Indonesian Journal of Electrical Engineering and Computer Science
Vol. 34, No. 2, May 2024, pp. 697∼710
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v34.i2.pp697-710

Support vector regression-based state of charge estimation for batteries: cloud vs non-cloud

Mohamed Ben Youssef[1], Imen Jarraya[2], Mohamed Ali Zdiri[1], Fatma Ben Salem[3]
[1]Control and Energy Management Laboratory, Sfax Engineering School, University of Sfax, Sfax, Tunisia
[2]College of Computer and Information Sciences, Prince Sultan University, Riyadh, Saudi Arabia
[3]Department of Electrical, Engineering School of Prince Sattam Bin Abdulaziz, Elkahrj, Saudi Arabia

Article Info

Article history:

Received Nov 3, 2023
Revised Feb 14, 2024
Accepted Feb 16, 2024

Keywords:

Basic cloud AWS EC2
Experimental results
Lithium-ion batteries
MATLAB production server
Non-cloud MATLAB
SOC estimation
Support vector regression

ABSTRACT

Embracing the potential of cloud technology in the field of electric vehicle advancements, this paper explores the application of support vector regression (SVR) for accurate state of charge (SOC) estimation of lithium-ion batteries in various computational landscapes. This study aims to scrutinize and compare the performance of SOC estimation, with a specific focus on precision, computational efficiency, and execution speed. The investigation is conducted across diverse environments, including a traditional non-cloud setup and two cloud-based platforms-a standard cloud environment employing Amazon web services (AWS) EC2 servers and an enhanced configuration utilizing the MATLAB production server. The investigation not only emphasizes the effectiveness of cloud integration but also provides valuable insights into the strengths and weaknesses of the proposed methodology. The experimental results contribute to a nuanced understanding of the methodology's performance, shedding light on its potential implications for advancing electric vehicle technologies. This study thus extends its significance beyond technical considerations, providing a broader perspective on its relevance to global electrification initiatives.

Corresponding Author:

Mohamed Ali Zdiri
Control and Energy Management Laboratory, Sfax Engineering School, University of Sfax
Sfax, Tunisia
Email: mohamed-ali.zdiri@enis.tn

1. INTRODUCTION

In recent years, the dynamic expansion of cloud computing has drastically revolutionized a variety of businesses, offering new levels of efficiency, scalability, and data management capabilities [1], [2]. This technological innovation is critical as the global energy landscape evolves toward sustainable sources. Cloud technology is pivotal in renewable energy, driven by environmental concerns and climate change urgency. It manages complex data from sources like solar and wind power, enabling real-time information exchange essential for energy output regulation [3]. The use of batteries offers significant advantages in energy management strategies. Batteries play a crucial role in storing excess energy during periods of low demand and releasing it when demand is high, contributing to load balancing and grid stability [4]–[8]. They serve as a valuable tool in optimizing the integration of renewable energy sources, allowing for better management of intermittent energy production [9]–[12]. In electric vehicles, cloud technology enhances connectivity and intelligent systems, facilitating battery management, vehicle-to-grid services, and predictive maintenance [13], and [14]. Additionally, cloud integration in unmanned aerial vehicles (UAVs) improves data processing, flight optimization, and real-time communication, expanding their applications in surveillance and monitoring [15]. This convergence is revolutionizing electric vehicle and UAV industries, contributing to a sustainable energy ecosystem [16].

Lithium-ion batteries, depicted in Figure 1, have become a cornerstone in the renewable energy sector, playing a significant role in ensuring grid stability and effective energy storage. The schematic illustrates the meticulous integration of lithium-ion batteries within the chassis of an electric vehicle, tracing the connections from the individual cells to the assembled modules, and culminating in the complete battery system-a testament to the intricacy and modular nature of these power units. With the advent of major advancements in cloud computing, these batteries are now increasingly accessible and are being utilized in a variety of applications, ranging from electric vehicles to portable electronic devices. These technological strides have enabled the development of more advanced battery management systems (BMS). These systems are crucial for monitoring battery performance, ensuring its longevity, and maintaining safety standards. Given the intricate nature of lithium-ion batteries and their sensitivity to operational issues like over-voltages and under-voltages, a sophisticated BMS is imperative. Such systems are particularly important for accurately tracking key battery metrics, including the state of charge (SOC), state of health (SOH), state of function (SOF), and end of life (EOL) [17]. These metrics are essential for evaluating the battery's performance and condition, thereby playing a pivotal role in the overall management of lithium-ion batteries in various applications.

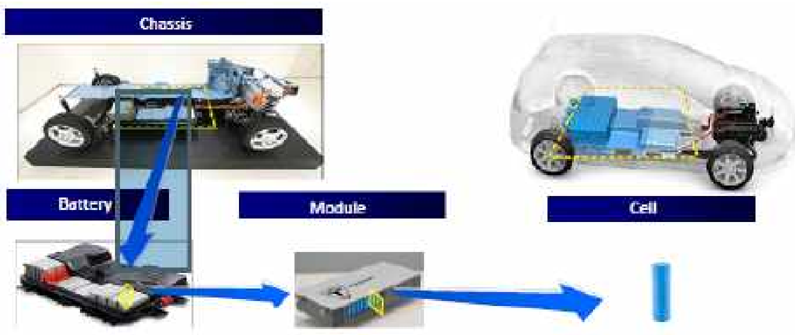


Figure 1. Lithium battery in electric vehicles

Accurately estimating the SOC in lithium-ion batteries is a critical yet complex aspect of advancing BMS [18]. Traditional methods for SOC estimation are somewhat effective but face numerous challenges. They require substantial computational resources and are sensitive to environmental variables, like temperature fluctuations. Additionally, these methods necessitate frequent recalibration to compensate for battery aging and performance decline [19]. These issues are magnified in large-scale applications such as electric vehicles and grid energy storage, where the demand for precision and reliability is exceptionally high.

The emergence of cloud computing offers a transformative solution to these challenges. With its robust computational power, scalability, and advanced data analytics capabilities, cloud computing provides an ideal platform for implementing more sophisticated, data-driven SOC estimation methods [20], [21]. This environment is conducive to the deployment of machine learning algorithms, which can handle complex, non-linear relationships and adapt to changing battery conditions without the need for extensive manual recalibration. Moreover, cloud computing enables real-time data processing and sharing, essential for applications where timely and accurate SOC information is critical for operational decision-making [22].

Incorporating cloud computing into BMS presents a set of distinct challenges and opportunities. It is essential to carefully address data security, mitigate latency issues, and ensure robust support from the cloud infrastructure for BMS demands [23]. Successfully managing these aspects can lead to significant advancements, including enhanced accuracy in SOC estimation, improved reliability of the systems, and the development of more sophisticated battery management strategies. These strategies would be capable of dynamically adapting to varying operational conditions, thereby elevating the efficacy of the BMS [24], [25].

Our contributions in the field of SOC estimation for battery and energy storage systems in vehicles span both cloud-based and non-cloud computing environments, demonstrating significant advancements in SOC estimation through the use of support vector machine (SVM) methodology across various computational settings. These settings encompass local MATLAB processing, Amazon web services (AWS) cloud infrastructure, and MATLAB production server environments.

– Utilizing MATLAB in a non-cloud environment, we establish a benchmark for SOC estimation. This foundational performance serves as a comparative point for cloud-integrated approaches.
– Through the AWS cloud environment, we harness EC2 instances' scalable computational capabilities, thereby refining the SVM method's efficiency and precision.
– The MATLAB production server offers an elevated cloud environment, which is instrumental in improving execution speed and streamlining the deployment of SVM models within operational frameworks.

The integration of cloud computing, illustrated in Figure 2, markedly enhances the accessibility and scalability of our SOC estimation methodology. The cloud paradigm we adopt is characterized by essential attributes, service models, and deployment models, which facilitate the delivery of information technology (IT)-enabled capabilities via the internet. Such technological progression merges the tangible and digital spheres, expediting the transference of theoretical SOC estimation into practical, cutting-edge BMS products. Moreover, it fosters the generation of data-centric battery life predictions, diminishes latency, and bolsters highly reliable edge computing systems-elements that present considerable implementation challenges in onboard systems.

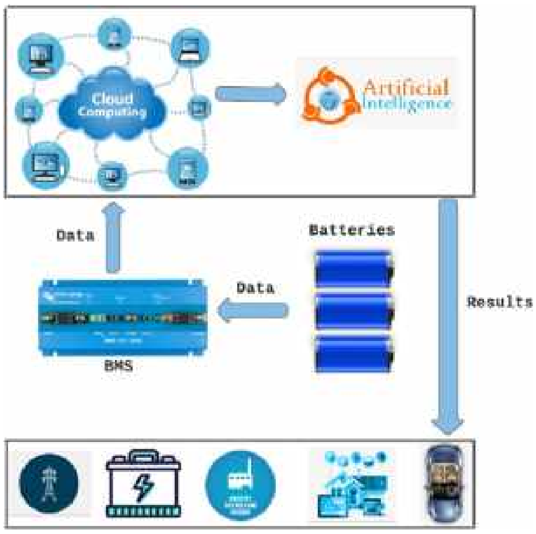


Figure 2. Integration of cloud computing in SOC estimation

This study is organized to provide a thorough examination of SOC estimation in lithium-ion batteries, exploring both non-cloud and cloud-based methodologies using support vector regression (SVR). Section 1 introduces the topic and sets the context for the study, emphasizing the importance of accurate SOC estimation. Section 2 examines existing SOC estimation approaches, highlighting their advantages and disadvantages, and includes a comparison between non-cloud and cloud-based methods. Section 3 delves deeper into machine learning approaches, particularly focusing on SVR and its application in both non-cloud and cloud environments. Section 4 explores various versions of cloud computing, detailing their integration and benefits specifically in the context of SOC estimation. Finally, section 5 serves as the conclusion, summarizing the major findings, presenting a comparative analysis of the different methodologies, and suggesting future research directions in this dynamic field.

2. SOC ESTIMATION: STATE-OF-THE-ART APPROACHES

The literature provides a variety of tried-and-true methods for reaching the best SOC estimation. On the other hand, each strategy needs to have its advantages and disadvantages carefully considered and supported by relevant arguments. We divide these contributions into three main categories in the following sections: the direct measurement approach, the model-based approach, and the data-driven approach. We give a brief synopsis of each strategy's foundations, characteristics, and constraints below.

2.1. Direct measurement approach

The direct measurement approach for SOC estimation in batteries utilizes observable physical parameters such as current and voltage. A prominent method within this approach is the ampere-hour counting (AHC). The AHC method integrates the current flow over time to calculate the SOC:

$$SOC(t) = SOC(t_0) - \frac{1}{Q_N} \int_{t_0}^{t} \eta I(\tau) d\tau \tag{1}$$

here, $SOC(t)$ is the SOC at time t, $SOC(t_0)$ is the initial SOC, η is the coulombic efficiency, $I(\tau)$ is the current, and Q_N is the nominal capacity.

The AHC method in the direct measurement approach is valued for its simplicity and cost-efficiency, making it ideal for applications like portable electronics, electric vehicles, and renewable energy systems. Its straightforward implementation outweighs the need for complex modeling. However, accuracy issues arise from potential errors in initial SOC estimations and current measurements. These inaccuracies, compounded over time, affect reliability. External factors such as temperature changes and battery aging also aren't accounted for. Despite these drawbacks, its practicality makes it a fundamental choice in situations where quick and uncomplicated estimations are preferable to high precision. The method's effectiveness is enhanced when applied to predefined load profiles and capacities, employing lookup tables for rapid SOC inference. Challenges like sensor noise and current loss during charge cycles are addressed in literature through modified algorithms and recalibrations, improving estimation accuracy.

2.2. Model-based approaches

Model-based approaches for SOC estimation rely on mathematical models of the battery system. These approaches often employ sophisticated algorithms to account for the non-linear and dynamic nature of battery behavior. By utilizing these models, researchers and engineers can accurately estimate the SOC of a battery in real-time, enabling effective monitoring and control of battery performance. These model-based methods have proven to be valuable in various applications, such as electric vehicles, renewable energy systems, and portable electronics. They play a crucial role in optimizing battery utilization, improving overall system efficiency, and extending battery lifespan.

2.2.1. Advanced Kalman filter variants

The Kalman filter (KF) and its variants are pivotal in estimating the SOC for battery systems. The standard KF excels in linear contexts, but battery dynamics often require more sophisticated approaches. The extended Kalman filter (EKF) linearizes non-linear systems, the unscented Kalman filter (UKF) employs deterministic sampling for greater accuracy in non-linear estimations, and the dual and adaptive EKFs offer enhanced state and parameter estimations with dynamic noise adjustments. Complementing these are the ensemble (EnKF) and cubature (CKF) KFs, which handle complex, non-linear scenarios. EKF, UKF, and particle filter (PF) stand out for their robust handling of non-linearities, each offering unique strengths and broad applicability in various SOC estimation scenarios, catering to the diverse needs of battery management systems.

The EKF is a recursive filter suitable for systems with non-linear dynamics, such as SOC estimation in batteries. EKF operates in two steps: prediction and update. The state equation for the system can be expressed as (2).

$$x_k = f(x_{k-1}, u_{k-1}) + w_{k-1} \tag{2}$$

Initially, set the state estimate $\hat{x}_0$ and estimate covariance P_0:

$$\hat{x}_0 = \text{initial state estimate} \tag{3}$$

$$P_0 = \text{initial estimate covariance} \tag{4}$$

in the prediction step:

$$\hat{x}_{k|k-1} = f(\hat{x}_{k-1|k-1}, u_{k-1}) \tag{5}$$

$$P_{k|k-1} = F_{k-1} P_{k-1|k-1} F_{k-1}^T + Q_{k-1} \tag{6}$$

in the update step:

$$K_k = P_{k|k-1}H_k^T(H_kP_{k|k-1}H_k^T + R_k)^{-1} \tag{7}$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k(z_k - h(\hat{x}_{k|k-1}, u_k)) \tag{8}$$

$$P_{k|k} = (I - K_kH_k)P_{k|k-1} \tag{9}$$

where w_{k-1} represents the process noise, F_{k-1} and H_k are the Jacobians of f and h respectively, and Q_{k-1} and R_k are the process and measurement noise covariances.

The EKF is suitable for systems with moderate non-linearities, offering a balance between accuracy and computational efficiency. It's often used in environments where real-time estimation is essential but computational resources are limited. The UKF, on the other hand, excels in handling highly non-linear systems, providing superior accuracy at the cost of increased computational demand. It is ideal for applications where precision is crucial. The PF provides a non-parametric solution, best suited for scenarios with complex, non-Gaussian noise and requiring robust state estimation. Each of these filters, including dual, adaptive, ensemble, and cubature Kalman filters, has distinct advantages, making them applicable across various SOC estimation scenarios, tailored to the unique dynamics and requirements of battery management systems.

2.2.2. Recursive algorithms

Recursive algorithms for SOC estimation, such as recursive least squares (RLS) and its variants, are essential in battery management systems for their ability to adapt and refine estimates over time. These algorithms include the standard RLS, RLS with forgetting factor (RLS-FF), adaptive RLS (ARLS), and regularized RLS (RRLS). Each variant offers unique features: RLS-FF adapts to changing system dynamics, ARLS adjusts its parameters based on the estimation error, and RRLS includes regularization to handle ill-conditioned data. These algorithms are particularly useful in environments where battery characteristics evolve, requiring continual adjustment of the estimation process.

The RLS-FF algorithm, an adaptive method, is tailored for SOC estimation in battery systems characterized by time-varying dynamics. This algorithm is adept at updating its estimates in response to recent data, making it highly suitable for environments where battery characteristics evolve. It aligns with the 1st order Thevenin model of a battery, which characterizes terminal voltage as a function of open-circuit voltage, internal resistance, current, and voltage across the RC element, to deliver efficient and dynamic SOC estimation. The 1st order Thevenin model of a battery, described by (10).

$$V(t) = OCV(SOC(t)) - R \cdot I(t) - V_{RC}(t) \tag{10}$$

Where $V(t)$ is the terminal voltage, OCV is the open-circuit voltage as a function of SOC, R is the internal resistance, $I(t)$ is the current, and $V_{RC}(t)$ is the voltage across the RC element.

In the RLS-FF algorithm, the following steps are meticulously executed for efficient SOC estimation in battery systems with dynamic properties:

1. Initialization:

$$\hat{x}_0 = \text{initial state estimate} \tag{11}$$

$$P_0 = \text{initial error covariance matrix} \tag{12}$$

2. Update state estimate:

$$\hat{x}_k = \hat{x}_{k-1} + K_k(y_k - H_k\hat{x}_{k-1}) \tag{13}$$

3. Update error covariance:

$$P_k = \lambda^{-1}(P_{k-1} - K_kH_kP_{k-1}) \tag{14}$$

4. Calculate gain:

$$K_k = P_{k-1}H_k^T(H_kP_{k-1}H_k^T + R)^{-1} \tag{15}$$

where λ is the forgetting factor, y_k is the new measurement, H_k is the measurement matrix, and R is the measurement noise covariance.

2.3. Data-driven approach and feed forward neural network

The data-driven approach in SOC estimation leverages advanced computational techniques and algorithms that learn from data. This approach includes methods such as machine learning and artificial intelligence, which are particularly adept at handling large datasets and complex, non-linear relationships inherent in battery behavior. Machine learning (ML) technologies, with their advancements in computing capacity and data storage, are revolutionizing state estimation in the battery industry. ML algorithms find applications in diverse fields and have shown exceptional performance, even outperforming human capabilities in certain scenarios. Particularly in battery state estimation, ML integration plays a pivotal role.

The FFNN is one of the simplest types of neural networks and is capable of performing non-linear mappings. It typically involves non-linear activation functions like the hyperbolic tangent or the rectified linear unit (ReLU) as (16) and (17). Feed forward neural networks (FFNNs) are used in various SOC estimation methods, including backpropagation neural networks and radial basis function neural networks.

$$F(x) = \frac{2}{1 + e^{-2x}} - 1 \tag{16}$$

$$F(x) = \max(0, x) \tag{17}$$

2.4. Recurrent neural network

Recurrent neural networks (RNNs) process sequences by maintaining a form of memory. They are particularly efficient for short-term dependencies but face challenges with long-term sequences, a common characteristic in battery systems. Advanced RNNs like long short-term memory (LSTM), bidirectional LSTM (BiLSTM), and gated recurrent unit (GRU) address this by incorporating mechanisms to remember long sequences. The LSTM, for instance, uses a memory cell with gates to control the flow of informationas (18) to (23). BiLSTM and GRU further refine the RNN's ability to handle sequential data, making them suitable for complex tasks like speech recognition and SOC estimation.

$$i_t = \sigma(W_i \cdot [h_{(t-1)}, x_t] + b_i) \tag{18}$$

$$o_t = \sigma(W_o \cdot [h_{(t-1)}, x_t] + b_o) \tag{19}$$

$$f_t = \sigma(W_f \cdot [h_{(t-1)}, x_t] + b_f) \tag{20}$$

$$\tilde{C}_t = \tanh(W_c \cdot [h_{(t-1)}, x_t] + b_c) \tag{21}$$

$$C_t = f_t \cdot C_{(t-1)} + i_t \cdot \tilde{C}_t \tag{22}$$

$$h_t = o_t \cdot \tanh(C_t) \tag{23}$$

2.5. Support vector machine

SVMs are used for classification and regression tasks. They work by finding a hyperplane in a high-dimensional space that best separates different classes of data. The SVM's effectiveness in generalized regression, especially in SVR, is well-documented for SOC estimation.

$$h_f = f(W_{f1} \cdot x_t + W_{f2} \cdot h_{(t-1)}) \tag{24}$$

$$h_b = f(W_{b1} \cdot x_t + W_{b2} \cdot h_{(t+1)}) \tag{25}$$

$$y_i = g(W_{o1} \cdot h_f + W_{o2} \cdot h_b) \tag{26}$$

The efficiency of SVR in SOC estimation is demonstrated in its ability to handle dynamic stress test cycles and varied data sets, with significant accuracy. Note: the equations are representations of the models' architectures and functionalities. They need to be contextualized and adapted according to the specific battery system and SOC estimation requirements.

The continuous increase in battery cell numbers and the complexity of algorithms challenge the capability of onboard BMS due to limited computational power and data storage. This has led to a collaboration between industry and academia to create scalable infrastructure platforms, aligning with future research directions. Machine learning, an integral part of this collaboration, offers insights that highlight the advantages of cloud computing. As defined by the National Institute of Standards and Technology (NIST), cloud computing enables on-demand access to a shared pool of configurable computing resources. This model promotes availability and consists of five essential characteristics, three service models, and four deployment models.

These IT-enabled capabilities, delivered over the Internet, bridge the physical and virtual worlds. They accelerate the application of optimal SOC estimation theories into new BMS products, paving the way for advanced battery monitoring. The shift to cloud computing is particularly advantageous for data-based lifetime prediction, low latency, and reliable edge-computing systems, which are challenging to implement onboard.

3. SVR FOR SOC ESTIMATION

SVR is a powerful machine learning technique used for predicting continuous outcomes. In the context of battery systems, SVR can be employed to estimate the SOC based on a set of predictor variables. These variables typically include voltage, current, and temperature readings obtained from the battery sensors. SVR leverages the principles of SVM to construct a regression model that accurately captures the relationship between the predictor variables and the SOC. By training the SVR model on historical data with known SOC values, it can then be applied to new data to predict the SOC value. This enables real-time monitoring and control of battery performance, facilitating optimal utilization and improved efficiency in various applications, such as electric vehicles, renewable energy systems, and portable electronics.

3.1. Data collection

Data collection is the first step, where real-time data from current and voltage sensors, as well as temperature readings, are gathered. This data forms the basis for feature vectors in the SVR model.

$$X = \begin{bmatrix} I(t_1) & V(t_1) & T(t_1) \\ I(t_2) & V(t_2) & T(t_2) \\ \vdots & \vdots & \vdots \\ I(t_n) & V(t_n) & T(t_n) \end{bmatrix} \tag{27}$$

Where I is the current, V is the voltage, T is the temperature, and t represents different time points.

3.2. Preprocessing

Data preprocessing is a crucial step in preparing the input data for the SVR model. It involves various techniques, including normalization, feature selection, and dimensionality reduction, to ensure that the SVR model operates on clean and relevant data. By applying data preprocessing techniques like normalization, feature selection, and dimensionality reduction, the SVR model operates on clean and relevant data. This enhances the model's performance, improves its ability to generalize to unseen data, and facilitates accurate SOC estimation in real-time applications.

3.3. SVR model training

The SVR model is trained using the collected data, where it learns to map the input features to the output SOC.

$$\text{SOC} = f(I, V, T) \tag{28}$$

The SVR function f is defined by the following optimization problem:

$$\min \frac{1}{2}||w||^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) \tag{29}$$

subject to:

$$\begin{aligned} y_i - \langle w, x_i \rangle - b &\leq \epsilon + \xi_i, \\ \langle w, x_i \rangle + b - y_i &\leq \epsilon + \xi_i^*, \\ \xi_i, \xi_i^* &\geq 0, \end{aligned} \tag{30}$$

where C is the penalty parameter, ϵ is the error margin, w is the weight vector, b is the bias, and ξ, ξ^* are the slack variables.

3.4. Model validation

Validation is a critical step in the SOC estimation process using the SVR model. It helps to assess the model's performance and ensure its accurate prediction on unseen data. One commonly used technique for validation is cross-validation. Cross-validation involves splitting the available data into multiple subsets or folds. The model is then trained on a subset of the data and evaluated on the remaining fold. This process is repeated multiple times, with each subset serving as the validation set at least once. The performance metrics, such as mean squared error or coefficient of determination, are calculated across all folds to provide an overall assessment of the model's predictive capabilities. By employing cross-validation, the SVR model can be evaluated on various subsets of the data, allowing for a robust assessment of its performance. This technique helps to mitigate issues like overfitting or underfitting, which can occur when a model performs well on the training data but fails to generalize to new, unseen data.

4. INTEGRATION OF CLOUD COMPUTING FOR SOC ESTIMATION IN LITHIUM-ION BATTERIES: IMPLEMENTATION AND RESULTS

This section details a study comparing three environments for implementing ML, specifically SVR, in the estimation of the SOC of lithium-ion batteries. The first environment uses only the computational resources of a standard desktop computer, without cloud support. The second and third environments, in contrast, implement SVR in MATLAB integrated with AWS cloud computing, encompassing both basic-cloud and advanced-cloud environments. The third environment additionally includes the integration of MATLAB server. A laboratory prototype, as shown in Figure 3, was used to validate the performance and efficiency of these environments in cloud and non-cloud settings for SOC estimation in lithium-ion batteries. The specifications of the Lithium-ion battery used in this study are outlined in Table 1. Each scenario is detailed with a focus on the execution environment and performance outcomes. Additionally, the study assessed the SVR model's performance by investigating four distinct datasets in an On-Premises environment. These datasets, summarized in Table 2, varied in size, allowing us to evaluate the model's execution time under different data loads and conditions.

4.1. Scenario 1: non-cloud environment (desktop computer)

In scenario 1, we established an on-premises model by installing MATLAB on a desktop computer equipped with 8 CPUs and 16 GB of RAM. The results and observations as OCV discharging mode and OCV method. Discharging mode as shown in Figure 4: in this mode, the green lines symbolize the measurement of the battery voltage. In our study, we employed the Coulomb counting method to determine the experimental SOC, which is shown in blue. Coulomb counting, based on (31), is a widely recognized and established method for estimating the SOC of a lithium-ion battery. It operates on the principle of measuring the current flowing into or out of the battery over time and integrating it to obtain the total charge transferred. The SOC estimate shown by the red SOC line closely matches the experimental value for this continuous discharge mode (stationary state), demonstrating the durability and accuracy of the SVR ML method. The SVR method exhibits a mean inaccuracy of just 1%, calculated as the average of the absolute differences between predicted and actual SOC values. The formula for mean absolute error (MAE) is given by:

$$\text{MAE} = \frac{1}{n} \sum |SOC_{\text{EXP}} - SOC_{\text{EST}}| \tag{31}$$

where n is the number of data points.

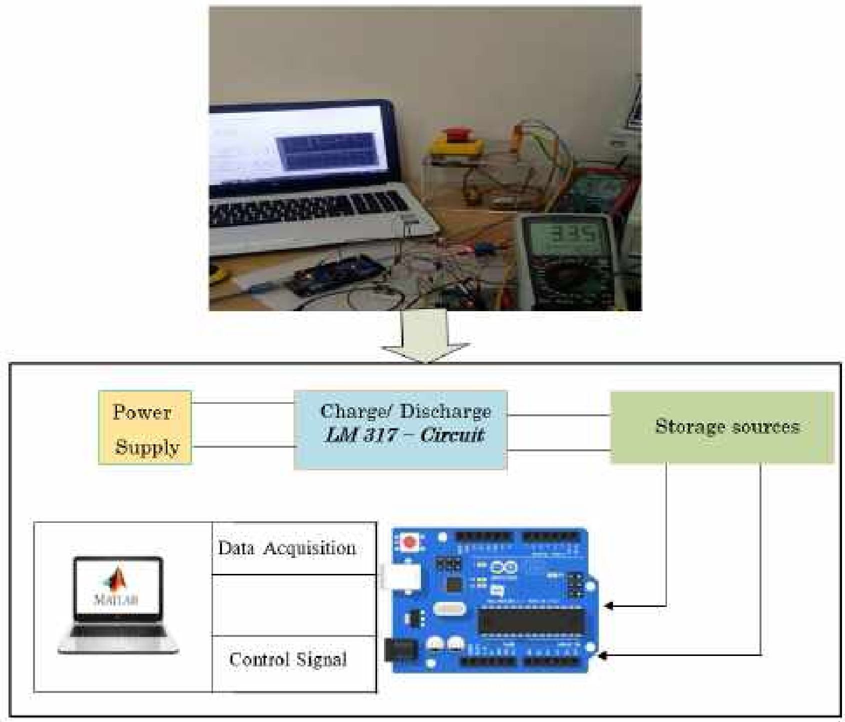


Figure 3. Experimental setup of test OCV-SOC bench

Table 1. Specifications for Camelion li-ion battery

Characteristic	Value
Name	ICR18650
Chemical system	Lithium-ion
Nominal voltage	3.7 V
Capacity	2600 mAh
Internal resistance	12 mΩ
Operating temperature range	-20 °C to 60 °C
Average weight	52 g

Table 2. Size of each datasets

Name	Size
Dataset 1	253 ko
Dataset 2	2,827 ko
Dataset 3	23,448 ko
Dataset 4	24,289 ko

OCV Method as shown in Figure 5: in this method, the green lines represent the battery voltage measurements. Notably, the SVR technique demonstrates superior accuracy in estimating SOC compared to the Coulomb experimental results. SVR's ability to leverage knowledge of the nonlinear relationship between the battery's input and output characteristics allows it to provide precise predictions. In contrast, the Coulomb counting method relies on measuring the battery's current and integrating it over time, making it susceptible to various factors such as temperature, aging, and environmental conditions. SVR excels in providing accurate and robust SOC estimates, particularly in cases where the battery's initial state of charge is low or zero. It's worth mentioning that the Coulomb counting method requires regular calibration to maintain accurate SOC estimations.

Run-time analysis reveals that the computational demand of the SVR method for SOC estimation escalates as the dataset size grows. For instance, estimating the SOC of a lithium battery in the first dataset, which is 253 KB in size, takes 16.6821 seconds. However, as the dataset size increases to 24,289 KB in the fourth dataset, the execution time extends to 21.7746 seconds (refer to the summary in Table 3). This escalation in execution time is primarily attributed to the SVR method's necessity to train the algorithm on a substantial amount of data, resulting in heightened computational intensity.

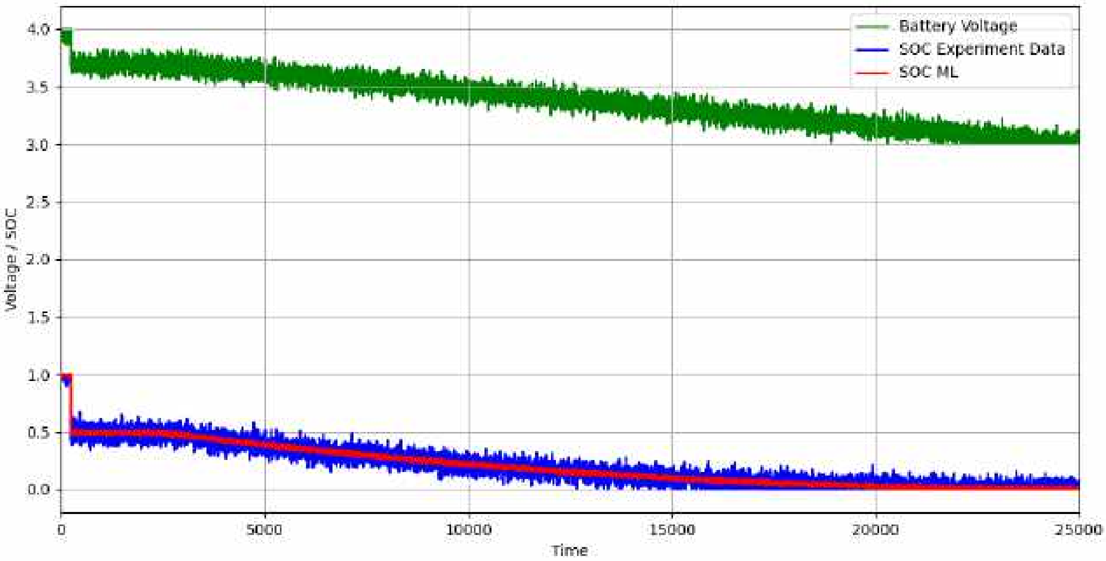


Figure 4. Model validation considering steady-state operation

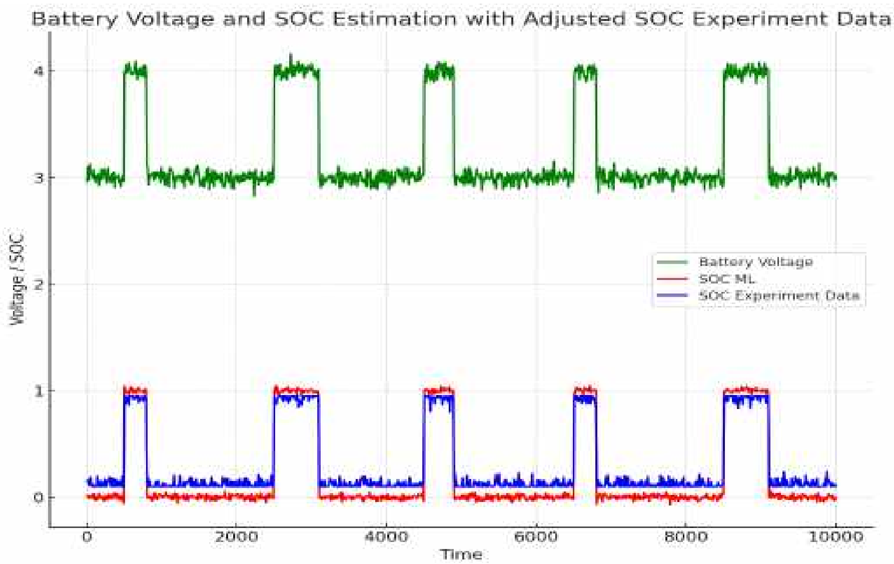


Figure 5. Model validation considering dynamic state

Table 3. Comparison of execution times across environments

Datasets	Non-cloud	Basic-cloud environment			MATLAB server
		C5.2xlarge	C5.12xlarge	p4d.24xlarge	
Dataset 1	16.6821 sec	12.5737 sec	9.5785 sec	6.4857 sec	4.5896 sec
Dataset 2	19.9576 sec	20.0963 sec	13.2293 sec	10.5526 sec	6.5245 sec
Dataset 3	20.3577 sec	20.8024 sec	13.8540 sec	10.1027 sec	6.7024 sec
Dataset 4	21.7746 sec	22.3273 sec	13.9949 sec	10.4836 sec	6.9653 sec

4.2. Scenario 2: basic-cloud environment (AWS EC2 servers)

Cloud computing, particularly leveraging AWS, offers robust solutions for scalable and reliable SOC estimation for battery systems. The integration of AWS services like EC2, S3, and CloudFormation enhances computational efficiency, data management, and model deployment. The AWS compute environment, specifically utilizing EC2 P4d instances, provides the necessary computational power for real-time SOC estimation. These instances come equipped with high-performance graphics processing units (GPUs) and ample network bandwidth to handle intensive ML and high-performance computing workloads. Given the substantial volume of data involved in SOC estimation, AWS's S3 service offers scalable and secure storage solutions. It enables efficient data retrieval and updates to model parameters, which are critical aspects of ML models as shown in Figure 6.

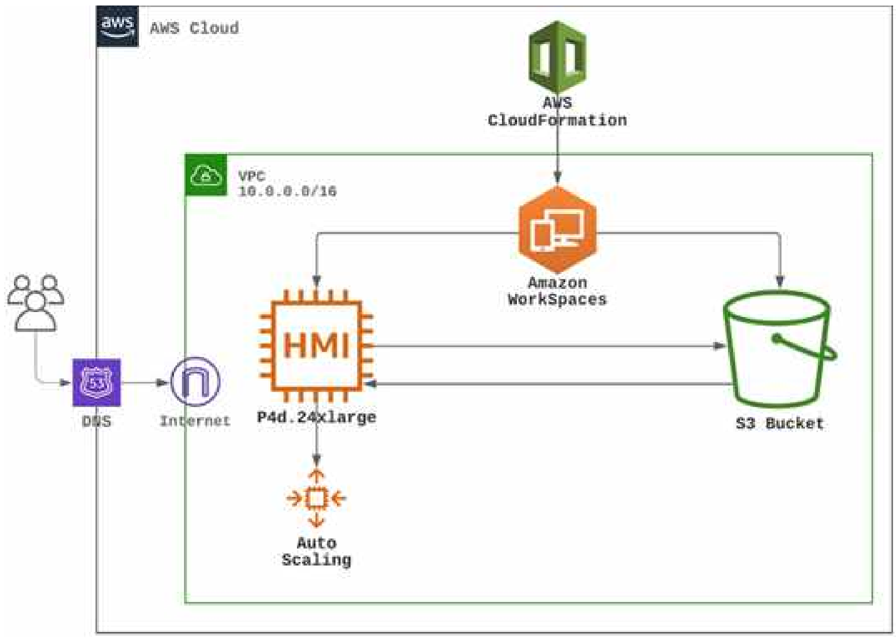


Figure 6. Cloud storage architecture using AWS S3 for SOC estimation data

AWS CloudFormation streamlines the setup of complex environments required for SOC estimation, enabling quick and automated provisioning of necessary resources as shown in Figure 7. Amazon route 53 ensures efficient and secure routing to the cloud resources, making SOC estimation services easily accessible and reliable as shown in Figure 8. Cloud resources offer a flexible and scalable alternative to on-premises technology. Performance tests on AWS EC2 instances yielded varied results:

i) The C5.2xlarge instance delivered comparable performance to the desktop computer, with some datasets processing more rapidly in the cloud.
ii) Upgrading to the C5.12xlarge instance resulted in a 35.7% reduction in execution time.
iii) The high-performance p4d.24xlarge instance exhibited a 48.14% faster processing time for large datasets.

These findings suggest that cloud computing can provide faster execution times, especially for larger datasets, without compromising the accuracy of SOC estimation.

4.3. Scenario 3: advanced-cloud environment (MATLAB production server)

The MATLAB production server offers a centralized, secure, and scalable environment that enables parallel execution and provides a wide range of valuable features. This advanced cloud-based environment represents a significant departure from traditional work paradigms, effectively breaking down the barriers that typically separate industry and academia by harnessing a robust workspace model. Powered by the high-performance 'p4d.24xlarge' instance, our research results unequivocally demonstrate that such a configuration, supported by its exceptional computational resources, delivers the fastest execution times. However, it's crucial to recognize that the exact improvement in execution time depends on various factors, including the complexity of the SVR model, dataset size, and the availability of computational resources. Moreover, the performance of MATLAB itself may exhibit variations when deployed in a cloud environment as opposed to a non-cloud environment, which can consequently impact execution time and algorithmic speed.

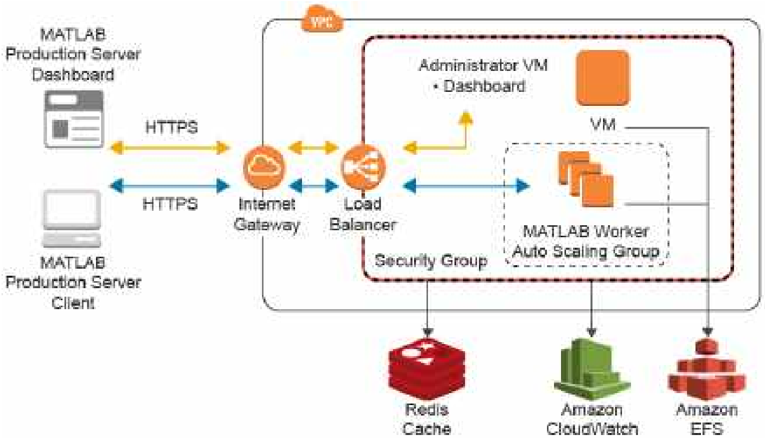


Figure 7. The automated deployment process facilitated by AWS CloudFormation

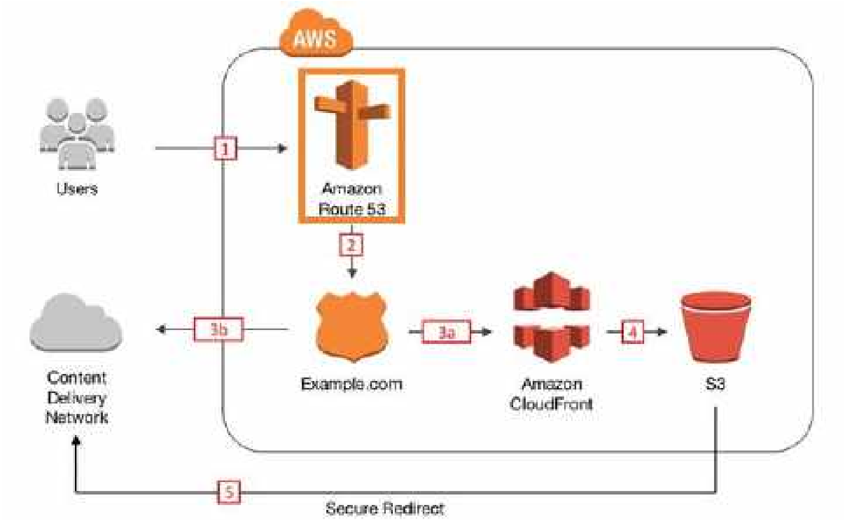


Figure 8. Routing and domain management using Amazon route 53

In this scenario, we have observed an average enhancement in execution time by 50% in comparison to utilizing ’p4d.24xlarge’ for the basic-cloud environment. In certain instances, this mean improvement surges to an impressive 70% when juxtaposed with the implementation of SOC estimation via SVR sans cloud integration. Significantly, notwithstanding the gains in execution time, we have sustained a consistent average SOC estimation error when juxtaposed with the previous scenarios. This implies that harnessing cloud resources for SOC battery estimation via the SVR technique might not intrinsically affect the precision of SOC estimation. Nonetheless, it exerts a substantial influence on execution time, effectively expediting the SOC estimation process.

We can encapsulate the contributions of these deployment models succinctly in Table 3, which provides a detailed comparison of execution times across different environments. In summation, cloud environments, particularly in advanced configurations, offer substantial advantages in terms of execution time and scalability for SOC estimation employing the SVR method. These advantages render cloud-based solutions an appealing choice for large-scale battery management applications.

5. CONCLUSION

Our study has highlighted the significant potential of cloud-based artificial intelligent (AI) methods, particularly emphasizing SVR, in enhancing the accuracy and efficiency of SOC estimation for Lithium-ion batteries. This research emphasizes the pivotal role of cloud computing in reshaping BMS, with far-reaching implications for energy storage and utilization. The integration of cloud computing and AI methods, coupled with the emerging concept of digital twins, represents an exciting frontier. This innovative approach not only holds the promise of further enhancing SOC estimation but also opens doors to predictive analytic and proactive battery management. The convergence of cloud computing and AI methods lays the foundation for a sustainable and technologically advanced future for BMS, providing abundant avenues for ongoing enhancements and innovation.

REFERENCES

[1] A. H. A. AL-Jumaili, R. C. Muniyandi, M. K. Hasan, J. K. S. Paw, and M. J. Singh, "Big data analytics using cloud computing based frameworks for power management systems: status, constraints, and future recommendations," *Sensors*, vol. 23, no. 6, p. 2952, Mar. 2023, doi: 10.3390/s23062952.

[2] O. Debauche, S. Mahmoudi, P. Manneback, and F. Lebeau, "Cloud and distributed architectures for data management in agriculture 4.0: review and future trends," *Journal of King Saud University-Computer and Information Sciences*, vol. 34, no. 9, pp. 7494–7514, Oct. 2022, doi: 10.1016/j.jksuci.2021.09.015.

[3] S. Long, Y. Li, J. Huang, Z. Li, and Y. Li, "A review of energy efficiency evaluation technologies in cloud data centers," *Energy and Buildings*, vol. 260, p. 111848, Apr. 2022, doi: 10.1016/j.enbuild.2022.111848.

[4] N. Guru, S. Patnaik, M. R. Nayak, and M. Viswavandya, "Wind generator and storage system scheduling for customer benefit and battery life," *Bulletin of Electrical Engineering and Informatics*, vol. 12, no. 5, pp. 2586–2594, Oct. 2023, doi: 10.11591/eei.v12i5.4661.

[5] K. Omiloli, A. Awelewa, I. Samuel, O. Obiazi, and J. Katende, "State of charge estimation based on a modified extended Kalman filter," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 13, no. 5, pp. 5054–5065, Oct. 2023, doi: 10.11591/ijece.v13i5.pp5054-5065.

[6] V. A. Kusuma, A. A. Firdaus, S. S. Suprapto, D. F. U. Putra, Y. Prasetyo, and F. Filliana, "Leveraging PSO algorithms to achieve optimal stand-alone microgrid performance with a focus on battery lifetime," *International Journal of Applied Power Engineering (IJAPE)*, vol. 12, no. 3, pp. 293–299, Jul. 2023, doi: 10.11591/ijape.v12.i3.pp293-299.

[7] V. Gopu and M. S. Nagaraj, "Power management algorithm for standalone operated renewable distribution generator with hybrid energy backup in microgrid," *International Journal of Power Electronics and Drive Systems (IJPEDS)*, vol. 14, no. 2, pp. 1249–1259, Jun. 2023, doi: 10.11591/ijpeds.v14.i2.pp1249-1259.

[8] I. Mansoury, D. El Bourakadi, A. Yahyaouy, and J. Boumhidi, "A novel decision-making approach based on a decision tree for micro-grid energy management," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 30, no. 2, pp. 1150–1158, May 2023, doi: 10.11591/ijeecs.v30.i2.pp1150-1158.

[9] S. A. Razzaq and V. Jayasankar, "Energy management system for AC/DC HMGS integrated with interconnected renewable sources and interlinking converter," *International Journal of Applied Power Engineering (IJAPE)*, vol. 12, no. 1, pp. 24–36, Mar. 2023, doi: 10.11591/ijape.v12.i1.pp24-36.

[10] M. Al-Flehawee and A. Al-Mayyahi, "Energy management for hybrid electric vehicles using rule based strategy and PI control tuned by particle swarming optimization algorithm," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 6, pp. 5938–5949, Dec. 2022, doi: 10.11591/ijece.v12i6.pp5938-5949.

[11] S. K. Bilgundi, H. Pradeepa, A. Kadam, and L. M. Venkatesh, "Energy management schemes for distributed energy resources connected to power grid," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 28, no. 1, pp. 30–40, Oct. 2022, doi: 10.11591/ijeecs.v28.i1.pp30-40.

[12] W. Srirattanawichaikul and P. Wirasanti, "Evaluation of lightweight battery management system with field test of electric bus in campus transit system," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 10, no. 6, pp. 6202–6213, Dec. 2020, doi: 10.11591/ijece.v10i6.pp6202-6213.

[13] Z. Teimoori, A. Yassine, and M. S. Hossain, "A secure cloudlet-based charging station recommendation for electric vehicles empowered by federated learning," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 9, pp. 6464–6473, Sep. 2022, doi: 10.1109/TII.2022.3148997.

[14] P. Dong *et al.*, "Practical application of energy management strategy for hybrid electric vehicles based on intelligent and connected technologies: development stages, challenges, and future trends," *Renewable and Sustainable Energy Reviews*, vol. 170, p. 112947, Dec. 2022, doi: 10.1016/j.rser.2022.112947.

[15] W. Meng, Y. Yang, J. Zang, H. Li, and R. Lu, "DTUAV: a novel cloud–based digital twin system for unmanned aerial vehicles," *Simulation*, vol. 99, no. 1, pp. 69–87, Jan. 2023, doi: 10.1177/00375497221109575.

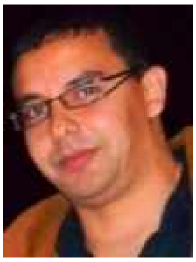
[16] B. B. Gupta, A. Gaurav, E. C. Marín, and W. Alhalabi, "Novel graph-based machine learning technique to secure smart vehicles in intelligent transportation systems," *IEEE Transactions on Intelligent Transport ation Systems*, vol. 24, no. 8, pp. 8483–8491, Aug. 2023, doi: 10.1109/TITS.2022.3174333.

[17] M. S. H. Lipu *et al.*, "Deep learning enabled state of charge, state of health and remaining useful life estimation for smart battery management system: methods, implementations, issues and prospects," *Journal of Energy Storage*, vol. 55, p. 105752, Nov. 2022, doi: 10.1016/j.est.2022.105752.

[18] R. R. Kumar, C. Bharatiraja, K. Udhayakumar, S. Devakirubakaran, K. S. Sekar, and L. Mihet-Popa, "Advances in batteries, battery modeling, battery management system, battery thermal management, SOC, SOH, and charge/discharge characteristics in EV applications," *IEEE Access*, vol. 11, pp. 105761–105809, 2023, doi: 10.1109/ACCESS.2023.3318121.

[19] S. Jiang and Z. Song, “A review on the state of health estimation methods of lead-acid batteries,” *Journal of Power Sources*, vol. 517, p. 230710, Jan. 2022, doi: 10.1016/j.jpowsour.2021.230710.
[20] D. Shi *et al.*, “Cloud-based artificial intelligence framework for battery management system,” *Energies*, vol. 16, no. 11, p. 4403, May 2023, doi: 10.3390/en16114403.
[21] X. Sun, Q. Chen, L. Zheng, and J. Yang, “Joint estimation of state-of-health and state-of-charge for Lithium-Ion battery based on electrochemical model optimized by neural network,” *IEEE Journal of Emerging and Selected Topics in Industrial Electronics*, vol. 4, no. 1, pp. 168–177, Jan. 2023, doi: 10.1109/JESTIE.2022.3148031.
[22] D.V.S.R. Sesidhar, C. Badachi, and R. C. Green II, “A review on data-driven SOC estimation with Li-Ion batteries: Implementation methods & future aspirations,” *Journal of Energy Storage*, vol. 72, p. 108420, Nov. 2023, doi: 10.1016/j.est.2023.108420.
[23] M. Waseem, M. Ahmad, A. Parveen, and M. Suhaib, “Battery technologies and functionality of battery management system for EVs: current status, key challenges, and future prospectives,” *Journal of Power Sources*, vol. 580, p. 233349, Oct. 2023, doi: 10.1016/j.jpowsour.2023.233349.
[24] H. Li, M. B. Kaleem, Z. Liu, Y. Wu, W. Liu, and Z. Huang, “IoB: internet-of-batteries for electric vehicles–architectures, opportunities, and challenges,” *Green Energy and Intelligent Transportation*, vol. 2, no. 6, p. 100128, Dec. 2023, doi: 10.1016/j.geits.2023.100128.
[25] O. Markogiannaki, S. Stefanidou, M.-A. Kourtis, A. Oikonomakis, D. Santorinaios, and G. Xilouris, “A survey on smart buildings integration with 5G and cloud technologies towards energy efficiency,” *Preprints*, p. 10, 2023, doi: 10.20944/preprints202311.1553.v1.

BIOGRAPHIES OF AUTHORS



Mohamed Ben Youssef was born in Tunisia. He received the BS degree in 2011 from the University of Sousse, Tunisia. He is currently working toward Ph.D. in Electrical Engineering at Sfax Engineering School (SES), University of Sfax, Tunisia. His main research interests include computer science applications, information technologies, video processing systems and network. He is a member of the research “CEMLab” laboratory of the SES. He can be contacted at email: mby.phd@gmail.com.

Imen Jarraya received her MS degree in Electrical Systems from the National School of Engineers (ENIS), Tunisia, in 2009, and her M.Sc. degree in Electrical Conversion and Renewable Energy from the same institution in 2010. She completed her Ph.D. in Electrical Engineering at the National Engineering School of Sfax (ENIS) in 2021. She is currently working as a postdoctoral researcher at the Robotics and Internet-of-Things Laboratory at Prince Sultan University. Her research primarily focuses on computer science applications in unmanned aerial vehicles (UAVs), contributing to advancements in this field. She can be contacted at email: ijarraya@psu.edu.mysa.

Mohamed Ali Zdiri obtained the Bachelor of Science degree in 2016 and completed the Ph.D. in 2019, both in Electrical Engineering at National Engineering School of Sfax (ENIS), University of Sfax, Tunisia. Currently, he serves as an Assistant Contractual in the Electrical Engineering Department at ENIS. He has held the position of contractual assistant in electrical engineering at ENIS since 2016. Additionally, he is a distinguished Member of the Control and Energy Management Laboratory (CEMLab) at ENIS. His primary research interests encompass the implementation and synthesis of investigative diagnosis, fault-tolerant methods for electrical power converters, power system stability, and energy management strategies tailored for renewable energy applications. He can be contacted at email: mohamed-ali.zdiri@enis.tn.

Fatma Ben Salem was born in Sfax, Tunisia, in 1978. She received the BS, the Master’s, the Ph.D., and the HCR degrees in 2002, 2003, 2010, and 2015, respectively, all in electrical engineering from the National Engineering School of Sfax, University of Sfax, Tunisia. She is a professor of electrical engineering at the High Institute of Industrial Management of Sfax, Tunisia. She is a member of the Control and Energy Management Laboratory (CEMLab) of the University of Sfax. She is an IEEE member. Her main research interests cover several aspects related to the advanced control and the diagnostic of electric machine drives and generators involved in automotive, renewable energy systems, and Smart grid technologies. She can be contacted at email: fatma.bensalem@isgis.usf.tn.