition only (1C, 25°C). Therefore, considering that the generation of a suitable dataset would require testing a significant number of conditions in (C_{rate}, T), it turns out that pulsed characterizations are time-consuming. For example, assuming a rest time of two hours between two consecutive discharges of 10% SoC, the basic pulse test requires an overall time of 20 hours plus the time needed for completely discharging the battery at the desired C_{rate}. On the other hand, considering the typical HPPC procedure illustrated in [34], a common testing time of about 23 hours is requested regardless of the desired discharging C_{rate}. Conversely, a single static test only requires the time needed for completely discharging the battery at the desired C_{rate}. It is important to highlight that all these testing times refer to a single C_{rate} and temperature condition for the battery under test.

B. BATTERY CHARACTERIZATION PROCEDURE FOR THE IDENTIFICATION OF THE PROPOSED BEHAVIORAL MODEL

The dataset needed to identify the proposed behavioral model can be generated by using only the static procedures described above, therefore with a huge reduction of the experimental characterization effort. Specifically, a static multi-rate test procedure is adopted, and herein discussed,

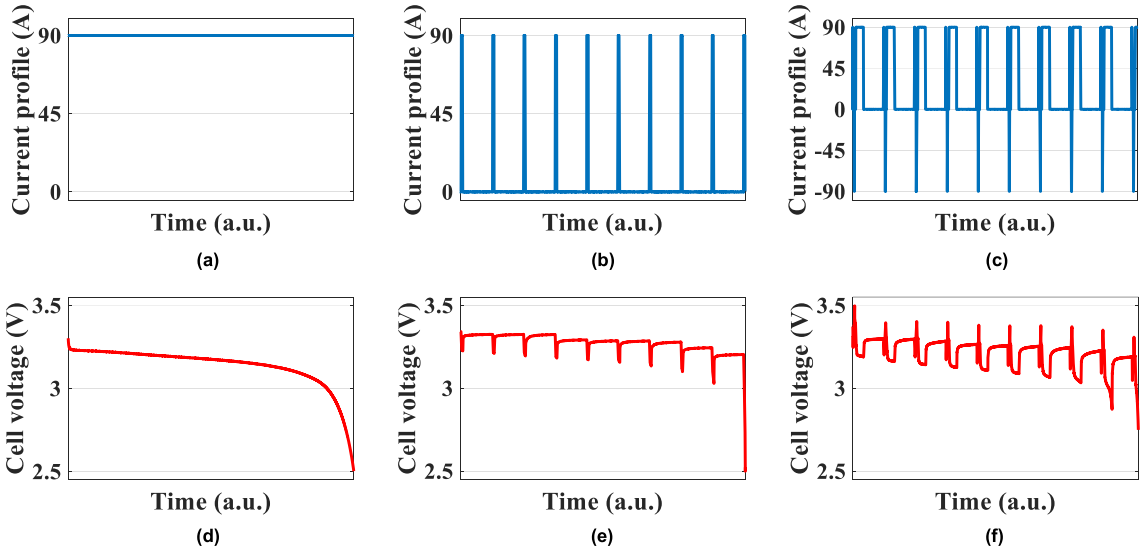


FIGURE 1. Current profiles for (a) static, (b) basic pulsed and (c) HPPC test procedures, with the related voltage responses (d, e, f).

to perform the characterization in the discharging mode of the 90-Ah LFP battery, whose main specifications are summarized in Table 1.

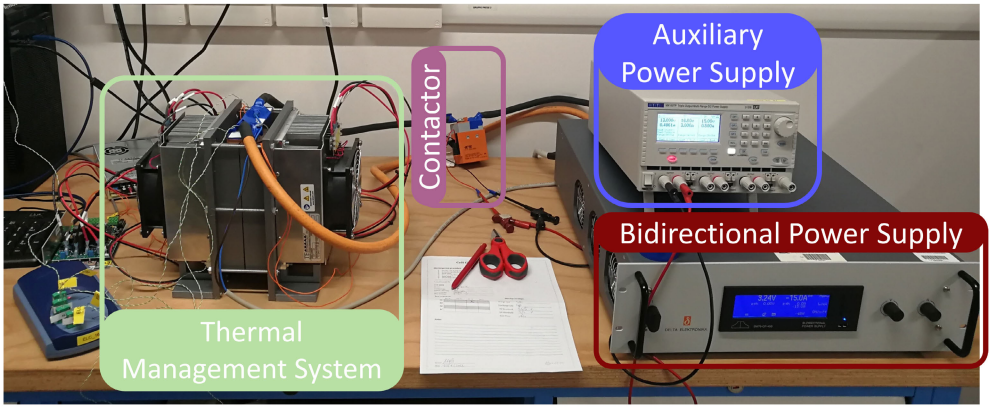
As previously said, the battery terminal voltage V_B depends on the given values of SoC, C_{rate}, and T. Accordingly, reference value ranges for these quantities must be selected to build a consistent modeling dataset.

As for the SoC, this quantity value is calculated using the Coulomb counting method, which integrates the discharging (charging) current over time [35]. A discretization procedure is consequently required to limit the number of SoC values comprised in a given reference range. Additionally, it is worth highlighting that a discharging–charging rate comprised between 20% and 80% of SoC represents the SoC area within which a Li-ion battery should operate [36]. In fact, batteries operating between 20% and 80% of SoC present an excellent cycling performance with essentially reduced capacity degradation [37], [38]. As a result, we take into account only SoC values with steps of 2.5 % in the range from 20 % to 80 % (namely, 25 SoC values).

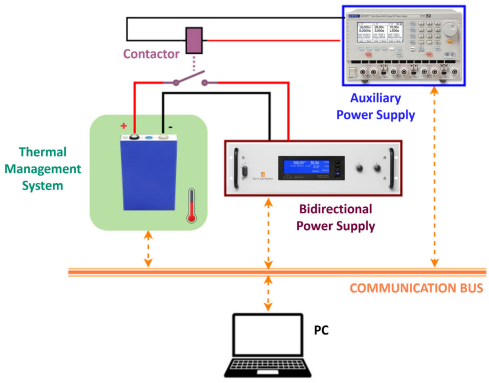
As for the C_{rate}, we consider four values: 0.25C, 0.33C, 0.5C, and 1C. Note that, although LFP batteries usually allow charging and discharging operations at higher C_{rate}, the ones chosen for this paper are limited to a maximum continuous discharging current of 1C, as declared by the manufacturer.

As for the temperature, we consider the values: 5°C, 15°C, 25°C and 35°C. Note that, to validate the proposed modeling approach in a temperature range as close as possible to common real-world applications, the dataset was specifically tailored between 5°C and 35°C. Indeed, temperatures outside this range significantly contribute to accelerate degradation phenomena, potentially impacting the long-term safety and operation of the battery.

A picture and a schematic representation of the experimental setup implemented and adopted for performing the static multi-rate test procedure are respectively shown in Figure 2(a) and Figure 2(b). The SM70-CP-450 Delta Elektronika bidirectional power supply is adopted for performing the static characterization, which integrates both the power supply for charging and the electronic load for discharging in a single unit, also resulting in less space-consuming and control complexity for the experimental setup. Moreover, considering its voltage and current resolutions, this bidirectional power supply also operates as a measurement instrument. In detail, the sensing features of the bidirectional power supply are enabled and the related sensing wires are directly connected to the battery terminals, to correctly measure the battery voltage during the experimental test and compensate for the voltage drop on the power lines due to the flow of the charging/discharging current. A normally-open contactor (model EVC250 by TE Connectivity), connected to the positive electrical interconnection between the bidirectional power supply and the battery, is used to ensure its complete disconnection during the rest time between two consecutive charging and discharging operations, thus resulting in a correct open circuit condition for the battery under test. The opening and closing of this contactor are controlled by the dedicated MX100TP Aim-TTi power supply. Moreover, a thermal management system (TMS) based on two CP-110 Peltier junction devices from TE Technology is adopted within the experimental setup, as shown in Figure 2(c), to maintain the temperature of the battery surface at the fixed and desired value during each test within the static characterization. A custom-made software is implemented for managing remotely both the bidirectional and the external power supply, thus properly reproducing the specific current profile while minimizing the acquisition time, as well as setting the temperature reference for the TMS. According to the experimental setup just described, the static multi-rate test procedure is operated to achieve the characteristic curves of the battery terminal voltage in discharging mode for all the combinations of C_{rate} (4 values) and T (4 values) previously



(a)



(b)

(c)

FIGURE 2. (a) Picture of the experimental setup implemented for battery characterization purposes. (b) Schematic representation of the experimental setup. (c) Detail of the disassembled thermal management system, based on two Peltier junction devices, with the battery under test.

TABLE 1. Specifications of the 90-Ah LFP battery from EVE energy [31].

Battery features	Values
Chemistry	Lithium Iron Phosphate (LFP)
Nominal capacity	90 Ah
Nominal voltage	3.2 V
Minimum voltage	2.5 V
Maximum voltage	3.65 V
Maximum charging/discharging continuous C_{rate}	1C
Maximum operating temperature	0°C – 55°C (Charging) -20°C – 55°C (Discharging)

illustrated, resulting in an overall number of 16 discharging characteristic curves. In detail, the operations illustrated in Table 2 are executed sequentially for each test.

Then, given all the 16 experimental battery terminal voltage characteristics collected during the characterization, the SoC discretization approach previously described is carried out to achieve the final modeling dataset, comprised of 400 overall operating conditions, namely combinations of the SoC (25 values), C_{rate} (4 values) and T (4 values) considered for the static multi-rate test procedure. All the experimental settings adopted for the GP-based modeling approach are summarized in Table 3.

Finally, the standard measurement uncertainty was estimated, based on the bidirectional power supply specifications. In particular, concerning the operating conditions defined and reported in Table 3, the expanded measurement uncertainty was calculated with a confidence level of about 98%, being the measured voltage associated with a uniform probability distribution. Accordingly, the measured battery voltage can be represented by a coverage interval with a confidence interval of about 98%, as shown later in Section IV (see Figure 9).

III. BEHAVIORAL MODELING BASED ON MULTI-OBJECTIVE GENETIC PROGRAMMING

This Section discusses the GP-based approach proposed to identify optimal behavioral models for the battery terminal voltage, given its SoC, $C_{rate,}$ and T values. Firstly, a brief introduction to the GP is provided, along with a description of the specific goal for the given case study and a reference to the multi-objective optimization approach adopted to improve the performance of the resulting GP solutions. Then, after some preliminary considerations on the reference datasets, optimal GP-based behavioral models are identified, based on absolute error metrics (e.g., RMS, mean, standard deviation, and maximum value of the absolute error values).

TABLE 2. Sequence of operations performed for each test within the static characterization.

#	Operation	Description
1	Definition of the test conditions	Specific values for the discharging C_{rate} and the surface temperature T are defined among those included in the static multi-rate test procedure.
2	Thermal conditioning of the battery	The battery is displaced within the TMS and conditioned until it reaches thermal equilibrium at the fixed surface temperature selected.
3	Initial charging phase	The battery is initially charged with a Constant-Current/Constant-Voltage (CCCV) profile at 0.1C, a value sufficiently low for ensuring its complete charging with a minimal perturbation to the electro-thermal behavior of the battery under test.
4	Rest time	The battery is completely disconnected from the bidirectional power supply and a rest time of 2 hours, suitable for extinguishing the electrochemical transients, is operated.
5	Discharging phase	A full discharge phase until reaching the minimum battery voltage (2.5 V) is performed at the specific discharging C_{rate} and surface temperature T previously defined.

TABLE 3. Adopted settings for GP-based modeling.

Settings	Conditions
4 C_{rate} values	0.25 C, 0.33 C, 0.5 C, 1 C
4 Temperature values	5 °C, 15 °C, 25 °C, 35 °C
25 SoC values	Minimum 20%, maximum 80%, steps of 2.5%

A. MULTI-OBJECTIVE GENETIC PROGRAMMING: SETUP PARAMETERS AND OBJECTIVES FUNCTIONS

The GP is an evolutionary algorithm where the population is composed of models. Each model is a mathematical function (symbolized by "f") that can be represented as a "tree": its internal nodes encode algebraic *operators* ($+$, $-$, $\times$, $\div$) or *basic analytic functions* (e.g., power, sine, exponential, logarithm, etc.), while its external nodes, also referred to as *leaves*, simply encode inputs or constants [27]. During its evolution, the GP applies the Darwinian principle of survival of the fittest, by applying classical genetic operations such as selection, cross-over, mutation, and elitism, to create a new offspring population from the current population of models. The effectiveness of these genetic operations has been extensively discussed in the literature [39], [40], and depends on the occurrence percentages of each of them. The typical flowchart of the GP is shown in [41], where the representation of a numeric expression using a tree structure is also provided.

Many GP software tools have been developed over the last thirty years [42]. Among them, several open-source tools are coded in MATLAB®, like GPTIPS2 [43] or GPLAB [43], mainly intended for symbolic regression problems, or CGP4MATLAB [44], specially developed for signal processing and image processing problems. Other popular GP tools are developed in Python, like GPLEARN [45] and PyGEP [46], or in Java, like GEP4J [47]. All these open-source tools are quite versatile and easily extendable, given a minimum knowledge of MATLAB, Python, or Java programming environments. Depending on the nature of the problem, many setup parameters can be configured and optimized differently. Additional multi-objective optimization tasks could be required, which are usually not implemented in basic GP codes. Accordingly, starting from such open-source GP tools, it is quite common to customize a GP code for a given application, with respect to multiple goals or objective functions. This is the case of this work, where GPTIPS2 is adopted as the reference GP software tool, while a dedicated multi-objective optimization infrastructure is built on top of it.

The goal of the proposed GP-based approach is to identify a behavioral ("*bhv*") analytical formula expressing the battery terminal voltage $V_{B,bhv}$ as a mathematical function f of SoC, $C_{rate,}$ and temperature, according to (1):

$$V_{B,bhv} = f\,[\mathrm{SoC}\ ,\ \ C_{rate}\ ,\ u_1,\ \ \ldots,\ u_n] \tag{1}$$

where $\mathbf{u} = [u_1, u_2, \ldots, u_n]$ is a vector of numerical coefficients (determined using nonlinear least square methods, as discussed in Appendix Section B), each one in turn expressed as a function of the temperature T (namely, as $\mathbf{u}(T) = [u_1(T), u_2(T), \ldots, u_n(T)]$). The number of these numerical coefficients can vary for different functions f. Generally speaking, the GP allows "discovering" a simple and accurate function f, such that the $V_{B,bhv}$ value returned by (1) is as close as possible to the true battery terminal voltage value, for each condition of the training dataset. The main concept embedded in the canonic expression (1) is that the SoC and the C_{rate} have a major effect on the $V_{B,bhv}$ value so that they appear as explicit variables in f. Conversely, the temperature T has a minor effect on V_B, so that it simply influences the $\mathbf{u}$ coefficients of the function f. In other words, the SoC and the C_{rate} are adopted as *primary variables,* and the temperature as a *secondary variable*.

For the case study under discussion, each GP individual represents a single-gene model, where a *bias* coefficient and a *scaling* coefficient can be forced as additive and multiplicative coefficients by the end-user when launching the GP, or can be let evolve freely during the GP generations. Additionally, the GP was configured based on the following assumptions:

- the best 1% of the GP population is subject to elitism and is directly copied into the next generation;

- the k-Selection Tournament with $k = 4$ is adopted, so that 4 individuals are randomly selected from the entire population: they compete against each other, and the individual with the highest fitness wins and is chosen as one of the two next-generation parents;
- the remaining part of the population not subject to elitism is obtained through crossover and mutation operations, which occur with 80% and 20% probability, respectively.

Three objective functions are identified and adopted for the behavioral modeling of this case study: the error between the outputs of the model and the training dataset, the complexity of the resultant model f, and the monotonicity of its **u** coefficients to temperature values. Accordingly, a multi-objective optimization problem is set up [48], whose fitness function F_{FIT} can be expressed as the weighted sum given in (2):

$$F_{FIT} = \alpha\ F_{rms} + \beta\ F_{cmp} + \gamma\ F_{mnt} \tag{2}$$

where α, β, and γ are weighting coefficients and F_{rms}, $F_{cmp,}$ and F_{mnt} are the objective functions for the RMS error, the model complexity, and the coefficients monotonicity, respectively. The weighting coefficients − whose value influences the presence of each GP model in the Pareto front [48] − are chosen so that $\alpha + \beta + \gamma = 1$. The objective functions F_{rms}, $F_{cmp,}$ and F_{mnt}, described in more detail in Appendix Section B, are given in the range [0, 1].

B. GENETIC PROGRAMMING: DATASET, METRICS, AND RESULTS

The 90-Ah LPF battery dataset described in Section II was adopted for the GP execution. Similarly to all algorithms based on data-driven predictions or decisions, these input data are used to build the model and are usually divided into multiple data sets. In particular, three datasets are commonly used in different stages of the model definition: training, validation, and test datasets. The training dataset consists of the data samples used to identify the model *during* the GP evolution. Then, the validation dataset is used to provide an unbiased assessment of how well a model fits the training data set, while still tuning the model's hyper-parameters. Finally, the test dataset consists of data samples used to provide an unbiased evaluation of the final discovered model, identified among all the models discovered thanks to the training dataset. For this battery case study, a total of 400 operating conditions, given as a combination of the SoC, $C_{rate,}$ and temperature values listed in Table 3, were considered as GP inputs. A set of 40 conditions (10% of the dataset) was randomly selected to create a test dataset $\mathbf{T_{ts}}$, common to all GP runs. Then, for each run, the remaining 360 conditions (90% of the dataset) were randomly divided into 288 conditions for the training dataset $\mathbf{T_{tr}}$ (80% of the remaining dataset) and 72 conditions for the validation dataset $\mathbf{T_{vs}}$ (20% of the remaining dataset).

Different fitness weighting coefficient setups were chosen (see Table 4), to explore solutions with the prevalence of different objective functions. Among the α, β, and γ coefficients, a higher weight was assigned to the RMS error objective function F_{rms}, to guarantee the occurrence of more individuals (models) with low RMS error. This choice is taken based on the preliminary analysis of the experimental data, evidencing an inherent smooth behavior of the battery voltage with respect to SoC, $C_{rate,}$ and T, which fairly yields simple functions f and monotonic **u** coefficients. Furthermore, the GP was set up with different combinations of bias and scaling coefficients, as shown in Table 4.

TABLE 4. Fitness weighting coefficients and bias/scaling coefficients.

α	β	γ	bias coefficient	scaling coefficient
0.9	0.05	0.05	forced	forced
0.85	0.075	0.075	forced	forced
0.8	0.1	0.1	forced	forced
0.9	0.05	0.05	free	free
0.85	0.075	0.075	free	free
0.8	0.1	0.1	free	free
0.33	0.33	0.34	forced	forced
0.75	0.2	0.05	forced	forced
0.5	0.3	0.2	forced	forced

The GP was executed over 50 runs per each setup included in Table 4, with a population of 100 models, evolving over 50 generations for each run. After every 10 generations, the RMS error of the best model on both the $\mathbf{T_{tr}}$ and $\mathbf{T_{vs}}$ datasets was evaluated. If the RMS error on the $\mathbf{T_{vs}}$ dataset is significantly worse than that on the $\mathbf{T_{tr}}$ dataset, then overfitting would be detected. In our case, the RMS errors on the $\mathbf{T_{vs}}$ and $\mathbf{T_{tr}}$ datasets were almost similar across generations, indicating that no overfitting was observed. At the end of each run, all GP models were pre-selected based on RMS error achieved on the $\mathbf{T_{tr}}$ dataset. An RMS error lower than 10 mV was imposed, which aligns with the level of accuracy required for modeling this type of battery and the measurement uncertainties associated with the adopted characterization method [50], as discussed in detail later (see Fig. 9).

The GP ended all the runs and compiled a list of suitable solutions (*i.e.*, analytical functions), which are all potential candidates for the problem we are investigating, despite their different values of fitness. Among these solutions, all non-dominated models in the (F_{cmp}, F_{rms}, F_{mnt}) domains were considered, then sorted based on their F_{FIT} values and labeled with increasing numbers (in our case, from #1 up to #16). These models are all listed in Table 5, along with their values of F_{FIT}, F_{cmp}, $F_{rms,}$ and F_{mnt}. It is worth noting that all **u** coefficients are purely numerical at this stage and were evaluated using a nonlinear least square optimization method, as discussed in Appendix, Section B.

For all these GP-based models, Table 6 summarizes the mean (μ_{err}), standard deviation (σ_{err}), maximum (err_{max}), and RMS values of the absolute error over both the training

TABLE 5. Expressions of best models obtained from GP.

Models	Expressions	F_{FIT}	F_{rms}	F_{cmp}	F_{mnt}	# coeff.
#1	$u_1 + u_2\left[\frac{1}{SoC}\sqrt{C_{rate}} + \sqrt{atan(C_{rate})}\right]$	0.0068	0.0030	0.0560	~0	2
#2	$u_1 + u_2\left[\frac{1}{SoC}\sqrt{C_{rate}} + \sqrt{\sqrt{C_{rate}}}\right]$	0.0082	0.0029	0.0761	~0	2
#3	$u_1 + u_2[u_3 + SoC + atan(C_{rate})]\frac{1}{SoC}$	0.0148	0.0028	0.0342	0.2099	3
#4	$\frac{1}{SoC}(u_1 C_{rate} + u_2) + u_3 C_{rate} + u_4$	0.0203	0.0026	0.1015	0.2583	4
#5	$SoC^2(u_1 SoC + u_2 C_{rate} + u_3) + C_{rate}(u_4 SoC + u_5 C_{rate} + u_6) + u_7 SoC + u_8 C_{rate} + u_9$	0.0207	0.0013	0.3890	0.0008	9
#6	$\frac{1}{SoC}(u_1 C_{rate} + u_2) + C_{rate}(u_3 C_{rate} + u_4) + u_5 C_{rate} + u_6$	0.0259	0.0023	0.2216	0.2559	6
#7	$u_1 + u_2\frac{1}{SoC}[atan(C_{rate}) + u_3]$	0.0304	0.0029	0.0227	0.5339	3
#8	$SoC(u_1 C_{rate} + u_2) + C_{rate}(u_3 C_{rate} + u_4) + \frac{u_5}{SoC} + u_6 C_{rate} + u_7$	0.0308	0.0028	0.2858	~0	7
#9	$\frac{1}{SoC}(-u_1 C_{rate} + u_2) + \frac{1}{C_{rate}}(-u_3 C_{rate} + u_4) - u_5 C_{rate} + u_6$	0.0311	0.0023	0.2768	0.1112	6
#10	$\frac{1}{SoC}(u_1 C_{rate} + u_2) + u_3 C_{rate} + C_{rate} + u_4$	0.0319	0.0025	0.1120	0.2845	4
#11	$\frac{1}{SoC}(u_1 C_{rate} + u_2) + C_{rate}(u_3 C_{rate} + u_4) + u_5 SoC + u_6 C_{rate} + u_7$	0.0331	0.0016	0.2733	0.1501	7
#12	$SoC\ C_{rate}(u_1 SoC + u_2) + (u_3 SoC + u_4)atan(C_{rate}) + C_{rate}(u_5 SoC + u_6) + u_7 SoC + u_8$	0.0376	0.0016	0.3632	~0	8
#13	$C_{rate}[SoC\ (u_1 SoC + u_2) + u_3 SoC + u_4] + SoC(u_5 SoC + u_6) + \frac{1}{SoC}u_7 + SoC + u_8$	0.0386	0.0014	0.3745	0.0003	8
#14	$ln(SoC)(u_1 SoC + u_2 C_{rate} + u_3) + u_4 SoC + u_5 C_{rate} + Soc + u_6$	0.0416	0.0022	0.2236	0.1749	6
#15	$C_{rate}[u_1 SoC + u_2 C_{rate} + u_3] + \frac{1}{SoC}(u_4 C_{rate} + u_5) + u_6 SoC + u_7 C_{rate} + u_8$	0.0428	0.0015	0.3291	0.4998	8
#16	$C_{rate}\left(u_1\frac{1}{SoC} + u_2 C_{rate} + u_3\right) + \frac{1}{SoC}(u_4 + C_{rate}) + u_5 + C_{rate}$	0.0475	0.0025	0.2178	0.3868	5

dataset $\mathbf{T_{tr}}$ and the test dataset $\mathbf{T_{ts}}$. For each column, a color scale from blue to red is adopted in Table 6 to highlight the minimum and maximum values of each quantity, respectively. All these models have a maximum RMS value of the absolute error lower than 9.5 mV over the training dataset and 9.8 mV over the test dataset. Among them, some models can be preferred to others. For example, models #1 and #2 are characterized by the lowest value of F_{FIT} (namely, the best trade-off in terms of the three objective functions) and have the clear advantage of relying on the minimum number of **u** coefficients (only two numerical coefficients, u_1 and u_2). Both are performing better on the test dataset (data not explored by the GP during its evolution), where the minimum err_{max} value is achieved. Conversely, models with a number of **u** coefficients greater than five show very good metrics on the training dataset (e.g., models #5, #11, #12, #13, #14, and #16) and an acceptable compromise on the test dataset. Model #15 (with eight numerical coefficients) instead shows excellent performance on the training dataset against the highest values of σ_{err}, err_{max} and RMS of the entire test dataset. A good compromise is achieved with model #5, with an average fitness value FFIT and quite low values of μerr, σerr and RMS.

Therefore, in the following, we consider the comparison between:

- model #1 (*i.e.*, the simplest model, with the minimum number of coefficients), with the best fitness value F_{FIT}, the highest RMS value ($F_{rms} = 0.003$) but a quite low complexity value ($F_{cmp} = 0.056$), and a negligible monotonicity factor ($F_{mon} = 0$);
- model #5 (*i.e.*, the most complex model, with the maximum number of coefficients), with a higher fitness value F_{FIT}, the lowest RMS value ($F_{rms} = 0.0013$) but the highest complexity value ($F_{cmp} = 0.389$), and a quite low level of monotonicity factor ($F_{mon} = 0.0008$).

It is worth noting that the lower the monotonicity factor, the higher the probability of easily expressing the **u** coefficients as a function of the temperature T. For example, models #6 and #16 could both appear as good compromising solutions, with a reduced number of coefficients with respect to model #5 and better error metrics with respect to model #1. However, models #6 and #16 have a quite high level of monotonicity factor, which can compromise the real chance of representing the **u** coefficients as a temperature-dependent function (as clarified in Section IV).

Figures 3 and 4 show the absolute error $E_a = V_{B,exp} - V_{B,bhv}$ between the experimental terminal battery voltage given into the GP dataset and the corresponding voltage predicted by using the GP models #1 and #5, respectively. In this analysis, we consider all the experimental samples adopted in both the training and validation datasets (circle markers) and the test dataset (star markers). Overall, these results confirm the good level of accuracy of behavioral model #1, especially for intermediate values of the SoC and lower values of C_{rate}. The highest error values are achieved at

TABLE 6. Metrics values for selected GP models.

Model	Training dataset T_{tr}				Testing dataset T_{ts}			
	μ_{err} (mV)	σ_{err} (mV)	err_{max} (mV)	RMS (mV)	μ_{err} (mV)	σ_{err} (mV)	err_{max} (mV)	RMS (mV)
#1	7.0	6.4	43.5	9.5	6.3	5.3	21.2	8.2
#2	7.0	5.9	35.2	9.2	6.7	5.7	21.8	8.9
#3	6.5	5.9	37.1	8.8	6.1	7.2	36.3	9.5
#4	5.9	5.7	45.4	8.2	6.4	6.2	31.9	8.9
#5	2.8	3.8	38.9	4.7	3.8	5.0	25.4	6.3
#6	5.5	5.0	37.0	7.4	5.1	5.3	29.3	7.3
#7	6.5	5.9	36.1	8.8	6.0	7.3	35.2	9.5
#8	5.9	6.5	58.9	8.8	6.3	6.3	29.1	8.9
#9	5.6	4.9	34.4	7.4	5.2	5.7	32.3	7.7
#10	6.0	5.5	39.7	8.1	6.7	6.5	27.4	9.3
#11	3.9	3.3	19.3	5.1	4.8	7.6	48.0	9.0
#12	3.3	4.3	31.8	5.4	5.6	6.5	26.7	8.6
#13	3.4	3.8	26.4	5.1	5.2	7.6	42.7	9.2
#14	4.7	5.0	27.4	6.8	5.9	6.3	29.5	8.6
#15	3.5	3.0	19.6	4.6	4.9	8.5	54.2	9.8
#16	5.6	4.7	28.7	7.3	5.6	5.8	30.9	8.1

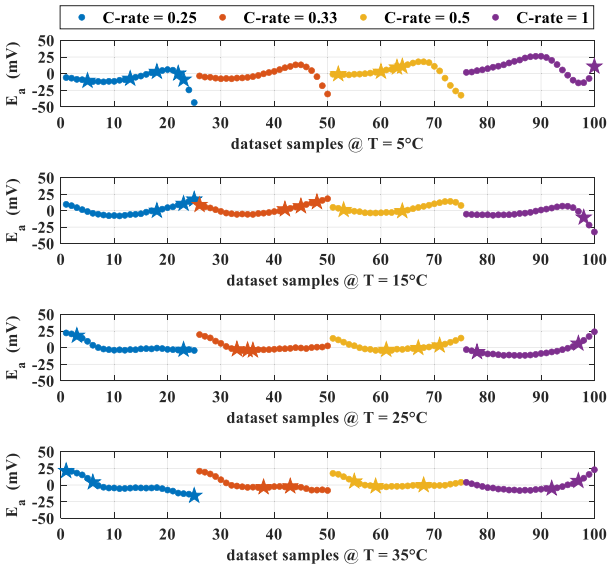


FIGURE 3. Absolute error between the experimental dataset and the battery terminal voltage predicted by using the GP model #1: circle markers = training dataset samples, star markers = test dataset samples. For each temperature value and C_{rate} subset (C_{rate} =0.25, C_{rate} =0.33, C_{rate} =0.50, C_{rate} =1), data samples are sorted from SoC = 80% to SoC = 20%.

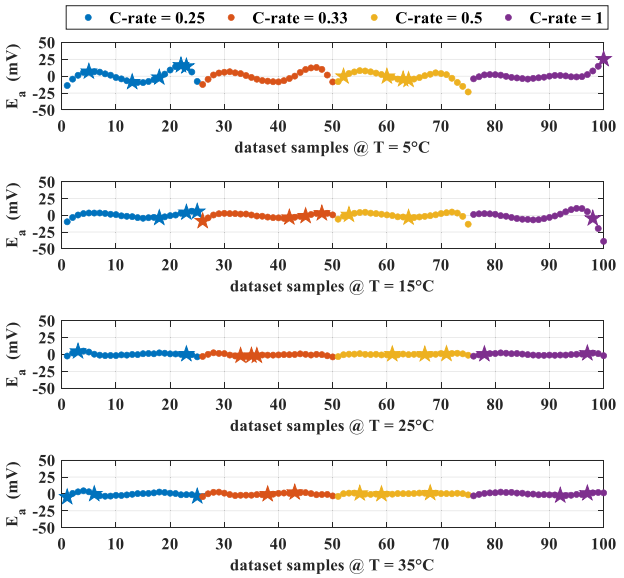


FIGURE 4. Absolute error between the experimental dataset and the battery terminal voltage predicted by using the GP model #5: circle markers = training dataset samples, star markers = test dataset samples. For each temperature value and C_{rate} subset (C_{rate} =0.25, C_{rate} =0.33, C_{rate} =0.50, C_{rate} =1), data samples are sorted from SoC = 80% to SoC = 20%.

lower values of the SoC (e.g., model #1 realizes the highest error err_{max} = 43.5 mV at C_{rate} = 0.25, SoC = 20%, and T = 5°C). Conversely, the reliability and good accuracy of behavioral model #5 are verified for almost all values of SoC, C_{rate}, and temperature (e.g., at C_{rate} = 0.25, SoC = 20%, and T = 5°C, the behavioral model #5 realizes the highest error err_{max} = 38.9 mV at C_{rate} = 1, SoC = 20%, and T = 15°C). To summarize, model #1 and model #5 respectively realize a maximum relative RMS error of 0.31% and 0.22% over the training and test datasets, thus both ensuring a good level of accuracy [51].

IV. FINAL RELIABILITY PROOF AND COMPARISONS

A. GP-BASED MODELING RELIABILITY

This Section aims to confirm the applicability and reliability of the proposed GP-based models in the whole considered range of investigation for the SoC, C_{rate}, and T parameters. For each given value of temperature T = {5, 15, 25, 35} °C, the numerical coefficients **u** were determined by the GP using a nonlinear least squares optimization method (see Appendix, Section B). This process allows us to obtain the $V_{B,bhv}$ versus

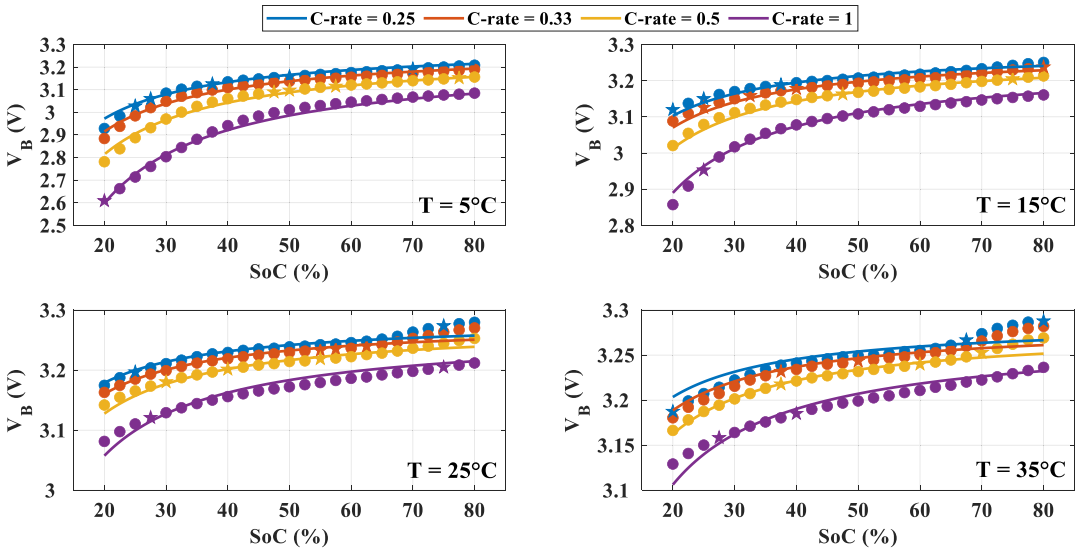


FIGURE 5. Battery terminal voltage experimental values (circle markers = training dataset samples, star markers = test dataset samples), and results based on model #1 with constant coefficients u (continuous lines).

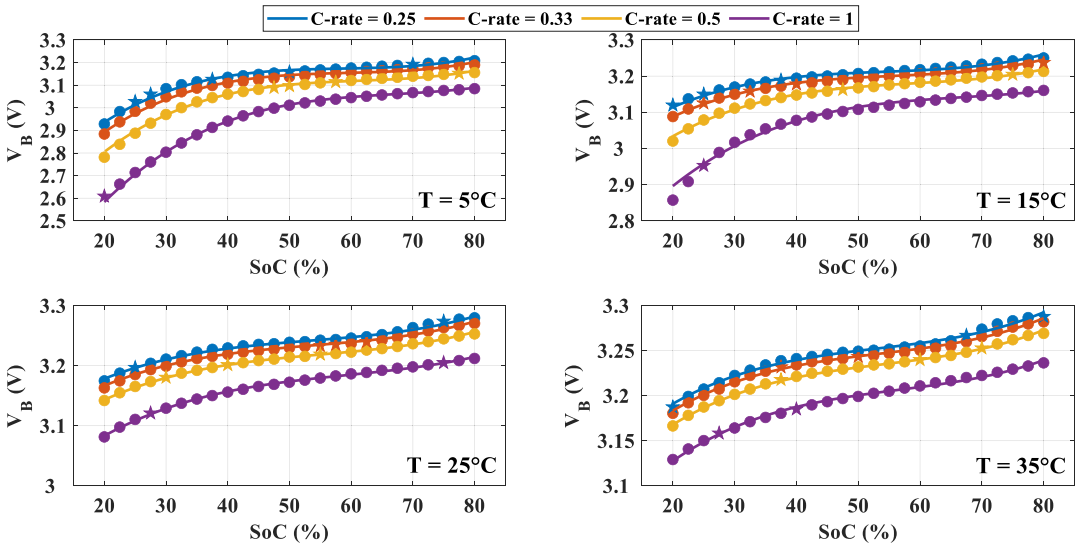


FIGURE 6. Battery terminal voltage experimental values (circle markers = training dataset samples, star markers = test dataset samples), and results based on model #5 with constant coefficients u (continuous lines).

SoC characteristics for the given values of C_{rate} and SoC. In fact, in the previous analysis, only 25 SoC values for each C_{rate} value were considered, ranging from SoC = 20% up to SoC = 80% with steps of 2.5% (see Figures 3 and 4).

Figures 5 and 6 show the fitting of the battery terminal voltage (continuous lines) for any SoC values ranging from 20% to 80%, as modeled by models #1 and #5. These are compared against all experimental samples from the training and validation datasets (circle markers) and test dataset (star markers). Both models accurately predict the battery terminal voltage trend for any SoC value within the original dataset. Specifically, model #1 effectively predicts the battery terminal voltage behavior for intermediate SoC values, although it shows a larger error at the lowest SoC values (the voltage scales in Figure 5 have been adjusted for clarity). In contrast, model #5 accurately follows the trend of experimental data samples across various SoC, Crate, and temperature conditions.

As an additional reliability proof of the proposed approach, the GP-based models were also verified for C_{rate} and T values included in the original range of definition, but not in the adopted GP dataset. To this end, the trend of coefficients **u** with respect to T was determined. The plots in Figure 7 show the coefficients u_1 and u_2 for T = {5, 15, 25, 35}°C (red dots), and the coefficient trend u_1(T) and u_2(T) for model #1. This trend can be expressed as in (3)

$$u_i(\mathrm{T}) = \mathrm{c_{i1}}\ \mathrm{T}^3 + \mathrm{c_{i2}}\ \mathrm{T}^2 + \mathrm{c_{i3}}\ \mathrm{T} + \mathrm{c_{i4}} \tag{3}$$

for $i = \{1, 2\}$ and coefficients $c_{i1}, \ldots, c_{i4}$ listed in Table 7.

Similarly, the plots in Figure 8 show the coefficient values $u_1, \ldots, u_9$ for T = {5, 15, 25, 35}°C (red dots), and the coefficient trend u_1(T), …, u_9(T) for model #5. Again, this coefficient trend can be expressed as in (3) for $i = \{1, \ldots, 9\}$, given the numeric coefficients $c_{i1}, \ldots, c_{i4}$ listed in Table 8. Accordingly, model #1 and model #5 with their coefficients

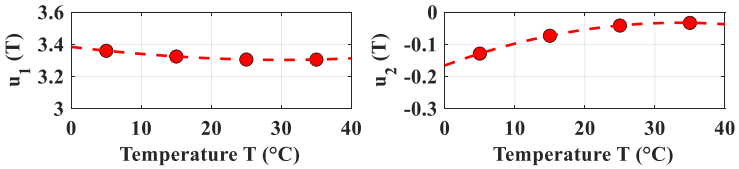


FIGURE 7. Coefficients values u_1 and u_2 for T = {5, 15, 25, 35}°C (red dots), and coefficients trend $u_1(T)$ and $u_2(T)$ for the GP model #1.

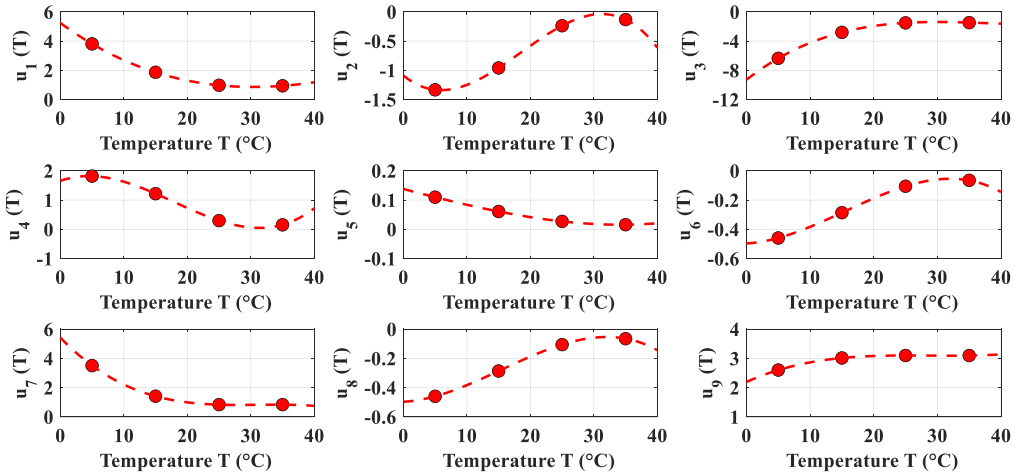


FIGURE 8. Coefficient values $u_1, \ldots, u_9$ for T = {5, 15, 25, 35}°C (red dots), and coefficient trend $u_1(T), \ldots, u_9(T)$ for the GP model #5.

trend (3) can be used to obtain the battery terminal voltage for any SoC values from 20% to 80%, C_{rate} values from 0.25 to 1, and temperature values from 5°C to 35°C.

Note that some of the previous metrics given in Table 6 can vary when model #1 and model #5 are used with $u_i(T)$. Table 9 lists the new resulting metrics for the original dataset. In this case, the new metrics are quite similar to those obtained with purely numerical coefficients **u** identified by the GP listed in Table 6. This is mainly due to the monotonic trend of the numerical coefficients **u**, which can easily be fitted with analytical functions.

As a final endorsement step, the experimental setup adopted for the previous datasets was adopted to perform a new characterization of the same 90-Ah LFP battery in discharging mode. For C_{rate} = 0.6C and T = 10°C, overall 55500 SoC values ranging from 20% to 80% were collected. Figure 9 shows the error between this new experimental battery terminal voltage dataset and the one predicted by using GP model #1 (blue line) and GP model #5 (red line) along with their coefficient trend **u**(T). The dashed black lines represent the measurement uncertainty that was calculated with a confidence level of about 98%, as detailed in Section II. Model #5 predicts the battery terminal voltage with a maximum absolute error lower than 15 mV, perfectly included in the given measurement uncertainty for C_{rate} = 0.6C and T = 10°C, and all the SoC values comprised between 20% and 80%. Conversely, the absolute error of model #1 does not always fall into the given measurement uncertainty but still ensures a good prediction of the battery terminal voltage with a maximum absolute error lower than 20 mV.

TABLE 7. Coefficient values for the behavioral model #1.

Coefficients for model #1	c_{i1}	c_{i2}	c_{i3}	c_{i4}
u_1	2.860e-07	7.169e-05	-0.510e-02	3.383
u_2	8.726e-08	-1.235e-04	8.004e-03	-1.662e-01

TABLE 8. Coefficient values for the behavioral model #5.

Coefficients for model #5	c_{i1}	c_{i2}	c_{i3}	c_{i4}
u_1	-3.360e-05	6.787e-03	-3.192e-01	5.241
u_2	-1.606e-04	8.963e-03	-8.965e-02	-1.088
u_3	1.638e-04	-1.861e-02	6.740e-01	-9.282
u_4	1.829e-04	-9.828e-03	7.706e-02	1.657
u_5	1.196e-06	2.242e-05	-5.736e-03	1.379e-01
u_6	-2.469e-05	1.149e-03	2.335e-03	-4.964e-01
u_7	-1.573e-04	1.467e-02	-4.518e-01	5.423
u_8	-2.466e-05	1.148e-03	2.365e-03	-4.965e-01
u_9	4.038e-05	-3.461e-03	9.741e-02	2.197

B. MACHINE LEARNING VERSUS GENETIC PROGRAMMING-BASED MODELING APPROACHES

This last paragraph compares the performance of the GP-based models derived in previous sections with the results of some of the most popular ML algorithms. Specifically, we refer to neural networks, linear regressors, and decision trees, which are well-known ML algorithms commonly used for their capability to create interpretable models [52]. Below,

TABLE 9. New metrics values for selected GP models, when coefficients trend (3) and values given in Table 7 and 8 are adopted.

Model & coefficients trend	Training dataset T_{tr}			
	μ_{err} (mV)	σ_{err} (mV)	err_{max} (mV)	RMS (mV)
Model #1 & $u_k(T)$	7.1	6.3	43.8	9.5
Model #5 & $u_k(T)$	3.3	3.7	38.5	4.9
Model & coefficients trend	Testing dataset T_{ts}			
	μ_{err} (mV)	σ_{err} (mV)	err_{max} (mV)	RMS (mV)
Model #1 & $u_k(T)$	6.3	5.3	21.1	8.2
Model #5 & $u_k(T)$	4.2	4.9	25.6	6.4

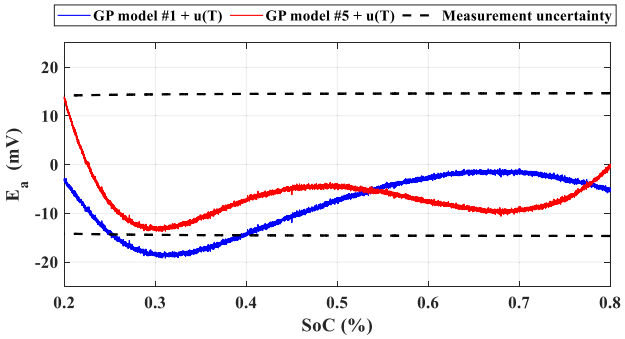


FIGURE 9. Absolute error between the experimental battery terminal voltage for $C_{rate} = 0.6C$ and $T = 10°C$ and the one predicted by using model #1 (blue line) and model #5 (red line) along with coefficients u(T).

we provide a brief overview of the ML algorithms used for the comparison, all implemented through the SciKit library [53] and applied to the same datasets described in Section III-B:

- the Multi-Layer Perceptron (MLP) regressor [54], which stands out for deep learning capabilities, with insights into complex patterns thanks to its layered architecture;
- the Cat-Boost (CBO) regressor [55], which allows predicting continuous data using decision trees;
- the Lasso (LAS) regressor [56], which is effective in feature selection, simplifying models, and enhancing their interpretability.

Table 10 shows the performance comparison of the GP-based and ML-based modeling approaches. These values clearly demonstrate the superior performance of the GP-based models over the ML-based ones. It is worth noting that the minimum amount of data required for good ML results greatly varies depending on the problem, model complexity, and data quality. In many scenarios, the cost of data acquisition can significantly impact modeling feasibility. An empirical approach, starting with a small dataset, gradually increasing its size, and continuously evaluating model performance, is often the only way to determine the minimum data volume needed to achieve satisfactory outcomes. Specialized techniques, like regularization and data augmentation, can improve the performance of ML-based modeling approaches. Given the constraints of battery characterization processes and the need for a significant reduction in

TABLE 10. Metrics For ML-based and GP-based modeling approaches over the testing dataset.

	Modeling approaches	μ_{err} (mV)	σ_{err} (mV)	err_{max} (mV)	RMS (mV)
GP	Model #1 & $u_k(T)$	6.3	5.3	21.1	8.2
	Model #5 & $u_k(T)$	4.2	4.9	25.6	6.4
ML	MLP	21.0	20.3	84.0	29.1
	CBO	15.3	26.0	133.1	29.9
	LAS	34.7	55.7	362.1	65.1

experimental effort (see Section II), the GP-based modeling approach represents a winning strategy for the behavioral modeling of Li-ion batteries.

Despite their different levels of complexity, the two GP models analyzed in the previous sections are both reliable analytical models. In a "direct problem," they can be similarly implemented into a BMS to predict the battery terminal voltage. The level of complexity can be increased, and all constraints on the coefficients released, if the ultimate goal is to achieve higher accuracy. However, a powerful aspect of the GP analytical representation lies in its reversibility, allowing it to be used in an "inverse problem," such as optimization or uncertainty analysis during the design phase. For these purposes, simpler analytical models with monotonic coefficients are of considerable added value.

V. CONCLUSION AND FUTURE WORKS

The paper proposes a novel methodology to derive analytical behavioral models for batteries using Genetic Programming (GP). Specifically, a 90-Ah Lithium Iron Phosphate (LFP) battery is modeled with the scope of describing its terminal voltage during a discharge phase as a function of the parameters State-of-Charge (SoC), charging/discharging rate (C_{rate}), and temperature (T), under realistic operating conditions: SoC between 20% and 80%, C_{rate} between 0.25C and 1.0C, and T between +5°C and +35°C. The methodology generates a variety of analytical models, each evaluated with metrics to select the final model based on a desired trade-off between accuracy and complexity. Two models of varying complexity and accuracy are extensively discussed and validated against experimental results, demonstrating maximum relative root mean square error values of 0.31% and 0.22% over the reference dataset, respectively.

The main merits of the new proposed approach are summarized as follows.

- The GP effectively identifies suitable behavioral models that describe the transient evolution of the battery terminal voltage as a function of SoC, $C_{rate,}$ and T. The modeling approach is versatile and can accommodate various types of datasets, including those incorporating environmental factors and aging dependencies.

- Only a few discharging curves obtained from full-discharge static characterization are required to generate a sufficient dataset for the GP.
- The GP-based models demonstrate high accuracy, performing well within the measurement uncertainties associated with the reference valuesof SoC, C_{rate}, and T.

Future work will be devoted to quantifying the advantages of this methodology over the existing modeling approaches, especially those relying on the popular Equivalent Circuit Models (ECMs). The robustness of the GP approach will also be tested on different types of batteries, with special emphasis on battery sizes (cylindrical, pouch) and chemistries, e.g. Nickel-Manganese-Cobalt Oxide (NMC) and Nickel-Cobalt-Aluminum Oxide (NCA).

APPENDIX
GP SETTINGS AND MULTI-OBJECTIVE OPTIMIZATION

As mentioned in Section III, the goal of our GP is to express the battery terminal voltage $V_{B,bhv}$ as a function of SoC, $C_{rate,}$ and temperature, according to (1). To evaluate the "quality" of each model discovered by the GP, three objective functions were adopted: the error between the outputs of the discovered model and the training data (evaluated by the objective function F_{rms}), the complexity of the model (evaluated by the objective function F_{cmp}), and the monotonicity of its **u** coefficients with respect to C_{rate} (evaluated by the objective function F_{mnt}). This results in a multi-objective optimization problem, whose fitness function F_{FIT} was expressed as the weighted sum given in (2). This Appendix aims to provide some clarifications on some GP settings (e.g., how the GP operators are set) and on the objective functions F_{cmp}, F_{cmp} and F_{mnt}.

A. GP SETTINGS

Table 11 summarizes the GP settings and main parameter values. Table 12 shows the elements of the terminal set (external nodes) and non-terminal set (internal nodes) used by the GP for models' generation. In this paper, the external nodes and internal nodes were assigned different complexity factors, with lower complexity factors given to the elements of the terminal set, to limit the vertical development of the models (i.e., involved functions of functions).

Then, the complexity c_f of a model was calculated as follows:

- each external node (input or coefficient) implies a 0.8 additive contribution to the overall complexity;
- each internal node implies an additive contribution to complexity, depending on the type of algebraic operator or basic function it implements;
- each internal node implies an extra additive contribution equal to the product of the complexity of the function and its argument, with different complexities for the function if the argument is an external or an internal node.

TABLE 11. GP settings and parameters.

Description	Value
Number of runs	50
Population size	100
Generation number	50
Maximum tree size	25 nodes
Individuals subjected to elitism	1% of the current population
Selection operator	Tournament ($k = 4$)
Crossover operator	subtree crossover
Mutation operator	subtree & node mutation
Crossover probability	0.80
Subtree mutation probability	0.20

All these choices limit the so-called bloat phenomenon [57], with models resulting in involved functions of functions or with many operations on simple functions.

B. GP OBJECTIVE FUNCTIONS

1) MODEL ERROR F_{rms}

Each element u_i ($i = 1,\ldots, n$) of the coefficients vector **u** was obtained by processing the m values u_{ij} ($j = 1, \ldots, m$), calculated using the Levenberg-Marquardt Non-Linear Least Square (NLLS) optimization method [58], over the m-size of the secondary variable data vector. In particular, for each $j = 1, \ldots, m$, the NLLS method provides the optimal values of each element u_{ij} minimizing the relative RMS error between the output values of the training dataset and those predicted by the model.

For example, given model #5 discussed in Section III, we have $n = 9$ and $m = 4$, for a training dataset length of $p_{Tr} = 288$. Given the numerical coefficients **u**, the global accuracy of each GP model must be estimated over the training dataset, by calculating the objective function for the relative RMS error as given in (4):

$$F_{rms} = \frac{1}{100}\sqrt{\frac{1}{p_{Tr}}\sum_{k=1}^{p_{Tr}}\left(\frac{V_{B,k}\text{-}V_{B,bhv,k}}{V_{B,k}}\right)^2} \tag{4}$$

2) MODEL COMPLEXITY F_{cmp}

There are many ways to classify the complexity of a model. The most common approach is to simply consider the depth and number of nodes that make up the tree. In this paper, the elements of the external nodes and internal nodes were assigned different complexity factors, as detailed in Table 12. The global complexity c_f of each constructed GP model was estimated accordingly (see Appendix A). The normalized complexity objective function F_{cmp} is then given by (5):

$$F_{cmp} = \frac{c_f - c_{f,\,min}}{c_{f,\,max} - c_{f,\,min}} \tag{5}$$

where c_f is the complexity factor value of a given model function, and $c_{f,min} = 7.5$ and $c_{f,max} = 80$ are the minimum and maximum levels of complexity, specifically identified

TABLE 12. GP external and internal node complexities.

Function	Arity	Description	External nodes complexity	Inner node complexity
sum	2	$h+g$	0.6	1
subtraction	2	$h-g$	0.6	1
multiplication	2	$h \cdot g$	0.75	1.1
division	2	h/g	0.85	1.2
reciprocal	1	$1/h$	0.85	1.2
negation	1	$-h$	0.6	1
square root	1	$\sqrt{h}$	1	1.5
square	1	h^2	1	1.5
natural log.	1	$\ln(h)$	1	1.5
natural exp.	1	e^h	1	1.5
sine	1	$\sin(h)$	1	1.5
cosine	1	$\cos(h)$	1	1.5
tangent	1	$\tan(h)$	1	1.5
inverse sine	1	$\sin^{-1}(h)$	1	1.5
inverse cosine	1	$\cos^{-1}(h)$	1	1.5
inverse tangent	1	$\tan^{-1}(h)$	1	1.5

for the case under study after some preliminary runs of the GP. These coefficients, like all the other assumptions for the configuration setup of the GP (e.g., the setting and parameters listed in Tables 11 and 12) can all be the object of a further level of optimization, as it usually happens for evolutionary algorithms intended for solving multi-objective problems [49].

It is worth noting that the function F_{cmp} refers solely to the complexity of the model function f and is then associated with its dependence on the primary variables SoC and C_{rate} only.

3) MODEL MONOTONICITY F_{mnt}

The objective function Fmnt expresses a qualitative characteristic of the u coefficients, namely their monotonicity with respect to temperature values for the case under study, and is calculated as in (5)-(9), according to the method detailed in [59]:

$$F_{mnt} = \frac{1}{k}\sum_{i=1}^{k} 2\min\left\{X^{(-)}, X^{(+)}\right\} \tag{6}$$

where:

$$X^{(+)} = \sum_{j=1}^{m-1}\left(C_{rate\,j+1} - C_{rate\,j}\right)\frac{\dot{u}_{i,j+1}^{(+)} + \dot{u}_{i,j}^{(+)}}{2} \tag{7}$$

$$X^{(-)} = \sum_{j=1}^{m-1}\left(C_{rate\,j+1} - C_{rate\,j}\right)\frac{\dot{u}_{i,j+1}^{(-)} + \dot{u}_{i,j}^{(-)}}{2} \tag{8}$$

$$\dot{u}_{i,j}^{(+)} = \max\left(\frac{u_{i,j} - u_{i,j-1}}{C_{rate\,j} - C_{rate\,j-1}}, 0\right) \tag{9}$$

$$\dot{u}_{i,j}^{(-)} = \max\left(\frac{u_{i,j-1} - u_{i,j}}{C_{rate\,j} - C_{rate\,j-1}}, 0\right) \tag{10}$$

ACKNOWLEDGMENT

The authors would like to thank Mario Molinara, whose expertise provided fundamental insight in the performance analysis of machine learning algorithms.

A minor part of C. Bourelly's dissertation, "Optimized data-driven measurement methods and devices for modeling diagnosis and prediction of batteries behavior", published on January 2024, University of Cassino and Southern Lazio, drew upon an early version of the present work.

REFERENCES

[1] M. Catenacci, E. Verdolini, V. Bosetti, and G. Fiorese, "Going electric: Expert survey on the future of battery technologies for electric vehicles," *Energy Policy*, vol. 61, pp. 403–413, Oct. 2013, doi: 10.1016/j.enpol.2013.06.078.

[2] C. P. Grey and D. S. Hall, "Prospects for lithium-ion batteries and beyond—A 2030 vision," *Nature Commun.*, vol. 11, no. 1, p. 6279, Dec. 2020, doi: 10.1038/s41467-020-19991-4.

[3] T. Kim, W. Song, D.-Y. Son, L. K. Ono, and Y. Qi, "Lithium-ion batteries: Outlook on present, future, and hybridized technologies," *J. Mater. Chem. A*, vol. 7, no. 7, pp. 2942–2964, Feb. 2019, doi: 10.1039/c8ta10513h.

[4] S. Tamilselvi, S. Gunasundari, N. Karuppiah, R. K. A. Razak, S. Madhusudan, V. M. Nagarajan, T. Sathish, M. Z. M. Shamim, C. A. Saleel, and A. Afzal, "A review on battery modelling techniques," *Sustainability*, vol. 13, no. 18, p. 10042, Sep. 2021, doi: 10.3390/su131810042.

[5] P. Schröer, H. van Faassen, T. Nemeth, M. Kuipers, and D. U. Sauer, "Challenges in modeling high power lithium titanate oxide cells in battery management systems," *J. Energy Storage*, vol. 28, Apr. 2020, Art. no. 101189, doi: 10.1016/j.est.2019.101189.

[6] H. Gabbar, A. Othman, and M. Abdussami, "Review of battery management systems (BMS) development and industrial standards," *Technologies*, vol. 9, no. 2, p. 28, Apr. 2021, doi: 10.3390/technologies9020028.

[7] P. Dini, A. Colicelli, and S. Saponara, "Review on modeling and SOC/SOH estimation of batteries for automotive applications," *Batteries*, vol. 10, no. 1, p. 34, Jan. 2024, doi: 10.3390/batteries10010034.

[8] A. Fotouhi, D. J. Auger, K. Propp, S. Longo, and M. Wild, "A review on electric vehicle battery modelling: From lithium-ion toward lithium–sulphur," *Renew. Sustain. Energy Rev.*, vol. 56, pp. 1008–1021, Apr. 2016, doi: 10.1016/j.rser.2015.12.009.

[9] P. Amiribavandpour, W. Shen, and A. Kapoor, "Development of thermal-electrochemical model for lithium ion 18650 battery packs in electric vehicles," in *Proc. IEEE Vehicle Power Propuls. Conf. (VPPC)*, Beijing, China, Oct. 2013, pp. 1–5, doi: 10.1109/VPPC.2013.6671675.

[10] P. Eleftheriadis, S. Giazitzis, S. Leva, and E. Ogliari, "Data-driven methods for the state of charge estimation of lithium-ion batteries: An overview," *Forecasting*, vol. 5, no. 3, pp. 576–599, Sep. 2023, doi: 10.3390/forecast5030032.

[11] J. F. Manwell and J. G. McGowan, "Lead acid battery storage model for hybrid energy systems," *Sol. Energy*, vol. 50, no. 5, pp. 399–405, May 1993, doi: 10.1016/0038-092x(93)90060-2.

[12] V. Rao, G. Singhal, A. Kumar, and N. Navet, "Battery model for embedded systems," in *Proc. 18th Int. Conf. VLSI Design Jointly 4th Int. Conf. Embedded Syst. Design*, Kolkata, India, Jan. 2005, pp. 105–110, doi: 10.1109/icvd.2005.61.

[13] M.-F. Ng, J. Zhao, Q. Yan, G. J. Conduit, and Z. W. Seh, "Predicting the state of charge and health of batteries using data-driven machine learning," *Nature Mach. Intell.*, vol. 2, no. 3, pp. 161–170, Mar. 2020, doi: 10.1038/s42256-020-0156-7.

[14] F. Heinrich, P. Klapper, and M. Pruckner, "A comprehensive study on battery electric modeling approaches based on machine learning," *Energy Informat.*, vol. 4, no. S3, Sep. 2021, Art. no. 17, doi: 10.1186/s42162-021-00171-7.

[15] L. Zhang, K. Li, D. Du, Y. Guo, M. Fei, and Z. Yang, "A sparse learning machine for real-time SOC estimation of li-ion batteries," *IEEE Access*, vol. 8, pp. 156165–156176, 2020, doi: 10.1109/ACCESS.2020.3017774.

[16] M. Thele, O. Bohlen, D. U. Sauer, and E. Karden, "Development of a voltage-behavior model for NiMH batteries using an impedance-based modeling concept," *J. Power Sources*, vol. 175, no. 1, pp. 635–643, Jan. 2008, doi: 10.1016/j.jpowsour.2007.08.039.

[17] H. He, R. Xiong, and J. Fan, "Evaluation of lithium-ion battery equivalent circuit models for state of charge estimation by an experimental approach," *Energies*, vol. 4, no. 4, pp. 582–598, Mar. 2011, doi: 10.3390/en4040582.

[18] M. Chen and G. A. Rincon-Mora, "Accurate electrical battery model capable of predicting runtime and I-V performance," *IEEE Trans. Energy Convers.*, vol. 21, no. 2, pp. 504–511, Jun. 2006, doi: 10.1109/TEC.2006.874229.

[19] A. Hentunen, T. Lehmuspelto, and J. Suomela, "Time-domain parameter extraction method for Thévenin-equivalent circuit battery models," *IEEE Trans. Energy Convers.*, vol. 29, no. 3, pp. 558–566, Sep. 2014, doi: 10.1109/TEC.2014.2318205. https://doi.org/10.1109/TEC.2014.2318205

[20] Z. M. Salameh, M. A. Casacca, and W. A. Lynch, "A mathematical model for lead-acid batteries," *IEEE Trans. Energy Convers.*, vol. 7, no. 1, pp. 93–98, Mar. 1992, doi: 10.1109/60.124547.

[21] L. Gao, S. Liu, and R. A. Dougal, "Dynamic lithium-ion battery model for system simulation," *IEEE Trans. Compon. Packag. Technol.*, vol. 25, no. 3, pp. 495–505, Sep. 2002, doi: 10.1109/TCAPT.2002.803653.

[22] M. C. Glass, "Battery electrochemical nonlinear/dynamic SPICE model," in *Proc. 31st Intersociety Energy Convers. Eng. Conf.*, Washington, DC, USA, Aug. 1996, pp. 292–297, doi: 10.1109/iecec.1996.552887.

[23] R. Xiong, J. Tian, W. Shen, and F. Sun, "A novel fractional order model for state of charge estimation in lithium ion batteries," *IEEE Trans. Veh. Technol.*, vol. 68, no. 5, pp. 4130–4139, May 2019, doi: 10.1109/TVT.2018.2880085. https://doi.org/10.1109/TVT.2018.2880085

[24] C. Bourelly, M. Vitelli, F. Milano, M. Molinara, F. Fontanella, and L. Ferrigno, "EIS-based SoC estimation: A novel measurement method for optimizing accuracy and measurement time," *IEEE Access*, vol. 11, pp. 91472–91484, 2023, doi: 10.1109/ACCESS.2023.3308029.

[25] D. Andre, M. Meiler, K. Steiner, C. Wimmer, T. Soczka-Guth, and D. U. Sauer, "Characterization of high-power lithium-ion batteries by electrochemical impedance spectroscopy. I. Experimental investigation," *J. Power Sources*, vol. 196, no. 12, pp. 5334–5341, Jun. 2011, doi: 10.1016/j.jpowsour.2010.12.102.

[26] S. Gold, "A PSPICE macromodel for lithium-ion batteries," in *Proc. 12th Annu. Battery Conf. Appl. Adv.*, Long Beach, CA, USA, Jan. 1997, pp. 215–222, doi: 10.1109/bcaa.1997.574106.

[27] J. Koza, "Genetic programming as a means for programming computers by natural selection," *Statist. Comput.*, vol. 4, no. 2, pp. 87–112, Jun. 1994, doi: 10.1007/bf00175355.

[28] K. Stoyka, G. D. Capua, and N. Femia, "A novel AC power loss model for ferrite power inductors," *IEEE Trans. Power Electron.*, vol. 34, no. 3, pp. 2680–2692, Mar. 2019, doi: 10.1109/TPEL.2018.2848109. https://doi.org/10.1109/TPEL.2018.2848109

[29] G. Di Capua, N. Femia, K. Stoyka, G. Di Mambro, A. Maffucci, S. Ventre, and F. Villone, "Mutual inductance behavioral modeling for wireless power transfer system coils," *IEEE Trans. Ind. Electron.*, vol. 68, no. 3, pp. 2196–2206, Mar. 2021, doi: 10.1109/TIE.2019.2962432. https://doi.org/10.1109/TIE.2019.2962432

[30] G. Di Capua, A. Maffucci, K. Stoyka, G. Di Mambro, S. Ventre, V. Cirimele, F. Freschi, F. Villone, and N. Femia, "Analysis of dynamic wireless power transfer systems based on behavioral modeling of mutual inductance," *Sustainability*, vol. 13, no. 5, p. 2556, Feb. 2021, doi: 10.3390/su13052556.

[31] Accessed: Jun. 26, 2023. [Online]. Available: https://www.lithium-solarbattery.com/sale-14082487-eve-3-2v-90ah-lfp-prismatic-lifepo4-rechargeable-battery-for-solar-energy.html

[32] *Secondary Cells and Batteries Containing Alkaline or Other Non-Acid Electrolytes—Secondary Lithium Cells and Batteries for Use in Industrial Applications*, IEC Standard 62620:2014, 2014. [Online]. Available: https://webstore.iec.ch/publication/7268

[33] *Secondary Lithium-Ion Cells for the Propulsion of Electric Road Vehicles—Part 3: Safety Requirements*, IEC Standard 62660-3:2022, 2022. [Online]. Available: https://webstore.iec.ch/publication/65084

[34] United States Advanced Battery Consortium (USABC) Manuals. (Oct. 2020). *Electric Vehicle Battery Test Procedures Manual Version 3.1*. Accessed: Mar. 6, 2024. [Online]. Available: https://uscar.org/usabc/

[35] S. Piller, M. Perrin, and A. Jossen, "Methods for state-of-charge determination and their applications," *J. Power Sources*, vol. 96, no. 1, pp. 113–120, Jun. 2001, doi: 10.1016/s0378-7753(01)00560-2.

[36] E. D. Kostopoulos, G. C. Spyropoulos, and J. K. Kaldellis, "Real-world study for the optimal charging of electric vehicles," *Energy Rep.*, vol. 6, pp. 418–426, Nov. 2020, doi: 10.1016/j.egyr.2019.12.008.

[37] J. Jiang, W. Shi, J. Zheng, P. Zuo, J. Xiao, X. Chen, W. Xu, and J.-G. Zhang, "Optimized operating range for large-format $LiFePO_4$/graphite batteries," *J. Electrochemical Soc.*, vol. 161, no. 3, pp. A336–A341, 2014, doi: 10.1149/2.052403jes.

[38] L. Lu, X. Han, J. Li, J. Hua, and M. Ouyang, "A review on the key issues for lithium-ion battery management in electric vehicles," *J. Power Sources*, vol. 226, pp. 272–288, Mar. 2013, doi: 10.1016/j.jpowsour.2012.10.060.

[39] S. Luke and L. Spector, "A comparison of crossover and mutation in genetic programming," *Genetic Program.*, vol. 97, pp. 240–248, Jul. 1997.

[40] A. Shukla, H. M. Pandey, and D. Mehrotra, "Comparative review of selection techniques in genetic algorithm," in *Proc. Int. Conf. Futuristic Trends Comput. Anal. Knowl. Manage. (ABLAZE)*, Feb. 2015, pp. 515–519, doi: 10.1109/ABLAZE.2015.7154916.

[41] M.-J. Willis, "Genetic programming: An introduction and survey of applications," in *Proc. 2nd Int. Conf. Genetic Algorithms Eng. Syst.*, Sep. 1997, pp. 314–319, doi: 10.1049/cp:19971199.

[42] D. Ari and B. B. Alagöz, "A review of genetic programming: Popular techniques, fundamental aspects, software tools and applications," *Sakarya Univ. J. Sci.*, vol. 25, no. 2, pp. 397–416, Apr. 2021, doi: 10.16984/saufenbilder.793333.

[43] Accessed: Mar. 6, 2024. [Online]. Available: https://sites.google.com/site/gptips4matlab/

[44] Accessed: Mar. 6, 2024. [Online]. Available: https://gplab.sourceforge.net/features.html

[45] Accessed: Mar. 6, 2024. [Online]. Available: https://github.com/tiagoinacio/cgp4matlab

[46] Accessed: Mar. 6, 2024. [Online]. Available: https://gplearn.readthedocs.io/en/stable/

[47] Accessed: Mar. 6, 2024. [Online]. Available: https://code.google.com/archive/p/pygep/

[48] Accessed: Mar. 6, 2024. [Online]. Available: https://code.google.com/archive/p/gep4j/

[49] C. A. C. Coello, G. B. Lamont, and D. A. Van Veldhuizen, *Evolutionary Algorithms for Solving Multi-Objective Problems*, 2nd ed., New York, NY, USA: Springer, 2007, p. 21 and 800, doi: 10.1007/978-0-387-36797-2.

[50] H. He, R. Xiong, H. Guo, and S. Li, "Comparison study on the battery models used for the energy management of batteries in electric vehicles," *Energy Convers. Manage.*, vol. 64, pp. 113–121, Dec. 2012, doi: 10.1016/j.enconman.2012.04.014.

[51] M. S. Hosen, D. Karimi, T. Kalogiannis, A. Pirooz, J. Jaguemont, M. Berecibar, and J. Van Mierlo, "Electro-aging model development of nickel-manganese-cobalt lithium-ion technology validated with light and heavy-duty real-life profiles," *J. Energy Storage*, vol. 28, Apr. 2020, Art. no. 101265, doi: 10.1016/j.est.2020.101265.

[52] C. Molnar, *Interpretable Machine Learning—A Guide for Making Black Box Models Explainable*, 2nd ed., Independently Published, Accessed: Jun. 26, 2024. [Online]. Available: https://christophm.github.io/interpretable-ml-book/cite.html

[53] *Scikit-Learn–Machine Learning in Python*. Accessed: Jun. 26, 2024. [Online]. Available: https://scikit-learn.org/

[54] R. Collobert and S. Bengio, "Links between perceptrons, MLPs and SVMs," in *Proc. 21st Int. Conf. Mach. Learn. (ICML)*, Jul. 2004, pp. 1–17, doi: 10.1145/1015330.1015415.

[55] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "CatBoost: Unbiased boosting with categorical features," 2017, *arXiv:1706.09516*.

[56] R. Tibshirani, "The lasso method for variable selection in the Cox model," *Statist. Med.*, vol. 16, pp. 385–395, Feb. 1997, doi: 10.1002/(SICI)1097-0258(19970228)16:4<385::AID-SIM380>3.0.CO;2-3.

[57] E. De Jong, R. A. Watson, and J. B. Pollack, "Reducing bloat and promoting diversity using multi-objective methods," in *Proc. 3rd Annu. Conf. Genetic Evol. Comput.*, Jul. 2001, pp. 11–18.

[58] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing*, 2nd ed., Cambridge, U.K.: Cambridge Univ. Press, 1992.

[59] Y. Davydov and R. Zitikis, "Quantifying non-monotonicity of functions and the lack of positivity in signed measures," *Modern Stochastics, Theory Appl.*, vol. 4, no. 3, pp. 219–231, Sep. 2017, doi: 10.15559/17-vmsta84.

GIULIA DI CAPUA (Senior Member, IEEE) received the B.Sc. and M.Sc. degrees in electronic engineering and the Ph.D. degree in information engineering from the University of Salerno, in 2006, 2009, and 2013, respectively.

She was a Postdoctoral Researcher (2013–2016) and a Research Assistant (2016–2020) with the University of Salerno. Since 2021, she has been with the Department of Electrical and Informational Engineering, University of Cassino and Southern Lazio, where she has been an Assistant Professor (2021–2024). She is currently an Associate Professor in electrotechnics. Her research interests include semiconductor and magnetic power device modeling and testing, switching-mode power supply design, wireless power transfer systems, and signal/power integrity.

Dr. Di Capua is also the Chair of the IEEE Power and Energy Circuits and Systems Technical Committee (PECAS-TC). She was an Associate Editor of IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I: REGULAR PAPERS (IEEE TCAS-I) (2020–2023).

FRANCESCO PORPORA (Member, IEEE) received the master's degree in electrical engineering and the Ph.D. degree in methods, models, and technologies for engineering from the University of Cassino and Southern Lazio, Italy, in 2017 and 2021, respectively.

He is currently a Research Associate with the Department of Electrical and Information Engineering, University of Cassino and Southern Lazio. His research interests include the development, prototyping, control of battery management systems and passive and active equalization techniques for li-ion packs, modeling, optimization, control of electrochemical energy storage systems, and the power converters needed for their integration in mobility and stationary applications.

FILIPPO MILANO (Member, IEEE) received the M.S. degree (summa cum laude) in electrical engineering and the Ph.D. degree in methods, models and technologies for engineering from the University of Cassino and Southern Lazio, Cassino, Italy, in 2018 and 2021, respectively. He is currently a Research Fellow with the Department of Electrical and Information Engineering, University of Cassino and Southern Lazio. His research interests involve the design, implementation and characterization of positioning systems for biomedical and industrial applications, the development of models and techniques for the predictive batteries diagnosis, and the application of eddy current techniques for the estimation of geometric and physical properties of conductive materials.

NUNZIO OLIVA received the B.Sc. and M.Sc. degrees in electronic engineering from the University of Salerno, Italy, in 2018 and 2020, respectively.

From 2020 to 2022, he held positions as a Research Fellow with the University of Salerno, conducting research on the analysis and design of switching mode power supplies, power magnetics, numerical techniques for identification, and optimization of behavioral models of power devices and systems. In 2022, he co-founded Exeling, a startup providing technical-scientific consultancy services and focusing on the development of advanced systems for power devices and systems characterization, where he currently leads the Research and Development Department.

ANTONIO MAFFUCCI (Senior Member, IEEE) received the Laurea degree (summa cum laude) in electronic engineering and the Ph.D. degree in electrical engineering from the University of Naples Federico II, Italy, in 1996 and 2000, respectively.

He is currently a Full Professor in electrotechnics with the Department of Electrical and Information Engineering, University of Cassino and Southern Lazio, Cassino, Italy, and a Research Associate with the National Institute of Nuclear Physics, INFN-LNF, Frascati, Italy. In 1997, he was with the Nuclear Fusion Laboratory JET, Culham, U.K. From 1998 to 2002, he was with the Department of Electrical Engineering, University of Naples Federico II. He is the author of about 180 technical papers in international journals, conference proceedings, and essays on books. He has co-authored the books *Transmission Lines and Lumped Circuits* (Academic Press, 2001), *Fundamentals of Applied Nanoelectromagnetics* (Springer, 2016 and 2019), *Carbon Nanotube for Interconnects* (Springer, 2016), and *Carbon-Based Nano-Electromagnetics* (Elsevier, 2019). His research interests include computational electromagnetics, electromagnetic compatibility, distributed circuits and systems, nanoelectronics, and quantum circuits. He is the Coordinator of the EU H2020-MSCA-RISE Project "Terasse" (2019–2023). He is an Associate Editor of IEEE TRANSACTIONS ON COMPONENTS, PACKAGING, AND MANUFACTURING TECHNOLOGY, a member of IEEE Nano-Packaging Council, and an Editorial Board Member of *Applied Sciences* and the *Journal of Nanoscience and Nanotechnology Applications*. He served as the General Chairperson for the Conferences IEEE SPI 2011 and 2012 and FANEM 2015 and 2018. He was a recipient of the Best Presentation Award at ENDE 2019, the Nanoscale Horizon Poster Prize at IEEE Nano 2015, the Best Paper Award at the IEEE Workshop SPI 2009, and the Outstanding Paper Award from "Literati Network Awards for Excellence 2008."

Open Access funding provided by 'Università degli Studi di Cassino e del Lazio Meridionale' within the CRUI CARE Agreement