Advancing Electric Vehicle Battery Management: A Data-Driven Digital Twin Approach for Real-Time Monitoring and Performance Enhancement

Advancing Electric Vehicle Battery Management: A Data-Driven Digital Twin Approach for Real-Time Monitoring and Performance Enhancement

Khaled S. S. Alamin, *Student Member, IEEE*, Yukai Chen, *Member, IEEE*, Enrico Macii, *Fellow, IEEE*, Massimo Poncino, *Fellow, IEEE*, Sara Vinco, *Senior Member, IEEE*

***Abstract*—The global shift towards electric vehicles (EVs) represents a critical strategy for fighting climate change and reducing both reliance on fossil fuels and CO_2 emissions. In this scenario, battery management systems (BMS) become crucial in EVs to ensure battery safety, reliability, and efficiency. Recently, data-driven estimation techniques have been proposed to estimate battery-related metrics, such as remaining capacity or aging condition. These techniques emerged as an answer to batteries' intrinsic variability, but they rely primarily on full charge/discharge cycles, overlooking the nuances of partial charging, which is prevalent in real-world usage. This paper presents a novel BMS architecture for EVs, leveraging digital twin technology and data-driven modeling to address these challenges. The proposed dynamic dual-model approach seamlessly integrates real-time monitoring with cloud-based analytics to continuously evaluate and predict battery capacity degradation. Key innovations include sophisticated feature engineering and segmentation strategies, which enable precise state of health (SoH) estimation across a wide range of driving conditions. Additionally, the architecture incorporates a dynamically retraining State of Charge/Energy (SoC/SoE) model that adapts to battery aging, thereby maintaining high accuracy throughout the battery's life cycle. Extensive validation using datasets from public institutions demonstrates the effectiveness and robustness of the proposed system.**

***Index Terms*—Partial discharge, Battery state estimation, Data-driven modeling, Battery management system (BMS), Digital twin, Feature extraction, Electric vehicle (EV)**

I. Introduction

The global transportation sector is experiencing a significant transformation, driven by the urgent need to address environmental challenges and to adopt sustainable practices [1]. The shift to Electric Vehicles (EVs) has become a key strategy in combating climate change and reducing dependence on finite fossil fuels and is thus not merely a trend but rather a fundamental adaptation to modern challenges [2]. Simultaneously, the growing demand for electricity across both consumer and industrial sectors highlights the pressing need for efficient, reliable, and safe energy storage systems [3].

Accepted at IEEE Transactions on Vehicular Technology

At the heart of this eco-friendly revolution lies a critical component: batteries. In EVs, batteries are not mere power sources; They are the cornerstone of the entire electric propulsion system. The *Battery Management Systems* (BMS) thus emerges as an essential component to ensure safe, reliable, and efficient battery utilization [4], [5], with special attention to estimating and evaluating metrics such as State of Charge (SoC, i.e., available charge stored in the battery relative to its full charge capacity), State of Health (SoH, i.e., ratio of the maximum available capacity during a full discharge cycle to the battery's nominal capacity), and State of Energy (SoE, i.e., ratio of the remaining energy to the total available energy) [6].

To accomplish these goals, the BMS continuously collects and analyzes data from the battery, including temperature, voltage, and current [7]. The architectural framework of a BMS is organized around all hierarchy levels (i.e., cell, module, and pack), as each layer includes specialized units that perform essential monitoring and control functions to ensure efficient operation at all levels of the battery system [8].

Despite the critical importance of SoH, SoC, and SoE estimation, these parameters cannot be directly measured from the battery terminals, emphasizing the *need for advanced estimation algorithms* [9]. Extensive research has been conducted on battery SoH, with a particular focus on developing online estimation methods tailored for EVs and smart grid applications [10]. However, many existing approaches to SoH estimation often overlook the complexity raised by the application to real driving conditions.

First of all, it is important to note that *SoC has a dynamic SoH-dependent nature*: when the SoH declines, the available capacity of the battery decreases, leading to shorter discharge times. This means that any SoC model that is accurate for a new battery may no longer be valid as the battery ages. To exemplify this, Figure 1 shows the evolution of SoC during full discharge cycles (100% to 0%) at three different SoH levels, as from real measurement data [11]: as the battery degrades, the discharge time decreases from 130 minutes at SoH = 96% (red), to 100 minutes at SoH = 80% (green), and to only 70 minutes seconds at SoH = 54% (purple). This variability poses a significant challenge to existing SoC models, which rely on static estimation frameworks that lose accuracy as battery aging affects discharge characteristics. It is thus clear that BMS must enforce multiple SoH-aware SoC models that dynamically

adapt to changing battery conditions.

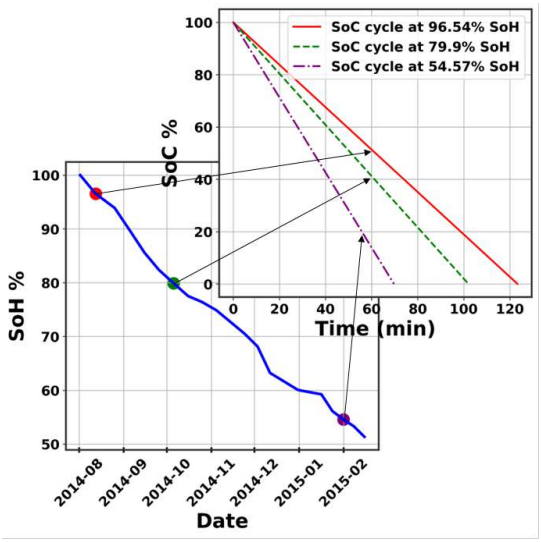

Figure 1: An analysis of full discharge cycles (SoC from 100% to 0%) at different SoH levels highlights that SoC behavior strictly depends on battery aging (data from [11]).

A second necessary consideration is that current data-driven approaches for battery monitoring largely rely on the availability of *full charge/discharge profiles*, ranging from 0% to 100% SoC and vice versa. Such profiles are typical of open source datasets used for SoH models training [11], [12]. However, in real-world driving scenarios, the assumption of frequent full cycles is highly restrictive as full charge/discharge cycles are rare in reality, making it impractical to depend on these profiles for regular model retraining. Figure 2 shows the SoC profile of a 60Ah battery system recorded during 72 real driving trips of a BMW i3 [13]. In this figure, green represents charging phases, red represents discharge phases, and white indicates rest phases. Notably, complete charge or discharge cycles are uncommon. This observation is crucial for battery modeling and state estimation. The rarity of full charge/discharge cycles in real-world conditions highlights the necessity of considering partial SoC patterns for model updates and battery management strategies. Real driving conditions thus necessitate dynamic model updating to account for incomplete and irregular SoC profiles.

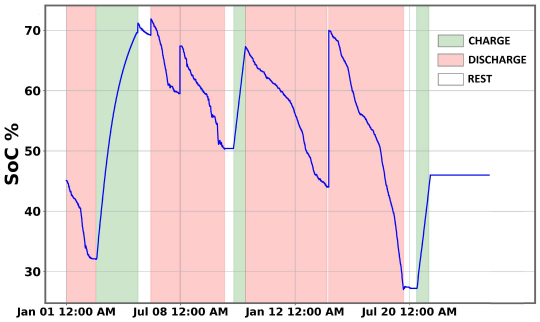

Figure 2: Real Driving Pattern of an EV [13].

Given these considerations, we present an innovative BMS solution for EVs, integrating digital twin technology with battery-tailored data-driven modeling:

- The proposed solution is a more flexible, adaptive approach to SoC and SoH estimation that can operate effectively across the full life-cycle of the battery and in realistic driving conditions, thus taking advantage of both rare full charge/discharge cycles and frequent partial charge/discharge cycles;
- The proposed solution accommodates well in a typical BMS configuration, exploiting both (i) cloud support to dynamically retrain SoC/SoE models whenever necessary to react to ageing, thus ensuring an accurate estimation of battery health across diverse driving conditions and over the battery's lifespan, and (ii) edge to achieve real-time monitoring and timely estimation of battery dynamics;
- We demonstrate the effectiveness, reliability, and generality of our approach through rigorous validation using datasets sourced from reputable institutions.

The paper is structured to provide a comprehensive understanding of the proposed digital twin solution. Section II lays the foundation by presenting essential background information. Section III elaborates on the methodology and architecture of the proposed digital twin, which is then detailed in Section IV. The dataset description is presented in Section V, and the experimental application is detailed in Section VI. Finally, Section VII describes a proof-of-concept digital twin, and Section VIII summarizes the key findings and implications of this study.

II. Foundations of Proposed Approach

A. Digital Twin Evolution

A digital twin serves as a *dynamic, real-time digital replica of a physical battery system*, with predictive capabilities driven by continuous, two-way data exchange between the physical battery and its digital counterpart. This close interaction enables accurate replication and forecasting of battery behavior under various conditions [6]. The adoption of digital twin technology represents a shift in BMS functionality, emphasizing real-time monitoring, precise control, and predictive analytics to optimize battery performance and extend its operational life.

Digital twins usually require massive computational resources to analyze real-time data and to run sophisticated algorithms to perform state estimation, health prediction, and optimization with reduced latency [6]. For these reasons, digital twin architectures adopt scalable solutions that integrate the real-time processing and decision-making capabilities of edge computing with the extensive computational power and storage resources of cloud systems [14]. This hybrid approach leverages the strengths of both edge and cloud computing. By distributing the workload, this architecture ensures efficient state estimation, health prediction, and optimization [15].

B. Battery Modeling Developments

Even industry leaders, such as Tesla, have encountered challenges with BMS calibration [16]: such real-world examples underscore the persistent difficulties in achieving accurate and reliable SoC and SoH estimations.

1) Battery models: Traditional *physics-based models*, such as equivalent circuit models or electrochemical models, come with significant drawbacks: sources of error include initial

inaccuracies in SoC estimation, particularly when *Coulomb Counting* techniques are used, as well as errors from imprecise measurements, prediction flaws in the models, and capacity estimation inaccuracies [14]. Moreover, the inherent complexity of EV battery systems further increases the number of parameters required for accurate modeling.

Recognizing such limitations, recent research has increasingly explored *data-driven methodologies* to enhance battery state estimation [17], [18], [19]. Even if recent works improve SoC estimation stability and accuracy, they often require extensive optimization efforts, not suitable for frequent updates required in realistic EV operational scenarios specially at real time [19]. Most works rely on specific charging and discharging patterns. As an example, [20] requires setting the initial SoC of the partial charge curve to a fixed value and charging the battery to 100% SoC during the estimation process, while [21] and [22] exploits gradual decreasing current in the constant current and constant voltage (CC-CV) charging phase, thus still requiring complete and structured charging cycles, that are rarely available in real-world scenarios. Such constraints are often impractical in real-world scenarios, where batteries are frequently charged in partial cycles and with heterogeneous usage patterns. This limits the flexibility and applicability of the algorithm, especially in dynamic environments where full charging is not feasible.

Lately, *hybrid approaches* have been proposed to pursue increased accuracy by combining a formal model, either physics- or math-based, with a model that is either data-driven or built on lab experiments [23], [24]. The combination exploits the known physical governing battery dynamics with the support for non-linear behaviors and variability. A successful example is Physical Informed Neural Networks (PINNs), which are trained to solve supervised learning tasks while respecting laws of physics (e.g., Fick's diffusion equation or Peukert's equation) described by general nonlinear PDEs [25], [26], [27]. It is important to note that the data-driven part of such models still relies on full charge/discharge curves for model training.

2) Data segmentation: Most existing data-driven SoH estimation methods employ *static segmentation*, which divides the battery's operating range into fixed intervals for analysis. This static approach fails to capture the dynamic behavior of batteries, particularly under varying operating conditions, and leads to inaccurate SoH estimations, especially when the battery deviates from the expected operating conditions.

To overcome the limitations of static approaches, *dynamic segmentation* has emerged as a promising alternative. Dynamic segmentation adjusts the segmentation points based on the real-time state of the battery. For example, [28] uses a recursive least squares algorithm to estimate the open-circuit voltage (OCV), which is then used to infer the SoC via look-up tables. However, the accuracy of this method diminishes as the correlation between OCV and SoC changes with battery aging, resulting in potential SoC estimation errors. [29] aligns variable cycle lengths with trained data using spatial and temporal correlations, but it works only for long cycles and heavily relies on datasets with specific degradation patterns, reducing its adaptability to diverse battery types and operational conditions.

III. Digital Twin for Battery Monitoring

This section outlines the key features of the digital twin framework proposed for EV BMS. The primary focus of this framework is the accurate estimation of SoC, SoE, and SoH, which are critical parameters influencing both vehicle performance and battery longevity. Our approach leverages data-driven models to capture the dynamic behavior of these parameters, facilitating more effective management of battery resources. By addressing the limitations of the current state-of-the-art solutions discussed in the previous section, our methodology aims to provide a more adaptive and accurate framework for battery monitoring and optimization.

A. Proposed solution and novelty

Our research acknowledges a notable void in the existing literature: the absence of a comprehensive BMS model that dynamically adapts to varying driving cycles while maintaining accurate estimation of SoC, SoE, and SoH. Most existing studies treat these parameters in isolation, lacking a holistic framework that accounts for the interplay between diverse driving behaviors and their cumulative effect on battery health. Moreover, many approaches rely on unrealistic assumptions, such as the availability of full charge and discharge data over time, which seldom occurs in real-world electric vehicle usage.

Unlike these methods, our proposed approach features an *adaptive framework* that captures the non-linear relationship between battery states and other critical parameters. This enables improved generalization and robustness across a wide range of operational conditions and real driving scenarios, providing a more realistic and effective approach to battery monitoring and management.

The main characteristics and innovative choices of our digital twin solution are the following:

a) Data-Driven Models: Both the SoC/E and SoH models are inherently *data-driven*, constructed using historical data traces. By leveraging various machine learning techniques, these models offer flexibility and adaptability to diverse battery types and usage scenarios. Our solution specifically utilizes a Light Gradient Boosting Model (LGBM), which effectively captures complex relationships in high-dimensional datasets [30].

b) Proactive Model Updates: Our method utilizes separate models for SoH, and SoC/SoE, allowing each to specialize and achieve improved accuracy. The SoH model, developed from historical data of aging batteries with similar characteristics to those in the vehicle, provides a long-term reference for capacity degradation. In contrast, the SoC/E model is initially trained under nominal battery conditions but undergoes continuous retraining with real-time data collected during vehicle operation. This *dynamic updating mechanism* ensures that the SoC/E model remains responsive to evolving battery conditions throughout its lifespan. A threshold-based triggering system updates the models when SoH degradation exceeds a predefined limit, ensuring sustained accuracy even in continuous operation.

c) Dynamic Segmentation: The construction of data-driven models *dynamically adjusts the segmentation points* based on the current battery state and driving pattern. This is necessary to allow the model to capture the evolving battery dynamics, and fully exploit available data, even when full charge/discharge cycles are absent. Although the training dataset may include full cycles, our solution deliberately splits them into segments to emulate real-life driving conditions, which typically involve partial SoC variations (e.g., from 63% to 43%, or 75% to 10%). This segmentation technique significantly enhances prediction accuracy in real-world scenarios.

d) Applicability to different scenarios: Being strictly data-driven, the proposed approach generalizes well, as will be proven by the experimental results. The approach is battery type and chemistry agnostic, as it relies on data measurements (collected in the training dataset) and it does not contain any explicit formal modeling of battery internal physical and chemical properties. This will be proven with experiments on different battery chemistry types (e.g., Lithium-Cobalt Oxide, Nickel Manganese Cobalt, Lithium Iron Phosphate). Additionally, the experiments will be applied to heterogeneous dataset conditions, including different charge and discharge patterns, and also laboratory setup versus real driving patterns. This will be highlighted with the adoption of different datasets from reputable institutions: NASA Randomized Battery Usage dataset [11], the Sandia National Laboratories (SNL) Battery dataset [31], and the Stanford Urban Dynamometer Driving Schedule (UDDS) Battery dataset [32]. Finally, the experiments will compare different data-driven algorithms, to discuss the extensibility of the proposed approach.

In summary, this paper presents an innovative approach to BMS, addressing the critical gap in the existing literature by proposing a dynamic, adaptive model capable of responding to varying driving cycles. Our methodology, grounded in advanced data analytics and the concept of a digital twin, marks a significant advancement in the field of EV battery management. We anticipate that our findings will provide a foundation for future research in this rapidly evolving domain.

B. Digital Twin Architecture and Operation

Figure 3 illustrates the architecture of our digital twin system seamlessly integrated within the BMS architecture. The figure highlights that the BMS has a hierarchical structure. At the foundation of this hierarchy, the Cell Monitoring Units (CMUs) are responsible for extracting raw data from individual battery cells (e.g., voltage, current, and temperature). The data collected by multiple CMUs are transmitted via a CAN bus to the Module Management Unit (MMU), that optimizes performance and longevity with fault detection, cell balancing, and temperature regulation at the module level. Finally, the Pack Management Unit (PMU), performs pack-wide monitoring and safety management and serves as the central hub for communication with vehicle controllers.

To integrate the digital twin system into the BMS, historical data from batteries of the same type as those used in the target EV is stored in the cloud. This data is used to train LightGBM models for SoH and SoC/E estimation ❶. These trained models

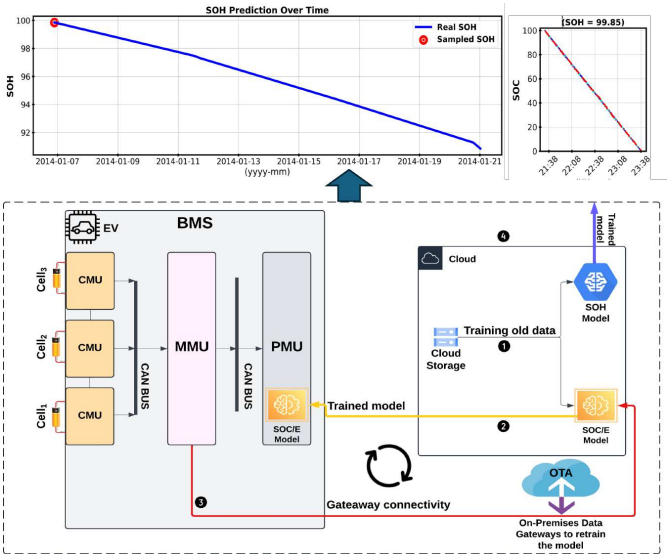

Figure 3: BMS integration with Digital Twin

form the basis for predictive analytics and real-time monitoring of the battery system.

Once the SoC/E model is trained, it is transmitted from the cloud to the PMU within the BMS of the EV ❷. The model is deployed on the PMU and runs continuously, predicting real-time SoC/E values during vehicle operation.

The EV periodically sends data from the BMS to the cloud via a secure gateway ❸. This data includes real-time sensor readings and operational parameters, which are stored in the cloud for further analysis.

The cloud system uses this collected data to periodically retrain the SoC/E model ❹. The SoH model is also periodically evaluated to track battery aging. Based on predefined time intervals (e.g., weekly, biweekly, monthly) or specific SoH degradation thresholds (e.g., 1%, 5%, 10%), the accumulated data is utilized to retrain the SoC/E model. This updated model is then redeployed on the BMS, ensuring continuous accuracy and adaptation to the evolving battery conditions.

The use of OTA technology facilitates seamless communication between BMS and cloud-based systems, overcoming potential challenges such as internet access issues and ensuring continuous data exchange in real-time. This allows the digital twin to remain synchronized with the battery pack's operational status, enabling timely interventions and optimization strategies [33].

In summary, the methodology outlined here establishes a robust framework for the integration of a digital twin with the BMS, enabling proactive monitoring, predictive maintenance, and optimization of EV battery performance. Through continuous data exchange and real-time analytics, this approach enhances the reliability, efficiency, and longevity of EV batteries across diverse operating conditions.

IV. Proposed Methodology Outline

This Section provides an overview of the necessary steps to construct the proposed data-driven model of battery state. The goal is to provide a discussion of the proposed methods and algorithms, and to ease reproducibility.

A. Assumptions on the Initial Dataset

The starting point of our approach is a dataset of battery conditions during operation: voltage, current, temperature, and relative time (necessary as embedding the temporal context essential for understanding the evolution of battery states), plus the corresponding SoC/E. Data collection frequency typically spans between 0.1Hz to 10Hz, to maintain sensitivity to battery internal dynamics. No assumption or constraint is necessary on the charge and discharge patterns, as sequences of full charge/discharge cycles are not mandatory if dataset entries are annotated with the corresponding values of SoH. Else, the dataset must include full charge/discharge cycles to allow the application methods like Coulomb Counting to integrate the current over a full discharge (or charge) cycle. Top of Figure 4 shows an example of data collected over time, both as table with one entry per instant (left) and as plot of one discharge cycle (right).

B. Dataset processing

The initial dataset is preprocessed thorough data segmentation and feature extraction, to allow effective battery state modeling. The result will be two datasets, one for SoC/SoE modeling and one for SoH modeling, to better target the effects and dependencies specific to each monitored aspect. In this section we do not discuss the cleaning steps, that are dataset-specific and thus will require dedicated discussion.

1) Data Segmentation: Even if working with full charge/discharge cycles is enabled from existing battery datasets, this prevents the possibility of performing any future retraining, as in the real life of the battery full charge/discharge cycles will be extremely rare, if available at all. To address this challenge, we introduce a segmentation strategy designed to closely represent the varying driving scenarios that EVs may encounter. The key idea is to divide the discharge cycles in the initial dataset into smaller segments, thereby training the models on partial cycles instead of relying solely on full cycles.

The dataset is thus segmented by dividing each discharge cycle throughout the battery's lifespan into non overlapping segments of varying ΔSoC (1%, 2%, 5%, 8%, 10%, 15%, 20%, 25%, 50%, and up to complete discharge, 100%-0%). For instance, when partitioning a full cycle with ΔSoC = 5%, segments become [100%-95%], [95%-90%], [90%-85%], and so on, until [5%-0%]. Similarly, ΔSoC = 25% yields segments [100%-75%], [75%-50%], etc. This allows to have more entries, that replicate portions of the initial discharge cycles to emulate the presence of partial cycles as well. This is exemplified in bottom left of Figure 4, where segments are portions of a full discharge cycle, partitioned every ΔSoC = 5%. The collection of such segments forms the resulting dataset for SoC and SoE models training.

2) Feature Extraction: The degradation rate of a battery does not depend only on quantities that can be measured instantaneously, but it is affected also by phenomena relative to the typical charge/discharge pattern, characteristics not of a specific instant, but rather of a whole discharge cycle. To model battery SoH, it is thus necessary to further elaborate the dataset to go beyond instantaneous measurements, by adding information that is relative to each charge/discharge cycle [34].

As a result of the former step, the initial dataset is split into portions, each representing a discharge cycle between variable values of SoC. Each of such portions is aggregated in only one row representing the whole portion as statistical and derived information, more aging-aware than instantaneous samples (bottom right of Figure 4).

The instantaneous values of each time instant composing the cycle are elaborated so that the new dataset contains the statistical descriptors of current, temperature, and SoC over the current discharge phase (i.e., mean, min, max, difference, range, standard deviation). This offers a comprehensive view over the cycle (bottom of Figure 4).

Then, cycle information is extended with the *discharge rate* over the cycle, defined as the rate of change of a battery state over time. Mathematically, it is expressed as:

$$\text{Discharge Rate} = \frac{1}{n}\sum_{i=1}^{n}\frac{\mathrm{X}_i - \mathrm{X}_{i-1}}{t_i - t_{i-1}} \tag{1}$$

where X_i and X_{i-1} represent state feature like SoC or energy (voltage multiplied by current) at consecutive time steps, t_i and t_{i+1} are the timestamps, and n denotes the total number of samples in the cycle. The discharge rate is particularly important because it quantifies how quickly battery state (i.e., SoC) changes over a consistent time interval. The discharge rate directly reflects the battery's ability to sustain energy delivery under stress. As the battery ages, its internal resistance increases, causing a steeper decline in SoC for the same load. This results in a higher discharge rate in the battery when it is aged compared to its nominal capacitance, thus reflecting battery degradation [35].

3) Resulting dataset processing algorithm: Algorithm 1 shows our approach. The starting point is the initial dataset D, and the outputs are two new datasets:

- $D_{\text{SoC/E}}$: result of the segmentation phase, it comprises segmented data specifically tailored to partial SoC/E transitions, enabling robust SoC/E modeling;
- D_{SoH}: feature-rich dataset keyed by cycle-level SoH for training advanced SoH estimation models, result of both the segmentation phase and the feature extraction process.

The algorithm illustrates how we isolate each discharge cycle and apply multiple segmentation thresholds, effectively partitioning long cycles into smaller segments suitable for diverse model requirements. For each discharge cycle, we apply multiple ΔSoC thresholds (e.g., 1%, 5%, 10%, etc., line 3), partitioning the cycle into smaller segments S_j. These segmented partial discharges are appended to $D_{\text{SoC/E}}$ (line 6). Each segmented portion S_j undergoes feature extraction, by generating one vector of statistical and derived features computation labeled with the SoH of the parent cycle C_i (lines 7-8). Accumulating the resulting feature vectors forms D_{SoH}, used to train and validate SoH estimation models (line 9).

C. Data-driven Battery State Estimation

In this study, we employed the Light Gradient Boosting Machine (LightGBM) algorithm, due to its capability to model

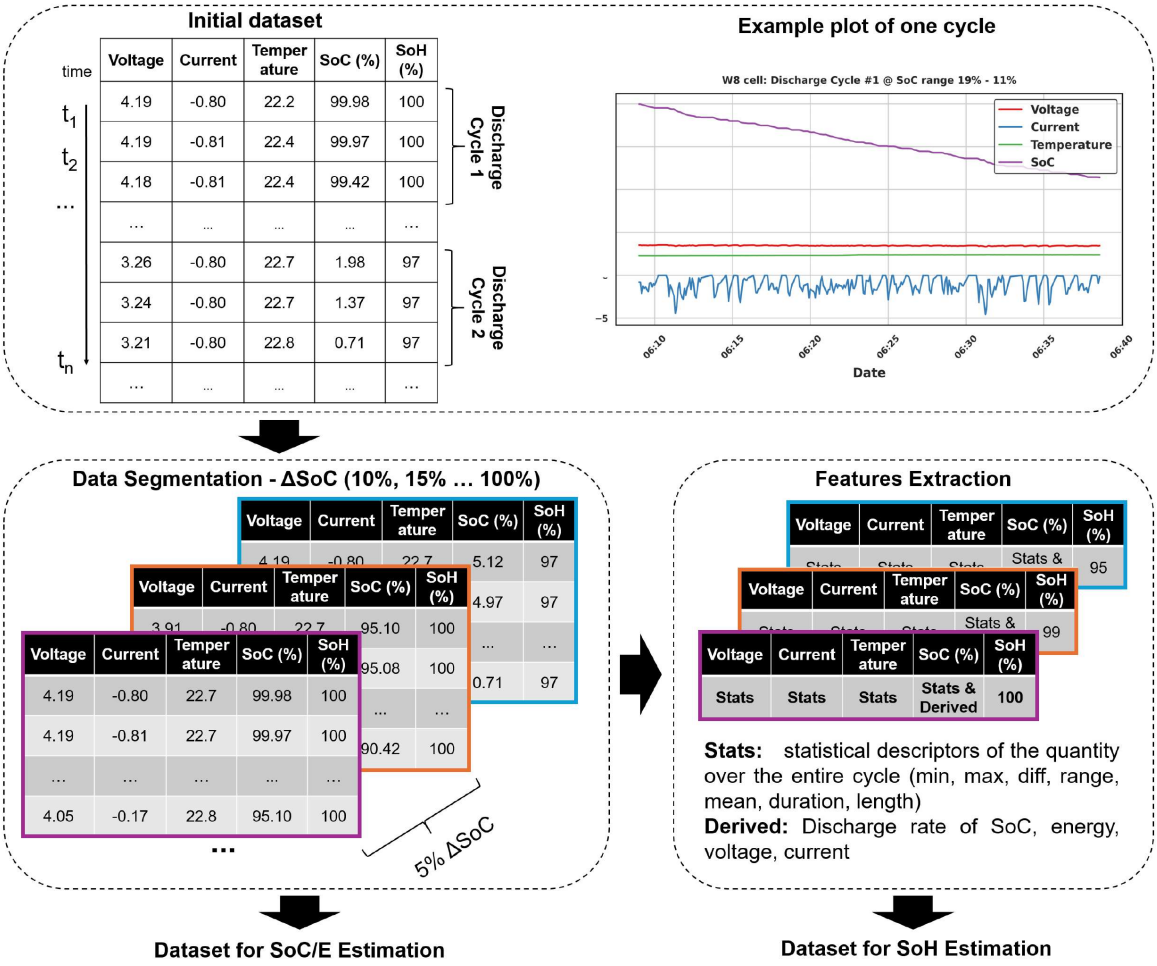

Figure 4: From initial dataset (top) to feature extraction (bottom). The initial dataset is reported as table and as plot of one discharge cycle (data from [32]). The discharge raw cycles are segmented based on the ΔSoC (in the Figure we show a few samples with ΔSoC=5%) and aggregated to train the SoC/E model (bottom left). Statistical and derived features are computed over each segment, forming the trained data for SoH (bottom right).

Algorithm 1 Integrated Pipeline for Segmentation and Feature Extraction

Input: Raw dataset $D = \{(V_t, I_t, T_t, \text{SoC}_t, \text{SoH}_t)\}_{t=1}^{N}$; Set of discharge cycles $\mathscr{C} = \{C_1, C_2, \ldots, C_k\}$; Segmentation thresholds $\Delta = \{\delta_1, \delta_2, \ldots\}$.

Output: $D_{\text{SoC/E}}$ (segmented data for SoC/E model); F_{SoH} (feature set for SoH model)

Initialize $D_{\text{SoC/E}} \leftarrow \varnothing$ and $D_{\text{SoH}} \leftarrow \varnothing$
for cycle $C_i \in \mathscr{C}$ **do**
  for $\delta \in \Delta$ **do**
    $\{S_1, S_2, \ldots, S_m\} \leftarrow \text{Segment}(C_i, \delta)$
    for segment S_j **do**
      $D_{\text{SoC/E}} \cup = S_j$
      $f_j \leftarrow \text{ExtractFeatures}(S_j)$
      $f_j.\text{SoH} \leftarrow \text{SoH}_i$
      $D_{\text{SoH}} \cup = f_j$
    end for
  end for
end for
return $D_{\text{SoC/E}}, F_{\text{SoH}}$

complex relationships within high-dimensional datasets efficiently. This makes LightGBM particularly suitable for battery health monitoring applications where real-time performance and accuracy are paramount [30]. Other models will be discussed in the experimental sections, to show the applicability of the proposed approach.

LightGBM is a gradient boosting framework that builds upon gradient boosting decision trees (GBDT). Unlike traditional GBDT methods that grow trees level-wise, LightGBM adopts a leaf-wise (best-first) growth strategy, selecting the leaf with the highest loss reduction to split. This approach allows LightGBM to achieve deeper trees with fewer iterations, leading to faster convergence and improved accuracy [36].

The objective of LightGBM is to find an optimal function $\hat{f}(x)$ that minimizes the empirical loss over the training dataset $\mathscr{D} = \{(x_i, y_i)\}_{i=1}^{N}$:

$$\hat{f} = \arg\min_{\hat{f}} \sum_{i=1}^{N} L(y_i, \hat{f}(x_i)) + \sum_{k=1}^{K} \Omega(f_k) \tag{2}$$

where:

- L is a differentiable convex loss function measuring the discrepancy between the predicted value $\hat{f}(x_i)$ and the true value y_i.
- $\Omega(f_k)$ is the regularization term that controls the complexity of each regression tree f_k, preventing overfitting.

LightGBM builds the model $\hat{f}(x)$ as an ensemble of K regression trees:

$$\hat{f}(x_i) = \sum_{k=1}^{K} f_k(x_i) \tag{3}$$

Each tree f_k is trained to minimize the residuals from the previous trees, refining the model iteratively. The regularization term Ω for each tree is defined as:

$$\Omega(f_k) = \gamma T + \frac{1}{2}\lambda \|w\|^2 \tag{4}$$

where:

- T is the number of leaves in the tree.
- w represents the leaf weights.
- γ and λ are regularization coefficients controlling the complexity and preventing overfitting.

The training objective for the k-th tree can be expressed as:

$$L^{(k)} = \sum_{i=1}^{N} L(y_i, \hat{f}^{(k-1)}(x_i) + f_k(x_i)) + \Omega(f_k) \tag{5}$$

where $\hat{f}^{(k-1)}(x_i)$ is the prediction from the ensemble of the first $k-1$ trees. LightGBM employs a greedy algorithm to find the best split at each node by maximizing the loss reduction, efficiently handling large datasets through its innovative sampling and bundling techniques.

The leaf-wise growth strategy and efficient tree structures of LightGBM facilitate rapid inference times, allowing real-time SoH predictions essential for dynamic battery management. Additionally, given our problem's dynamic nature with periodic model updates, we require models capable of adapting to online data. The LightGBM framework stands out by enabling seamless retraining and updates on streaming data, making LightGBM a superior choice for our evolving needs. Finally, empirical evaluations demonstrated that the final LightGBM models achieved high accuracy in SoH estimation with minimal computational overhead, confirming their practicality for real-time BMS applications [37], [38].

V. Datasets and Dataset Preparation

Accurate battery State-of-Health (SoH) estimation relies heavily on the quality and comprehensiveness of the underlying datasets. In this study, we utilized three distinct datasets to ensure robustness and general applicability of our models: the NASA Randomized Battery Usage dataset [11], the Sandia National Laboratories (SNL) Battery dataset [12], and the Stanford Urban Dynamometer Driving Schedule (UDDS) Battery dataset [32]. Each dataset offers unique insights and challenges, reflecting different chemistry, operational conditions and data collection methodologies. Additionally, while the former are built from lab measurements, the latter is derived from real driving cycles, thus highlighting the applicability of the proposed solution. The intended scope of this paper is not an exhaustive exploration of the entire digital twin architecture integrated with BMS but rather a discussion of the effectiveness of the SoC/E-SoH model organization within the realm of constructing a battery digital twin.

A. Datasets description

1) NASA Randomized Battery Usage Dataset: The NASA Randomized Battery Usage dataset [11] is a widely recognized benchmark in battery health monitoring research. The dataset includes detailed operational data from 28 LCO 18650 batteries, each possessing a nominal capacity of 2.1Ah and a rated voltage of 4.2V. These batteries were subjected to a series of charge and discharge cycles under varying temperature conditions to simulate real-world usage patterns through a method known as *random-walk* operations. In this process, each cell was fully charged to 4.2V and then discharged to 3.2V, following randomized sequences of current levels. The duration of these cycles varied unpredictably, embodying the characteristics of a random walk.

The dataset meticulously records key parameters, including current, voltage, temperature, and time stamps for each charging and discharging event, sampled at 1Hz. Periodic reference charge and discharge cycles were conducted to establish baseline measurements essential for calculating the total available capacity and determining the SoH at different stages of the batteries' life cycles. The 28 battery cells are organized into seven distinct subgroups, each characterized by unique combinations of random-walk cycle features and reference charge/discharge parameters. For instance:

- Subgroup RW11 involves random walks with 5-minute intervals and reference discharges at 2A;
- Subgroup RW23 features random walks with 1-minute intervals and reference discharges at 1A;
- Subgroups RW1, RW2, RW7 and RW8 cycle through random walk discharging with 0.5A–4A at 3.2V, with random charging durations from 0.5 to 3 hours long. Every 50 cycles, reference tests battery health;
- Subgroup RW13, RW14, RW15 and RW16 feature charge cycles to 4.2V and discharge cycles to 3.2V using current values between 0.5A and 5A, randomly changed every 5 minutes.

Such diverse conditions across the subgroups provide a rich dataset ideal for analyzing battery performance under a wide range of operational scenarios and effectively simulates real-world driving patterns.

2) Sandia National Laboratories (SNL) Battery Dataset: The second dataset utilized in our study is sourced from Sandia National Laboratories [12]. This dataset provides extensive empirical data on lithium-ion batteries, encompassing three distinct types of cells: Lithium Iron Phosphate (LFP), Nickel Manganese Cobalt (NMC), and Lithium Nickel Cobalt Aluminum Oxide (NCA).

The dataset includes in-cycle measurements such as date-time, voltage, current, energy, and temperature, along with per-cycle metrics like charged/discharged capacity, sequential cycle numbering, and a notation of the cycle state (either charge *C*, discharge *D*, or rest *R*), sampled at 0.1Hz. For the purpose of our study, we primarily focus on charge and discharge cycles,

as the rest periods typically involve minimal activity and hence have a negligible impact on the observed parameters.

Each battery cell in the dataset undergoes various operational scenarios, reflecting a broad spectrum of conditions. These include discharge currents of 0.5C, 1C, 2C and 3C, temperatures of 15°C, 25°C and 35°C, and discharge depth (DoD) ranges from 0–100%, 20–80%, and 40–60%. After completing a predefined number of operational cycles (either 500 or 1,000 cycles), the dataset includes a capacity check conducted through three full reference charge and discharge cycles. These reference cycles (RCs) are performed at a fixed current of 0.5C and serve as synchronization points, allowing for accurate measurement of the SoH of the battery. The SoH is extrapolated from the time required to charge the cell, serving as an indicator of capacity degradation over time

3) Stanford UDDS Battery Dataset: The Stanford Urban Dynamometer Driving Schedule (UDDS) Battery dataset provides empirical data from 10 INR21700-M50T NMC battery cells subjected to real-driving patterns and varying charging rates over a period of 28 months [32]. It allows thus to validate the proposed solution over real driving conditions, rather than on a laboratory setup.

The Stanford UDDS dataset reproduces realistic electric vehicle (EV) usage patterns, including discharge cycles transition of battery SoC from 80% to 20% (no full charge/discharge cycles) and Constant Current (CC)-Constant Voltage (CV) charging at rates ranging from C/4 to 3C. The dataset features also periodic capacity tests, used to evaluate cell degradation.

The dataset includes comprehensive measurements such as voltage, current, temperature, and energy (collected at 1Hz), along with diagnostic capacity tests to keep tracks of the SoH value and cycling data contains the UDDS (collected at 10Hz). Given its real-world nature, this dataset has some issues due to challenges encountered during data collection: data is missing for some cells (W4, W5, W7, and G1), some cycling data exhibit irregularities, such as temperature readings exceeding 400°C and abnormal voltage profiles, and some irregularities in the resting times between diagnostic cycles.

B. Data Preparation and Sampling Frequency

This Section discusses the specific challenges of each dataset and the preparation work that was necessary to apply the proposed approach.

1) Data Selection: As an initial pre-processing step, we selected suitable cells from all three datasets by excluding those with excessive missing samples. This selection criterion ensures a robust evaluation of model performance, particularly in critical battery states. For the NASA Dataset, out of the 28 available battery cells, 24 were retained after excluding four cells due to significant irregularities in measurements. In the Sandia Dataset, focus was placed on Nickel Manganese Cobalt (NMC) type cells, comprising 31 cells subjected to diverse operational conditions. Finally, we considered all INR21700-M50T NMC cells of the Stanford UDDS Dataset, with further later exclusions based on data integrity issues, as will be outlined in Section V-A3.

2) Reference Discharge Cycle Extraction: None of the datasets explicitly provides SoH annotations for each measurement. Therefore, we estimated SoH exclusively during reference discharge cycles for Sandia and NASA, and during the cycling data used for by capacity check for the Stanford Dataset.

Within each reference discharge cycle, we conduct the Coulomb Counting method to calculate the residual capacity corresponding to each time step in the discharge phase and to compute the total available discharge capacity at the end of the discharge cycle. Then we convert the residual capacity during one reference discharge to the corresponding value of SoC and SoE (which integrate the product of the voltage and the current with time), and use each reference discharge total available capacity to derive the SoH.

It is important to note that this step is necessary due to the limitations of the datasets: since they do not provide SoH annotations for each measurement, we can estimate SoH only during the reference discharge cycles. If the datasets had included SoH for every measurement, our methodology could have utilized all cycles. Notice that the SoH value in one discharge cycle remains constant; As it only updates from reference discharge cycle to reference discharge cycle.

3) Data Cleaning: Upon calculating SoH values for each reference discharge cycle, we identified and excluded irregularities that deviated from expected monotonic degradation trends. These anomalies were likely caused by measurement errors or equipment malfunctions and could compromise model accuracy.

Specific data cleaning actions included exclusion of anomalous cycles. For instance, in the NASA RW9 cell, 9 out of 79 reference discharge cycles exhibited irregular behavior as significant differences in SoH values in consecutive cycles did not align with the expected degradation trajectory; Such cycles were subsequently removed. Discharge cycles with temperature measurements exceeding 400°C were either removed or replaced with the mean temperature of the corresponding cycle. Such extreme temperature readings were deemed implausible and indicative of sensor malfunctions or data recording errors. Finally, we noted some issues in the Stanford UDDS Dataset, where variable resting times between cycles were noted, particularly between Diagnostic Tests 5 and 8 for cells W4, W5, W7, W9, W10, and between Diagnostic Tests 3 and 4 for cells G1, V4, and V5. These irregularities were accounted for during data segmentation to maintain cycle integrity, ensuring that each discharge cycle was processed as a distinct and complete operational event. The last challenge was due to the presence of different data sampling, that was fixed by down-sampling the Stanford Dataset to 0.2Hz.

These data cleaning steps were essential to mitigate the impact of misleading measurements and ensure that the dataset accurately reflected the true degradation patterns of the battery cells under various operational conditions.

4) Degradation Analysis: Having calculated battery state over time, we could perform a preliminary analysis on battery degradation reproduced by each dataset. The degradation observed in the Stanford dataset is less pronounced compared to the Sandia and NASA datasets. Specifically, the minimum

degradation was observed in cell W8, which retained 91% of its capacity after aging, while NASA degraded to approximately 40% and Sandia to approximately 80%. This variability is particularly significant for NASA and Sandia, as most non-stationary applications, especially in EVs, typically exhibit degradation around 80%.

5) Data Splitting: To evaluate model performance comprehensively, the cleaned datasets were partitioned into training, validation, and testing subsets:

- The NASA Dataset was partitioned into 12 cells for training (RW3, RW6, RW8, RW10, RW11, RW15, RW17, RW18, RW19, RW23, RW26, RW28), 7 cells for validation (RW1, RW5, RW7, RW12, RW13, RW24, RW27), and 4 cells for testing (RW2, RW9, RW14, RW16);
- The Sandia Dataset was partitioned into 17 NMC cells for training, 7 for validation, and 6 for testing (including one unseen operation condition, i.e., discharge cycle at 2C performed at 35°C);
- The Stanford Dataset diagnostic tests were partitioned into 5 cycles for training (W3, W4, V5, W7, W8), and 3 for validation (V9, W10, G1), while for testing we used 2 diagnostic tests (W9, W5) plus the ordinary urban driving data traces.

The splitting strategy ensures that models are trained on a diverse range of operational conditions while being validated and tested on unseen data to assess generalizability.

VI. Experimental Results

A. Performance Metrics

We employed key performance metrics to quantify accuracy and robustness of predicted values ($\hat{y}$) over actual ones (y). The Mean Absolute Error (MAE) provides a straightforward indication of the average absolute difference between $\hat{y}$ and y. The Root Mean Squared Error (RMSE) offers a comprehensive evaluation by penalizing larger errors more heavily. Finally, the coefficient of determination (R^2 score) was utilized to gauge the proportion of variance in the target variable explained by the model. These metrics collectively offer a comprehensive understanding of the predictive capabilities of the models, emphasizing not only the average accuracy but also the extent and distribution of prediction errors.

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \qquad \text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

All experiments were conducted on a machine equipped with AMD processor with 16 GB RAM with 3.4 GHz, and no GPU was utilized. These specifications facilitated rapid hyperparameter optimization and model training, although the final models are lightweight enough for deployment on standard BMS hardware.

B. SoH Estimation

The first constructed model is the model of SoH, trained and optimized by using Optuna [39]. The result on the validation set demonstrates good performance, with overall MAE over training and validation sets reported in Table I for the three datasets.

Table I: MAE for SoH Model Estimation Error with LightGBM and SVR

Dataset	Validation set		Test set	
	LightGBM	SVR	LightGBM	SVR
NASA	1.35	1.52	0.88	1.23
Sandia	0.75	0.58	0.73	0.99
Stanford	0.52	0.38	0.19	0.17

To perform a generality analysis, we calculated the performance indicators on different segment sizes (i.e., ΔSoC), and the results are reported in Table II. Figure 5 visually illustrates the SoH results in the test set for the three datasets. The graphs showcase model precision in tracking the real SoH of the battery, even in segments with ΔSoC = 1%. This is a testament to the model's resilience and efficiency in capturing variations in battery health, even when the available data is limited. The high and stable R^2 across all datasets shows that our models consistently capture the full fluctuation of SoH, regardless of whether the degradation span is large or small. In contrast, MAE and MSE reflect the magnitude of prediction deviations in absolute or squared terms, respectively. For instance, although Sandia's MAE might appear lower than NASA's, both datasets still exhibit high R^2 values because the model effectively describes SoH variability in each scenario.

To discuss the applicability of the proposed strategy to other data-driven models, we compare the performance of LightGBM with Support Vector Regression (SVR). SVR is an extension of Support Vector Machines that aims at finding a function that deviates from the actual target values by a value no greater than a specified threshold ε for all training data points. SVR is particularly effective in capturing complex, non-linear patterns through the use of kernel functions, enabling it to perform well even with limited training data [40]. Its robustness and flexibility make it a suitable alternative for SoH prediction in battery management applications where accuracy and reliability are critical. Tables I and II and Figure 5 show that SVR has performance comparable to Light GBM, with higher accuracy on the Stanford dataset and lower accuracy on the Sandia and NASA datasets.

C. SoC and SoE Estimation

We employed a multi-regression model developed using Scikit-Learn [41], incorporating a LightGBM model to estimate both SoC and SoE. The key objectives of this analysis were as follows:

- Assessing the effectiveness of an updatable model compared to a fixed model for SoC/E estimation under varying levels of battery aging (SoH);
- Analyzing the impact of different update intervals on estimation performance;

Table II: SoH Model Estimation Error for LightGBM and SVR

Model	ΔSoC of Segments	NASA			Sandia			Stanford		
		MAE (%)	MSE (%)	R2	MAE (%)	MSE (%)	R2	MAE (%)	MSE (%)	R2
LightGBM	1%	0.91	2.21	0.99	0.62	0.78	0.98	0.258	0.104	0.984
	2%	0.89	2.08	0.99	0.70	0.86	0.98	0.250	0.096	0.985
	5%	0.88	1.97	0.99	0.73	0.87	0.98	0.256	0.103	0.984
	10%	0.86	1.91	0.99	0.75	0.93	0.98	0.242	0.097	0.985
	20%	0.88	1.93	0.99	0.71	0.83	0.98	0.239	0.095	0.985
	25%	0.90	1.98	0.99	0.73	0.87	0.98	0.237	0.093	0.986
	50%	0.88	1.91	0.99	0.68	0.78	0.98	0.242	0.095	0.985
	100%	0.82	1.51	0.99	0.72	0.76	0.98	0.236	0.090	0.986
SVR	1%	1.58	4.83	0.98	1.22	6.27	0.86	0.177	0.067	0.990
	2%	1.75	5.57	0.98	1.20	6.07	0.86	0.176	0.066	0.990
	5%	1.65	4.30	0.98	1.16	5.23	0.88	0.176	0.066	0.990
	10%	1.55	3.75	0.98	1.16	4.60	0.89	0.175	0.064	0.990
	20%	1.46	3.42	0.98	0.81	1.06	0.98	0.174	0.062	0.991
	25%	1.55	3.87	0.98	0.78	0.92	0.98	0.171	0.061	0.991
	50%	1.48	3.74	0.98	0.76	0.88	0.98	0.170	0.055	0.992
	100%	1.62	4.58	0.98	0.88	1.28	0.97	0.148	0.047	0.993

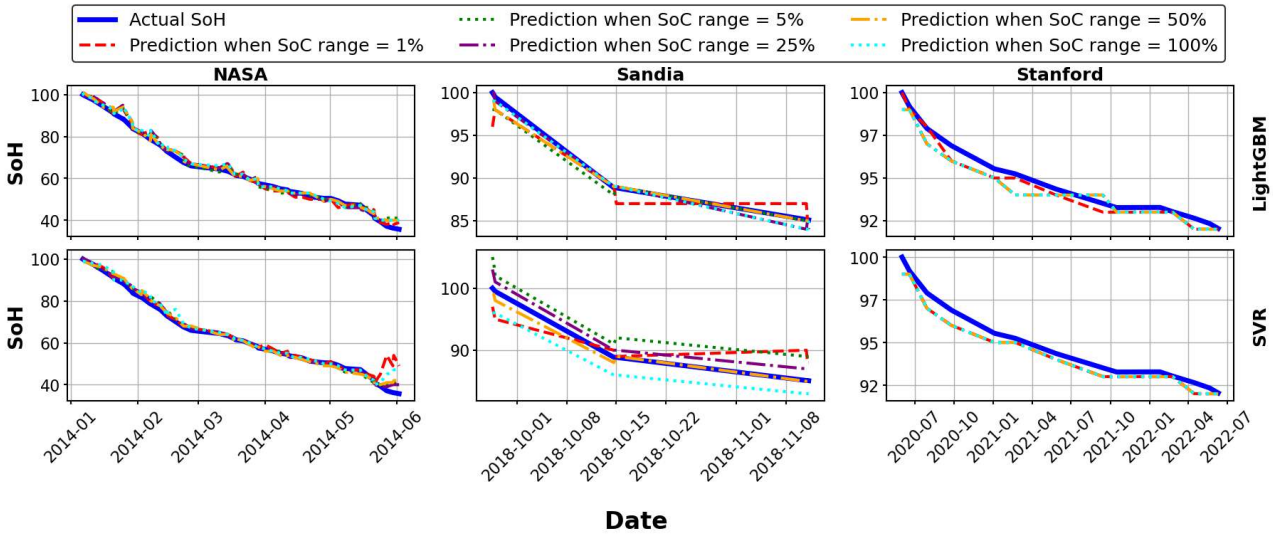

Figure 5: SoH estimation under different segmentation with LightGBM and SVR

.

- Investigating the model's ability to accurately estimate SoC/E in partial charging and discharging scenarios.

The primary aim was to develop a model that adapts to changing battery conditions and maintains accurate SoC/E predictions, even when batteries undergo partial charge and discharge cycles.

For each dataset, we considered different partial SoC/E sequences within the test set to represent various discharging scenarios. We then trained two types of models:

- Fixed Model: a static model trained once using the initial training dataset, with no further updates;
- Updatable Models: dynamic models that are retrained based on two different update strategies:
  - Time Interval-Based Updating: the model was retrained at regular intervals, such as one day, one week, two weeks, one month, and two months.
  - Degradation Level-Based Updating (ΔSoH): the model was retrained when the SoH degraded by certain percentages, specifically 1%, 5%, 10%, 15%, and 20%.

These strategies allowed us to evaluate the impact of model retraining and the effectiveness of different retraining activation criteria on SoC/E estimation accuracy. However, we encountered challenges with missing data for some intervals, particularly in the time interval-based updating strategy. For instance, the 1-day interval had approximately 86% missing values. This indicates that, when extracting daily data, certain days lacked recorded measurements. For example, if the SoH was recorded as 100% on one day, data might be missing for the following days, only reappearing when the SoH had dropped to 97%. This issue persisted across different intervals, including one week and 1% ΔSoH drops.

In Table III, each row represents a different discharge segment in the test set, and each column represents the performance of the updated model relative to the fixed

Table III: Comparison of Testing MAE between Fixed Model and Model Updating

	SoC/E Segment(%)	Fixed (MAE%)	SoC/E Model Update ($\frac{(MAE_{fix}-MAE_{update})}{MAE_{fix}}$ * 100%)									
			Interval					SoH drops				
			1 day	1 week	2 weeks	1 month	2 months	1%	5%	10%	15%	20%
NASA	100% - 0%	1.240	21.95	17.83	16.86	13.15	8.61	21.35	18.72	15.93	16.25	14.26
	62% - 25%	15.610	59.95	58.10	55.35	46.63	39.72	58.41	49.65	42.65	35.91	31.60
	36% - 8%	26.327	81.09	79.01	75.63	66.04	54.79	80.12	72.88	66.22	56.35	51.96
	53% - 2%	18.412	67.41	64.30	61.39	53.18	46.10	65.26	56.86	50.57	43.43	39.09
	74% - 27%	10.997	49.58	47.79	46.03	38.48	34.79	46.99	37.50	31.38	26.98	24.91
	80% - 69%	10.254	52.46	50.48	48.46	42.26	38.93	49.28	40.36	34.65	30.90	30.06
	35% - 17%	26.661	84.01	81.63	77.40	65.95	53.32	83.35	75.83	67.53	57.13	49.92
	96% - 12%	2.151	23.60	20.32	17.33	11.94	5.01	23.10	17.50	13.50	13.73	12.44
Sandia	100% - 0%	12.442	45.30	40.64	40.64	34.86	20.92	46.79	40.96	32.20	27.15	04.40
	62% - 25%	6.212	43.16	38.52	38.52	35.47	25.77	31.91	25.01	18.21	19.06	01.37
	36% - 8%	7.164	14.56	10.70	10.70	12.95	11.87	19.94	14.42	12.21	10.15	5.62
	53% - 2%	12.401	46.22	35.96	35.96	33.95	23.03	44.13	37.24	29.88	23.39	4.26
	74% - 27%	05.634	35.79	31.75	31.75	28.75	20.64	27.09	19.47	12.68	14.75	0.26
	80% - 69%	4.037	16.22	16.31	16.31	16.20	12.21	11.13	9.76	7.62	3.18	-1.81
	35% - 17%	5.351	8.11	6.15	6.15	10.26	11.81	9.45	5.48	5.90	7.55	7.16
	96% - 12%	8.240	32.63	28.16	28.16	24.79	17.50	33.35	27.90	22.14	17.11	2.58
Stanford	UDDS cycle (80%-20%)	11.22	48.64	51.03	52.95	52.95	53.01	38.37	N/A	N/A	N/A	N/A

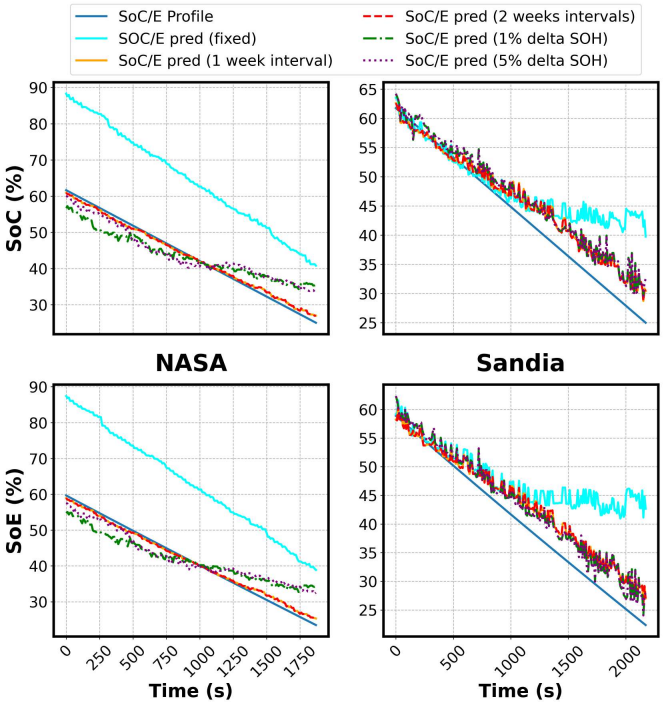

Figure 6: Adaptable SoC/E model comparison under different intervals/%ΔSOH drops level.

model for different updated scenarios. The values indicate the percentage improvement in the performance of the updatable model (MAE_{update}) compared to the MAE of the fixed model (MAE_{fix}):

$$\frac{MAE_{fix} - MAE_{update}}{MAE_{fix}} \times 100$$

For NASA, the updatable model showed a significant improvement over the fixed model across all partial SoC/E scenarios. The improvements ranged from 8.61% to 84.06% for the test set. For instance, in the SoC segment 36-8%, the updatable model improved by 75.63% after two weeks and 80.12% after a 5% ΔSoH drops. For Sandia, the updatable model demonstrated notable improvements over the fixed model, with improvements ranging from 0.25% to 46.22%. For the SoC segment 53-2%, the updatable model improved by 35.96% after two weeks and 44.13% after a 5% ΔSoH drops. The study also demonstrated that updating the model after longer intervals (such as two months or 20% ΔSoH drop) showed less improvement in general: this indicates that the longer the interval between updates, the closer the results were to the fixed one (with the same data complexity). This is likely due to the increased complexity of the time series data, as the model needs to learn all the SoH levels over two months. To visualize and give more focus on the performance of the SoC model, Figure 6 illustrates the results for the SoC segment 62-25%, showing the real SoC/E values against the fixed model and updatable models with one-week and two-week intervals and 1% and 5% ΔSoH drops. For this segment, the updatable model after two weeks and a 5% ΔSoH drops outperformed the fixed model by approximately 55.35% and 49.65% for the NASA dataset, and 38.52% and 25.01% for the Sandia dataset. The SoH levels in this segment were 67% for NASA and 83% for Sandia.

D. SoC and SoE Estimation over the Stanford Dataset

To evaluate performance under realistic driving profiles, we trained the SoC and SoE models on the Stanford Dataset, and we tested them on the ordinary driving data traces subset. This constitutes a more challenging environment, with frequent partial cycles that reflect normal EV operation.

First of all, we applied the same analysis as for the other datasets with the LightGBM model, as reported in Table III. Note that it was possible to evaluate model updates only with timed updates or every 1% SoH drop, due to the limited degradation level of the dataset. Overall, the performance improvement is significant, ranging from 38.37% (with updates every 1% SoH drop) to values consistently around 50% for the timed updates, with an overall MAE of 0.62%.

To further discuss the generality of the approach, we evaluated the behavior of SoC/E models built by using the Passive-Aggressive (PA) algorithm. PA is an online learning method designed to handle large or sequentially arriving datasets effectively [42]. Unlike conventional batch learning algorithms that require retraining from scratch upon receiving new data, PA algorithms update their model parameters only if the current sample prediction incurs a loss above a predefined threshold. This selective update strategy allows the algorithm to adapt quickly to new information without suffering excessive computational overhead. PA is promising and outperforms the fixed algorithm, with improvements of up to 14% (in detail: 8.02% with daily updates, 14.36% with weekly updates, 13.17% with bi-weekly updates, 11.45% with monthly updates, and 13.46% with bi-monthly updates). To elaborate more, we focused on an example cell W5, operating from 80%–20% SoC, and on a two-weeks long simulation with daily updates. PA outperformed the fixed model in 85.71% of cases, with mean improvement of 68.95% and a maximum improvement of 94.83%. LightGBM performed even better, surpassing the fixed model in 13 out of 14 updates (i.e., 92.86% of the cases), with improvements between 3.47% and 94.83%.

The results suggest the importance of regularly updating battery models to maintain accuracy and reliability, especially in scenarios involving partial charging or discharging with battery aging.

E. Computational Complexity and Memory Overhead

This Section briefly discusses the computational requirements for training and inference when applying our approach to the three datasets using two models for SoH estimation (LightGBM and SVR) and two models for SoC/E updates (LightGBM and PA). We focus on two main aspects: (i) training and inference time, and (ii) memory usage, to approximate memory footprint of the model during training and inference, including space for storing intermediate data structures and learned parameters. The most critical dimensions in our opinion are model buffer size and inference time, that affect real-time behavior.

Table IV summarizes these metrics for each model-dataset combination. Training time is influenced by both the model's intrinsic complexity and the dataset size (i.e., number of partial discharge segments, length of each segment, and feature dimension). LightGBM commonly benefits from efficient parallel tree-boosting, enabling relatively fast training times even for large or repeated datasets (as in the NASA and Stanford cases for SoH estimation). By contrast, SVR can become computationally more demanding in terms of both memory and training duration, particularly when the dataset includes multiple updates or high-dimensional feature spaces.

Regarding SoC/E, the Sandia dataset demonstrates a reduced overhead due to its smaller dataset following data cleaning and cycle removal; Training or updating the model thus proceeds more quickly with fewer valid segments. PA training time can vary widely depending on hyperparameters (e.g., network depth, optimizer settings), yet remains relatively modest in these experiments. Inference overhead likewise remains low across most configurations, reflecting the suitability of these models for online or near-real-time battery management applications.

Table IV: Computational Overhead for SoH and SoC/E Models Across Three Datasets

Metrics	SoH Models		SoC/E Models	
	LightGBM	**SVR**	**LightGBM**	**PA**
NASA				
Training (s)	0.522	281.116	10.732	3.407
Inference + FE (s)	2.217	2.145	0.008	0.003
Memory (MB)	0.010	10.237	5.775	18.590
Sandia				
Training (s)	0.196	178.460	171.554	5.215
Inference + FE (s)	0.492	0.574	0.001	10^{-5}
Memory (MB)	10.237	8.276	10^{-7}	10^{-7}
Stanford				
Training (s)	0.264	13.770	47.504	23.579
Inference + FE (s)	0.924	0.927	0.009	0.002
Memory (MB)	0.125	0.101	0.150	0.064

F. Comparative analysis with the state-of-the-art methods

Table V presents a consolidated comparison between our proposed methodology and several state-of-the-art techniques for battery state estimations using the NASA and Sandia battery datasets. The table encompasses multiple dimensions to provide a perspective on each method's strengths and limitations.

For the NASA dataset, prior methods such as [43] and [44] demonstrate moderate RMSE values between 1.36% and 1.91%, and MEA between 1.088% and 1.528%. Although these techniques employ innovative time-series embeddings and voltage variance health indicators, they do not emphasize online or frequent model updating under partial discharges. In contrast, our proposed framework achieves a competitive MAE of 0.88% and R^2 of 0.99 using LightGBM, still maintaining an inference time of 2.217 seconds and a memory usage of 10.237 MB. This balance between accuracy and computational cost underscores its suitability for dynamic EV scenarios where continuous updates are crucial.

For the Sandia dataset, the Table shows that existing works, including [45] and [46], achieve MAE values ranging from 1.79% to 6.00%. Our proposed-based solution reports an MAE of 0.736% with R^2 of 0.984, outperforming these benchmarks by a substantial margin while retaining minimal inference time and low memory usage.

None of the referenced SoTA methods incorporate a robust segmentation framework coupled with digital twin updates to track both SoH and SoC/E across partial cycles. Consequently, the comparative results illustrate that our strategy effectively merges accuracy with computational feasibility, facilitating real-time model adaptation and broader applicability in EV battery management.

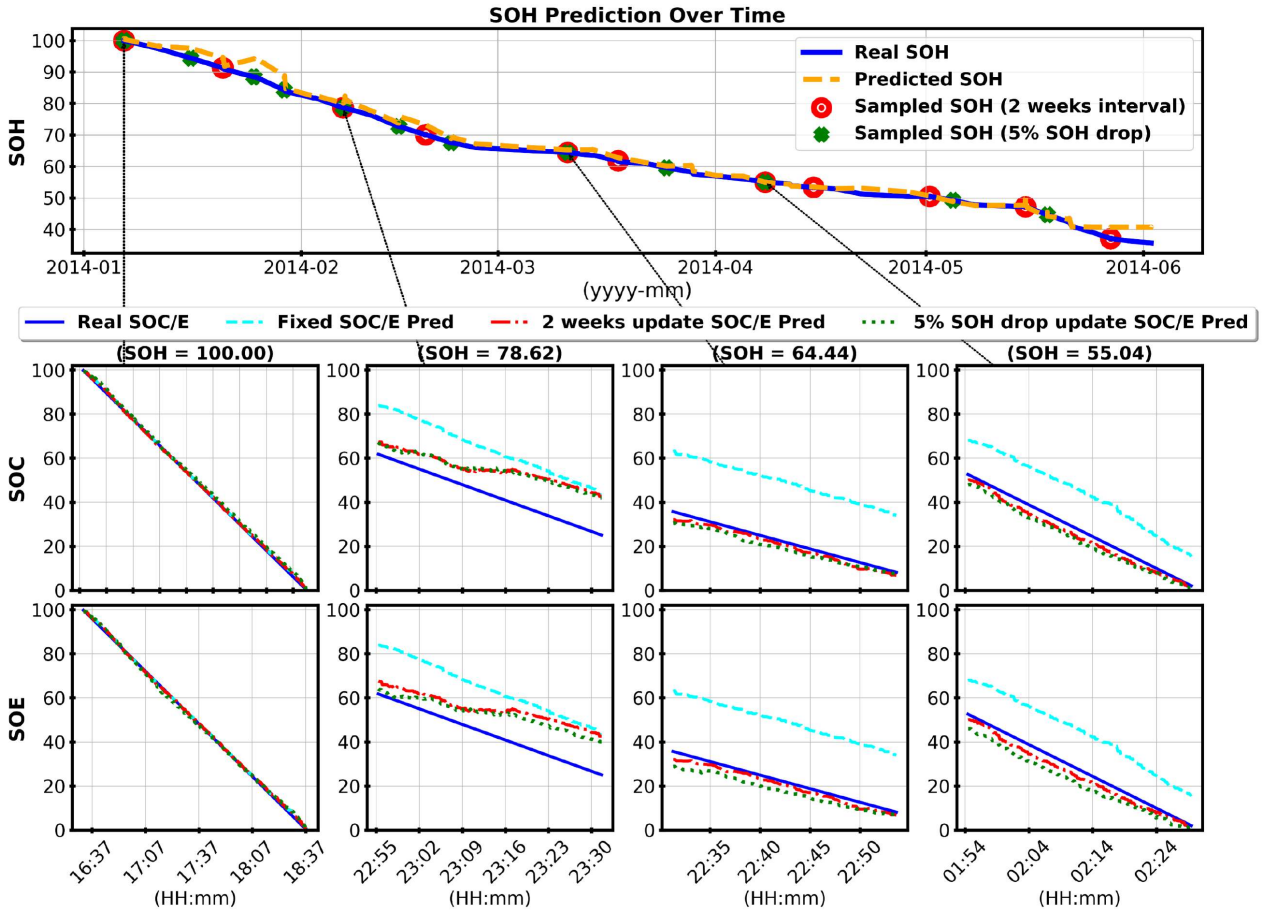

Figure 7: Digital Twin Proof of Concept.

Table V: Comparison with State-of-the-Art Techniques for SoH Estimation on NASA and Sandia Datasets

Dataset	Methodology	Metrics
NASA	[43]: biGRU + SKI algorithms; focus on SoH estimation, no SoC/E estimation; Time-series embedding with graph representation	RMSE 1.91% MAE 1.528%
	[44]: Hierarchical Ensemble ELM; Focus on SoH estimation, no SoC/E estimation; Health indicator: Voltage Variance during Equal Time Interval (VVETI)	RMSE 1.36% MAE 1.088%
	Our Proposed Method: Variable discharge cycles, dynamic update of SoC/E; Segmentation and time-series feature extraction, discharge rate	MAE 0.88% R^2 0.99
Sandia	[45]: SVR, KNN, RF alghoritms; focus on SoH, no model update; Uses direct measurable data (V, I, T)	MAE 1.79%
	[46]: SVR, MLR, ANN algorithms; focus on SoH, no model update; Uses direct measurable data	MAE 6.00%
	Our Proposed Method: Variable discharge cycles, dynamic update of SoC/E; Segmentation and time-series feature extraction, discharge rate	MAE 0.736% R^2 0.984

VII. Digital Twin Proof-of-Concept

Figure 7 presents a proof of concept for the digital twin framework outlined in the methodology section, demonstrating its application to a specific test cell within the NASA dataset. This example illustrates the behavior and advantages of the digital twin architecture described in Section III.

The process begins by establishing foundational models for SoH and SoC/E using historical data from battery cells of the same type but subjected to different operating conditions. These initial models serve as reference points for estimating SoH across various battery SoC segments. The dynamic digital twin process is then initiated by transferring raw data from the vehicle's BMS to the cloud after each discharge cycle. This data undergoes pre-processing and feature extraction, as detailed in the feature extraction subsection, enabling the cloud-based model to perform SoH predictions across different segmentation and driving-cycle patterns.

The segmentation of the SoC evolves over successive months, adapting to real-world usage patterns. As depicted in the first row of Figure 7, the segmentation starts with a full cycle (January-February) covering the entire range (100%-0%), followed by partial cycles in subsequent months: (February-March): (62%-25%), (March-April): (26%-8%), and (April-May): (53%-2%). This approach allows the model to accommodate diverse driving scenarios and battery usage conditions.

The orange dashed line represents the SoH prediction, while the blue line shows the actual SoH values. Concurrently, an updatable SoC/E model is deployed to the edge, using a two-week interval and a 5% ΔSoH drop strategy for model updates.

The first plot in the second row illustrates the initial SoC/E predictions, made before any new data is available. As new data is collected from the vehicle over two-week intervals or 5% ΔSoH drops, the model undergoes retraining on the cloud using this fresh data, which is also utilized by the SoH model to refine SoH predictions. The updated SoC/E model is then transferred back to the edge, replacing the old model for continued SoC/E estimation (circles).

Each predicted SoC/E value from the updatable model is compared against both the actual SoC/E values (solid blue line) and the estimates from a fixed model trained solely on the initial training data (light blue dashed lines). This comparative analysis highlights the superior performance of the updatable model in accurately tracking battery states over time. The results underscore the pivotal role of digital twin technology in enhancing and extending the functionality of the BMS, demonstrating its potential to provide more accurate and adaptive battery management in real-world applications.

VIII. Conclusions

In this work, we addressed a critical gap in the current literature by introducing a dynamic, adaptive model that accurately estimates the SoC, SoE, and SoH of EV batteries across various driving cycles. Our approach effectively bridges the limitations of existing methodologies, which often fail to consider the complex interplay between different driving behaviors and their impact on battery states.

The key strength of our methodology lies in its comprehensive approach to battery management. By dynamically adjusting segmentation points based on real-time battery states and driving patterns, our model captures the nonlinear relationships between battery parameters more effectively. This has been validated on three public datasets, including real driving conditions and comparing different data driven models. Overall, our method could achieve MAE lower than 1% on all datasets (0.87% for Nasa, 0.73% for Sandia, and 0.24% for Stanford) and improved SoC/E estimation by overall mean MAE of 44.14%, 21.13% and 50% on the three dataset using different updating mechanism. This proves that our adaptive strategy ensures enhanced generalization and robustness across diverse operational conditions. Additionally, our framework incorporates proactive model updates triggered by predefined degradation thresholds, maintaining accuracy and reliability even under continuous operation. This approach is crucial for optimizing battery performance and extending its lifespan in real-world applications.

For future work, exploring edge computing capabilities for on-device model inference can reduce dependence on cloud resources, thereby improving system responsiveness and reliability in varied operational conditions. Furthermore, implementing advanced cybersecurity measures, such as federated learning, will be vital for ensuring data integrity and privacy. This will be especially important for scalable deployment in commercial EV fleets, ensuring secure and efficient battery management.

References

[1] S. Bilgen, "Structure and environmental impact of global energy consumption," *Renewable and Sustainable Energy Reviews*, vol. 38, pp. 890–902, 2014.

[2] A. C. R. Teixeira and J. R. Sodré, "Impacts of replacement of engine powered vehicles by electric vehicles on energy consumption and CO2 emissions," *Transportation Research Part D: Transport and Environment*, vol. 59, pp. 375–384, 2018.

[3] IEA, "Electricity 2024 - executive summary," https://www.iea.org/reports/electricity-2024/executive-summary, 2024.

[4] 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 Industrial Electronics Magazine*, vol. 7, no. 2, pp. 4–16, 2013.

[5] M. Kumar, K. P. Panda, R. T. Naayagi, R. Thakur, and G. Panda, "Comprehensive review of electric vehicle technology and its impacts: Detailed investigation of charging infrastructure, power management, and control techniques," *Applied Sciences*, vol. 13, no. 15, p. 8919, 2023.

[6] F. Naseri, S. Gil, C. Barbu, E. Cetkin, G. Yarimca, A. Jensen, P. Larsen, and C. Gomes, "Digital twin of electric vehicle battery systems: Comprehensive review of the use cases, requirements, and platforms," *Renewable and Sustainable Energy Reviews*, vol. 179, 2023.

[7] M. T. Lawder, B. Suthar, P. W. C. Northrop, S. De, C. M. Hoff, O. Leitermann, M. L. Crow, S. Santhanagopalan, and V. R. Subramanian, "Battery energy storage system (BESS) and battery management system (BMS) for grid-scale applications," *Proceedings of the IEEE*, vol. 102, no. 6, pp. 1014–1030, 2014.

[8] M. Lelie, T. Braun, M. Knips, H. Nordmann, F. Ringbeck, H. Zappen, and D. U. Sauer, "Battery management system hardware concepts: An overview," *Applied Sciences*, vol. 8, no. 4, p. 534, 2018.

[9] X. Hu, F. Feng, K. Liu, L. Zhang, J. Xie, and B. Liu, "State estimation for advanced battery management: Key challenges and future trends," *Renewable and Sustainable Energy Reviews*, vol. 114, p. 109334, 2019.

[10] A. W. Thompson, "Economic implications of lithium ion battery degradation for vehicle-to-grid (V2X) services," *Journal of Power Sources*, vol. 396, pp. 691–709, 2018.

[11] B. Bole, C. Kulkarni, and M. Daigle, "Randomized battery usage data set (dataset 11, last accessed: 26/04/2024)," https://www.nasa.gov/intelligent-systems-division/discovery-and-systems-health/pcoe/pcoe-data-set-repository/, NASA Ames Research Center, 2022.

[12] "Sandia national laboratories study overview," https://www.batteryarchive.org/snl_study.html, BatteryArchive.org, 2024.

[13] M. Steinstraeter, J. Buberger, and D. Trifonov, "Battery and heating data in real driving cycles," https://dx.doi.org/10.21227/6jr9-5235, 2020.

[14] P. Shen, M. Ouyang, X. Han, X. Feng, L. Lu, and J. Li, "Error analysis of the model-based state-of-charge observer for Lithium-Ion batteries," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 9, pp. 8055–8064, 2018.

[15] Y. Zhang, *Edge Computing for Digital Twin*. Springer Nature, 2024, pp. 41–59.

[16] J. Gibbs, "Tesla battery management system (BMS) calibration," https://tesla-info.com/guide/tesla-bms-calibration.php, Apr 2023.

[17] 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 Machine Intelligence*, vol. 2, no. 3, pp. 161–170, 2020.

[18] L. Chen, Z. Wang, Z. Lü, J. Li, B. Ji, H. Wei, and H. Pan, "A novel state-of-charge estimation method of Lithium-Ion batteries combining the grey model and genetic algorithms," *IEEE Transactions on Power Electronics*, vol. 33, no. 10, pp. 8797–8807, 2017.

[19] B. Fu, W. Wang, Y. Li, and Q. Peng, "An improved neural network model for battery smarter state-of-charge estimation of energy-transportation system," *Green Energy and Intelligent Transportation*, vol. 2, no. 2, p. 100067, 2023.

[20] R. Xiong, Y. Zhang, J. Wang, H. He, S. Peng, and M. Pecht, "Lithium-Ion battery health prognosis based on a real battery management system used in electric vehicles," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 5, pp. 4110–4121, 2019.

[21] C. Zhang, L. Luo, Z. Yang, S. Zhao, Y. He, X. Wang, and H. Wang, "Battery SOH estimation method based on gradual decreasing current, double correlation analysis and GRU," *Green Energy and Intelligent Transportation*, vol. 2, no. 5, p. 100108, 2023.

[22] C. She, Y. Shen, G. Bin, J. Hong, and Y. Peng, "Accurate state of health estimation of battery system based on multistage constant current charging and behavior analysis in real-world electric vehicles," *IEEE*

Transactions on Transportation Electrification, vol. 11, no. 1, pp. 97–108, 2025.

[23] T. Kim, W. Qiao, and L. Qu, "An enhanced hybrid battery model," *IEEE Transactions on Energy Conversion*, vol. 34, no. 4, pp. 1848–1858, 2019.

[24] X. Shu, Z. Chen, J. Shen, F. Guo, Y. Zhang, and Y. Liu, "State of charge estimation for Lithium-Ion battery based on hybrid compensation modeling and adaptive H-Infinity filter," *IEEE Transactions on Transportation Electrification*, vol. 9, no. 1, 2023.

[25] S. Singh, Y. E. Ebongue, S. Rezaei, and K. P. Birke, "Hybrid modeling of lithium-ion battery: Physics-informed neural network for battery state estimation," *Batteries*, vol. 9, no. 6, 2023.

[26] R. G. Nascimento, M. Corbetta, C. S. Kulkarni, and F. A. C. Viana, "Li-ion battery aging with hybrid physics-informed neural networks and fleet-wide data," *Annual Conference of the PHM Society*, vol. 13, 2021.

[27] G. Pollo, A. Burrello, E. Macii, S. V. Massimo Poncino, and D. J. Pagliari, "Coupling neural networks and physics equations for Li-Ion battery state-of-charge prediction," in *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2025.

[28] X. Hu, H. Yuan, C. Zou, Z. Li, and L. Zhang, "Co-estimation of state of charge and state of health for Lithium-Ion batteries based on fractional-order calculus," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 11, pp. 10 319–10 329, 2018.

[29] Y. Qin, A. Arunan, and C. Yuen, "Digital twin for real-time Li-Ion battery state of health estimation with partially discharged cycling data," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 5, 2023.

[30] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "LightGBM: A highly efficient gradient boosting decision tree," *Advances in neural information processing systems*, vol. 30, 2017.

[31] Y. Preger, H. M. Barkholtz, A. Fresquez, D. L. Campbell, B. W. Juba, J. Romàn-Kustas, S. R. Ferreira, and B. Chalamala, "Degradation of commercial Lithium-Ion cells as a function of chemistry and cycling conditions," *Journal of The Electrochemical Society*, vol. 167, no. 12, p. 120532, sep 2020.

[32] G. Pozzato, A. Allam, and S. Onori, "Lithium-Ion battery aging dataset based on electric vehicle real-driving profiles," *Data in Brief*, vol. 41, p. 107995, 2022.

[33] S. Yang, Z. Zhang, R. Cao, M. Wang, H. Cheng, L. Zhang, Y. Jiang, Y. Li, B. Chen, H. Ling, Y. Lian, B. Wu, and X. Liu, "Implementation for a cloud battery management system based on the chain framework," *Energy and AI*, vol. 5, p. 100088, 2021.

[34] K. Alamin, D. J. Pagliari, Y. Chen, E. Macii, S. Vinco, and M. Poncino, "Model-driven feature engineering for data-driven battery SOH model," in *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2024.

[35] S. Yang, C. Zhang, J. Jiang, W. Zhang, L. Zhang, and Y. Wang, "Review on state-of-health of lithium-ion batteries: Characterizations, estimations and applications," *Journal of Cleaner Production*, vol. 314, p. 128015, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0959652621022332

[36] J. Barbier, M. Dia, N. Macris, F. Krzakala, T. Lesieur, and L. Zdeborová, "Mutual information for symmetric rank-one matrix estimation: A proof of the replica formula," in *Advances in Neural Information Processing Systems*, vol. 29, 2016.

[37] H. Liu, Q. Xiao, Z. Jiao, J. Meng, Y. Mu, K. Hou, X. Yu, S. Guo, and H. Jia, "LightGBM-based prediction of remaining useful life for electric vehicle battery under driving conditions," in *IEEE Sustainable Power and Energy Conference (iSPEC)*, 2020, pp. 2577–2582.

[38] H. Qin, X. Fan, Y. Fan, R. Wang, Q. Shang, and D. Zhang, "A computationally efficient approach for the state-of-health estimation of Lithium-Ion batteries," *Energies*, vol. 16, no. 14, p. 5414, 2023.

[39] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A next-generation hyperparameter optimization framework," in *ACM SIGKDD international conference on knowledge discovery & data mining*, 2019, pp. 2623–2631.

[40] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, p. 273–297, 1995.

[41] L. Buitinck, G. Louppe, M. Blondel, F. Pedregosa, A. Mueller, O. Grisel, V. Niculae, P. Prettenhofer, A. Gramfort, J. Grobler, R. Layton, J. VanderPlas, A. Joly, B. Holt, and G. Varoquaux, "API design for machine learning software: experiences from the scikit-learn project," in *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, 2013, pp. 108–122.

[42] K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, Y. Singer, and M. K. Warmuth, "Online passive-aggressive algorithms." *Journal of Machine Learning Research*, vol. 7, no. 3, 2006.

[43] Y. Xiang, W. Fan, J. Zhu, X. Wei, and H. Dai, "Semi-supervised deep learning for lithium-ion battery state-of-health estimation using dynamic discharge profiles," *Cell Reports Physical Science*, vol. 5, no. 1, 2024.

[44] W. Liu, Y. Xu, and X. Feng, "A hierarchical and flexible data-driven method for online state-of-health estimation of li-ion battery," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 12, pp. 14 739–14 748, 2020.

[45] K. Das, R. Kumar, and A. Krishna, "Analyzing electric vehicle battery health performance using supervised machine learning," *Renewable and Sustainable Energy Reviews*, vol. 189, p. 113967, 2024.

[46] A. Arif, M. Hassaan, M. Abdullah, A. Nadeem, and N. Arshad, "Estimating battery state of health using machine learning," in *International Conference on Smart Grid and Clean Energy Technologies (ICSGCE)*, 2022, pp. 72–77.