Structural Analysis in Reconfigurable Battery Systems for Active Fault Diagnosis

Michael Schmid, *Student Member, IEEE*, Emanuel Gebauer, and Christian Endisch, *Member, IEEE*

***Abstract*—Conventional automotive battery systems consisting of a large number of battery cells pose a variety of challenges in terms of safety, reliability, lifetime, and energy efficiency. Reconfigurable battery systems (RBSs) are a promising solution to these issues of conventional battery systems. However, the large number of components in RBS also increases the fault probability. To meet this challenge on the way to fault tolerance, this article addresses fault isolation in an RBS, which comprises two switches per cell. Based on an electrothermal model, a structural analysis is performed and a sensor set with optimal fault isolation properties is found. Since the system consists of many equations, a novel algorithm is introduced to efficiently calculate minimal structurally overdetermined (MSO) subsystems for fault diagnosis. For each fault, the algorithm allows determining the MSO set that has the least number of equations. A complexity analysis of the algorithm reveals that the proposed algorithm is computationally significantly less expensive for systems with high redundancy, such as RBS, than existing algorithms that compute all MSO sets. Since the algorithm considers the switch states, it is suitable for active fault isolation through switches. The application to the RBS shows that the electrical equations are prioritized over the thermal equations due to the model uncertainties. A video file demonstrating the proposed graph-based algorithm with an example is attached to this article.**

***Index Terms*—Active fault diagnosis, electrothermal battery model, minimal sensor set, reconfigurable battery, structural analysis.**

I. Introduction

IN ORDER to meet the global climate targets, hybrid and electric vehicles have been widely recognized as an integral part of a sustainable transportation in the sense of environmental impacts [1]. Due to their long lifespan, high power, and energy density, lithium-ion (Li-ion) cells are predestined for this application [2]. To meet the range and performance requirements, a large number of Li-ion cells are connected in parallel and serial to form a battery pack. Most of the literature on automotive battery packs deals with battery management systems [3], [4] and the related signal processing [5], [6], fast charging [7], cell balancing strategies [8], [9], and battery safety [10], [11]. In particular, the safety of battery systems is of crucial importance, since faulty operation of Li-ion cells can lead to the thermal runaway of a cell. In battery packs, the thermal runaway may also propagate [12], [13].

As a promising solution for the issues in conventional battery systems, reconfigurable battery systems (RBSs) have received increasing attention in the literature. Using dc–dc converters or switches, RBSs are dynamically reconfigured and adapted to current conditions [14], [15]. Depending on the topology used, RBSs allow modifying the cell level topology and actively adapt the battery to accommodate cell characteristics. This enables a continuous balancing of the cells with respect to the state of charge [16] and the state of health [17] as well as an energy efficient operation [18]. In case of faults, RBSs enable to bypass faulty cells in order to achieve fault tolerance (*bad block management*) [19]. In addition, RBSs are suitable for an application as hybrid storage by high frequency switching [14]. RBSs can also be used to generate a sine wave voltage by sequentially switching cells from active to bypass. A further application is therefore the alternating current battery in the form of a modular multilevel converter [20]–[23], which can be advantageously combined with the bad block management.

In RBS, however, the number of components of the battery system increases, as there are switches and electronics in addition to cells. As the number of components increases, so does the probability that one of the components will fail. For this reason, sophisticated fault diagnosis is essential in order to increase the robustness of the system. There is a wide range of literature on fault diagnosis in battery systems. A review of recent work is presented in [24] and [25]. In general, a distinction can be made between knowledge-based, data-driven, and model-based approaches. While the results of knowledge-based fault diagnosis are usually well interpretable, they require in advance a thorough understanding of the underlying fault mechanisms. Data-driven methods analyze the sensor data directly to diagnose faults. According to Tran and Fowler [24], the algorithms applied for data-driven fault diagnosis are based on Shannon entropy [26], correlation coefficient [27], wavelet transform [28], or sensor topology [29]. However, a large amount of historical data is often required and the diagnostic results are difficult to interpret because the underlying fault mechanism is neglected. Model-based approaches usually compare the sensor measurements with signals generated by a system model. The deviations, so-called *residuals*, are analyzed for fault diagnosis. Both Tran and Fowler [24] and Hu *et al.* [25] subdivide the model-based

Manuscript received October 2, 2020; revised November 20, 2020; accepted January 2, 2021. Date of publication January 6, 2021; date of current version May 5, 2021. This work was supported by the AUDI AG, Ingolstadt. Recommended for publication by Associate Editor C. N. M. Ho. *(Corresponding author: Michael Schmid.)*

The authors are with the Institute of Innovative Mobility, Technische Hochschule Ingolstadt, 85049 Ingolstadt, Germany (e-mail: michael.schmid@thi.de; emanuel.gebauer@thi.de; Christian.Endisch@thi.de).

This article has supplementary material provided by the authors and color versions of one or more figures available at https://doi.org/10.1109/TPEL.2021.3049573.

Digital Object Identifier 10.1109/TPEL.2021.3049573

methods based on state or parameter estimation [30], [31], parity space [32], [33], or structural analysis theory [34]–[36]. Liu *et al.* apply structural analysis to a battery pack [34], [35] and to cylindrical Li-ion cells [36] in order to generate residuals for fault diagnosis. However, in these previous works, only sensor faults and cooling system faults in conventional non-RBS are considered. Chen *et al.* [37] and Sampath *et al.* [38] analyze their system as a finite-state machine (FSM). It allows developing a method for active structural fault diagnosis. Both interpret all possible switch states and all possible faults as states of the FSM. Although only two possible faults per cell are considered in [37], the approach results in an exponential complexity regarding the number of cells in our case. This challenge makes it unsuitable for a large-scale RBS, whose model consists of a large number of equations. Furthermore, no validity conditions are considered and the method has not been validated experimentally. While our subsequent work [19] focuses on active fault diagnosis using state and parameter estimation, this article concentrates on the generation of residuals using structural analysis. The resulting residuals often form the basis for model-based fault diagnosis, as in [19]. Since switches are available in RBS, a promising approach is to use these switches advantageously for fault diagnosis. Fault diagnosis through the use of switches is referred to below as *active* fault diagnosis. Except for Schmid *et al.* [19] and Chen *et al.* [37], active fault diagnosis has not yet been applied to RBS, to the best of our knowledge. Even though active fault diagnosis in battery systems is a new field of application, there are approaches in other areas [39]–[41]. Gelso and Blanke [39] apply their approach exemplarily to a positioning control system of an offshore vessel. In [40], a synthetic example system is chosen and in [41], the approach is applied to a three-tank benchmark system. In all the aforementioned approaches, an active fault diagnosis is achieved by modifying the input signal of the system. However, the introduction of an additional input signal is disadvantageous in battery systems as it causes additional losses during operation of the energy storage. In order to avoid the exponential complexity of existing approaches and to solve the present issues in RBS, a novel algorithm for active structural fault diagnosis is introduced. The main contributions of this article are summarized as follows.

1) A novel algorithm for structural analysis is presented and applied to find the smallest minimal structurally overdetermined (MSO) subsystems in a module consisting of reconfigurable battery cells.
2) The complexity issues of existing algorithms for large systems with high redundancy are solved.
3) The algorithm offers the possibility to consider validity conditions that are present in RBS due to the switches.
4) The structural analysis provides the basis for a holistic fault diagnosis that considers sensor faults, actuator faults, battery system faults, and cell faults.

The remainder of this article is organized as follows. In Section II, the underlying electrothermal model of the RBS is presented. The chosen topology allows a modular design of the battery system with two switches per cell. In Section III, general definitions commonly used for structural methods are introduced and an isolability analysis is performed. The structural analysis of the RBS also allows finding a minimal sensor set with optimal structural isolation properties for the RBS. Based on the structural analysis, an algorithm for active fault isolation is first introduced using an example and then formally described (see Section IV). A heuristic is presented to reduce the computing time. In Section V, the presented algorithm is applied to the RBS and a complexity analysis is performed. Finally, Section VI concludes this article.

II. Model of the Reconfigurable Battery Module

A coupled electrothermal model is used to describe the RBS. To model the electrical behavior of the Li-ion cell, a first-order equivalent circuit model (ECM) is chosen. The thermal behavior is modeled using a second-order thermal ECM.

A. Electrical Model

Fig. 1 shows the electrical model of each cell in the RBS. The Li-ion cell is modeled by an ideal voltage source E_m, an ohmic resistance R_o, and a single resistor–capacitance element (*RC*-element) with R_p, C_p, and the time constant $\tau_\mathrm{p} = R_\mathrm{p}C_\mathrm{p}$ to take the cell dynamics into account. The parameters $E_\mathrm{m}, R_\mathrm{o}, R_\mathrm{p}$, and C_p are dependent on the state of charge SoC and temperature T_c based on a lookup table. The state-space representation

$$\begin{bmatrix} \dot{\mathrm{SoC}} \\ \dot{v_\mathrm{p}} \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & -\tau_\mathrm{p}^{-1} \end{bmatrix} \cdot \begin{bmatrix} \mathrm{SoC} \\ v_\mathrm{p} \end{bmatrix} + \begin{bmatrix} Q^{-1} \\ C_\mathrm{p}^{-1} \end{bmatrix} \cdot i_\mathrm{cell} \tag{1}$$

$$\begin{bmatrix} v_\mathrm{cell} \end{bmatrix} = \begin{bmatrix} E_\mathrm{m}/\mathrm{SoC} & 1 \end{bmatrix} \cdot \begin{bmatrix} \mathrm{SoC} \\ v_\mathrm{p} \end{bmatrix} + \begin{bmatrix} R_\mathrm{o} \end{bmatrix} \cdot i_\mathrm{cell} \tag{2}$$

of the electrical model has the SoC and the polarization voltage v_p as states. The input is the cell current i_cell, where $i_\mathrm{cell} > 0$ indicates a charging current. The output is the cell voltage v_cell [42]–[44]. The AS and the BS are modeled by ideal power switches with a small ON-state resistance $R_\mathrm{act} = R_\mathrm{act,on}, R_\mathrm{byp} = R_\mathrm{byp,on}$ and a large OFF-state resistance $R_\mathrm{act} = R_\mathrm{act,off}$, $R_\mathrm{byp} = R_\mathrm{byp,off}$. The voltage of the bypass path is given by

$$v_\mathrm{byp} = v_\mathrm{cell} + R_\mathrm{act} i_\mathrm{cell} = R_\mathrm{byp} i_\mathrm{byp}. \tag{3}$$

In order to ensure a safe operation, only one of the two switches can be in ON-state. Each terminal of the switchable cell has contact resistances, which are summarized in R_con. Six cells, as shown in Fig. 1, are interconnected to a larger RBS. Fig. 2 shows the connection of six cells, which form a module with three cells in parallel and two cells in serial (2s3p configuration). By connecting the cells, additional resistances R_bp, R_bn arise due to the busbars.

B. Thermal Model

To account for the thermal dependence of the cell parameters, the thermal behavior of the cell is also modeled. Fig. 3 shows the ECM representation of the two-state thermal model [45], [46]. T_c, T_s, and T_amb are the cell core temperature, the cell surface temperature, and the ambient temperature, respectively. C_c and C_s represent the heat capacities of the core and the surface,

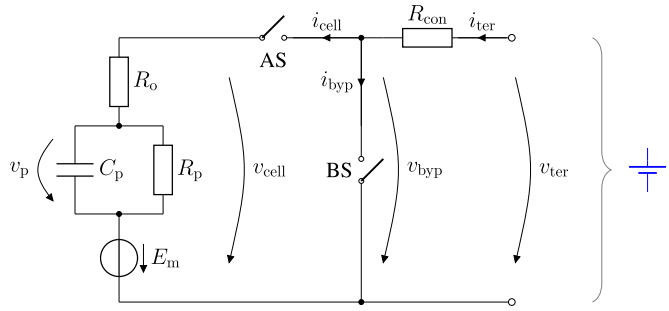

Fig. 1. RBS cell: Single *RC* model of one Li-ion cell with an active switch (AS) and a bypass switch (BS) [19].

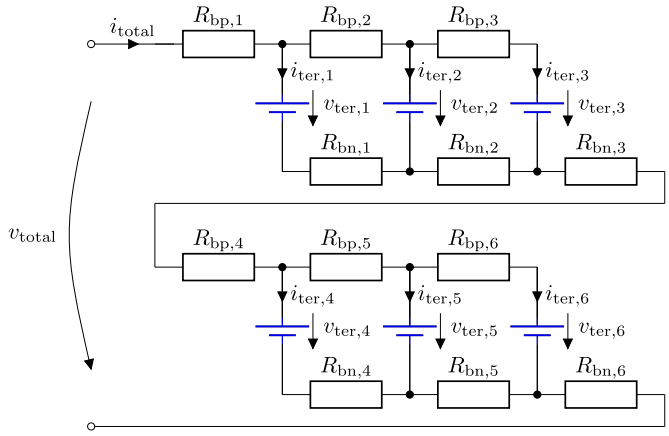

Fig. 2. Electrical connection of six RBS cells (2s3p) [19].

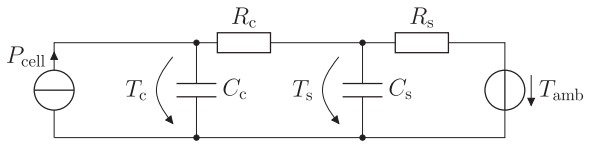

Fig. 3. Two-state thermal model for the cell core temperature T_{c} and surface temperature T_{s} with the heat generation P_{cell} and ambient temperature T_{amb} as inputs.

respectively. The heat conduction resistance of the core and the convection resistance of the surface are given by R_{c} and R_{s}, respectively. Based on the electrical model, the joule heating and dissipated power in the electrode overpotential is given as

$$P_{\text{cell}} = R_{\text{o}} i_{\text{cell}}^2 + \frac{v_{\text{p}}^2}{R_{\text{p}}}. \tag{4}$$

The governing differential equation for the two states T_{c} and T_{s} is

$$\begin{bmatrix} \dot{T}_{\text{c}} \\ \dot{T}_{\text{s}} \end{bmatrix} = \begin{bmatrix} -\tau_{\text{c}}^{-1} & \tau_{\text{c}}^{-1} \\ \tau_{\text{cs}}^{-1} & -\tau_{\text{s}}^{-1} - \tau_{\text{cs}}^{-1} \end{bmatrix} \begin{bmatrix} T_{\text{c}} \\ T_{\text{s}} \end{bmatrix} + \begin{bmatrix} C_{\text{c}}^{-1} & 0 \\ 0 & \tau_{\text{s}}^{-1} \end{bmatrix} \begin{bmatrix} P_{\text{cell}} \\ T_{\text{amb}} \end{bmatrix} \tag{5}$$

with $\tau_{\text{c}} = R_{\text{c}} C_{\text{c}}$, $\tau_{\text{s}} = R_{\text{s}} C_{\text{s}}$, and $\tau_{\text{cs}} = R_{\text{c}} C_{\text{s}}$.

III. Structural Analysis

The structural analysis of the model is motivated by the complexity of the battery system. It provides a fast and simple tool for approximating analytical solutions and allows a quick insight into a system. For model-based fault diagnosis, minimal overconstrained (MO) subsystems are often used for residual construction [47]. In the literature, there are a few algorithms to compute MOs [48]–[51]. For large systems however, these algorithms run into complexity problems [52], since the problem of finding MO subsystems is closely related to variable elimination algorithms [53]. Structural analysis provides a fast tool for approximations of MO subsystems.

A. Preliminaries and Definitions

The system $\mathcal{F} = \{\mathcal{V}, \mathcal{E}, \mathcal{K}\}$ under consideration consist of a set of unknown variables $v_i \in \mathcal{V}, i = 1, \ldots, n_{\mathcal{V}}$, equations $e_i \in \mathcal{E}, i = 1, \ldots, n_{\mathcal{E}}$, and known variables $\overline{k_i} \in \mathcal{K}, i = 1, \ldots, n_{\mathcal{K}}$. $\mathcal{F}$ is structurally represented by a bipartite graph $\mathcal{G}(\mathcal{U} \cup \mathcal{E}, \mathcal{A})$ with the unknown and known variables $\mathcal{U} = \mathcal{V} \cup \mathcal{K}$ and the set of equations $\mathcal{E}$ as nodes. $\mathcal{A}$ is the set of edges between the nodes. An edge $a_{ij} \in \mathcal{A}$ between equation $e_i \in \mathcal{E}$ and variable $v_j \in \mathcal{V}$ or $\overline{k_j} \in \mathcal{K}$ is added when v_j or $\overline{k_j}$ is included in equation e_i. As an example, consider the system

$$\begin{array}{l} e_1 : 0 = x_1^2 + 2x_2^3 + \overline{a} \\ e_2 : 0 = x_1 + 4x_2 - \overline{b} \\ e_3 : 0 = x_1 + 2(\overline{a} + \overline{b}) \\ e_4 : 0 = \dot{x}_1 + x_1 \\ e_5 : 0 = \dot{x}_1 - \frac{\delta}{\delta t} x_1 \end{array} \longrightarrow \begin{array}{ccc|cc} x_1 & \dot{x}_1 & x_2 & \overline{a} & \overline{b} \\ \hline 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{array} \tag{6}$$

with unknown variables $\mathcal{V} = \{x_1, \dot{x}_1, x_2\}$, equations $\mathcal{E} = \{e_1, \ldots, e_5\}$, and known variables $\mathcal{K} = \{\overline{a}, \overline{b}\}$. The adjacency matrix $\mathbf{A} = [\mathbf{H}, \mathbf{L}] \in \mathbb{R}^{n_{\mathcal{E}} \times (n_{\mathcal{V}} + n_{\mathcal{K}})}$ has nonempty entries for the variables of $\mathcal{V}$ and $\mathcal{K}$, which are included in an equation e_i. Similar to the work in [53]–[55], it is assumed that each fault can only violate one equation. This assumption is made without loss of generality since if a fault appears in more than one equation, it can be replaced by a new variable x_f and a new equation $f = x_f$. To be in line with Krysander [53], the structural redundancy of a system is defined as

$$\varphi_s = \text{s-rank}([\mathbf{H}, \mathbf{L}]) - \text{s-rank}(\mathbf{H}) \tag{7}$$

where $\text{s-rank}(\mathbf{A})$ is the structural rank of the matrix $\mathbf{A}$ and is defined as the cardinality of the maximal matching of the bipartite graph with the adjacency matrix $\mathbf{A}$. A subsystem $\mathcal{M} \subseteq \mathcal{E}$ is properly structurally overdetermined (PSO) if and only if

$$\varphi_s(\mathcal{M} \setminus \{e\}) < \varphi_s(\mathcal{M}) \quad \forall e \in \mathcal{M} \tag{8}$$

holds. The PSO part of $\mathcal{M}$, denoted as $\mathcal{M}^+$, is found with the Dulmage–Mendelsohn decomposition [52], [56]. A subsystem $\mathcal{M}$ is *MSO* if and only if it is PSO and $\varphi_s(\mathcal{M}) = 1$. A fault f_i is *structurally detectable* in a model $\mathcal{M}$ if $e_{f_i} \in \mathcal{M}^+$. e_{f_i} is the equation that is violated by the fault f_i [54]. For the example in (6), the MSO sets would be $\{e_1, e_2, e_3\}$, $\{e_1, e_2, e_4, e_5\}$, and $\{e_3, e_4, e_5\}$. Throughout this article, it is assumed that $x_1(0)$ is

known, thus x_1 is unambiguously calculable from $\dot{x}_1$ [57]. By forbidding the transition $\dot{x}_1 \rightarrow x_1$ in the algorithm in Section IV-B, this assumption can however be avoided. A fault f_i in model $\mathcal{M}$ is *structurally isolable* under the single fault assumption from a fault f_j if $f_j \in (\mathcal{M} \setminus \{e_{f_i}\})^+$ [54]. It should be noted that structural isolability is only an approximation for analytical isolability [53].

B. Isolability Analysis and Minimal Sensor Set of an RBS

In order to perform a full structural fault analysis, any sensor, battery, or cell component and actuator may become faulty. Faults are modeled as

$$\theta = \theta_0 + f_\theta \tag{9}$$

with θ_0 being the fault-free parameter or sensor value and f_θ the respective fault amplitude. The set of all potentially faulty variables is composed of $\theta = \theta_{\text{sensor}} \cup \theta_{\text{param}} \cup \theta_{\text{actor}}$ with the sensors

$$\theta_{\text{sensor}} \in \{i_{\text{cell},i}, i_{\text{total}}, v_{\text{cell},i}, v_{\text{total}}, T_{\text{s},i} \mid i \in [1,6]\} \tag{10}$$

the cell or system parameter

$$\theta_{\text{param}} \in \{E_{\text{m},i}, R_{\text{o},i}, R_{\text{con},i}, R_{\text{bn},i}, R_{\text{bp},i} \mid i \in [1,6]\} \tag{11}$$

and the actuators

$$\theta_{\text{actuator}} \in \{R_{\text{act},i}, R_{\text{byp},i} \mid i \in [1,6]\}. \tag{12}$$

Since no assumptions regarding the fault amplitude are made in the structural analysis, both gain and offset sensor faults are included [55]. Faults in the cell or system parameters are abnormal changes in the corresponding parameters. Similarly, actuator faults contain fault states, such as stuck-closed or stuck-open. From the set θ result the axes of the structural isolability matrix in Fig. 4. The ordinate and abscissa are identical and contain every possible fault $\mathrm{f\_}\theta$.

As in [58] and [59], it is assumed that only a single fault occurs at a time (single-fault assumption) and that there are no other faults than the ones considered (*closed-world assumption*) for the subsequent isolability analysis [54]. At first, the isolability of faults is investigated without considering the thermal model. Fig. 4, Case 1 shows the maximal structural isolability of six RBS cells. A dot (or 1 in the corresponding isolability matrix) indicates that a fault on the ordinate is not isolable from a fault on the abscissa. A unit matrix would promise perfect structural isolability. Without the thermal model, a fault isolation between the cells and the switches is possible. The contact resistances are structurally not isolable from some busbar resistances. Including the thermal model into the analysis, the structural isolability is improved, as shown in Fig. 4, Case 2. It can be seen that structurally even parameter faults inside the cell are isolable using the thermal model. The result of the structural analysis should be considered as an approximation for the fault isolation in practical applications as done in [19]. Commonly, fault isolation possibilities decrease, as there are always model uncertainties and sensor noise. Both, the ECM of the RBS cell and the modeled thermal connection of the cells, are simplified representations of the reality. Especially the possible residuals, including the temperature model, react slowly to faults in parameters or sensors.

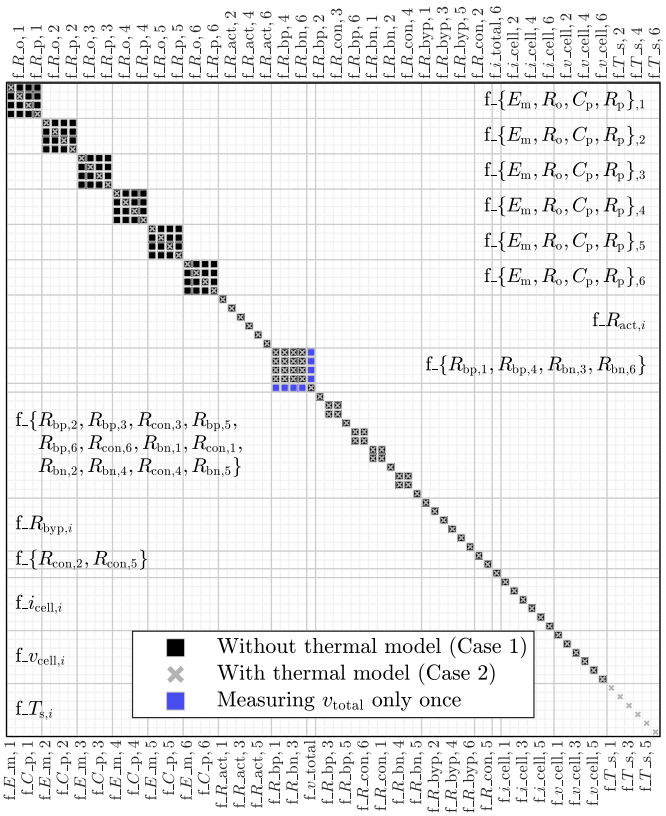

Fig. 4. Structural isolability analysis of a 2s3p battery module with all AS in ON-state and all BS in OFF-state. The labels of the ordinate and the abscissa are identical. For better readability, the faults of the abscissa are shown on two axes and the faults of the ordinate are summarized by sets and $i = 1, \ldots, 6$.

Frisk and Krysander [54] introduced a graph-based method to calculate the minimal sensor set with maximal structural isolability. The minimal sensor set $\mathcal{S}$ is defined as the multiset of measurable variables that give maximal fault isolability and every proper submultiset $s \subset \mathcal{S}$ does not. We define $i_{\text{ter},i}$, $i_{\text{cell},i}$, $v_{\text{byp},i}$, $v_{\text{cell},i}$, and $T_{\text{s},i}$, $i = 1, \ldots, 6$ as possible measurable variables by a sensor (see Figs. 1 and 3). In Case 1, the smallest minimal sensor set contains the voltage $v_{\text{cell},i}$ for each cell, the total voltage v_{total}, measured twice with two independent sensors, and a measurement of the total current i_{total}. Including the thermal model, it is sufficient to measure the temperature $T_{s,i}$ of every cell, the total voltage v_{total} measured twice and the total current i_{total} to obtain the structural isolability, as shown in Fig. 4, Case 2. However, assuming nonperfect system knowledge, the prognosis of the minimal sensor set provided by the algorithm is limited. The noise and the uncertainties can quickly overpower the influence of variables and faults.

Based on the structural analysis, the following sensor measurements are considered for all subsequent investigations: the cell voltages $v_{\text{cell},i}$, the cell currents $i_{\text{cell},i}$, and the surface temperatures $T_{\text{s},i}$ for each RBS cell $i = 1, \ldots, 6$ along with the total voltage v_{total} and total current i_{total}. Since the total voltage is only measured once, this sensor setting results in a

slight loss in structural isolability, as shown in red in Fig. 4. However, as shown in a subsequent publication [19], this can partially be restored since information on the fault sign is available.

IV. Calculation of MSO Subsystems

Residuals are required to detect a fault in the system. For the construction of residuals, overdetermined subsystems are often used in the literature [47]. Krysander *et al.* [52] show that overdetermined subsystems with redundancy $\varphi_s = 1$ are best suited for this task. Structural analysis provides a powerful tool to approximately find those systems by searching for MSO sets. There are a number of different algorithms for calculating MSO subsystems [51], [52], [60], [61]. A comparison between these different approaches is found in [47]. The mentioned approaches calculate all MSO subsystems. As Krysander *et al.* [52] point out, the maximum number of MSO subsystem increases combinatorially by $\sum_{k=n_\varepsilon-\varphi_s+1}^{n_\varepsilon} \binom{n_\varepsilon}{k}$ with $n_\mathcal{E}$ being the number of equations and the structural redundancy φ_s. For large systems with a high degree of redundancy, such as battery systems, these results in a computational exhaustive number of MSO subsystems.

Each equation in a model is affected by some model uncertainties concerning model structure and parameter uncertainties. The objective is to find the best candidates of MSO subsystems for the residual generation with respect to the minimal model uncertainties. Assuming that every equation in the system is affected by the same amount of model uncertainty, the optimal MSO subsystem to construct a residual with the least amount of model uncertainty is the one with the smallest number of equations. This assumption may generally not always be true. However, it is still intuitive to assume that the best MSO subsystems for residual generation are among the candidates with the fewest equations. Subsequently, an algorithm is proposed that guarantees to output the MSO subsystems with the fewest equations. Furthermore, the algorithm provides the possibility to include validity conditions for each equation. The following notation is used: each equation has a requirement attached, which represents its validity condition. A "1" on the ith position means that the equation is only valid if the ith switch is closed, an equation with state "0" holds only if the ith switch is open. An asterix "*" on the ith place means that the equation is independent of the state of the ith switch in the system. Therefore, an equation with a "1" on the ith position in its validity condition cannot be combined with an equation with a "0" on the ith position in its validity condition. However, it can be combined with an equation that has an asterix at the ith position.

The inclusion of validity conditions to the equations in the algorithm is motivated by the binary outcome of the switch states for the system equations. Depending on their state, some equations are simplified as currents are set to zero. An AS in OFF-state results in $i_{\mathrm{cell},i} \approx 0$—assuming the corresponding power switch is not faulty. Similarly setting the BS in OFF-state results in $i_{\mathrm{byp},i} \approx 0$. Since the proposed framework actively tries to isolate the fault, the algorithm is classified as active (structural) fault diagnosis.

Computing MSO subsystems not only allows to use previously developed residual-based schemes, such as observers [62]–[64], Kalman filter [65], [66], or parity equations [67]–[69] but also reduces the complexity of the fault diagnosis.

A. Introduction of the Greedy Algorithm

A greedy algorithm is developed on an example. The objective of the algorithm is to calculate the smallest MSO subset for each possible fault in the system. Unlike Trave-Massuyes *et al.* [61], the approach is inspired by breadth-first search instead of depth-first search. Starting from an initial fault-variable f_0, the system equations and their variables are explored. In each step, a visited variable and its occurrence in the system equations are analyzed. All nonprocessed variables in each equation are stored as visited variables. Then, the currently visited variables are rated based on the best-case estimate for the cardinality of an MSO set containing both, the respective variable and the initial fault f_0. In each subsequent step, the visited variable with the lowest MSO set estimate is processed. This is repeated, until an MSO subsystem with the initial fault is found. Thus, the proposed algorithm for MSO set calculation is similar to Dijkstra's algorithm.

Additionally, case distinctions for different equation requirements are introduced. When processing a variable x_i, each equation e_j containing x_i is analyzed separately, thus allowing to include the requirements for e_j. The variable x_i is only reachable from the fault f_0 under certain requirements. Therefore, it is calculated if the requirement of e_j is combinable with the requirement to reach the variable x_i. If it cannot be combined, the path is omitted. If it is combinable, all nonprocessed and nonvisited variables in e_j are stored as newly visited variables. For clarity, the algorithm is introduced based on the following example system.

Example IV.1:

$$
\begin{array}{lrcl}
 & & & \text{req.} \\
e_1: & 0 = x_1^2 + 2x_2 + \overline{a} & & [*,*] \\
e_2: & 0 = x_2 + 4x_3 - \overline{b} & & [1,*] \\
e_3: & 0 = x_1 + x_3 - \overline{b} & & [0,*] \\
e_4: & 0 = x_1 + x_2^3 + \overline{a}/2 & & [*,1] \\
e_5: & 0 = x_1 + x_3^2 + 2\overline{a} & & [*,0] \\
e_6: & 0 = x_2 - \overline{x_2} & - f_{x_2} & [*,*] \\
e_7: & 0 = x_3 - \overline{x_3} & - f_{x_3} & [*,*].
\end{array}
$$

The switch states (requirements, req.) where the equations $e_{\{1,\ldots,7\}}$ are valid are shown on the right. The known variables are $\mathcal{K} = \{\overline{a}, \overline{b}, \overline{x_2}, \overline{x_3}\}$. The smallest MSO subset containing the fault f_{x_3} shall be generated.

Step 1: f_{x_3} is only part of equation e_7. x_3 is the only nonvisited variable in e_7. Table I lists the data of the algorithm stored within Step 1. Since e_7 is valid for all switch states in the system, no additional requirements are needed ($[*,*]$). $\mathrm{Eq}_{\mathrm{aux}}$ are auxiliary equations needed to calculate the best-case MSO subsystem for each visited variable. For each visited, but not processed variable x_i, a temporary auxiliary equation $e_{\mathrm{aux},x_i} : x_i = \overline{x_i}$ is introduced, where x_i is the newly discovered variable and

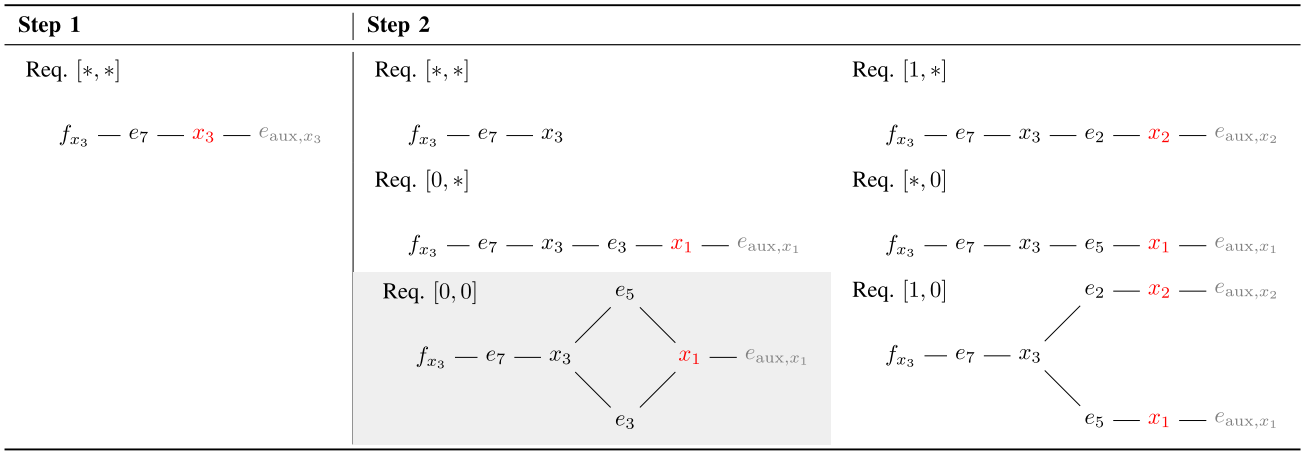

Fig. 5. Visualization of the steps of the greedy algorithm for finding the MSO subsystems with the fewest equations for the example system (see Example IV.1). Visited variables are marked with red and auxiliary equations with grey. The termination step is marked with a gray box.

TABLE I
STORED DATA OF THE ALGORITHM AFTER STEP 1

Req.	Equations	$\text{Var}_{\text{visited}}$	Eq_{aux}	MSO_{temp}
$[*, *]$	e_7	x_3	e_{aux,x_3}	$\{e_7, e_{\text{aux},x_3}\}$

TABLE II
STORED DATA OF THE ALGORITHM AFTER STEP 2

Req.	Equations	$\text{Var}_{\text{visited}}$	Eq_{aux}	MSO_{temp}
$[*, *]$	e_7			$\emptyset$
$[1, *]$	e_7, e_2	x_2	e_{aux,x_2}	$\{e_7, e_2, e_{\text{aux},x_2}\}$
$[0, *]$	e_7, e_3	x_1	e_{aux,x_1}	$\{e_7, e_3, e_{\text{aux},x_1}\}$
$[*, 0]$	e_7, e_5	x_1	e_{aux,x_1}	$\{e_7, e_5, e_{\text{aux},x_1}\}$
$[0, 0]$	e_7, e_3, e_5	x_1	e_{aux,x_1}	$\{e_7, e_3, e_5\}$
$[1, 0]$	e_7, e_2, e_5	x_2, x_1	e_{aux,x_2}, e_{aux,x_1}	$\{e_7, e_2, e_{\text{aux},x_2}\}$

Note: The resulting MSO and the termination step are marked with a gray box.

$\overline{x_i}$ its known value. The auxiliary equation Eq_{aux} represents that x_i is measured in the system, thus its best-case scenario. When the variable is processed, the belief is updated, and new equations and new visited variables needed are added. Within Step 1, the best guess for x_3 is that x_3 is measured in the system, represented by the auxiliary equation $e_{\text{aux},x_3} : x_3 = \overline{x_3}$ with $\overline{x_3}$ being a known value. The equation e_{aux,x_3} is stored in the column Eq_{aux} in Table I. The currently best-case estimate for an MSO subsystem containing f_{x_3} is therefore $\{e_7, e_{\text{aux},x_3}\}$ with the cardinality $|\{e_7, e_{\text{aux},x_3}\}| = 2$. x_3 is part of MSO_{temp}, therefore it is weighted with the respective cardinality. The current progress of the algorithm is shown in Fig. 5, Step 1. For the next step of the algorithm, the visited variable with the smallest rating is processed next. Since x_3 is the only visited variable, the choice is x_3.

Step 2: The exploration status under the different requirements is shown in Fig. 5 (Step 2) and Table II. x_3 is part of e_2, e_3, and e_5. Starting from f_{x_3}, only x_3 is reached via e_7 under the requirement $[*, *]$. e_2 requires the first switch to be closed ($[1, *]$). This path is allowed since no special switch state is needed for x_3 to be reachable starting from f_{x_3}. Therefore, the variable x_2 is added as a new visited variable under the requirement $[1, *]$. Similarly, e_3 is only valid for $[0, *]$ (first switch open). Therefore, x_1 is the only new variable in e_3 and is added to the visited variables under the requirement $[0, *]$. With e_5, the variable x_1 is further reached under the requirement $[*, 0]$. Combining the requirements $[0, *]$ and $[*, 0]$ to [0,0] allows to reach x_1 via both e_3 and e_5. Analogously, the combination of $[1, *]$ and $[*, 0]$ results in x_2 and x_1 being reachable under the requirement [1, 0] by e_2 and e_5, respectively. The algorithm terminates since the MSO set $\{e_7, e_3, e_5\}$ without an auxiliary equation is found and no MSO set estimate has a smaller cardinality. Formally with Algorithms 1 and 2, the steps are summarized in the following: In Algorithm 1, line 7, the variable becomes $Equations = \{e_2, e_3, e_5, e_7\}$. e_7 in $Equations$ is ignored, since it has already been processed (see Algorithm 2, line 10). The new requirement $Rcom$ when processing e_2 is the combination of $[*, *]$ and $[1, *]$, therefore $[1, *]$. $[1, *]$ is added to the list of requirements $\mathcal{R}$ (see Algorithm 2, line 7) and x_2 is added to the list of visited variable: $visitedVars = \{x_2\}$. Combining $V.Requirements = \{[*, *], [1, *]\}$ with the requirement of e_3 $[0, *]$ results in $[0, *]$. The variable x_1 is added to the visited variables $visitedVars = \{x_2, x_1\}$. When processing e_5, $[*, 0]$ is combined with $V.Requirements = \{[*, *], [1, *], [0, *]\}$ to $R = \{[*, 0], [0, 0], [1, 0]\}$. x_1 is already part of $visitedVars$. Similar to Step 1, for every requirement, the lower bound for the smallest MSO set estimate is calculated via Algorithm 3 (see Table II).

Remark IV.1: At first glance, this may seem like an exponential growth of possible switch states and, therefore, computational resources. However, taking a closer look, one can notice that all the equations and all visited variables of the first four requirements in the table above are part of the latter two requirements. Therefore, the smallest MSO set estimate of the latter two requirements gives a lower bound for the cardinality of the best-case MSO set estimates for the upper four requirements. Taking these estimates into account, the MSO_{temp} for the first four requirements does not need to be calculated.

Algorithm 1: `find min-MSO`($fault$).

1: $visitedVars \leftarrow [fault]$
2: $\mathcal{R} \leftarrow \{[*,*,*,\ldots,*]\}$ ▷ Set of all requirements
3: $MSOfound \leftarrow false$
4: **while** $\neg MSOfound \vee visitedVars \neq \emptyset$
5: $V \leftarrow visitedVars$. `firstElement`
6: $visitedVars$. `delete` (V)
7: $Equations \leftarrow V$. `Equations`()
8: `EvalVar` ▷ Algorithm 2
9: **for all** $\{Ri \in \mathcal{R} : Ri.newEq \wedge Ri.msoLn <$`min`$(\mathcal{R}.msoLn)\}$ **do**
10: `EvalRequirement` ▷ Algorithm 3
11: $Ri.newEq \leftarrow false$
12: $MoreGeneralR \leftarrow$ `findSubset`(Ri) ▷ Remark IV.1
13: $MoreGeneralR.msoLn \leftarrow Ri.msoLn$
14: **end for**
15: $visitedVars \leftarrow$ `sort`($visitedVars$, 'ascend,' 'by msoLn')
16: **if** MSO_{temp} with minimal length has no auxiliary equations **then**
17: $MSOfound \leftarrow true$
18: **end if**
19: **end while**

Algorithm 2: `EvalVar`.

1: **for all** $Eq \in Equations$ **do**
2: $Rcom \leftarrow$ `combine`$(Eq.Requirement, V.Requirements)$
3: **If** `isnull` $(Rcom)$ **then**
4: continue
5: **end if**
6: $\mathcal{R} \leftarrow \mathcal{R} \cup \{Rcom\}$
7: V.`Requirements`$\leftarrow V$.`Requirements`$\cup\{Rcom\}$
8: $S.newEq \leftarrow true$
9: **for all** $\tilde{R} \in Rcom$ **do**
10: **If** $Eq \notin \tilde{R}$.`Equations` **then**
11: $\tilde{R}$. `Equations` $\leftarrow \tilde{R}$. `Equations` $\cup Eq$
12: **for all** $W \in \{Eq$.`Variables`$\} \setminus \{\tilde{R}$.`Variables`$\}$ **do** ▷ W: not visited variables
13: $visitedVars \leftarrow visitedVars \cup \{W\}$
14: $\tilde{R}.Variables \leftarrow \tilde{R}.Variables \cup \{W\}$
15: **end for**
16: **end if**
17: **end for**
18: **end for**

B. Formal Description of the Algorithm

Algorithm 1 shows the formal description of the proposed algorithm. It calculates the smallest MSO subsystem for a given fault $fault$ as input. In Algorithm 1, lines 1–3, all variables are initialized. The default requirement is $[*,*,*,\ldots,*]$ (do not care). Within the while-loop (see Algorithm 1, lines 4–19), the first element in the list of currently only visited variables is taken and processed by the algorithm `EvalVar` (see Algorithm 2). In Algorithm 1, lines 9–14, a lower bound for the smallest MSO subsystems for all requirements is initially calculated or refreshed. To this, also note Remark IV.1. The algorithm terminates if an MSO set without auxiliary equation has been found and every other current MSO set estimate (with or without auxiliary equation) has a greater cardinality. Note that Algorithms 2 and 3 are seen as scripts and are presented separately for reasons of clarity.

As shown in Example IV.1, by means of Algorithm 2, every feasible equation within which the current variable occurs is visited. An equation is feasible if the requirement of the equation fits to the requirement for the processed variable to be reachable from the initial fault and it has not been visited before. In the function `combine`, additional requirements may be created if they do not already exist. The equations and variables for each new requirement are copied from the respective original requirement, e.g., in Example IV.1, Step 2: $V.Requirements = [*,*]$ with equation $\{e_7\}$, combined with $e_2.Requirements = [1,*]$, results in an additional requirement $[1,*]$ with equations $\{e_7\}$. Note that e_2 is added to the requirement later in Algorithm 2, line 11. If the requirement of the equation does not fit to the requirement of the current variable, $Rcom$ is set to $null$ and the loop terminates. Each requirement has the visited equations and visited variables saved.

The function `EvalRequirement` is inspired by Gelso *et al.* [60]. It is important to notice that for the application in this article only, the smallest MSO set containing the root-fault in every step is needed. Under this premise, the algorithm is implemented more efficiently. Gelso *et al.* [60] combine so-called basic MSO sets generated by a maximal-matching algorithm with the following rule: Let a and b be some combination of basic MSO subsystems or single basic MSO sets. A new combination c of MSO subsystems is written as

$$c = (a \cup b) \setminus ab \qquad \text{with } ab \subseteq a \cap b. \tag{13}$$

Therefore, a lower (and upper) bound for the cardinality of c can be introduced

$$|c| \geq |(a \cup b) \setminus (a \cap b)| = |a| + |b| - 2|(a \cap b)|. \tag{14}$$

Gelso *et al.* [60] show that all MSO subsystems in a system can be generated in this way. If the lower bound calculated by (14) is greater than the smallest basic MSO set containing the root fault, the combination is discarded. The lower bound computation takes very little computational power comparing to the computation of the actual MSO sets. With this lower bound, the combination of the basic MSO sets are sorted and processed in a prioritized order until the smallest MSO set has been found. The formal description of the implementation is found in Algorithm 3. In line 17, any of the algorithms presented in [51], [52], [60], and [61] can be used to calculate the MSO subsystems of the basic MSO set combination. Krysander *et al.* [52] is used in the present algorithm since it seems to be the most efficient one and allows specifying equations that need to be included in the MSO sets. Since the structural redundancy for the combination of basic MSO sets and the amount of their equations is low, the computational effort in this step is reasonable compared to the whole system. The smallest MSO subsystem or at least a lower bound for the cardinality of the respective subsystem on the current step has to be calculated for all current requirements. Note that in Algorithm 3, line 22, the length is set to ∞ if no MSO set containing the $faultEq$ could be found.

Algorithm 3: `EvalRequirement`.

1: $Eqs \leftarrow Ri.$ `Equations()` $\cup$ $Ri.$`AuxEquations()`
2: $basicMSOList \leftarrow$ `calcBasicMSO`(Eqs)
▷ maximal matching [60]
3: $MSOList \leftarrow \emptyset$
4: $faultEq \leftarrow fault.$ `Equation()`
5: $mL \leftarrow$ minimal length of basic MSO set with $faultEq$
6: **for all combinations** $\{m_1, \ldots, m_n\}$ **of** m_i **in** $basicMSOList$
▷ Each m_i is a basic MSO set → a set of equations
7: $lB \leftarrow \begin{cases} |m_1| & n = 1 \\ \sum_{i=1}^{n} |m_i| - n| \bigcap_{i=1}^{n} m_i| & n \neq 1 \end{cases}$
▷ Calculation of lower bound, see (14)
8: **If** $lB < mL \wedge faultEq \in \bigcup_{i=1}^{n} m_i$ **then**
9: $MSOList.$ `append` $(\bigcup_{i=1}^{n} m_i)$
10: $lowerBound.$ `append`(lB)
11: **end if**
12: **end for**
13: $[lowerBound, sortIdx] \leftarrow$ `sort`$(lowerBound,$ 'ascend')
14: $MSOList \leftarrow MSOList(sortIdx)$
15: $minMSOfound \leftarrow false$
16: **while** $\neg minMSOfound \vee$ `all`$(lowerBound == \infty)$**do**
17: $allMSO \leftarrow$ `calcAllMSO` ...
$(MSOList.$`firstElement`$,$ 'contains $faultEq$')
▷ use any [51], [52], [60], [61]
18: $[lB, idxMin] \leftarrow$ `min`(`ElementLength`$(allMSO)$)
19: $MSOList.$ `firstElement` $\leftarrow allMSO(idxMin)$
20: **If** $lB == lowerBound(1)$
▷ Guaranteed smallest MSO set for requirement Ri
21: $minMSOfound \leftarrow true$
22: $Ri.\text{MSO}_{temp} \leftarrow allMSO(idxMin)$
23: $Ri.msoLn \leftarrow lB$
24: $Ri.noEq\_aux \leftarrow$ `any`$(Ri.$`AuxEquations`$(\notin Ri.MSO)$
25: **else**
26: $lowerBound$ `firstElement` $\leftarrow lB$
27: **end if**
28: $[lowerBound, sortIdx] \leftarrow$ `sort`$(lowerBound,$ 'ascend')
29: $MSOList \leftarrow MSOList(sortIdx)$
30: **end while**

C. Heuristic for Faster Calculation Time

After several steps, each requirement in the algorithm can have numerous equations. For big systems, there tend to be many basic MSO subsystems in the function `EvalRequirement`. Since they are combined in the next step, this results in a combinational growth. We consider that the more basic MSO subsystems are combined into another MSO subsystem, the bigger this MSO subsystem on average tends to be. The lower bound for the combination of two different MSO sets a and b is given by (14). It is stated that the lower bound for the combination d of three basic MSO subsystems a, b, and c is

$$|d| \geq |a| + |b| + |c| - 3|(a \cap b \cap c)| > |a| + |b| - 2|(a \cap b)|. \tag{15}$$

Equality cannot hold, since $c = a \cap b$ should apply for equality, however a and b are two different MSO subsystems, and therefore, $a \cap b$ cannot be a further MSO subsystem. Hence, the lower bound rises with each additional combination. From this observation, it is stated that the more basic MSO subsystems are combined to form another MSO subsystem, the less likely it is that the respective subsystem is the overall smallest MSO subsystem. Thus, a possible heuristic is to stop after a certain number of combinations. This is done by limiting the number n of basic MSO sets in Algorithm 3, line 6. In the subsequent implementation of the algorithm, n is set to $n = 5$.

V. Application to the Reconfigurable Battery Module

The algorithm for finding the MSO sets with the fewest equations is applied to the RBS with six cells in 2s3p configuration introduced in Section II. The applicability of the MSO sets for residual generation is demonstrated on an experimental setup. Finally, the present approach is compared with the existing approaches in the form of a complexity analysis.

A. Equations

The model equations for the six cells are shown in Table III. In Section II, the switches are represented by a high or low resistance value. In the following, open switches are assumed to be ideal. That is, an open switch corresponds to an ideal open circuit. Within the module, two further constraints are added. If the BS of a cell is closed, all ASs of the cells parallel to it must be open in order not to cause a low-impedance short circuit. The second constraint is that at least one switch (AS or BS) within a parallel group of cells must always be closed, otherwise there is an open circuit in the power path. Furthermore, at least one cell in the whole battery has to be active.

The equation $e_{1,i}$ (see Table III) corresponds to the second system equation of (1) with closed AS. The additional subscript i is used to summarize the equations for each of the $i = 1, \ldots, 6$ cells. Due to the assumption regarding the ideal switches, $i_{\text{cell}} = 0$ applies with an open AS, which leads to $e_{2,i}$. From the output equation (2) for each cell follows $e_{3,i}$ when the AS is closed and $e_{4,i}$ when the AS is open. The equations $e_{5,i}$, $e_{6,i}$, and $e_{7,i}$ follow by Kirchhoff's voltage law (KVL) (see Fig. 1). The current $\widehat{i}_{\text{act,off},i}$ is zero in the fault-free case and only deviates from zero in the case of a fault (see $e_{\theta,i}$). In the fault-free case with closed AS, $i_{\text{ter},i} = i_{\text{cell},i}$ applies in accordance with Kirchhoff's current law (KCL). For a leakage current in case of a fault, $\widehat{i}_{\text{byp,off},i}$ is introduced ($e_{8,i}$). With KCL follows $e_{9,i}$ if both switches are open and $e_{10,i}$ if the AS is open. From the system equation (5) of the thermal model, $e_{11,i}$ and $e_{12,i}$ follow. The variables $\overline{R_{\text{c},i}}$, $\overline{C_{\text{c},i}}$, $\overline{R_{\text{s},i}}$, $\overline{C_{\text{s},i}}$, and $\overline{T_{\text{amb}}}$ correspond to known variables from the model or a measurement and are marked by an overline. The heat generation of each cell depends on the switch state: If the AS is closed, $e_{13,i}$ applies; if the AS is open, $e_{14,i}$ applies. By KVL follows $e_{15,i}$ and e_{16} (see Fig. 2). n_{cell} is the total number of cells, $n_{\text{par}} = 3$ the number of cells in parallel and $n_{\text{ser}} = 2$ the number of cells in serial. Depending on the position of the voltage loop either $e_{15,\alpha}$ ($\alpha \in \{i \mid \text{mod}(i, n_{\text{par}}) \neq 0\}$) or $e_{15,\beta}$ ($\beta \in \{i \mid \text{mod}(i, n_{\text{par}}) = 0\}$) applies. In equation $e_{\theta,i}$, all additional fault equations are combined according to the principle $\theta = \overline{\theta} + f_\theta$. $\overline{\theta}$ denotes the fault-free variables of θ known from the model or a measurement and f_θ denotes the fault. In this way, the equations $e_{i_{\text{cell},i}}$, ..., $e_{i_{\text{byp,off},i}}$ on the abscissa in Fig. 6 follow. In the presented system, 177 equations are thus obtained. The $3 \cdot 6 = 18$ differential relations for $v_{\text{p},i}$, $T_{\text{c},i}$, and $T_{\text{s},i}$ are not shown here.

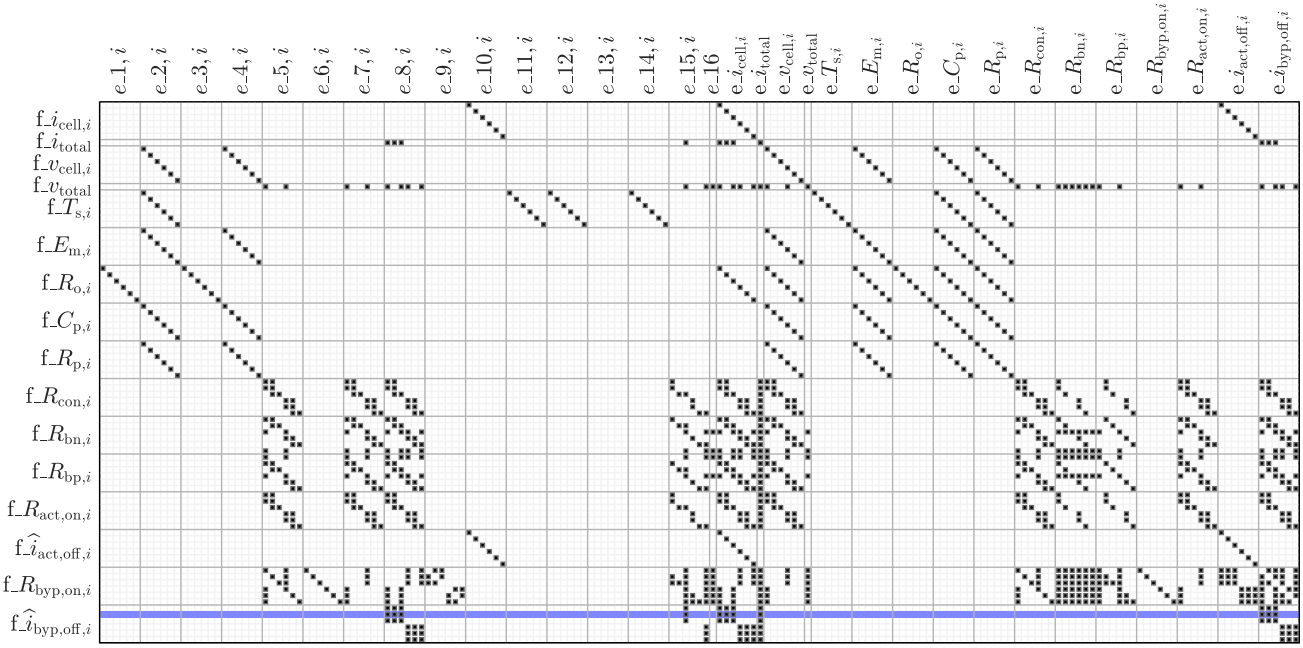

Fig. 6. MSO subsystems with the smallest number of equations for each fault: The faults are represented on the ordinate and the equations from Table III on the abscissa (summarized by $i = 1, \ldots, 6$). A black square indicates that the equation belongs to the respective MSO set of the fault. The experimentally studied fault f_$\widehat{i}_{\mathrm{byp,off},2}$ is marked in blue.

The MSO sets found require certain switch states in the system to be valid. These are either actively induced for fault diagnosis or in combination with other algorithms, such as balancing.

B. Resulting MSO Subsystems

Based on the equations in Table III, the presented algorithm is applied. For each of the 74 faults (see Fig. 4), the algorithm is used to find the smallest ten MSO subsystems. Due to the consideration of ideal switches, the leakage currents f_$\widehat{i}_{\mathrm{act,off},i}$ and f_$\widehat{i}_{\mathrm{byp,off},i}$ are added as additional faults. The 86 possible faults and the smallest MSO subsystem for each fault are shown in Fig. 6. The equations for each MSO set are shown on the abscissa.

To analyze the resulting MSO subsystems, the equations $e_{\{1,\ldots,16\},i}$ are considered. The main observation is that the equations of the thermal model $e_{\{11,\ldots,14\},i}$ are rarely used. They are only found in the MSO sets for fault diagnosis of the temperature sensor $T_{\mathrm{s},i}$. This is explained by the differential relationship via heat generation between the electrical and thermal variables. MSO subsystems with the thermal equations consist of more equations and, therefore, have larger model uncertainties. Thus, MSO subsystems without thermal equations are prioritized by the algorithm. The current sensors $i_{\mathrm{cell},i}$ and the voltage sensors $v_{\mathrm{cell},i}$ are diagnosed when the AS is open. The system equation $e_{2,i}$ of the cell is added for the voltage sensor due to the cell dynamics. Faults f_$E_{\mathrm{m},i}$, f_$R_{\mathrm{o},i}$, f_$C_{\mathrm{p},i}$, and f_$R_{\mathrm{p},i}$ within the cell are isolated only by means of the cell-specific current and voltage sensors and corresponding switch states ($e_{\{1,\ldots,4\},i}$). This is intuitive, since additional model uncertainties would be added via the remaining sensors. The fault diagnosis of the switches, contact resistances, and busbars is performed using similar subsets of equations. In addition to the cell-specific sensors and switches, the sensors of the neighboring cells ($e_{15,i}$) and the total voltage sensor (e_{16}) are used to diagnose the variables.

C. Experimental Validation

Six cylindrical Samsung cells (INR18650-25R) with a capacity of 2.5 Ah are used for the experimental tests. The cells are connected via busbars to form a 2s3p module, as shown in Fig. 2. The tests are performed in a temperature chamber at a constant temperature of 25 °C. The switches (see Fig. 1) are realized by power MOSFETs. From each cell, the current ($i_{\mathrm{cell},i}$), voltage ($v_{\mathrm{cell},i}$), and temperature ($T_{\mathrm{cell},i}$) are measured. The current measurement is shunt based. The module is excited using a Chroma Module Tester by a dynamic current profile based on the worldwide harmonized light vehicles test procedure driving cycle. Logging of the sensors is done by a Hioki 8432 data logger. To validate the presented approach, a stuck-closed fault in the BS of cell $i = 2$ is emulated. This corresponds to the fault f_$\widehat{i}_{\mathrm{byp,off},2}$ marked blue in Fig. 6. Instead of becoming high-impedance in the open state, the faulty BS has a resistance of 3.3 Ω in the open state. The fault is emulated at time $t = 610$ s. Fig. 7(a)–(c) shows the measured currents, voltages, and temperatures of each cell. Fig. 7(d) shows the residuals for each fault, which are formed from the corresponding MSO sets presented in Fig. 6. The calculation of the MSO sets and the generation of the residuals are performed offline in advance. In each system, this has to be done only once initially based on the system equations. The online fault diagnosis itself with

TABLE III
MODEL EQUATIONS FOR THE $n_{\mathrm{cell}} = 6$ CELLS $i \in \{1, 2, \ldots, n_{\mathrm{cell}}\}$, $\alpha \in \{i \mid \mathrm{mod}(i, n_{\mathrm{par}}) \neq 0\}$, $\beta \in \{i \mid \mathrm{mod}(i, n_{\mathrm{par}}) = 0\}$ WITH $n_{\mathrm{par}} = 3$ BEING THE NUMBER OF CELLS IN PARALLEL, $p_0(i) = \mathrm{floor}((i-1)/n_{\mathrm{par}}) \cdot n_{\mathrm{par}}$ AND $n_{\mathrm{ser}} = 2$ BEING THE NUMBER OF CELLS IN SERIAL, $s_0(i) = (i-1) \cdot n_{\mathrm{par}} + 1$

Equation	AS_i	BS_i
$e_{1,i}: 0 = i_{\mathrm{cell},i} C_{\mathrm{p},i}^{-1} - \dot{v}_{\mathrm{p},i} - v_{\mathrm{p},i} R_{\mathrm{p},i}^{-1} C_{\mathrm{p},i}^{-1}$	1	0
$e_{2,i}: 0 = -\dot{v}_{\mathrm{p},i} - v_{\mathrm{p},i} R_{\mathrm{p},i}^{-1} C_{\mathrm{p},i}^{-1}$	0	$*$
$e_{3,i}: 0 = E_{\mathrm{m},i} + v_{\mathrm{p},i} - v_{\mathrm{cell},i} + R_{\mathrm{o},i}\, i_{\mathrm{cell},i}$	1	0
$e_{4,i}: 0 = E_{\mathrm{m},i} + v_{\mathrm{p},i} - v_{\mathrm{cell},i}$	0	$*$
$e_{5,i}: 0 = v_{\mathrm{byp},i} - v_{\mathrm{ter},i} + R_{\mathrm{con},i}\, i_{\mathrm{ter},i}$	$*$	$*$
$e_{6,i}: 0 = v_{\mathrm{byp},i} - R_{\mathrm{byp,on},i}(i_{\mathrm{ter},i} - \widehat{i}_{\mathrm{act,off},i})$	0	1
$e_{7,i}: 0 = v_{\mathrm{cell},i} - v_{\mathrm{byp},i} + R_{\mathrm{act,on},i}\, i_{\mathrm{cell},i}$	1	0
$e_{8,i}: 0 = i_{\mathrm{ter},i} - i_{\mathrm{cell},i} - \widehat{i}_{\mathrm{byp,off},i}$	1	0
$e_{9,i}: 0 = i_{\mathrm{ter},i} - \widehat{i}_{\mathrm{byp,off},i} - \widehat{i}_{\mathrm{act,off},i}$	0	0
$e_{10,i}: 0 = i_{\mathrm{cell},i} - \widehat{i}_{\mathrm{act,off},i}$	0	$*$
$e_{11,i}: 0 = \dot{T}_{\mathrm{c},i} - P_{\mathrm{cell},i} \overline{C_{\mathrm{c},i}}^{-1} + (T_{\mathrm{c},i} - T_{\mathrm{s},i}) \overline{C_{\mathrm{c},i}}^{-1} \overline{R_{\mathrm{c},i}}^{-1}$	$*$	$*$
$e_{12,i}: 0 = \dot{T}_{\mathrm{s},i} - (T_{\mathrm{c},i} - T_{\mathrm{s},i}) \overline{C_{\mathrm{s},i}}^{-1} \overline{R_{\mathrm{c},i}}^{-1} - (\overline{T_{\mathrm{amb}}} - T_{\mathrm{s},i}) \overline{C_{\mathrm{s},i}}^{-1} \overline{R_{\mathrm{s},i}}^{-1}$	$*$	$*$
$e_{13,i}: 0 = P_{\mathrm{cell},i} - R_{\mathrm{o},i} i_{\mathrm{cell},i}^2 - v_{\mathrm{p},i}^2 R_{\mathrm{p},i}^{-1}$	1	0
$e_{14,i}: 0 = P_{\mathrm{cell},i} - v_{\mathrm{p},i}^2 R_{\mathrm{p},i}^{-1}$	0	$*$
$e_{15,\alpha}: 0 = v_{\mathrm{ter},i} - v_{\mathrm{ter},i+1} + \sum_{j=p_0(i)}^{i} \left(R_{\mathrm{bn},j} \sum_{k=p_0(i)}^{j} i_{\mathrm{ter},k} \right) - \sum_{j=p_0(i)}^{i} R_{\mathrm{bp},j} \left(i_{\mathrm{total}} - \sum_{k=p_0(i)}^{j} i_{\mathrm{ter},k} \right)$	$*$	$*$
$e_{15,\beta}: 0 = i_{\mathrm{total}} - \sum_{j=p_0(i)}^{i} i_{\mathrm{ter},j}$	$*$	$*$
$e_{16}: 0 = v_{\mathrm{total}} - \sum_{i=1}^{n_{\mathrm{cell}}} R_{\mathrm{bn},i} \sum_{k=p_0(i)}^{i} i_{\mathrm{ter},k} - \sum_{i=1}^{n_{\mathrm{ser}}} v_{\mathrm{ter},s_0(i)} - i_{\mathrm{total}} \sum_{i=1}^{n_{\mathrm{ser}}} R_{\mathrm{bp},s_0(i)}$	$*$	$*$
$e_{\theta,i}: 0 = \theta - \overline{\theta} - f_\theta,$ with $\theta \in \{ i_{\mathrm{cell},i}, i_{\mathrm{total}}, v_{\mathrm{cell},i}, v_{\mathrm{total}}, T_{\mathrm{s},i}, E_{\mathrm{m},i}, R_{\mathrm{o},i}, C_{\mathrm{p},i}, R_{\mathrm{p},i}, R_{\mathrm{con},i}, R_{\mathrm{bn},i}, R_{\mathrm{bp},i}, R_{\mathrm{byp,on},i}, R_{\mathrm{act,on},i}, \widehat{i}_{\mathrm{act,off},i}, \widehat{i}_{\mathrm{byp,off},i} \}$	$*$	$*$

Note: The requirements for the equations are shown on the right. Known variables are marked by an overline. $\overline{\theta}$ denotes the known fault-free variables of θ.

the help of the residuals requires little computing effort and is therefore real-time capable. The faulty switch results in an additional current flow of approximately 1A, which is divided between the three cells $i = 1, 2,$ and 3 [see the 0.33-A lower cell current in Fig. 7(a)]. In Fig. 7(b), a lower voltage of cells $i = 1, 2,$ and 3 is observed in comparison to cells $i = 4, 5,$ and 6. This is due to the additional load of cells $i = 1, 2,$ and 3 and the resulting voltage drop and faster discharge. The temperature remains largely unchanged. The emulated fault at $t = 610\,\mathrm{s}$ causes the residual

$$\overline{i_{\mathrm{total}}} - \sum_{j=1}^{3} \overline{i_{\mathrm{cell},j}} \neq 0 \tag{16}$$

for the fault f_$\widehat{i}_{\mathrm{byp,off},2}$ to become unequal to zero. Hereby, $\overline{i_{\mathrm{total}}}$ and $\overline{i_{\mathrm{cell},j}}$ are the known, measured variables of the corresponding sensors. Equation (16) is nonzero, because an additional current arises, which is not measured by the sensor $\overline{i_{\mathrm{total}}}$. Accordingly, a significant deviation of the residual for fault f_$\widehat{i}_{\mathrm{byp,off},2}$ is observed in Fig. 7(d). In general, the emulated fault could cause further residuals to deviate from zero, when the current $\widehat{i}_{\mathrm{byp,off},2}$ is also included. In this case, however, a clear deviation from zero is only observed for one residual. This confirms that the assumption is justified that the MSO subsystem with the smallest number of equations also has a good sensitivity to the fault.

D. Discussion

In the following, the computational complexity for the calculation of the MSO subsets is discussed in comparison to existing algorithms that compute all MSO subsets in the system as in [51], [52], [60], and [61]. Fig. 8 shows in blue the number of steps for each fault needed by the algorithm presented in Section IV. Hereby, a step is defined as each time the algorithm passes line 5 in Algorithm 1. It can be seen that the number of steps for most faults is independent on the number of cells in the system. This is the strength of the algorithm, since due to the bottom-up approach, it is inherently local to the fault under investigation. If an MSO subset is found close to the fault under investigation, the algorithm quickly terminates and does not consider the rest of the system. However, for faults that need knowledge about the whole system as faults in the total voltage or current sensor or faults on the busbars in between the rows, the algorithm has to consider all equations in the system, thus resulting in an increased computational load. For systems without validity conditions in the equations, such as conventional non-RBS, the bottleneck in this calculation is the combination of basic MSO sets in line 10 in Algorithm 3. By limiting the number of basic MSO sets to be combined to $n = 1$ (see Section IV-C), this bottleneck can be overcome, but with compromises in the accuracy of the algorithm, since it can no longer be guaranteed that the resulting MSO set is the smallest MSO set for the examined fault. In Fig. 8, the resulting number of steps for a heuristic of $n = 1$ are plotted in dotted lines.

For systems with validity conditions, such as RBS, another bottleneck arises: The number of requirements stored in the algorithm can grow very fast for the nonlocal faults (e.g., f_v_{total} or _i_{total}). Although not all MSO sets need to be evaluated (see Remark IV.1), this can quickly lead to storage problems. Each active and BS of each cell has three possible requirements (I): closed (1), open (0), or do not care ($*$). Some requirements are not valid, as not all cells can be in bypass mode (II), not all cells in a row are allowed to be in idle (III), and if a cell in one row is in active mode, none of the other cells in the same row is allowed to be in bypass mode (IV). Thus, each row has

$$n_{\mathrm{row}}(p) = \underbrace{3^{2p}}_{\mathrm{I}} - \underbrace{\sum_{i=1}^{p} \left(\binom{p}{i} \sum_{j=1}^{p} \binom{p}{j} \right)}_{\mathrm{IV}} - \underbrace{1}_{\mathrm{III}} \tag{17}$$

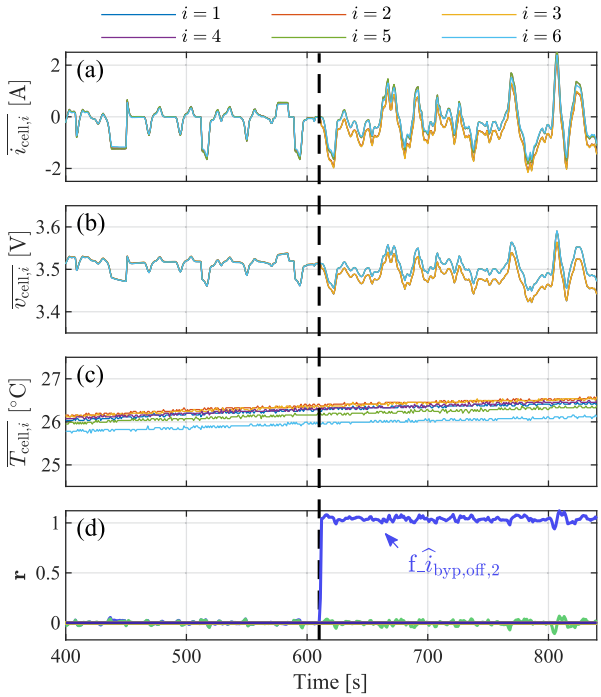

Fig. 7. Experimental results for the emulated fault f_$\widehat{i}_{\mathrm{byp,off},2}$. From top to bottom: (a) measured cell currents, (b) cell voltages, (c) cell temperatures, and (d) value of the residuals $\mathbf{r}$.

different possible requirements. With s rows, this results in

$$n_{\mathrm{req}}(s,p) = (n_{\mathrm{row}}(p))^s - \underbrace{1}_{\mathrm{II}} \tag{18}$$

different requirements. However, the computational restrictions should be seen in context to the conventional algorithms that calculate all MSO sets in the system [51], [52], [60], [61]. The exact number of all MSO sets in the system is difficult to calculate analytically. However, a lower bound for the RBS with p cells in parallel and s rows in series presented in Section II can be estimated. The estimate is based on the assumption that at least one MSO set results from each loop (for example, via KCL or KVL) containing two sensors. This lower bound is shown as a black dashed line in Fig. 8. It can be observed that the total number of MSO sets in the system quickly surpasses the number of steps needed by the algorithm proposed in this article. Thus, leading to the conclusion that for systems with a large redundancy as the RBS presented in Section II, the bottom-up approach presented in this article is computationally significantly less expensive than conventional algorithms that calculate all MSO sets in the system, even though only very few are actually needed for fault diagnosis. However, if the redundancy in the system is low, the conventional algorithms presented in [51], [52], [60], and [61] may show a better performance than the algorithm presented here.

The RBS presented here has two switches per cell (see Fig. 1). In an industrial application, however, it would also be conceivable to use only one BS for the cells connected in parallel, as this would provide the same functionality. However, it should be considered that the BS must then be able to carry the total current of the battery system, also from a thermal point of view. If this requires a parallel connection of several switches, the component saving is reduced or is not present. A further advantage of the two-switch solution is the modular design of the system (see Figs. 1 and 2), which allows, for example, electrically intrinsically safe cells to be implemented.

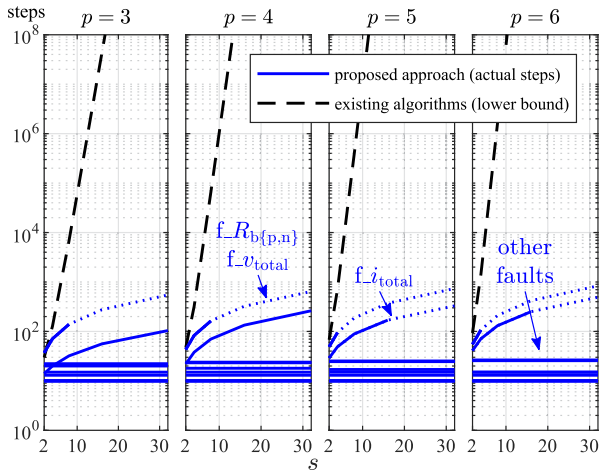

Fig. 8. Complexity analysis for battery systems: Steps needed by the algorithm to calculate the MSO sets dependent on the number of parallel (p) and serial (s) cells in the battery system (blue) and lower bound for the number of MSO sets computed by the existing algorithms (black).

VI. Conclusion

In this article, a structural analysis for fault diagnosis in an RBS based on an electrothermal model was presented. The isolability analysis has shown that the use of a thermal model increases the fault isolability from a structural point of view. However, it must be taken into account that the added equations of the thermal model contain additional parameters and model uncertainties. Based on the identified isolability matrix, a minimal sensor set with optimal isolability properties is presented. This consists of a measurement of each cell voltage, each cell current, each cell temperature, and the double measurement of the total current and total voltage in the system. An algorithm was developed to find the best MSO set for fault diagnosis in RBS. The algorithm efficiently identifies the MSO sets with the fewest equations for each fault. A comparison with existing approaches to find MSO subsystems shows the superiority of the presented algorithm in systems with many equations and high redundancy. This property opens up the algorithm beyond its application in RBS and makes it promising for all systems that have a large number of model equations and high redundancy, resulting in many MSO subsystems. Assuming that each model equation has the same uncertainty, the MSO sets found are optimal. However, if model uncertainties are known for each equation, the algorithm can be further developed by taking into account the actual model uncertainties. The application of the algorithm for finding the smallest MSO sets has shown that the thermal model equations are not used for fault diagnosis, even if valid MSO sets exist. This is because the MSO sets containing the thermal equations are larger than those containing the electrical equations. Therefore, an additional uncertainty is to be expected.

In a subsequent publication [19], residuals are generated from the resulting MSO sets and used for active fault isolation in RBS. Based on real measurement data, it will be shown that the fault isolation capability is increased by switches.

ACKNOWLEDGMENT

The authors would like to thank Dr. M. Hinterberger (AUDI AG, Ingolstadt) for his constructive support and valuable expert advice during the planning and development of the research work presented in this article.

REFERENCES

[1] "A European strategy for low-emission mobility," European Commission, Brussels, Belgium, Jul. 20, 2016.
[2] C. Wu, C. Zhu, Y. Ge, and Y. Zhao, "A review on fault mechanism and diagnosis approach for Li-ion batteries," *J. Nanomater.*, vol. 2015, pp. 1–9, 2015.
[3] L. Lu, X. Han, J. Li, J. Hua, and M. Ouyang, "A review on the key issues for lithium-ion battery management in electric vehicles," *J. Power Sources*, vol. 226, pp. 272–288, 2013.
[4] W. Waag, C. Fleischer, and D. U. Sauer, "Critical review of the methods for monitoring of lithium-ion batteries in electric and hybrid vehicles," *J. Power Sources*, vol. 258, pp. 321–339, 2014.
[5] D. Schneider, U. Vögele, and C. Endisch, "Model-based sensor data fusion of quasi-redundant voltage and current measurements in a lithium-ion battery module," *J. Power Sources*, vol. 440, 2019, Art. no. 227156.
[6] B. Liebhart, L. Komsiyska, and C. Endisch, "Passive impedance spectroscopy for monitoring lithium-ion battery cells during vehicle operation," *J. Power Sources*, vol. 449, 2020, Art. no. 227297.
[7] D. Anseàn *et al.*, "Fast charging technique for high power $LiFePO_4$ batteries: A mechanistic analysis of aging," *J. Power Sources*, vol. 321, pp. 201–209, 2016.
[8] M. M. Hoque, M. A. Hannan, A. Mohamed, and A. Ayob, "Battery charge equalization controller in electric vehicle applications: A review," *Renewable Sustain. Energy Rev.*, vol. 75, pp. 1363–1385, 2017.
[9] J. Qi and D. Dah-ChuanLu, "Review of battery cell balancing techniques," in *Proc. Australas. Univ. Power Eng. Conf.*, 2014, pp. 1–6.
[10] X. Feng, M. Ouyang, X. Liu, L. Lu, Y. Xia, and X. He, "Thermal runaway mechanism of lithium ion battery for electric vehicles: A review," *Energy Storage Mater.*, vol. 10, pp. 246–267, 2018.
[11] S. M. Rezvanizaniani, Z. Liu, Y. Chen, and J. Lee, "Review and recent advances in battery health monitoring and prognostics technologies for electric vehicle (EV) safety and mobility," *J. Power Sources*, vol. 256, pp. 110–124, 2014.
[12] X. Feng *et al.*, "Thermal runaway propagation model for designing a safer battery pack with 25 Ah $LiNi_x Co_y Mn_z O_2$ large format lithium ion battery," *Appl. Energy*, vol. 154, pp. 74–91, 2015.
[13] X. Feng, L. Lu, M. Ouyang, J. Li, and X. He, "A 3D thermal runaway propagation model for a large format lithium ion battery module," *Energy*, vol. 115, pp. 194–208, 2016.
[14] T. Zimmermann, P. Keil, M. Hofmann, M. F. Horsche, S. Pichlmaier, and A. Jossen, "Review of system topologies for hybrid electrical energy storage systems," *J. Energy Storage*, vol. 8, pp. 78–90, 2016.
[15] S. Ci, N. Lin, and D. Wu, "Reconfigurable battery techniques and systems: A survey," *IEEE Access*, vol. 4, pp. 1175–1189, 2016.
[16] N. Bouchhima, M. Schnierle, S. Schulte, and K. P. Birke, "Active model-based balancing strategy for self-reconfigurable batteries," *J. Power Sources*, vol. 322, pp. 129–137, 2016.
[17] N. Bouchhima, M. Gossen, S. Schulte, and K. P. Birke, "Lifetime of self-reconfigurable batteries compared with conventional batteries," *J. Energy Storage*, vol. 15, pp. 400–407, 2018.
[18] N. Bouchhima, M. Schnierle, S. Schulte, and K. P. Birke, "Optimal energy management strategy for self-reconfigurable batteries," *Energy*, vol. 122, pp. 560–569, 2017.
[19] M. Schmid, E. Gebauer, C. Hanzl, and C. Endisch, "Active model-based fault diagnosis in reconfigurable battery systems," *IEEE Trans. Power Electron.*, vol. 36, no. 3, pp. 2584–2597, Mar. 2021.
[20] S. D'Arco, L. Piegari, and P. Tricoli, "A modular converter with embedded battery cell balancing for electric vehicles," in *Proc. Elect. Syst. Aircraft, Railway, Ship Propulsion*, 2012, pp. 1–6.
[21] S. D'Arco, L. Piegari, and P. Tricoli, "Power and balancing control considerations on modular multilevel converters for battery electric vehicles," in *Proc. 15th Eur. Conf. Power Electron. Appl.*, 2013, pp. 1–9.
[22] M. Quraan, P. Tricoli, S. D Arco, and L. Piegari, "Efficiency assessment of modular multilevel converters for battery electric vehicles," *IEEE Trans. Power Electron.*, vol. 32, no. 3, pp. 2041–2051, Mar. 2017.
[23] F. Helling, J. Glück, A. Singer, H.-J. Pfisterer, and T. Weyh, "The ac battery—A novel approach for integrating batteries into ac systems," *Int. J. Elect. Power Energy Syst.*, vol. 104, pp. 150–158, 2019.
[24] M.-K. Tran and M. Fowler, "A review of lithium-ion battery fault diagnostic algorithms: Current progress and future challenges," *Algorithms*, vol. 13, no. 3, 2020, Art. no. 62.
[25] X. Hu, K. Zhang, K. Liu, X. Lin, S. Dey, and S. Onori, "Advanced fault diagnosis for lithium-ion battery systems: A review of fault mechanisms, fault features, and diagnosis procedures," *IEEE Ind. Electron. Mag.*, vol. 14, no. 3, pp. 65–91, Sep. 2020.
[26] Z. Wang, J. Hong, P. Liu, and L. Zhang, "Voltage fault diagnosis and prognosis of battery systems based on entropy and z -score for electric vehicles," *Appl. Energy*, vol. 196, pp. 289–302, 2017.
[27] B. Xia, Y. Shang, T. Nguyen, and C. Mi, "A correlation based fault detection method for short circuits in battery packs," *J. Power Sources*, vol. 337, pp. 1–10, 2017.
[28] X. Kong, Y. Zheng, M. Ouyang, L. Lu, J. Li, and Z. Zhang, "Fault diagnosis and quantitative analysis of micro-short circuits for lithium-ion batteries in battery packs," *J. Power Sources*, vol. 395, pp. 358–368, 2018.
[29] Y. Kang, B. Duan, Z. Zhou, Y. Shang, and C. Zhang, "A multi-fault diagnostic method based on an interleaved voltage measurement topology for series connected battery packs," *J. Power Sources*, vol. 417, pp. 132–144, 2019.
[30] A. Singh, A. Izadian, and S. Anwar, "Fault diagnosis of Li-ion batteries using multiple-model adaptive estimation," in *Proc. 39th Annu. Conf. IEEE Ind. Electron. Soc.*, 2013, pp. 3524–3529
[31] A. Sidhu, A. Izadian, and S. Anwar, "Adaptive nonlinear model-based fault diagnosis of Li-ion batteries," *IEEE Trans. Ind. Electron.*, vol. 62, no. 2, pp. 1002–1011, Feb. 2015.
[32] J. Marcicki, S. Onori, and G. Rizzoni, "Nonlinear fault detection and isolation for a lithium-ion battery management system," in *Proc. ASME Dyn. Syst. Control Conf.*, 2010, pp. 607–614. [Online]. Available: https://cecas.clemson.edu/ sonori/Publications/DSCC2010-4085.pdf
[33] W. Lombardi, M. Zarudniev, S. Lesecq, and S. Bacquet, "Sensors fault diagnosis for a BMS," in *Proc. Eur. Control Conf.*, 2014, pp. 952–957.
[34] Z. Liu, Q. Ahmed, G. Rizzoni, and H. He, "Fault detection and isolation for lithium-ion battery system using structural analysis and sequential residual generation," in *Proc. ASME Dyn. Syst. Control Conf.*, 2014, vol. 2, pp. 1–10.
[35] Z. Liu, H. He, Q. Ahmed, and G. Rizzoni, "Structural analysis based fault detection and isolation applied for a lithium-ion battery pack," *IFAC-PapersOnLine*, vol. 48, no. 21, pp. 1465–1470, 2015.
[36] Z. Liu, Q. Ahmed, J. Zhang, G. Rizzoni, and H. He, "Structural analysis based sensors fault detection and isolation of cylindrical lithium-ion batteries in automotive applications," *Control Eng. Pract.*, vol. 52, pp. 46–58, 2016.
[37] Z. Chen, F. Lin, C. Wang, Y. Le Wang, and M. Xu, "Active diagnosability of discrete event systems and its application to battery fault diagnosis," *IEEE Trans. Control Syst. Technol.*, vol. 22, no. 5, pp. 1892–1898, Sep. 2014.
[38] M. Sampath, S. Lafortune, and D. Teneketzis, "Active diagnosis of discrete-event systems," *IEEE Trans. Autom. Control*, vol. 43, no. 7, pp. 908–929, Jul. 1998.
[39] E. R. Gelso and M. Blanke, "Structural analysis extended with active fault isolation—Methods and algorithms," *IFAC Proc. Vol.*, vol. 42, no. 8, pp. 597–602, 2009.
[40] N. Poulsen and H. Niemann, "Active fault diagnosis based on stochastic tests," *Int. J. Appl. Math. Comput. Sci.*, vol. 18, no. 4, pp. 487–496, 2008.
[41] A. Mesbah, S. Streif, R. Findeisen, and R. D. Braatz, "Active fault diagnosis for nonlinear systems with probabilistic uncertainties," *IFAC Proc. Vol.*, vol. 47, no. 3, pp. 7079–7084, 2014.
[42] M. Schmid, U. Vögele, and C. Endisch, "A novel matrix-vector-based framework for modeling and simulation of electric vehicle battery packs," *J. Energy Storage*, vol. 32, 2020, Art. no. 101736.
[43] T. Bruen and J. Marco, "Modelling and experimental evaluation of parallel connected lithium ion cells for an electric vehicle battery system," *J. Power Sources*, vol. 310, pp. 91–101, 2016.
[44] T. O. Ting, K. L. Man, N. Zhang, C.-U. Lei, and C. Lu, "State-space battery modeling for smart battery management system," in *Proc. Int. Multiconf. Eng. Comput. Scientists*, 2014, vol. 2, pp. 12–15.

[45] C. Forgez, D. Vinh Do, G. Friedrich, M. Morcrette, and C. Delacourt, "Thermal modeling of a cylindrical $LiFePO_4$/graphite lithium-ion battery," *J. Power Sources*, vol. 195, no. 9, pp. 2961–2968, 2010.
[46] X. Lin *et al.*, "A lumped-parameter electro-thermal model for cylindrical batteries," *J. Power Sources*, vol. 257, pp. 1–11, 2014.
[47] J. Armengol *et al.*, "Minimal structurally overdetermined sets for residual generation: A comparison of alternative approaches," *IFAC Proc. Vol.*, vol. 42, no. 8, pp. 1480–1485, 2009.
[48] M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki, *Diagnosis and Fault-Tolerant Control*. Berlin, Germany: Springer, 2016.
[49] M. Krysander and M. Nyberg, "Structural analysis utilizing MSS sets with application to a paper plant," in *Proc. 13th Int. Workshop Princ. Diagnosis*, 2002, pp. 1–7.
[50] S. Ploix, M. Désinde, and S. Touaf, "Automatic design of detection tests in complex dynamic systems," *IFAC Proc. Vol.*, vol. 38, no. 1, pp. 478–483, 2005.
[51] B. Pulido and C. A. Gonzalez, "Possible conflicts: A compilation technique for consistency-based diagnosis," *IEEE Trans. Syst., Man, Cybern.*, *Part B*, vol. 34, no. 5, pp. 2192–2206, Oct. 2004.
[52] M. Krysander, J. Aslund, and M. Nyberg, "An efficient algorithm for finding minimal overconstrained subsystems for model-based diagnosis," *IEEE Trans. Syst., Man, Cybern., Part A: Syst. Humans*, vol. 38, no. 1, pp. 197–206, Jan. 2008.
[53] M. Krysander, "Design and analysis of diagnosis systems using structural methods," Ph.D. dissertation, *Linköpings Universitet*, Dept. Institutionen för systemteknik, Linköping, Sweden, 2006.
[54] E. Frisk and M. Krysander, "Sensor placement for maximum fault isolability," in *Proc. 18th Int. Workshop Princ. Diagnosis*, 2007, pp. 106–113.
[55] M. Krysander and E. Frisk, "Sensor placement for fault diagnosis," *IEEE Trans. Syst., Man, Cybern., Part A: Syst. Humans*, vol. 38, no. 6, pp. 1398–1410, Nov. 2008.
[56] A. L. Dulmage and N. S. Mendelsohn, "Coverings of bipartite graphs," *Can. J. Math.*, vol. 10, pp. 517–534, 1958.
[57] D. Düştegör, E. Frisk, V. Cocquempot, M. Krysander, and M. Staroswiecki, "Structural analysis of fault isolability in the DAMADICS benchmark," *Control Eng. Pract.*, vol. 14, no. 6, pp. 597–608, 2006.
[58] Z. Liu and H. He, "Sensor fault detection and isolation for a lithium-ion battery pack in electric vehicles using adaptive extended Kalman filter,"*Appl. Energy*, vol. 185, pp. 2033–2044, 2017.
[59] D. Jung, H. Khorasgani, E. Frisk, M. Krysander, and G. Biswas, "Analysis of fault isolation assumptions when comparing model-based design approaches of diagnosis systems," *IFAC-PapersOnLine*, vol. 48, no. 21, pp. 1289–1296, 2015.
[60] E. R. Gelso, S. M. Castillo, and J. Armengol, "An algorithm based on structural analysis for model-based fault diagnosis," in *Proc. Conf. Artif. Intell. Res. Develop.*, pp. 138–147, vol. 2008, 2008.
[61] L. Trave-Massuyes, T. Escobet, and X. Olive, "Diagnosability analysis based on component-supported analytical redundancy relations," *IEEE Trans. Syst., Man, Cybern., Part A: Syst. Humans*, vol. 36, no. 6, pp. 1146–1160, Nov. 2006.
[62] S. Dey, S. Mohon, P. Pisu, and B. Ayalew, "Sensor fault detection, isolation, and estimation in lithium-ion batteries," *IEEE Trans. Control Syst. Technol.*, vol. 24, no. 6, pp. 2141–2149, Nov. 2016.
[63] C. Edwards, S. K. Spurgeon, and R. J. Patton, "Sliding mode observers for fault detection and isolation," *Automatica*, vol. 36, no. 4, pp. 541–553, 2000.
[64] J. Xu, J. Wang, S. Li, and B. Cao, "A method to simultaneously detect the current sensor fault and estimate the state of energy for batteries in electric vehicles," *Sensors*, vol. 16, no. 8, 2016, Art. no. 1328.
[65] Z. Liu and H. He, "Model-based sensor fault diagnosis of a lithium-ion battery in electric vehicles," *Energies*, vol. 8, no. 7, pp. 6509–6527, 2015.
[66] C. Zheng, Y. Ge, Z. Chen, D. Huang, J. Liu, and S. Zhou, "Diagnosis method for Li-ion battery fault based on an adaptive unscented Kalman filter," *Energies*, vol. 10, no. 11, 2017, Art. no. 1810.
[67] R. J. Patton and J. Chen, "A review of parity space approaches to fault diagnosis," *IFAC Proc. Vol.*, vol. 24, no. 6, pp. 65–81, 1991.
[68] I. Fagarasan and S. St Iliescu, "Parity equations for fault detection and isolation," in *Proc. IEEE Int. Conf. Autom., Qual., Testing, Robot.*, 2008, vol. 1, pp. 99–103.
[69] E. Chow and A. Willsky, "Analytical redundancy and the design of robust failure detection systems," *IEEE Trans. Autom. Control*, vol. AC-29, no. 7, pp. 603–614, Jul. 1984.

Michael Schmid (Student Member, IEEE) was born in Germany. He received the B.Sc. and M.Sc. degrees in electrical and computer engineering in 2014 and 2017, respectively, from the Technical University of Munich (TUM), Munich, Germany, where he is currently working toward the Ph.D. degree in electrical and computer engineering.

Since October 2017, he has been with the Institute of Innovative Mobility, Technische Hochschule Ingolstadt, Ingolstadt, Germany, where he is currently a Research Associate in a research project with the AUDI AG, Ingolstadt, Germany. His current research interests include data-driven modeling, system identification, simulation, and machine learning applied in battery systems.

Emanuel Gebauer was born in Buchloe, Germany, in 1995. He received the B.Sc. and M.Sc. degrees in electrical and computer engineering from the Technical University of Munich, Munich, Germany, in 2016 and 2020, respectively. He is currently working toward the Ph.D. degree in electrical and computer engineering with the Institute of Innovative Mobility, Technische Hochschule Ingolstadt, Ingolstadt, Germany.

He is currently a Research Associate with the Institute of Innovative Mobility, Technische Hochschule Ingolstadt, working in a research project with the AUDI AG, Ingolstadt, Germany. His current research interests include data-driven system modeling and identification, artificial intelligence applied to production optimization, and fault diagnosis.

Christian Endisch (Member, IEEE) received the Engineering and Ph.D. degrees in electrical engineering from the Technical University of Munich (TUM), Munich, Germany, in 2003 and 2009, respectively.

In 2010, he joined the AUDI AG, Ingolstadt, Germany, where he became a Project Manager in 2011. Since 2011, he has been researching and teaching in the field of artificial intelligence with the Chair for Electrical Drive Systems and Power Electronics, TUM. Since 2013, he has been with the Institute of Innovative Mobility (IIMo), Technische Hochschule Ingolstadt (THI), Ingolstadt, Germany, where he was appointed to a Professor in 2013 and a Research Professor in 2014. Since 2016, he has been the Head of the IIMo, THI. His research interests include learning battery systems, system identification, optimization strategies, networked mobility, predictive operating strategies, innovative manufacturing, and testing techniques with learning systems.