



HAL
open science

Onto-SoC: An ontology-based approach to battery pack SoC estimation

Ala Eddine Hamouni, Franco Giustozzi, Ahmed Samet, Ali Ayadi, Slimane Arbaoui, Tedjani Mesbahi

► **To cite this version:**

Ala Eddine Hamouni, Franco Giustozzi, Ahmed Samet, Ali Ayadi, Slimane Arbaoui, et al.. Onto-SoC: An ontology-based approach to battery pack SoC estimation. 27th International Conference on Knowledge-Based and Intelligent Information and Engineering Systems (KES 2023), Sep 2023, Athènes, Greece. pp.2254-2263, 10.1016/j.procs.2023.10.216 . hal-04236470

HAL Id: hal-04236470

<https://hal.science/hal-04236470>

Submitted on 8 Jul 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial - NoDerivatives 4.0 International License

27th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems (KES 2023)

OntoSoC : An ontology-based approach to battery pack SoC estimation

Ala Eddine Hamouni^{a,*}, Franco Giustozzi^{a,*}, Ahmed Samet^a, Ali Ayadi^a, Slimane Arbaoui^a, Tedjani Mesbahi^a

^aINSA Strasbourg, University of Strasbourg, ICube laboratory, CNRS (UMR 7357), 67000 Strasbourg, France

Abstract

A critical aspect of managing lithium-ion battery packs in electric vehicle applications is accurately determining the State of Charge (SoC). There are several methods available to estimate it, including coulomb counting with direct evaluation, Open circuit voltage, kalman filter with adaptive approach, particle filter, as well as fuzzy logic and data-based approach. In this paper, we use the state of charge data already computed by a data-driven approach and combine it with an ontology of a battery pack. The built ontology models the battery pack, taking into account the topology, types of cells and their organization inside. To make an exact estimation, different strategies of balance control of the cells are considered. SWRL rules are used to compute the state of charge of the whole battery pack. Matlab Simulink multi-physics model of a lithium-ion battery is used to provide simulated data for the experiments. The given model is evaluated based on regression metrics showing its performance.

© 2023 The Authors. Published by Elsevier B.V.

This is an open access article under the CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0>)

Peer-review under responsibility of the scientific committee of the 27th International Conference on Knowledge Based and Intelligent Information and Engineering Systems

Keywords: Li-ion battery-pack; Ontology; SWRL Rules; State of Charge (SoC).

1. Introduction

Reducing environmental pollution and combating climate change has never been more challenging. Lithium-ion (Li-ion) batteries are increasingly being used in electric vehicles and renewable energy applications [14]. To meet the demands for high energy and power requirements in various applications, lithium-ion battery packs are often composed of multiple cells that are connected in parallel and series. This configuration not only provides a durable and carbon-free solution in usage, but also allows for the optimization of performance by adjusting the pack voltage and capacity to meet specific application requirements [4]. In order to maintain the reliability and safety of the entire battery pack system, a battery management system (BMS) is implemented. This system is responsible for monitoring and controlling various parameters such as cell voltage, temperature, and SoC (State-of-Charge), as well as balancing the

* Corresponding author. Tel.: +0-000-000-0000 ; fax: +0-000-000-0000.

E-mail address: franco.giustozzi@insa-strasbourg.fr

charge and discharge of individual cells within the pack [7]. There are various factors that can impede the performance of the battery pack, including chemical impurities and unequal utilization of individual cells.

To control the cells in their operation, the BMS requires several pieces of information related to the battery. While voltage, current and temperature are easily accessible via a series of sensors, SoC and SoH (State-of-Health) are cell state quantities that cannot be directly measured.

The SoC refers to the remaining amount of energy in the battery cells of the battery energy storage system [29]. SoC cannot be measured directly as a physical quantity. Rather, it can only be estimated by measuring strongly correlated proxy quantities such as current, voltage, and temperature, with current being the most influential [20]. Typically, SoC is expressed as a value in the range of 0 to 1, or 0 to 100%, and it is defined in the literature as the ratio of the available amount of charge to the maximum amount of charge of the battery. The prediction of the SoC for the whole pack is more difficult because it must take into account several other constraints not considered on the cell level. Indeed, BMS operate SoC balancing among the battery cells to ensure the high operability of the system by avoiding overcharging/discharging of cells. An accurate estimation of the SoC of a battery can increase its remaining useful life (RuL) and prevent high Depth of Discharge (DoD) to avoid potential battery failure in the future [30]. The capacity of the whole battery pack is determined by the cell with the lowest capacity. This is because the cell with the lowest capacity will be the first to be completely discharged when the pack is discharged. Similarly, during charging, charging stops when the cell with the maximum available capacity is full, even if the other cells are not yet fully charged. Therefore, the problem of cell measure inconsistencies limits the total and the remaining capacity of the pack, which affects the SoC of the entire battery pack. Imbalanced cells are one of the most important parameters in the life of the battery to monitor, since without a balancing system, the voltages of the individual cells move away from each other over time [25].

To the best of our knowledge, there is no proposed approach using the existing battery domain ontologies for the prediction of the SoC value under several configurations. In this paper, we provide the following contributions: (i) a new ontological-based model that extends BattINFO ontology [1] with new properties; (ii) New SWRL (Semantic Web Rule Language) rules based on the SoC prediction equations; (iii) a prediction model that exploits these SWRL rules to compute the SoC value under several configurations (e.g. active or passive BMS); (iv) the OntoSoC is experimentally evaluated and validated on simulated data.

2. Background and related works

2.1. Data driven related works for State of Charge modeling

Car manufacturers must take into account the battery's lifetime when developing clean vehicles. This involves studying the aging of electric vehicle batteries, which is a crucial aspect. Battery aging can be caused by various factors such as usage patterns, temperature, and the state of charge. Therefore, it is important to understand the aging process and its impact on the battery's performance over time. This knowledge can help manufacturers optimize the battery design and management strategies to extend its lifespan and ensure optimal performance throughout the vehicle's lifetime. A few machine-learning approaches have been proposed to analyze battery signals to predict SoC, SoH and battery capacity. Most of these approaches rely on recurrent [10, 21] or convolutional neural network [27] to predict the SoC, the RUL (Remaining Useful Life) or the future SoH curves. The main limit of these approaches is that they rely on previous cycle information to make the future prediction. Additionally, these approaches claim a large volume of historical data to be accurate in their prediction, and they focus on single cells rather than at the battery pack level.

To effectively use battery packs in electric vehicles, it is important to accurately estimate their SoC and SoH, which directly affects their performance. However, due to the inherent variability in battery cells, estimating the SoC and SoH of a battery pack is more complex than for a single cell. In this direction, several works attempted to predict the SoC and SoH of the battery pack. Among these works, Zang et al. [31] propose the usage of a filter-unscented Kalman-based filter algorithm to identify the parameters of the battery pack and forecast current SoC. In particular, the extended Kalman filter is applied to update the battery pack parameters by real-time measured data, while the unscented Kalman filter is used to predict the battery pack SoC. Mawonou et al. have proposed another strategy to compute the battery pack SoC [22]. The assumption of the paper is that for an operating battery pack, only the

limiting cells' SoCs and voltages are relevant. The first approach consists of the detection of limiting cells based on voltage and current measurement to reduce the computational burden. The second is an improvement of the existing "bar-delta" approach developed. According to the authors, both of these approaches lead to significant improvement in limiting cell detection, computational burden reduction and accuracy. All the aforementioned approaches rely on time series data to predict battery SoH or SoC. Their main limitation is that do not rely on knowledge model to capture all the electronic specificity of battery to predict. In this paper, we intend to extend existing battery ontology to help in SoC prediction. To the best of our knowledge, there has been relatively little development of ontologies dedicated to batteries and battery manufacturing. Currently, two coupled ontologies exist. The Battery Interface Ontology (BattINFO) [1] and the Battery Value Chain Ontology (BVCO) [3] are both the main ontologies of the domain. BattINFO is designed to cover all knowledge related to the battery cell itself. This encompasses various aspects such as electrochemistry, battery characteristics, characterization techniques, observations, and modeling approaches. The BVCO describes the processes, materials, and equipment used in the value chain related to battery manufacturing and recycling. BVCO imports BattINFO to provide a single consistent description of a battery cell and supplements it with knowledge related to battery materials mining and processing, the battery manufacturing process steps, as well as battery second life and recycling processes. Both BattINFO and BVCO use the top-level European Materials and modeling Ontology (EMMO), which allows them to integrate with other domain ontologies stemming from EMMO.

2.2. State of Charge computing related works

In the literature, battery modeling can be classified into various categories depending on the level of detail and complexity. Some of the common categories include lumped parameter models [23], electrochemical models [2], equivalent circuit models [9], and physics-based models [19]. Another category, data-driven models, has recently grasped attention. Data-driven models cut across all areas and have been demonstrated in the context of materials discovery and lifetime estimation [12]. An alternative but complementary perspective, which captures the multi-scale nature of the tools available, considers that models can be used to predict properties from structure, predict performance from properties, or make decisions off the basis of predicted performance [13]. The model of a battery pack can be simplified by a practical model, assuming that the intrinsic imbalances of the cells and elements of the pack do not affect the characteristics of the battery. It is also possible to model a battery pack, by assembling the series-parallel configuration of a single cell's model [15].

The SoC of a battery at a given moment is the ratio between the available charge at that moment and the rated capacity [18]. Efficient use of the battery requires monitoring the different parameters, such as the SoC and the temperature. The SoC of a battery can not be measured with sensors, thus a variety of approaches are proposed to obtain accurate and robust SoC estimation in real-time, we list the most popular approaches in the literature : The coulomb counting method is the most commonly used method for SoC's estimation. The SoC is estimated by measuring the discharge current of the battery and integrating it over time [24]. The SoC is computed by the following equation:

$$SoC(t) = SoC_0(t_0) - \frac{\eta}{C_n} \int_{t_0}^t I(t) dt \quad (1)$$

where $SoC_0(t_0)$ is the initial SoC, C_n represents the battery capacity, η represents the coulombic efficiency, and $I(t)$ is the instantaneous discharge current of the battery.

Open circuit voltage (OCV) method has high precision and is easily implementable for SoC estimation. In this method, the OCV SoC relation is derived from the stepwise measurement of OCV for different values of SoC [16].

Two main classes of battery ageing exist : model-based methods, aiming to model the physical aspects of the battery to predict its behaviour, and data-driven approaches, which use historical ageing data in order to make a prediction. Data-driven models are currently thriving, as more and more data become available. The main advantage of these approaches are their simplicity, since no deep understanding of the electro-chemical processes of the battery is needed. The model-based methods are used in conjunction with adaptive filters and state estimation algorithms. The most prominent algorithms include Kalman filters with all its variants of dual Kalman filtering: Extended Kalman Filter (EKF), Unscented Kalman Filter (UKF), Particle Filter (PF) [28].

The data-driven approaches often require the use of machine learning (ML) models to establish a correlation from the data. Among well-used models, the neural network model[5]. The fuzzy logic approach is an extension of Boolean

logic that offers flexibility in a statement. This approach formalizes human reasoning using causal rules [6]. Despite, being mentioned, the problem of Cells' SoC inequality has not been seriously investigated. Ko et al. [15] have shown the possibility of accurate modeling of a battery pack by taking into account the global influences of unbalances and packaging elements. The proposed method adopts a practical model structure, which gives a fast computation in the battery management system. In addition, since the suggested method uses cell information without a manufactured battery pack, it can be useful for modeling optimal battery packs.

2.3. Battery State of Charge computing methods

Several Battery Management System (BMS) controls the batteries by estimating the SoC. BMS also applies control techniques to equalize the SoC of unbalanced cells. To avoid any overcharge or over-discharge of the batteries connected in series, it is necessary to limit the SoC between 0 and 100 %. This condition must be ensured for all the batteries to guarantee safety and long life. The relation between the pack's SoC and the cell parameters with the condition of the different balance control strategies is explained as the cell with the minimum remaining capacity is the first to be over-discharged and the cell with the minimum chargeable capacity is the first to be overcharged. Assuming $p1$ the first cell to be discharged and $p2$ the first cell to be charged, the SoC and the capacity of the cells connected in series are computed as follows [32]:

$$SoC_s = \begin{cases} \frac{SoC_{p1} \cdot C_{p1}}{SoC_{p1} \cdot C_{p1} + (1 - SoC_{p2}) \cdot C_{p2}} & \text{without balance} \\ SoC_{p1} & \text{passive balance} \\ \frac{SoC_{p1} + SoC_{p2}}{2} & \text{active balance} \end{cases} \quad (2)$$

$$C_s = \begin{cases} SoC_{p1} \cdot C_{p1} + (1 - SoC_{p2}) \cdot C_{p2} & \text{without balance} \\ C_{p1} & \text{passive balance} \\ \frac{C_{p1} + C_{p2}}{2} & \text{active balance} \end{cases} \quad (3)$$

where SoC_{p1} and SoC_{p2} are respectively the SoC of $p1$ and $p2$. C_{p1} and C_{p2} is the capacity of $p1$ and $p2$. $min(C_r)$ is the remaining cell capacity, $SoC(t)$ is the state of charge at time t , $C(t)$ is the cell capacity.

In the case of a parallel connection, we suppose that each row connected in series is a supercell. The state of charge of the pack in parallel, denoted SoC_p , is computed as follows:

$$SoC_p = \frac{\sum_{i=1}^{np} SoC_s C_s}{\sum_{i=1}^{np} C_s} \quad (4)$$

The battery pack capacity, denoted C_p , is computed as follows:

$$C_p = \sum_{i=1}^{np} C_s \quad (5)$$

3. OntoSoC : an ontology-based approach for battery pack SoC estimation

This section first provides an overview of the general architecture of the proposed approach and then details its main components as well as the interaction among them.

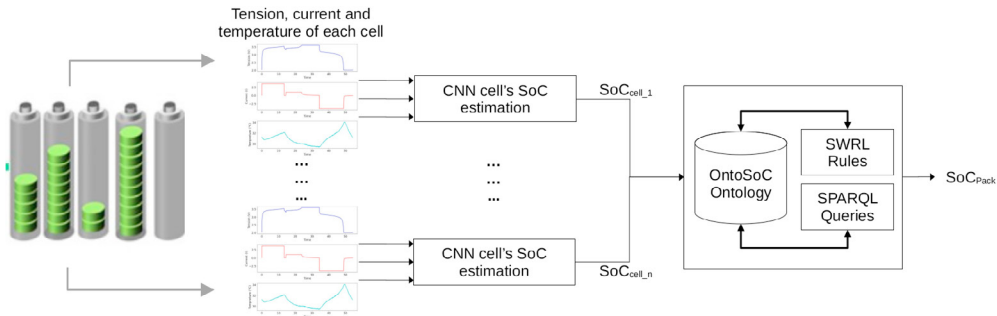


Fig. 1: The OntoSoC approach.

3.1. Overview of the proposed approach

The main objective of OntoSoC is the estimation of the battery pack SoC from voltage, current and temperature values. The approach combines data-driven methods as well as knowledge-based methods to represent the battery pack structure as well as known equations to determine the SoC of cell modules. At first, the SoC of each cell is estimated by a convolutional neural network (CNN) model [11] using as input the voltage (V), current (I) and temperature (T) of each cell. Once all SoCs are calculated, these values are inserted into our ontology and processed by a reasoner using SWRL rules, based on the existing equations introduced in section 2.3, to determine the battery pack SoC. Furthermore, the SoC of the battery pack is calculated considering the different possible configurations of the BMS (passive balance, active balance or no balance). The main components of our proposal are shown in Figure 1. In the following, the proposed semantic model is detailed, as well as the implemented SWRL rules.

3.2. OntoSoC ontology for battery pack SoC estimation

The semantic model shown in Figure 2 is based on the Battery Interface Ontology (BattINFO) [8]. BattINFO provides a chemistry-neutral description of Li-ion batteries to support data interoperability and artificial intelligence workflows. It also provides concepts and relations to describe the architecture of a battery pack. The BattINFO also allows representing how the cells are organized inside a battery pack through object properties.

Although BattINFO is mainly focused on the representation of chemical aspects of batteries, this ontology provides concepts and relations that can be reused to represent certain information about batteries, mainly regarding the architecture of a battery pack. In the following, we present these concepts and relations and also detail the concepts and relations added to the model to allow the estimation of the SoC of the battery pack. In particular, the added concepts serve to consider whether the BMS applies a control balance strategy as well as properties to represent the state of charge of the cells as well as the pack. In red font, the concepts and relations added for our purpose can be seen and are detailed below, while in black font the concepts reused from the BattINFO ontology are shown. For reasons of space and ease of reading, it was decided not to include the BattINFO prefixes in the figure or in the rules.

The word battery could be used to designate a battery cell or to designate a battery pack. Humans can easily understand the difference from the context, while machines are not able to make this difference. To distinguish these two concepts, BattINFO defines both `BatteryPack` and `BatteryCell` a subclass of `Battery` with the relation `hasPart` which indicates that a `BatteryPack` contains `BatteryCell`. The `BatteryCell` class is further subdivided into subclasses based on cell shape such as `ButtonCell`, `CylindricalCell`, `DryCell`, etc. The `Connection` class contains two instances, `Serie` and `Parallel` that allow representing how the cells or modules are connected to each other in a pack. Another important concept is the `BMS` class which represents the BMS that controls the battery, this fact is represented using the `has_Battery` relation. The BMS may or may not implement a control balancing strategy, and this control can be `Passive` or `Active`, as explained in section 2. The equations previously presented for the calculation of the SoC of a battery pack require knowing which cell has the maximum SoC value as well as which cell has the minimum SoC value. This is represented by the relations `minimum_SoC` and `maximum_SoC`, respectively. In order to represent the way cells or modules are connected within a pack, the relation `has_connection` is used. Finally,

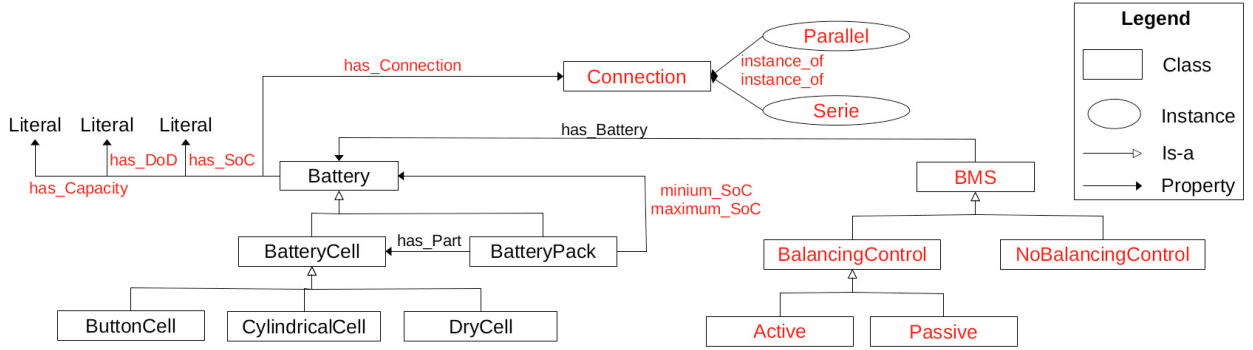


Fig. 2: Main concepts and relations of the OntoSoC ontology (in red font the concepts and relations added and in black font the concepts reused from BattINFO).

the `has_DoD`, `has_Capacity` and `has_SoC` data properties are key to our approach as they are used to indicate and calculate the SoC of each cell but also of the whole pack. The first one represents the DoD of a battery which is detailed in the next subsection, the second one allows for representing the capacity of a battery and the last one allows for representing the SoC of a battery. In particular, the data property `has_SoC` is the one used when inserting in our ontology the SoC value estimated by the CNN models for each cell that composes a battery pack.

The OntoSoC ontology is evaluated to check that it does not contain pitfalls and that it covers all the requirements identified in the introduction to calculate the SoC of a battery pack. In order to detect common mistakes done when developing ontologies we have used OOPS! [26]. The OntoSoC ontology is evaluated according to the following three categories: (i) Structural dimension focuses on mistakes detection on syntax and formal semantics; (ii) Functional dimension considers the intended use and functionality of the proposed ontology; and (iii) Usability-profiling dimension evaluates the level of ease of communication when different users use the same ontology. The evaluation of the OntoSoC ontology with OOPS! has yield some minor pitfalls, that do not affect the consistency, reasoning or applicability of the ontology.

3.3. OntoSoC SWRL rules for battery pack SoC estimation

SWRL rules facilitate expert knowledge representation in terms of OWL concepts to deliver more powerful deductive reasoning capabilities than those of OWL [17]. The SWRL rules used for battery pack SoC estimation are described below. These rules are based on the equations presented in the previous section and differentiate the cases in which different balancing modes are implemented, as well as the cell capacity and the type of connection present in the pack. These rules can be integrated into the BMS to perform state of charge estimation of the pack by combining a predictive model to determine the SoC of each cell.

The SWRL rule 6 allows calculating the value of depth of discharge (DoD) of a cell in a pack. DoD is equal to $1 - \text{SoC}$ e.g. if the $\text{SoC} = 0.6$ then $\text{DoD} = 0.4$. As long as the value of the SoC of a cell is known, then DoD can be easily calculated with a simple subtraction: $\text{DoD} = 1 - \text{SoC}$.

$$\text{BatteryCell}(\?x) \wedge \text{has_SoC}(\?x, \?w) \wedge \text{subtract}(\?b, 1, \?w) \rightarrow \text{has_DoD}(\?x, \?b) \quad (6)$$

The SWRL rule 7 allows the estimation of the SoC and capacity of a battery pack whose cells or modules are connected in series and the BMS controlling the pack does not implement any balancing control. This rule is based on the first case of equation 2 and the first case of equation 3.

$$\begin{aligned} & \text{BatteryPack}(\?p) \wedge \text{has_Connection}(\?p, \text{Serie}) \wedge \text{has_Battery}(\?bms, \?p) \wedge \\ & \text{NoBalancingControl}(\?bms) \wedge \text{maximum_SoC}(\?p, \?cmax) \wedge \text{minimum_SoC}(\?p, \?cmin) \wedge \\ & \text{has_SoC}(\?cmax, \?socmax) \wedge \text{has_SoC}(\?cmin, \?socmin) \wedge \text{has_Capacity}(\?cmin, \?capcellmin) \wedge \\ & \text{multiply}(\?m1, \?capcellmin, \?socmin) \wedge \text{has_Capacity}(\?cmax, \?capcellmax) \wedge \\ & \text{has_DoD}(\?cmax, \?DoD) \wedge \text{multiply}(\?m2, \?capcellmax, \?DoD) \wedge \text{add}(\?r, \?m1, \?m2) \wedge \\ & \text{divide}(\?SoCP, \?m1, \?r) \rightarrow \text{has_SoC}(\?p, \?SoCP) \wedge \text{has_Capacity}(\?p, \?r) \end{aligned} \quad (7)$$

In order to determine the capacity and SoC of a pack whose cells are connected in series and whose BMS implements a passive balancing control, only the SoC and capacity values of the least charged cell are needed. The rule 8 implements this scenario which corresponds to the second cases of equations 2 and 3.

$$\begin{aligned} & \text{BatteryPack}(\text{?p}) \wedge \text{minimum_SoC}(\text{?p}, \text{?cmin}) \wedge \text{has_Capacity}(\text{?cmin}, \text{?c}) \wedge \text{has_SoC}(\text{?cmin}, \text{?soc}) \\ & \wedge \text{has_Battery}(\text{?bms}, \text{?p}) \wedge \text{Passive}(\text{?bms}) \wedge \text{has_Connection}(\text{?p}, \text{Serie}) \\ & \rightarrow \text{has_SoC}(\text{?p}, \text{?soc}) \wedge \text{has_Capacity}(\text{?p}, \text{?c}) \end{aligned} \quad (8)$$

Finally, to determine the SoC and capacity of a pack whose cells or modules are connected in series and whose BMS implements an active balance control, the rule 9 is used. This rule uses the parameters of the most and least charged cells to determine their capacity and SoC, as shown in the third case of equations 2 and 3.

$$\begin{aligned} & \text{BatteryPack}(\text{?p}) \wedge \text{minimum_SoC}(\text{?p}, \text{?cmin}) \wedge \text{has_Capacity}(\text{?cmin}, \text{?cmi}) \wedge \\ & \text{has_SoC}(\text{?cmin}, \text{?smin}) \wedge \text{maximum_SoC}(\text{?p}, \text{?cmax}) \wedge \text{has_Capacity}(\text{?cmax}, \text{?cma}) \wedge \\ & \text{has_SoC}(\text{?cmax}, \text{?smax}) \wedge \text{has_Battery}(\text{?bms}, \text{?p}) \wedge \text{Active}(\text{?bms}) \wedge \\ & \text{has_Connection}(\text{?p}, \text{Serie}) \wedge \text{add}(\text{?r}, \text{?smin}, \text{?smax}) \wedge \text{divide}(\text{?SoCP}, \text{?r}, 2) \wedge \\ & \text{add}(\text{?w}, \text{?cmi}, \text{?cma}) \wedge \text{divide}(\text{?CapP}, \text{?w}, 2) \rightarrow \text{has_SoC}(\text{?p}, \text{?SoCP}) \wedge \text{has_Capacity}(\text{?p}, \text{?CapP}) \end{aligned} \quad (9)$$

The associated rules for determining the SoC and capacity of cells or modules connected in series were presented before (SWRL rules 8 and 9). Listing 1 shows the SPARQL¹ query for determining the SoC and capacity of a pack whose cells or modules are connected in parallel. It is based on the equations 4 and 5.

```

1 PREFIX ex: <http://www.semanticweb.org/OntoSoC/#>
2 SELECT (?vt/?ct AS ?SoCP) ?ct
3 WHERE {
4   { SELECT ?b (sum(?v*?cc) AS ?vt)
5     WHERE {
6       ?b a ex:BatteryPack .
7       ?b ex:has_Part ?m .
8       ?m ex:has_Soc ?v .
9       ?m ex:has_Capacity ?cc .
10    }
11   GROUP BY(?b)
12  }
13  { SELECT ?b (sum(?c) AS ?ct)
14    WHERE {
15      ?b a ex:BatteryPack .
16      ?b ex:has_Part ?m .
17      ?m ex:has_Capacity ?c .
18    }
19    GROUP BY(?b)
20  }
21 }

```

Listing 1: SPARQL query for determining the SoC and capacity of a pack whose cells or modules are connected in parallel.

The following section presents an illustrative case study that shows the application of the proposed approach and compares it with another approach.

4. Illustrative case study

In this study, we employed a model-based approach to compare it with the proposed data-based and knowledge-based OntoSoC approach. The model-based approach uses an interactive method for battery model parameter estimation, assuming that the observed battery parameters are equivalent to the actual battery parameters without any measurement errors. We tested our approach on a 2016 BMW i8 plug-in hybrid vehicle with a 96 kW electric motor.

¹ <https://www.w3.org/TR/rdf-sparql-query/>

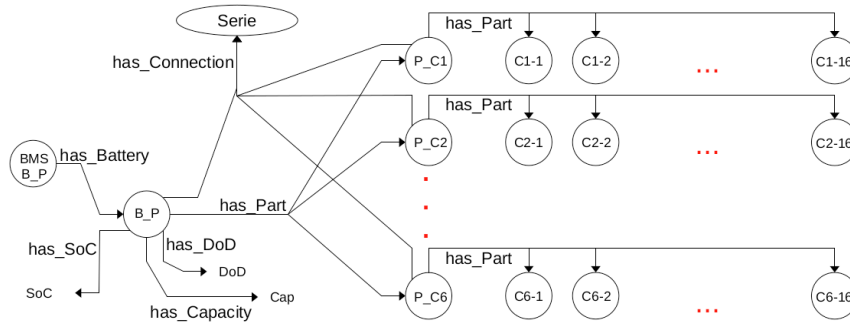


Fig. 3: Representation of the scenario presented in the case study using the OntoSoC ontology.

Figure 3 shows the instantiation of this case using the OntoSoC ontology (for reasons of space and clarity not all relations are shown). The high-voltage battery pack (B_P) has a capacity of 21 Ah (7.1 kWh). It contains a total of 96 cells, arranged in one string (1P 96S), and subdivided into 6 series-connected modules (P_C1, . . . , P_C6) of 16 series-connected cells (C1-1, . . . , C6-16). The nominal voltage of the battery pack is 355 V. The cooling is achieved by refrigerant (air conditioning), which also doubles to deliver the cabin climatic control.

The model parameters were obtained using the NEDC and WLTC driving cycles, and the accuracy and robustness of the model were evaluated by comparing the simulated results with the experimental data from a different cycle (WLTC) [9]. This allowed us to validate the model's behavior under different current profiles. The SoC, battery voltage, and battery current can be calculated using battery cell and pack models that combine mathematical and multi-physics models. These models were developed in MATLAB Simulink. We assumed that every cell in the pack was identical, balanced in climatic control, and connected in both series and parallel directions. In this study, we considered the model-based approach as the ground truth, since it is highly reliable when the parameters are accurately fixed.

The estimation error of the SoC for different balance control methods using the OntoSoC approach and the simulation made on Matlab Simulink are shown in Table 1. The errors are computed using Root Mean Square Error (RMSE), Mean Absolute Error (MAE) and the Mean Absolute Percentage Error (MAPE) measures [10]. These measures are shown in the formulas below:

$$RMSE = \sqrt{\sum_{i=1}^n \frac{(y_i - \hat{y}_i)^2}{n}} \quad (10)$$

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (11)$$

$$MAPE = \frac{1}{n} \sum_{i=1}^n \frac{|y_i - \hat{y}_i|}{y_i} \quad (12)$$

The comparison is performed on 100 battery pack SoC values that were calculated from battery cell SoC values set by an expert, i.e., 100 executions of the Matlab Simulink model and of our proposed OntoSoC model were made, based on 96 SoC values for each cell (96 different values for each of the 100 executions), to obtain the SoC of the battery pack. Once the 96 SoC values of each cell composing the B_P battery pack are determined, they are inserted into the OntoSoC ontology. Then it is determined and represented for each P_C1 to P_C6 modules what is the maximum and minimum SoC values using the `minimum_SoC` and `maximum_SoC` properties. When this information is in the OntoSoC ontology in addition to the balancing mode which is determined by the type of BMS (BMS_B_P a `Active`, BMS_B_P a `Passive` and BMS_B_P a `NoBalancingControl` for each control mode), the reasoner is executed which implies the execution of the SWRL rules presented in the previous section for the calculation of the SoC of the B_P battery pack, according to the balancing mode set up in each case (the three balancing modes were tested: without, passive and active). At the end of the reasoning process, the SoC value for each P_C1 to P_C6 modules is obtained, and also

Table 1: The estimated battery pack's SoC prediction errors.

Balance control	MAE	RMSE	MAPE
Without	0.0608	0.1903	0.0256
Active	0.0898	0.4135	0.0167
Passive	0.0412	0.1485	0.0891

the SoC of the B_P battery pack. This is represented using the data property `has_SoC`. In Figure 3, it can be seen that the value of this property for B_P is SoC, no concrete value is given since it was executed 100 times.

Table 1 compares the prediction made by OntoSoC to the prediction made by the model based approach previously described [9]. Our postulate is that the model based approach is reliable but unfortunately very time consuming and specific to only battery sizing. The MAE, RMSE and MAPE are used to capture how OntoSoC is able to follow the model based approach. By comparing the obtained errors with the simulation values presented in Table 1 we can see a difference in errors among the three control modes, with the lowest value for the passive mode. This difference is due to the number of parameters involved in each mode, for example, the estimation of SoC's pack with the active mode and without control depends on two parameters: the first overcharged cell and the first over-discharged cell. While with the passive mode, the SoC is calculated only from the parameters of the first overcharged cell as shown in equations 2 and 3, which means that few SWRL rules are used in this method, compared to the other methods, which reduces the calculation error. In general, it can be seen that our approach combining data-driven and knowledge-based methods demonstrates acceptable accuracy compared to model-based methods that need to know certain parameters beforehand to be highly reliable, a task that is not easy to perform.

5. Conclusion and future works

In this paper, an approach, named OntoSoC, to estimate the state of charge of a battery pack from voltage, current and temperature values was developed. The OntoSoC approach combines data-driven methods such as CNN models which determine the SoC of cells, and knowledge-based methods such as the OntoSoC ontology to model the battery pack architecture. Inspired by the existing battery SoC computing equations [32], the OntoSoC approach uses the OntoSoC ontology with SWRL rules and queries to compute the SoC and capacity of a battery pack whose cells or modules are connected in serie or in parallel. Furthermore, these rules and queries take into account whether or not the battery BMS implements a balancing control strategy, and whether this control is active or passive.

The proposed OntoSoC approach was tested on a real use case, a 2016 BMW i8 plug-in hybrid vehicle with a 7 kWh battery and a 96 kW electric motor. Then, we compare it with an existing physical simulation model developed in Matlab Simulink by the expert. Our use case demonstrates the feasibility of the OntoSoC approach to compute the SoC and capacity of a high-voltage battery pack with a total of 96 cells, subdivided into 6 series-connected modules of 16 series-connected cells, under different balance control modes (without, passive, and active). As well as, we evaluate the accuracy and robustness of our approach using regression metrics.

Future works will focus on improving the performance of our proposed approach and on its application to other battery models and under different usage scenarios. Hence, in addition to the proposed SWRL rules, the reasoning process of the OntoSoC ontology will be enhanced by adding new SWRL rules to consider other parameters that have an impact on the state of charge, such as the SoH of the battery pack. Additionally, another direction consists on combining our knowledge based model to trainable model based on transformer to increase the model's precision.

Acknowledgments

This project has received funding from the European Union's Horizon Europe research and innovation programme under grant agreement No GAP-101103667.

This research was funded in part by the French National Research Agency (ANR) under the project "ANR-22-CE92-0007-02"

References

- [1] BattINFO, accessed 28-March-2023. <https://github.com/BIG-MAP/BattINFO>.

- [2] Boovaragavan, V., Ramadesigan, V., Panchagnula, M.V., Subramanian, V.R., 2010. Continuum representation for simulating discrete events of battery operation. *Journal of The Electrochemical Society* 157, A98.
- [3] BVCO, accessed 28-March-2023. <https://github.com/Battery-Value-Chain-Ontology/ontology>.
- [4] Chang, F., Roemer, F., Baumann, M., Lienkamp, M., 2018. Modelling and evaluation of battery packs with different numbers of paralleled cells. *World Electric Vehicle Journal* 9, 8.
- [5] Charkhgard, M., Farrokhi, M., 2010. State-of-charge estimation for lithium-ion batteries using neural networks and ekf. *IEEE Transactions on Industrial Electronics* 57, 4178–4187.
- [6] Chau, K.T., Wu, K.C., Chan, C.C., 2004. A new battery capacity indicator for lithium-ion battery powered electric vehicles using adaptive neuro-fuzzy inference system. *Energy Conversion and Management* 45, 1681–1692.
- [7] Cheng, K.W.E., Divakar, B., Wu, H., Ding, K., Ho, H.F., 2010. Battery-management system (bms) and soc development for electrical vehicles. *IEEE transactions on vehicular technology* 60, 76–88.
- [8] Clark, S., Bleken, F.L., Stier, S., Flores, E., Andersen, C.W., Marcinek, M., Szczesna-Chrzan, A., Gaberscek, M., Palacin, M.R., Uhrin, M., et al., 2022. Toward a unified description of battery data. *Advanced Energy Materials* 12, 2102702.
- [9] Ghoulam, Y., Mesbahi, T., Wilson, P., Durand, S., Lewis, A., Lallement, C., Vagg, C., 2022. Lithium-ion battery parameter identification for hybrid and electric vehicles using drive cycle data. *Energies* 15, 4005.
- [10] Hannan, M.A., How, D.N., Lipu, M.H., Mansor, M., Ker, P.J., Dong, Z., Sahari, K., Tiong, S.K., Muttaqi, K.M., Mahlia, T.I., et al., 2021. Deep learning approach towards accurate state of charge estimation for lithium-ion batteries using self-supervised transformer model. *Scientific reports* 11, 19541.
- [11] How, D.N., Hannan, M., Lipu, M.H., Ker, P.J., 2019a. State of charge estimation for lithium-ion batteries using model-based and data-driven methods: A review. *Ieee Access* 7, 136116–136136.
- [12] How, D.N.T., Hannan, M.A., Hossain Lipu, M.S., Ker, P.J., 2019b. State of charge estimation for lithium-ion batteries using model-based and data-driven methods: A review. *IEEE Access* 7, 136116–136136.
- [13] Howey, D., Roberts, S., Viswanathan, V., Mistry, A., Beuse, M., Khoo, E., DeCaluwe, S., Sulzer, V., 2020. Free radicals: Making a case for battery modeling. *Electrochemical Society Interface* 29, 30.
- [14] Hu, X., Zou, C., Zhang, C., Li, Y., 2017. Technological developments in batteries: a survey of principal roles, types, and management needs. *IEEE Power and Energy Magazine* 15, 20–31.
- [15] Ko, S.T., Lee, J., Ahn, J.H., Lee, B.K., 2019. Innovative modeling approach for li-ion battery packs considering intrinsic cell unbalances and packaging elements. *Energies* 12.
- [16] Lee, S., Kim, J., Lee, J., Cho, B., 2008. State-of-charge and capacity estimation of lithium-ion battery using a new open-circuit voltage versus state-of-charge. *Journal of Power Sources* 185, 1367–1373.
- [17] Levy, M.A., O'Connor, M.J., Rubin, D.L., 2009. Semantic reasoning with image annotations for tumor assessment, in: *AMIA Annual Symposium Proceedings*, American Medical Informatics Association. p. 359.
- [18] Li, J., Mazzola, M.S., 2013. Accurate battery pack modeling for automotive applications. *Journal of Power Sources* 237, 215–228.
- [19] Li, X., Huang, Z., Tian, J., Tian, Y., 2021. State-of-charge estimation tolerant of battery aging based on a physics-based model and an adaptive cubature kalman filter. *Energy* 220, 119767.
- [20] Lipu, M.H., Hussain, A., Saad, M., Ayob, A., Hannan, M., 2018. Improved recurrent narx neural network model for state of charge estimation of lithium-ion battery using pso algorithm, in: *IEEE Symposium on Computer Applications & Industrial Electronics*, IEEE. pp. 354–359.
- [21] Liu, K., Shang, Y., Ouyang, Q., Widanage, W.D., 2021. A data-driven approach with uncertainty quantification for predicting future capacities and remaining useful life of lithium-ion battery. *IEEE Transactions on Industrial Electronics* 68, 3170–3180.
- [22] Mawonou, K.S.R., Eddahech, A., Dumur, D., Godoy, E., Beauvois, D., Mensler, M., 2018. Li-ion battery pack soc estimation for electric vehicles, in: *IECON 2018 - 44th Annual Conference of the IEEE Industrial Electronics Society*, pp. 4968–4973.
- [23] Nejad, S., Gladwin, D., Stone, D., 2016. A systematic review of lumped-parameter equivalent circuit models for real-time estimation of lithium-ion battery states. *Journal of Power Sources* 316, 183–196.
- [24] Ng, K.S., Moo, C.S., Chen, Y.P., Hsieh, Y.C., 2009. Enhanced coulomb counting method for estimating state-of-charge and state-of-health of lithium-ion batteries. *Applied Energy* 86, 1506–1511.
- [25] Omariba, Z.B., Zhang, L., Sun, D., 2019. Review of battery cell balancing methodologies for optimizing battery pack performance in electric vehicles. *IEEE Access* 7, 129335–129352.
- [26] Poveda-Villalón, M., Gómez-Pérez, A., Suárez-Figueroa, M.C., 2014. OOPS! (Ontology Pitfall Scanner!): An On-line Tool for Ontology Evaluation. *International Journal on Semantic Web and Information Systems (IJSWIS)* 10, 7–34.
- [27] Ren, L., Dong, J., Wang, X., Meng, Z., Zhao, L., Deen, M.J., 2021. A data-driven auto-cnn-lstm prediction model for lithium-ion battery remaining useful life. *IEEE Transactions on Industrial Informatics* 17, 3478–3487.
- [28] Shrivastava, P., Soon, T.K., Idris, M.Y.I.B., Mekhilef, S., 2019. Overview of model-based online state-of-charge estimation using kalman filter family for lithium-ion batteries. *Renewable and Sustainable Energy Reviews* 113, 109233.
- [29] Wang, W., Wang, X., Xiang, C., Wei, C., Zhao, Y., 2018. Unscented kalman filter-based battery soc estimation and peak power prediction method for power distribution of hybrid electric vehicles. *Ieee Access* 6, 35957–35965.
- [30] Xia, Q., Wang, Z., Ren, Y., Sun, B., Yang, D., Feng, Q., 2018. A reliability design method for a lithium-ion battery pack considering the thermal disequilibrium in electric vehicles. *Journal of Power Sources* 386, 10–20.
- [31] Zhang, X., Wang, Y., Yang, D., Chen, Z., 2016. An on-line estimation of battery pack parameters and state-of-charge using dual filters based on pack model. *Energy* 115, 219–229.
- [32] Zhong, L., Zhang, C., He, Y., Chen, Z., 2014. A method for the estimation of the battery pack state of charge based on in-pack cells uniformity analysis. *Applied Energy* 113, 558–564.