Physics-Informed Operator Learning for Parameter Estimation in Lithium-Ion-Battery Models Enhanced by Global Experimental Design and Local Identifiability Analysis

Philipp Brendel*[,1], Christopher Straub[1], Andreas Rosskopf[1], Vincent Lorentz[1,2], and Felix Dietrich[3]

[1]Fraunhofer Institute for Integrated Systems and Device Technology IISB, 91058 Erlangen, Germany
[2]University of Bayreuth, Chair of Electronics for Electrical Energy Storage, 95440 Bayreuth, Germany
[3]Technical University of Munich, TUM School of Computation, Information and Technology, 85748 Garching, Germany

January 29, 2026

Abstract

Accurate parameter estimation in lithium-ion battery models such as the Single-Particle-Model (SPM) enables accurate state estimation in lithium-ion batteries but is often hindered by a lack of parameter identifiability under certain operating conditions. This paper presents a parametrized physics-informed deep operator network (PI-DeepONet) that is trained to predict solutions of the SPM for various current profiles and varying electrochemical parameters such as diffusivities, active material volume fractions and stoichiometric limits with an average Root-Mean-Squared-Error (RMSE) in voltage responses below 1 mV – even when extrapolating to unseen drive cycles. Based on this result, an iterative parameter estimation algorithm is proposed that combines global optimal experimental design, local identifiability analysis, and the differential evolution algorithm, and only becomes computationally tractable by exploiting the inference speed and differentiability of PI-DeepONet. The proposed algorithm recovers a synthetic ground truth for seven degradation-related parameters with an averaged percentage error of 3.4 % and offers a speedup factor of 265 on a standard processor and more than 30,000 on a state-of-the-art graphics processor compared to a finite-difference approach based on the reference solver *PyBaMM*. This result paves the way for more wholistic and reliable state estimation algorithms for lithium-ion batteries.

1 Introduction

Accurate State-of-Health (SOH) estimation in lithium-ion batteries (LIB) is essential for predicting remaining useful life, preventing unexpected failures, and ensuring safe and reliable operation in various applications from electric vehicles to energy storage systems [1, 2]. To tackle this, several modeling approaches have been introduced, from lightweight Equivalent Circuit Models (ECM), to the physics-based Single Particle Model (SPM), its electrolyte-enhanced version (SPMe), and the Doyle-Fuller-Newman (DFN) model, each offering a certain trade-off between numerical simplicity and electrochemical fidelity [3, 4, 5]. Previous works have shown that SOH and specific degradation modes such as Loss-of-Active-Material (LAM) or Loss-of-Lithium-Inventory (LLI) can be correlated with the change in model parameters like stoichiometries at 0 % and 100 % State-of-Charge (SOC), active material volume fractions or the Open-Circuit-Voltage (OCV) [6, 7, 8, 9]. However, estimating such parameters in practice

*corresponding author: philipp.brendel@iisb.fraunhofer.de

can be difficult, as measurable quantities such as cell voltage or temperature may yield little to no information about these parameters under certain operating conditions. In [10], Andersson et al. have given a thorough review on parameter estimation from "input-output data" in battery models, concluding that wholistic procedures are needed that cover various aspects such as sensitivity and identifiability analyses, optimal design of experiments (DoE), and suitable optimization methods. More specifically, they emphasize that many of these methods are commonly applied only locally in parameter space, i.e., right at or close to some ground truth values of the considered parameters, which makes results difficult to compare, especially when dealing with non-linear models. This problem is amplified by varying assumptions on the chosen model, the considered range of parameters and applied current profiles and causes a lack of robust methods that can easily be applied to new scenarios with different operating conditions or other assumptions. Global approaches on sensitivity analysis and experimental design address these issues by analyzing the parameter space more rigorously and are starting to gain more traction in context of LIB model parametrization [11, 12, 13]. However, these approaches require significantly more computational efforts, which generally hinders their direct applicability for parameter estimation in realistic environments like Battery-Management-Systems (BMS).

The advancements of Machine-Learning (ML) and, more recently, Physics-Informed Neural Networks (PINN) have shown a lot of potential to reduce the computational efforts in classical simulations due to their generalizability and high inference speed after training [14, 15]. While the canonical PINN approach is trained to approximate individual solutions to a given partial differential equation (PDE), the addition of PDE parameters as explicit inputs to the network yields parametrized PINNs that approximate a family of PDE solutions [16]. Similarly, the concept of Operator Learning aims at the approximation of operators, i.e., mappings between function spaces, and has been tackled in a physics-informed manner via the notions of Physics-Informed Deep Operator Networks (PI-DeepONet) and Physics-Informed Neural Operators (PINO) [17, 18]. Input functions for Operator Learning are commonly tailored to represent application-specific boundary or initial conditions [19, 20]. In the context of LIB models, the applicability of PINNs has been shown for the ECM, the SPM, the SPMe, and even the DFN model given suitable architectural modifications and advanced training schemes [21, 22, 23, 24, 25]. Moreover, Operator Learning for LIB models has been applied in a data-driven, purely physics-informed, or hybrid manner, e.g., for DoE methodologies and parameter or state estimation [24, 26, 27, 28, 29, 30].

This work aims to close the gap between these previous efforts and the application of Operator Learning for LIB parameter estimation by considering a high-dimensional parameter space and introducing a novel estimation approach that exploits the inference speed and differentiability of PI-DeepONets. More precisely, a parametrized PI-DeepONet is trained to approximate solutions to a parametric formulation of the SPM with varying electrode-specific solid-phase diffusivities, active material volume fractions, stoichiometric limits at 0 % and 100 % SOC, and randomly generated current profiles. After an initial – significant, but one-time – training effort, the PI-DeepONet can be used as a fast surrogate model for the SPM and also to efficiently infer gradient information in a global manner throughout the parameter domain – even for application-specific current profiles that have not been part of the training process. Finally, an iterative parameter estimation algorithm is proposed that includes global optimal experimental design, population-based parameter estimation via Differential Evolution (DE) and local identifiability analysis based on Fisher-Information. The algorithm shows more robustness than standard, non-iterative procedures and only becomes computationally tractable by relying on the trained PI-DeepONet and its automatic differentiability paving the way for more wholistic and reliable state estimation in LIBs.

2 Methodology

In Section 2.1, the SPM as the fundamental model to this work is introduced, and, in Section 2.2, the PI-DeepONet that is trained to solve the SPM for different current profiles and electrochemical parameters. Moreover, in Section 2.3, optimality criteria based on Fisher-Information are introduced as local or global metrics for parameter sensitivity, identifiability, and experimental design. Finally, a novel parameter estimation algorithm enhanced by global experimental design and local identifiability analyis is proposed in Section 2.4.

2.1 Single-Particle-Model (SPM)

The SPM is derived from Fick's law of diffusion and describes the time-dependent lithium concentrations $c_j(r,t)$ in two spherically symmetric particles with radial dimensions $r \in [0,\, R_j]$ for anode ($j = n$) and cathode ($j = p$), respectively [4]. The SPM is governed by no-flux boundary conditions at the particle centers ($r = 0$) and Neumann boundary conditions at the particle surfaces ($r = R_j$) describing the flux of lithium in dependence on the applied current profile $I(t)$, i.e.,

$$\frac{\partial c_j(r,t)}{\partial t} = \frac{1}{r^2}\frac{\partial}{\partial r}\left(D_j r^2 \frac{\partial c_j(r,t)}{\partial r}\right) , \tag{1}$$

$$\frac{\partial c_j(0,t)}{\partial r} = 0 , \tag{2}$$

$$\frac{\partial c_j(R_j,t)}{\partial r} = \pm\frac{I(t)}{AL_jFD_ja_j} , \tag{3}$$

where D_j denotes the electrode-specific solid-phase diffusivity of lithium, A the electrode surface area, L_j the electrode thickness, F Faraday's constant, and $a_j = 3\frac{\varepsilon_j}{R_j}$ the specific interfacial area calculated from active material volume fractions ε_j and radii R_j. Following convention, positive currents $I(t)$ discharge the cell, and 100 % SOC is defined by a fully lithiated anode and fully de-lithiated cathode. This is reflected in Eq. (3) by a negative sign for $j = n$, and a positive sign for $j = p$. The SPM is completed by initial conditions that are related to the initial SOC, denoted as SOC_0, and the four stoichiometric limits $\theta_j^{0\%}, \theta_j^{100\%}$ that define the fraction of the maximum concentration $c_j^{\max}$ at 0 % and 100 % SOC, respectively:

$$c_n(r,0) = c_n^0 = \left(\theta_n^{0\%} + SOC_0\left(\theta_n^{100\%} - \theta_n^{0\%}\right)\right) c_n^{\max} , \tag{4}$$

$$c_p(r,0) = c_p^0 = \left(\theta_p^{100\%} + SOC_0\left(\theta_p^{0\%} - \theta_p^{100\%}\right)\right) c_p^{\max} . \tag{5}$$

The time-dependent voltage response is derived from the concentrations at the particle surfaces $c_j^{\text{surf}}(t) = c_j(r = R_j, t)$ and the applied current profile $I(t)$ via the material-specific open-circuit-potentials (OCP) U_j^{OCP} and overpotentials η_j as

$$V = U_p^{\text{OCP}}\left(c_p^{\text{surf}}\right) - U_n^{\text{OCP}}\left(c_n^{\text{surf}}\right) - \eta_p - \eta_n , \tag{6}$$

$$\eta_j = \frac{2RT}{F}\sinh^{-1}\left(\frac{I}{a_j\overline{j}_{0,j}L_j}\right) , \tag{7}$$

$$\overline{j}_{0,j} = (c_j^{\text{surf}})^{\frac{1}{2}}(c_j^{\max} - c_j^{\text{surf}})^{\frac{1}{2}} , \tag{8}$$

where R denotes the universal gas constant and T the temperature which is assumed to be constant throughout this work. Unless specified otherwise, all reference parameters in this work are obtained from experimental results by Chen et al., cf. [31], who characterized a cylindrical 21700 cell with a NMC811 cathode and a Graphite-SiO anode, as described in Tab. A1 in the Appendix.

A common assumption in battery modeling identifies 0 % and 100 % SOC with predefined voltage limits that aim to protect the battery from excessive degradation and safety risks due to overcharging or deep discharging. In context of the SPM, this means that the stoichiometric limits ($\theta_n^{0\%}$, $\theta_n^{100\%}$, $\theta_p^{0\%}$, $\theta_p^{100\%}$) do not only affect the initial conditions in Eq. (4) and Eq. (5), but must also yield voltage responses that respect the voltage cut-offs at 0 % and 100 % SOC via Eq. (6). This is commonly achieved by scaling and shifting the underlying OCPs to fit a given voltage response, which is often referred to as OCV-fitting and allows to relate the change in stoichiometric limits with specific degradation mechanisms like LLI or LAM, cf. [9, 32, 33, 34]. Following these lines of thoughts, we scale and shift the OCPs reported by Chen et al., cf. [31], according to stoichiometric limits ($\theta_n^{0\%}, \theta_n^{100\%}, \theta_p^{0\%}, \theta_p^{100\%}$) such that the reported voltage limits of 2.5 V and 4.2 V are matched at 0 % and 100 % SOC, respectively, cf. Eqs. (A1)–(A4) in the Appendix. Furthermore, as the LLI during a single cycle is expected to be negligible, the amount of

cyclable lithium between stoichiometric limits in both electrodes must match at all times to be physically meaningful, i.e.,

$$A_n c_n^{\max} L_n \varepsilon_n (\theta_n^{100\%} - \theta_n^{0\%}) = A_p c_p^{\max} L_p \varepsilon_p (\theta_p^{0\%} - \theta_p^{100\%}) \ . \tag{9}$$

Parameters $(\varepsilon_n, \varepsilon_p, \theta_n^{0\%}, \theta_n^{100\%}, \theta_p^{0\%}, \theta_p^{100\%})$ have recently been referred to as electrode-specific SOH (eSOH) parameters due to their direct and linear relation to SOH, LLI, and LAM [6, 7]. Motivated by this, the remainder of this work is focused on estimating these parameters as accurately as possible to enable accurate prediction of degradation mechanisms from voltage responses in realistic application scenarios of LIBs. Note that we also aim to estimate diffusivity values D_n, D_p as they play a crucial role in the SPM and have been shown to span several orders of magnitude, e.g., depending on SOC, temperature, or degradation mechanisms [6, 35].

2.2 Parametrized PI-DeepONet

Figure 1 shows the parametrized Physics-Informed Deep Operator Network (PI-DeepONet) that is trained to approximate solutions of the SPM for spatio-temporal coordinates $(r,\ t)$, scalar parameters $\mathbf{\Theta} = (D_n,\ D_p,\ \varepsilon_n,\ \varepsilon_p,\ \theta_n^{100\%},\ \theta_n^{0\%},\ \theta_p^{100\%},\ SOC_0)$ and current profiles $I(t)$ for a predefined time interval of 600 seconds. PI-DeepONet is composed of two sub-networks referred to as "branch" and "trunk", re-

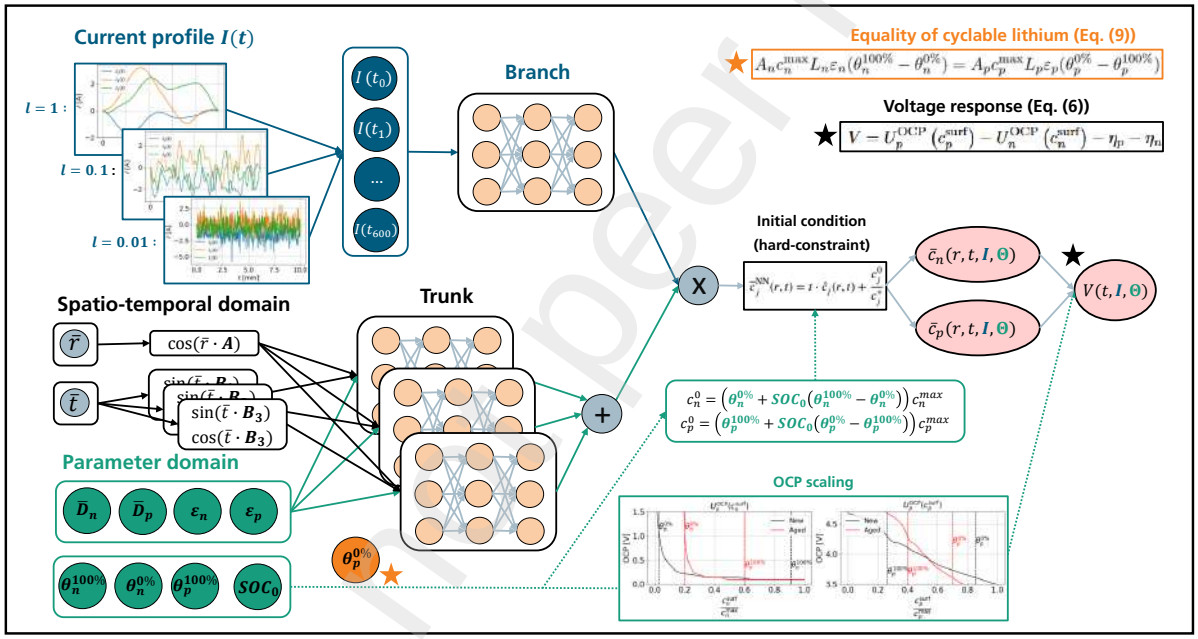

Figure 1: Parametrized PI-DeepONet trained to solve the SPM for varying current profiles generated as Gaussian Random Fields (top left) and scalar parametric inputs including diffusivities, active material volume fractions, and stoichiometric limits (bottom left).

spectively, cf. [19]. The branch is used to encode the discretized input function space for current profiles $I(t)$ and the trunk to encode the coordinate and parameter space. As we assume fixed values for A_j, L_j, and $c_j^{\max}$ throughout this work, one of the eSOH parameters, in this case $\theta_p^{0\%}$, highlighted in orange, can be determined from the remaining eSOH parameters via Eq. (9) and is therefore removed from the input domain of PI-DeepONet. In the following subsections, we describe the relevant architectural features with respect to a vanilla DeepONet and our previous work on parametrized PI-DeepONets in [28].

2.2.1 Input function space

In contrast to [28], we train a single PI-DeepONet with branch inputs solely based on Gaussian Random Fields (GRF), which are generated from the covariance kernel

$$k(t_i, t_j) = \exp\left(-\frac{2\,\sin^2\left(\pi\frac{\|t_i - t_j\|_2}{p}\right)}{l^2}\right) \,. \tag{10}$$

The length scale l is randomly chosen from $\{1.0, 0.1, 0.01\}$ and determines the time scale at which values within the profile are correlating to each other. Therefore, smaller values of l yield faster oscillations and higher values in the corresponding frequency spectrum, cf. top left in Fig. 1. Branch inputs are discretized at a resolution of 1 Hz, i.e., each profile is represented by 601 values which are interpolated linearly during training of PI-DeepONet. As we show in Section 3, this input function space is sufficient to achieve a good generalization and extrapolation accuracy at inference for battery-specific current profiles that were not part of the training profiles generated from Eq. (10).

2.2.2 Fourier feature embeddings

Similar to our previous work, cf. [28], and first introduced by Wang et al., cf. [36], we use spatio-temporal multi-scale Fourier feature embeddings in the trunk of PI-DeepONet. Due to the increased complexity and diversity of branch inputs and corresponding solutions, the temporal Fourier feature matrix B is split into three matrices B_1, B_2, B_3 which are sampled from Gaussian distributions with standard deviations $\sigma_{B_1} = 10$, $\sigma_{B_2} = 100$ and $\sigma_{B_3} = 1000$, and connected to three individual sub-networks, cf. Fig. 1. This adaption accounts for the more oscillatory dynamics in solutions to high-frequency inputs (e.g., $l = 0.01$ in Eq. (10)), most notably at the particle surfaces ($r = R_j$) where the Neumann boundary condition in Eq. (3) scales linearly with $I(t)$.

2.2.3 Hard-constraining techniques and data-free training

Due to the linearity of the SPM, a considerable amount of inputs, more precisely $(\theta_n^{100\%}, \theta_n^{0\%}, \theta_p^{100\%}, SOC_0)$, is only affecting the initial conditions in Eq. (4) and Eq. (5). As first introduced by Lu et al., cf. [37], such initial conditions can be hard-constrained into the architecture of PI-DeepONet by transforming the outputs of the last layer $\hat{c}_j(r,t)$ via

$$\bar{c}_j^{\mathrm{NN}}(r,t) = t \cdot \hat{c}_j(r,t) + \frac{c_j^0}{c_j^*} \,. \tag{11}$$

Technically, the aforementioned parameters could also be merged into lumped inputs c_n^0, c_p^0 to reduce the dimensionality of the input parameter space of PI-DeepONet, but as we aim to estimate all stoichiometric parameters directly in the scope of this work, we stick to the high-dimensional input domain presented in Fig. 1. A further crucial adaption is the hard-constraining of the homogeneous Neumann boundary condition in Eq. (2). This property can be enforced by only sampling cosine terms in the spatial Fourier feature embedding, as depicted in Fig. 1, such that spatial derivatives vanish at $r = 0$ by design, cf. recent work by Straub et al. [38]. This approach removes a further loss term from the training process of PI-DeepONet that is thereby only governed by minimizing the purely physics-informed, data-free loss $\mathcal{L}_{\mathrm{tot}}$ that is composed of the residuals of Eq. (1) and Eq. (3) for each electrode:

$$\mathcal{L}_{\mathrm{PDE}}^j = \frac{1}{N_I N_\Omega} \sum_{m=0}^{N_I} \sum_{i=0}^{N_\Omega} \left| \frac{\partial \bar{c}_j(\overline{x}_i,\, I_m)}{\partial \bar{t}} - \frac{2}{\bar{r}} \frac{\partial \bar{c}_j(\overline{x}_i,\, I_m)}{\partial \bar{r}} - \frac{\partial^2 \bar{c}_j(\overline{x}_i,\, I_m)}{\partial^2 \bar{r}} \right|^2 \,, \tag{12}$$

$$\mathcal{L}_{\mathrm{BC}}^j = \frac{1}{N_I N_{\mathrm{BC}}} \sum_{m=0}^{N_I} \sum_{i=0}^{N_{\mathrm{BC}}} \left| \frac{\partial \bar{c}_j(\overline{x}_i,\, I_m)}{\partial \bar{r}} \pm \frac{I_m(\bar{t})\; r^*}{A L_j F D_j a_j c^*} \right|_{\bar{r}=1}^2 \,, \tag{13}$$

$$\mathcal{L}_{\mathrm{tot}} = \sum_{j \in \{n,\, p\}} \lambda_{\mathrm{PDE}} \mathcal{L}_{\mathrm{PDE}}^j + \lambda_{\mathrm{BC}} \mathcal{L}_{\mathrm{BC}}^j \,. \tag{14}$$

The notation above with nondimensionalized collocation points $\overline{x} = (\bar{r},\, \bar{t},\, \overline{\Theta})$ and non-dimensionalized concentrations $\bar{c}_j$ follows the nondimensionalization scheme discussed more thoroughly in our previous work, cf. [28]. We refer to the Appendix A.2 for the hyperparameters used in the training of PI-DeepONet, such as network sizes, optimizer settings, amounts of sampled input functions (N_I), collocation points on the $r = R_j$ boundary (N_{BC}), and within the full domain of collocation points (N_Ω).

2.2.4 Differentiable voltage post-processing and OCP scaling

Equations (6)–(8) relate the predicted surface concentration profiles $c_j(r = R_j, t)$ to the voltage response $V(t)$ and can be added as a further transformation layer into PI-DeepONet, cf. Fig. 1. Similarly, the scaling of OCPs based on stoichiometric inputs $(\theta_n^{0\%}, \theta_n^{100\%}, \theta_p^{0\%}, \theta_p^{100\%})$ can be enforced in the same manner via the transformation described in Appendix A.1. As the loss in Eq. (14) is independent of voltage, these transformation layers do not affect training performance, but enable fast inference of voltage responses after training. Moreover, this procedure enables automatic differentiation of voltage responses with respect to the parametric inputs, which is paramount for the following subsections that are focused on parameter sensitivities, identifiabilities and optimal design of experiments for parameter estimation.

2.3 Parameter sensitivity, identifiability and global experimental design

The implementation of PI-DeepONet, cf. Appendix A.2, allows automatic differentiation (AD) of the networks outputs with respect to its inputs including parameters $\mathbf{\Theta}$. Hence, the trained PI-DeepONet can be used to efficiently analyze parameter sensitivities, identifiabilities, and various experimental designs. We consider $I(t)$ and SOC_0 as experimental design variables in this work as they are two of the main influence factors for parameter sensitivity and identifiability [39, 40]. Hence, they can and should be used for optimizing design of experiments or, e.g., in an on-board automotive application, for selecting the most informative sequences from a larger set of measurement data. Fisher-Information-Matrices (FIMs) provide a mathematical framework that quantifies the information provided by a given experiment for a set of parameters $\mathbf{\Theta}$. They are commonly computed from sensitivity matrices $\mathbf{S}$, composed of derivatives of some measurable outputs with respect to $\mathbf{\Theta}$, and the covariance matrix of the measurement error $\mathbf{Q}$ as

$$\mathbf{F} = \mathbf{S}^T \mathbf{Q} \mathbf{S} \ . \tag{15}$$

While our approach in earlier work was limited to analysing FIMs based on surface concentrations c_j^{surf}, cf. [28], we now consider voltage responses $\mathbf{V}(I,\, SOC_0,\, \Theta) = [V(t_0), V(t_1), ..., V(t_{600})]$ that are discretized at a resolution of 1 Hz over the considered time interval of 600 seconds. Hence, the FIM can be computed for any experimental design (I, SOC_0) and parameters $\mathbf{\Theta}$ as

$$\mathbf{F}_{i,\,j}(I,\, SOC_0,\, \mathbf{\Theta}) = \left[\left(\frac{\partial \mathbf{V}(I,\, SOC_0,\, \mathbf{\Theta})}{\partial \Theta_i} \right)^T \left(\frac{\partial \mathbf{V}(I,\, SOC_0,\, \mathbf{\Theta})}{\partial \Theta_j} \right) \right] \ , \tag{16}$$

where we assume no covariance in the output error, i.e., $\mathbf{Q} = \mathbb{1}$, for the scope of this work. In this case, $\mathbf{F}$ is equivalent to the Hessian of the cost function for parameter estimation based on voltage responses, and can be used to quantify practical parameter identifiability according to Andersson et al. [10]. More specifically, a high value on the FIM diagonal $\mathbf{F}_{i,\,i}$ indicates a high sensitivity of parameter Θ_i, whereas large values on off-diagonals $\mathbf{F}_{i,\,j},\ i \neq j$, indicate a high correlation between parameters $(\Theta_i,\, \Theta_j)$. In order to optimize the experimental design based on Fisher-Information, common optimality criteria, cf., e.g., [10], aim at maximizing certain scalar properties of FIMs, such as:

$$D_{\text{opt}}(I,\, SOC_0,\, \mathbf{\Theta}) := \log_{10}(\det(\mathbf{F})) \ , \tag{17}$$

$$A_{\text{opt}}(I,\, SOC_0,\, \mathbf{\Theta}) := -\log_{10}(\text{trace}(\mathbf{F}^{-1})) \ , \tag{18}$$

$$E_{\text{opt}}(I,\, SOC_0,\, \mathbf{\Theta}) := \log_{10}(\lambda_{\min}(\mathbf{F})) \ , \tag{19}$$

$$E^*_{\text{opt}}(I,\, SOC_0,\, \mathbf{\Theta}) := \log_{10}\left(\frac{\lambda_{\min}(\mathbf{F})}{\lambda_{\max}(\mathbf{F})} \right) \ , \tag{20}$$

where $\lambda_{\min}$ and $\lambda_{\max}$ denote the smallest and largest eigenvalue of $\mathbf{F}$, respectively. Note that these criteria are often equivalently stated as minimizable quantities, e.g., by considering the inverse FIM or its eigenvalues, but the above definitions are aligned to be maximizable quantities for better comparability throughout this work. Definitions (17)–(20) are based on local FIMs and therefore biased by the values of $\mathbf{\Theta}$, which are commonly unavailable – or at least uncertain – in realistic parameter estimation scenarios. To account for this, concepts like Global Sensitivity Analysis (GSA) [12, 13], Global Optimal

Experimental Design (GOED) [11, 41, 42] or Bayesian Experimental Design (BED) [43, 44] aim at covering the parameter space more rigorously to draw more robust conclusions for parameter sensitivities and experimental design. In early work by Dette et al., cf. [45], Bayesian D-optimality is introduced as the expectation of D-optimalities based on parameter priors $\xi(\Theta)$, i.e.,

$$E_\xi[\log_{10}(\det(\mathbf{F}(\Theta)))] = \int_\Theta \log_{10}(\det(\mathbf{F}(\Theta)))\, d\xi(\Theta) \ . \tag{21}$$

More recently, Rainforth et al., cf. [44], comment on the connection between BED and Fisher-Information, and, among other things, conclude that averaging alphabetic optimalities, such as Eqs. (17)–(20), is "arguably Bayesian" in the sense of assuming uniform priors $\xi(\Theta)$ and using Fisher optimalities as utility function for quantifying the information gain of an experiment. We follow these lines of thoughts and average optimalities (17)–(20) throughout the parameter space to obtain, e.g., for the E^*-optimality criterion, the global $\overline{E}^*$-optimality

$$\overline{E}^*_{\text{opt}} = \frac{1}{N^F_\Theta} \sum_{i=0}^{N^F_\Theta} E^*_{\text{opt}}(I, SOC_0, \mathbf{\Theta}_i)) \ , \tag{22}$$

where $\overline{D}_{\text{opt}}$, $\overline{A}_{\text{opt}}$ and $\overline{E}_{\text{opt}}$ can be defined analogously and N^F_Θ is the amount of parameter samples $\mathbf{\Theta}_i = (D_n,\, D_p,\, \varepsilon_n,\, \varepsilon_p,\, \theta_n^{100\%},\, \theta_n^{0\%},\, \theta_p^{100\%})_i$ sampled from a uniform grid within the application-driven bounds

$$D_n \in [10^{-15},\, 10^{-13}] \ , \tag{23}$$
$$D_p \in [10^{-15},\, 10^{-13}] \ , \tag{24}$$
$$\varepsilon_n \in [0.8 \cdot \hat{\varepsilon}_n,\, \hat{\varepsilon}_n] = [0.6,\, 0.75] \ , \tag{25}$$
$$\varepsilon_p \in [0.8 \cdot \hat{\varepsilon}_p,\, \hat{\varepsilon}_p] = [0.532,\, 0.665] \ , \tag{26}$$
$$\theta_n^{100\%} \in [\hat{\theta}_n^{0\%} + 0.9 \cdot (\hat{\theta}_n^{100\%} - \hat{\theta}_n^{100\%}),\ \hat{\theta}_n^{100\%}] \approx [0.82,\, 0.91] \ , \tag{27}$$
$$\theta_n^{0\%} \in [\hat{\theta}_n^{0\%},\, \hat{\theta}_n^{0\%} + 0.1 \cdot (\hat{\theta}_n^{100\%} - \hat{\theta}_n^{0\%})] \approx [0.026,\, 0.115] \ , \tag{28}$$
$$\theta_p^{100\%} \in [\hat{\theta}_p^{100\%} - 0.1 \cdot (\hat{\theta}_p^{0\%} - \hat{\theta}_p^{100\%}),\, \hat{\theta}_p^{100\%}] \approx [0.205,\, 0.264] \ . \tag{29}$$

The diffusivity ranges are aligned with the training domain of PI-DeepONet in Section 2.2, and motivated by the aforementioned different orders of magnitude reported in literature, see, e.g., work by O'Regan et al. [35]. Active material volume fractions are bounded between their nominal values from Tab. A1 and a 20 % lower value, reflecting the common End-of-Life criterion for aged battery cells at 80 % SOH. The stoichiometric limits are varied in relatively narrow bounds reflecting the minor changes expected due to LLI while ensuring that the resulting cell balancing and OCV fitting remain physically valid for all parameter combinations, i.e., ensuring $\theta_p^{100\%} < \theta_p^{0\%} \leq 1$. To balance parameter space exploration and computational tractability, we sample $\mathbf{\Theta}_i$ from a grid of three equidistant values per dimension including lower bound, center, and upper bound of the corresponding ranges in Eqs. (23)–(29), i.e., $N^F_\Theta = 3^7 = 2{,}187$.

2.4 Iterative parameter estimation based on global experimental design and local identifiability analysis

While both local and global optimalities can be used to maximize information in terms of a given optimality, such experimental designs can still fall short during parameter estimation, e.g., due to insensitive parameters being estimated poorly or correlated parameters not being separately identifiable. To address this, we propose an iterative parameter estimation algorithm that is depicted in Fig. 2 and based on three steps: (I) Global optimal experimental design by means of optimizing global optimalities such as Eq. (22), (II) parameter estimation based on a Differential Evolution (DE) algorithm, and (III) local identifiability analysis based on the FIM evaluated at the resulting parameter values $\mathbf{\Theta}$ from the previous step. For step (III), we propose a simple threshold approach that would fix only those parameters whose corresponding diagonal entries (i.e., parameter sensitivity) or off-diagonals to a sensitive parameter (i.e., correlation to a sensitive parameter) are above a relative threshold of 0.75 compared to the maximum

value of the respective FIM. The Cramer-Rao bound states that these parameters are most likely to be estimated correctly after step (II) [46]. After fixing locally identifiable parameters $\boldsymbol{\Theta}_{\text{fixed}}$, the procedure is repeated from step (I) by computing FIMs only for the uncertain parameter set $\boldsymbol{\Theta}_{\text{uncertain}} = \boldsymbol{\Theta} \setminus \boldsymbol{\Theta}_{\text{fixed}}$ and averaging the resulting optimalities across the uncertain parameter ranges. In this way, the algorithm aims at iteratively reducing the curse of dimensionality in parameter space while performing the estimation on the most informative experimental design for the remaining, most uncertain parameters in each iteration. Due to large design spaces derived from continuous inputs like initial SOC or the

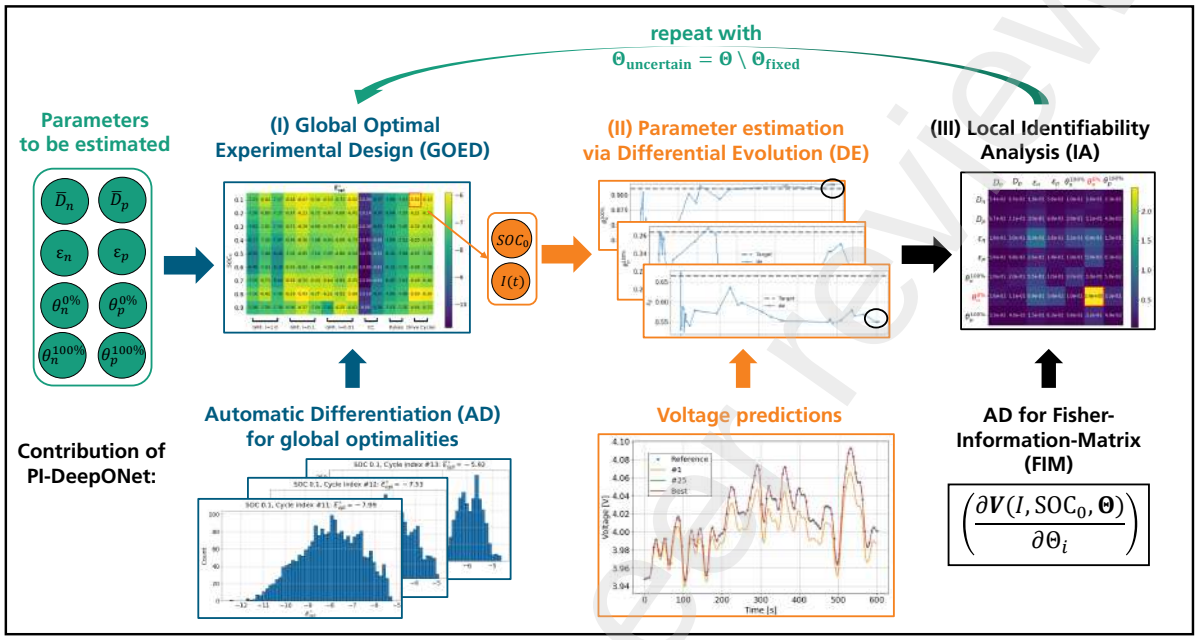

Figure 2: Proposed parameter estimation scheme based on the trained PI-DeepONet: (I) The GOED is determined for the parameters to be estimated (left) and used in (II) to estimate them via the DE algorithm and the corresponding voltage predictions from PI-DeepONet. (III) A local identifiability analysis is performed to fix the estimated values for the parameters that are most likely to be identified correctly and the procedure is repeated for the remaining set of parameters $\boldsymbol{\Theta}_{\text{uncertain}}$.

applied current profile, it is evident that an efficient surrogate model for Step (II), but also an efficient way of retrieving Fisher-Information in Steps (I) and (III) is paramount for this workflow to be both computationally tractable and experimentally meaningful. In Section 3, we show that the PI-DeepONet introduced in Section 2.2 is an ideal choice for providing the required efficiency in terms of surrogate modeling and approximation of Fisher-Information and corresponding global optimalities.

3 Results and discussion

In this section, PI-DeepONet is evaluated in terms of its accuracy as a surrogate model for the SPM, its accuracy for approximating Fisher-Information after training, and its viability for the parameter estimation algorithm proposed in Section 2.4. This evaluation is based on SPM reference solutions obtained via the open-source battery modeling framework *PyBaMM*, cf. [47], with settings as described in Appendix A.2.

3.1 Surrogate model accuracy

The deviation between outputs c_j^{NN}, V^{NN} predicted by PI-DeepONet and reference solutions c_j^{ref}, V^{ref} obtained from *PyBaMM* is quantified by two metrics. First, a normalized Mean-Absolute-Percentage-Error (NMAPE) on the surface concentration profiles and, second, the Root-Mean-Squared-Error (RMSE) on

the resulting voltage responses, where both metrics are averaged uniformly in time at a resolution of 1 Hz, i.e.,

$$\mathrm{NMAPE}^{\mathrm{surf}}(I,\, SOC_0,\, \boldsymbol{\Theta}) = \frac{1}{2} \sum_{j \in \{n,\, p\}} \frac{1}{N_t} \sum_{i=0}^{N_t-1} \frac{\left| c_j^{\mathrm{NN}}(R_j, t_i) - c_j^{\mathrm{ref}}(R_j, t_i) \right|}{\max\limits_t(c_j^{\mathrm{ref}}(R_j, t)) - \min\limits_t(c_j^{\mathrm{ref}}(R_j, t))} \cdot 100\ \%\ , \tag{30}$$

$$\mathrm{RMSE}(I,\, SOC_0,\, \boldsymbol{\Theta}) = \sqrt{\frac{1}{N_t} \sum_{i=0}^{N_t-1} \left| V(\mathbf{c}^{\mathrm{NN}},\, t_i) - V(\mathbf{c}^{\mathrm{ref}},\, t_i) \right|^2}\ , \tag{31}$$

where $N_t = 601$ and the dependence of $c_j(r,t)$ and $V(t)$ on the inputs $(I,\, SOC_0,\, \boldsymbol{\Theta})$ has been omitted for the sake of readability. As the input parameters $(\theta_n^{100\%},\, \theta_n^{0\%},\, \theta_p^{100\%}, SOC_0)$ are hard-constrained via Eq. (11) and the voltage post-processing layer, these parameters are fixed to their nominal values from Tab. A1, and we set $SOC_0 = 50$ % for the evaluation of surrogate model accuracy. Both metrics are further averaged over N_I^{test} current profiles and $N_\Theta^{\mathrm{test}} = 3^4$ parameter samples which are analogously sampled at three equidistant values for $\varepsilon_j \in [\varepsilon_j^{\min},\, \varepsilon_j^{\max}]$ and $D_j \in [D_j^{\min},\, D_j^{\max}]$, respectively, with bounds as specified in Eqs. (23)–(29), i.e.,

$$\mathrm{NMAPE}_{\mathrm{avg}}^{\mathrm{surf}} = \frac{1}{N_I^{\mathrm{test}} \cdot N_\Theta^{\mathrm{test}}} \sum_{i=1}^{N_I^{\mathrm{test}}} \sum_{j=1}^{N_\Theta^{\mathrm{test}}} \mathrm{NMAPE}^{\mathrm{surf}}(I_i,\, 0.5,\, \Theta_j)\ , \tag{32}$$

$$\mathrm{RMSE}_{\mathrm{avg}} = \frac{1}{N_I^{\mathrm{test}} \cdot N_\Theta^{\mathrm{test}}} \sum_{i=1}^{N_I^{\mathrm{test}}} \sum_{j=1}^{N_\Theta^{\mathrm{test}}} \mathrm{RMSE}(I_i,\, 0.5,\, \Theta_j)\ . \tag{33}$$

To quantify the accuracy of PI-DeepONet for different interpolation and extrapolation use cases, six groups of current profiles with a total of 15 profiles are generated, cf. Fig. A1 in the Appendix. These include three groups of GRF profiles with $l \in \{1.0, 0.1, 0.01\}$ representing interpolation based on profiles similar to those used during training and three battery-specific profile groups representing extrapolation based on constant-current profiles, pulse profiles, or common drive cycles for simulating automotive applications. Figure 3 depicts the evolution of the acurracy metrics for each profile group

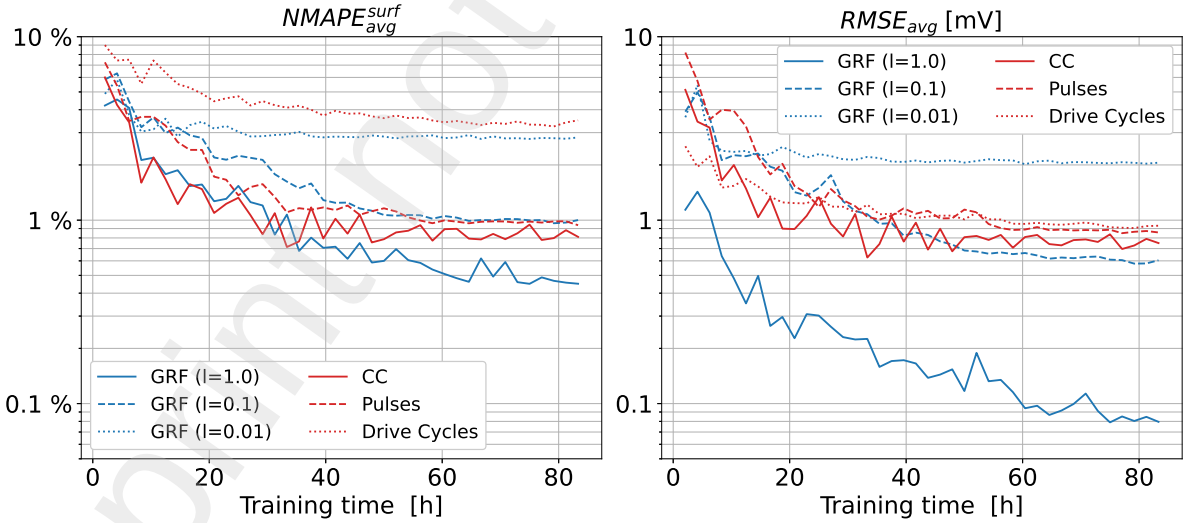

Figure 3: Evolution of accuracy metrics for different types of current profiles during training of PI-DeepONet. (Blue=Interpolation): GRF profiles with different length scales l. (Red=Extrapolation): Battery-specific profiles including constant-current discharge (CC), pulse profiles (Pulses), and representative profiles for automotive applications (Drive Cycles).

during training of PI-DeepONet, which was performed on a NVIDIA H200 GPU with hyperparameters

as provided in the Appendix A.2. For both interpolation and extrapolation, the accuracy levels in surface concentration are in line with the complexity of the applied profiles as both high-frequency GRF profiles (l=0.01) and the more dynamic drive cycles remain at a higher level of around 3 % $\text{NMAPE}_{\text{avg}}^{\text{surf}}$ until the end of training. To put these results into perspective, Fig. 4 shows three representative current profiles, the corresponding predictions of PI-DeepONet, and the reference obtained from *PyBaMM* for varying diffusivities D_n, D_p and active material volume fractions ε_n, ε_p. While both the medium-frequency GRF profile and the considered pulse profile show high levels of accuracy with $\text{NMAPE}_{\text{avg}}^{\text{surf}} < 1$ % and $\text{RMSE}_{\text{avg}} < 1$ mV, a closer look at drive cycle #13 in the bottom row of Fig. 4 reveals that the surface concentration dynamics are not approximated as accurate for high-frequency inputs. The inset plot for c_n in the second column of that bottom row demonstrates that PI-DeepONet smoothly approximates the more dynamic oscillations that are present in the reference solution. This indicates that – despite the Fourier feature embeddings introduced in Section 2.2.2 – some spectral bias remains leaving room for improvement in training PI-DeepONet to approximate the highest frequency components of such solutions. Despite these inaccuracies in surface concentration, the corresponding voltage response in the last column of the bottom row in Fig. 4 is fairly accurate and yields a $\text{RMSE}_{\text{avg}} < 1$ mV for the considered drive cycles. This is due to the overpotentials η_j in Eqs. (6)–(8) that depend more significantly on the input current and overshadow the error in surface concentration in this case. This effect is also SOC-dependent as a higher slope in U_j^{OCP} causes the errors in c_j^{surf} to propagate more significantly into the voltage response. For example, the GRF profiles at length scale $l = 0.01$ feature a higher current throughput, i.e., cover a wider SOC range, and therefore maintain a higher RMSE_{avg} of around 2 mV in Fig. 3, despite showing higher accuracy than the drive cycles in terms of $\text{NMAPE}_{\text{avg}}^{\text{surf}}$. Considering the fast inference times of only a few milliseconds per SPM evaluation in PI-DeepONet –

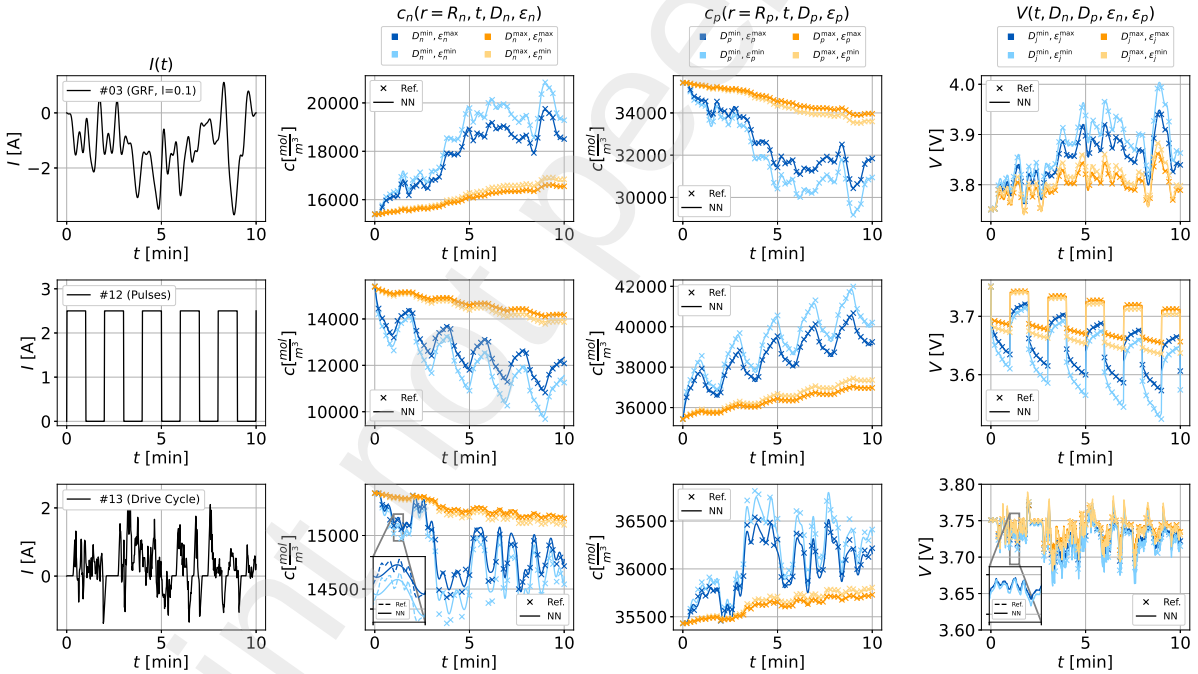

Figure 4: Accuracy of PI-DeepONet in surface concentration profiles and voltage responses compared to the reference solution from *PyBaMM* for three different current profiles and varying parameters (D_n, D_p, ε_n, ε_p) at upper and lower bounds as specified in Eqs. (23)–(29).

see the upcoming Section 3.3 for more details – these accuracy levels are within the expected order of magnitude of measurement errors in realistic applications suggesting that PI-DeepONet can be a viable surrogate model for the parameter estimation algorithm proposed in Section 2.4.

3.2 Parameter identifiability and global optimal experimental design

3.2.1 Local identifiability

The accuracy of gradient information obtained via AD on the trained PI-DeepONet is evaluated through a comparison of FIMs, cf. Eq. (16), and resulting optimality metrics from Eqs. (17)–(20) and a reference based on *PyBaMM*. For this reference, a finite difference 5-point stencil is applied on *PyBaMM* solutions using relative parameter perturbations of 10^{-3} as discussed more thoroughly in our previous work [28]. Figure 5 shows the local FIM at ground-truth parameters $\hat{\boldsymbol{\Theta}}$ for the experimental design using profile #09 in Fig. A1 (CC discharge at 0.1C) and $SOC_0 = 0.5$. Although the most important trends in terms of high and low parameter sensitivities (diagonals) and correlations (off-diagonals) are accurately predicted by PI-DeepONet, the rather insensitive parameters (D_n, ε_p, $\theta_p^{100\%}$) cause the FIM to be ill-conditioned for this design. In Fig. 6, optimalities from Eqs. (17)–(20) are compared for varying initial SOCs and four representative profiles including the aforementioned low-current CC profile #09. Due to the ill-

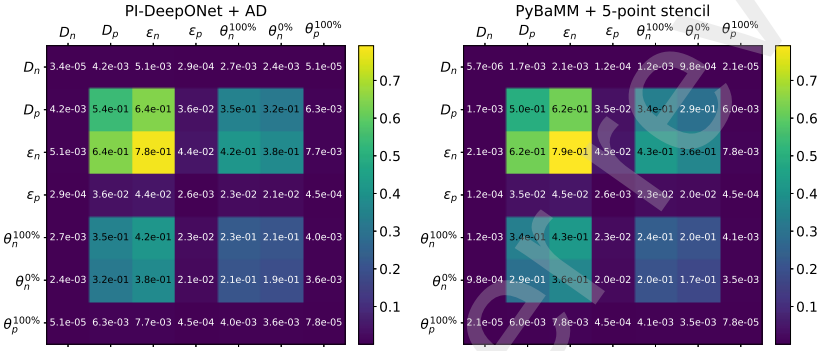

Figure 5: Comparison of local FIMs at ground-truth parameters $\hat{\boldsymbol{\Theta}}$ for profile #09, cf. Fig. A1, and $SOC_0 = 0.5$ obtained from PI-DeepONet with AD (left) and the based on the reference solve *PyBaMM* using a 5-point finite-difference stencil (right). A good agreement is observed specifically for the most sensitive parameters (D_p, ε_n, $\theta_n^{100\%}$, $\theta_n^{0\%}$) for this experimental design.

conditioning of FIMs for this profile, cf. Fig. 5, significant deviations are observed in all optimalities, which are based on the determinants, traces, or eigenvalues of the corresponding FIMs. Despite these inaccuracies for ill-conditioned FIMs, we generally observe good agreement across the SOC range between PI-DeepONet and *PyBaMM* for different GRF profiles, cf. #03 and #07, and even on extrapolation represented by drive cycle #13 in Fig. A1. Note that the maximum step size of *PyBaMM*'s default solver had to be decreased to 0.01 s to obtain the results in Fig. 6, and no further changes could be observed by decreasing the step size further or adapting other parameters related to meshing or the order of the underlying finite difference stencil, cf. [28]. Therefore, these deviations are likely either a result of PI-DeepONet providing imperfect gradients close to zero, or intrinsic instabilities due to the comparison of almost zero quantities like determinants or eigenvalues of ill-conditioned FIMs. Another observation is that the three optimalities $D_{\text{opt}}, A_{\text{opt}}, E_{\text{opt}}$ agree with each other qualitatively, whereas E^*_{opt} shows a quite different picture due to its different algebraic roots in the FIMs condition number, cf. Eq. (20). Due to this observation, only D_{opt} and E^*_{opt} are considered in the following sections.

3.2.2 Global experimental design

The trained PI-DeepONet and its automatic differentiability allows rapid inference of FIM gradients throughout the parameter space and therefore a fast inference of global optimalities such as $\overline{D}_{\text{opt}}$ or $\overline{E}^*_{\text{opt}}$ as defined in Eq. (22). As this approach would be computationally prohibitive using *PyBaMM* – see the discussion in the upcoming Section 3.3 – we report results only obtained via AD on the trained PI-DeepONet in this section. Figure 7 depicts the two global optimalities $\overline{D}_{\text{opt}}$ and $\overline{E}^*_{\text{opt}}$ evaluated for a total of 135 experimental designs (I, SOC_0) that arise by combining the 15 profiles $I(t)$ from Fig. A1 with 9 initial SOCs that are distributed uniformly from $SOC_0 = 0.1$ to $SOC_0 = 0.9$. The results on $\overline{D}_{\text{opt}}$ show the main shortcoming of this metric, as the determinant of FIMs tends to overemphasize single parameter sensitivities, in this case ε_n, which is most sensitive in low SOC regimes, where U_n^{OCP}

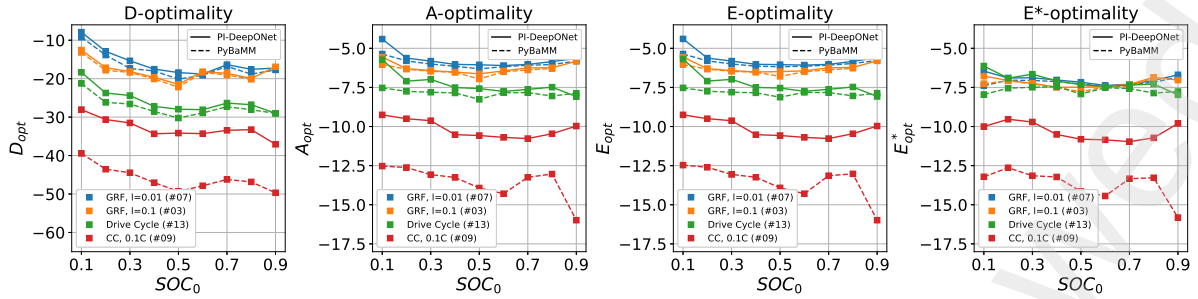

Figure 6: Local optimalities evaluated at ground-truth parameters $\hat{\boldsymbol{\Theta}}$ for varying initial SOCs and four representative profiles from Fig. A1 showing good agreement between PI-DeepONet and *PyBaMM* except for the most ill-conditioned FIM corresponding to the least informative experimental design (CC discharge at 0.1C).

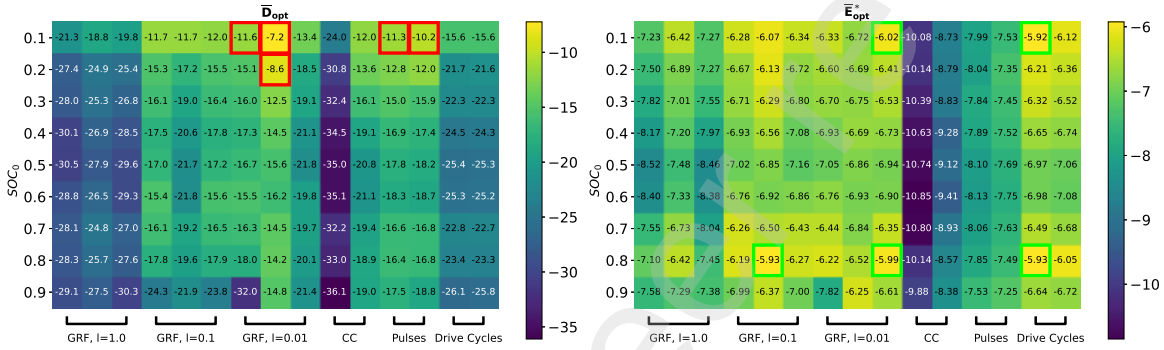

Figure 7: Global D- and E*-optimalities for different experimental designs (I, SOC_0) obtained via AD on the trained PI-DeepONet. The five most informative experimental designs are highlighted for each optimality metric and for further reference in Section 3.3.

shows the steepest slope, cf. Fig. 1. Hence, the GRF profile with the most net discharge, i.e., profile #07 in Fig. A1, turns out to be $\overline{D}$-optimal at $SOC_0 = 0.1$ as it covers – on average – the steepest parts of U_n^{OCP} that yield the highest sensitivity in ε_n. Due to this bias on the most sensitive parameters, we also observe that $\overline{D}_{\mathrm{opt}}$, much like $\overline{A}_{\mathrm{opt}}$ and $\overline{E}_{\mathrm{opt}}$, yields similar results throughout the parameter region, which is indicated by the re-occuring trend towards low SOC regimes and further demonstrated in the upcoming Section 3.3. In contrast to this, $\overline{E}^*_{\mathrm{opt}}$ yields a more diverse result with many designs of similar optimality, and less bias towards single parameter sensitivities. In the following section, we empirically show that $\overline{E}^*_{\mathrm{opt}}$ is the preferable optimality among the presented options to identify global optimal experimental designs in different regions of the parameter space, especially when aiming for estimation of all parameters $\boldsymbol{\Theta}$ and not only the most sensitive subset of parameters.

3.3 Iterative parameter estimation

Finally, we evaluate the performance of the parameter estimation procedure introduced in Section 2.4 to estimate parameters $\boldsymbol{\Theta} = (D_n,\, D_p,\, \varepsilon_n,\, \varepsilon_p,\, \theta_n^{100\%},\, \theta_n^{0\%},\, \theta_p^{100\%})$ based on various experimental designs (I, SOC_0). The underlying estimation is driven by the DE algorithm with a total budget of 100,000 model evaluations to minimize the deviation between predicted voltages $V^{\mathrm{NN}}(\boldsymbol{\Theta})$ and the voltage response $V^{\mathrm{ref}}(\hat{\boldsymbol{\Theta}})$ obtained with *PyBaMM* and ground truth parameters $\hat{\boldsymbol{\Theta}}$ from Tab. A1 in the Appendix, i.e.,

$$\min_{\boldsymbol{\Theta}} \ \mathrm{RMSE} = \sqrt{\frac{1}{N_t} \sum_{i=0}^{N_t-1} |V^{\mathrm{NN}}(t_i, \boldsymbol{\Theta}) - V^{\mathrm{ref}}(t_i, \hat{\boldsymbol{\Theta}})|^2} \,, \tag{34}$$

where $N_t = 601$ again discretizes the voltage response at a resolution of 1 Hz. The feasible ranges of each parameter $\Theta_i \in \left[\Theta_i^{\min}, \Theta_i^{\max}\right]$ are defined by extending the bounds specified in Eqs. (23)–(29) by 10 % in both directions to not bias the optimization towards ground-truth values that are located on a boundary of the search space. A further NMAPE is defined to measure the accuracy of the estimated parameters $\boldsymbol{\Theta}^{\text{est}}$ with respect to the ground truth $\hat{\boldsymbol{\Theta}}$, i.e.,

$$\text{NMAPE}^{\Theta} = \frac{1}{7}\sum_{i=1}^{7}\left|\frac{\Theta_i^{\text{est}} - \hat{\Theta}_i}{\Theta_i^{\max} - \Theta_i^{\min}}\right| \cdot 100\% \ . \tag{35}$$

Figure 8 shows numerous parameter estimation results based on the trained PI-DeepONet for four different algorithmic scenarios in each of the 135 total experimental designs: a) 100,000 model evaluations via DE without further adjustments, b) 50,000 evaluations, followed by a local identifiability analysis (IA) to determine $\boldsymbol{\Theta}_{\text{uncertain}} = \boldsymbol{\Theta} \setminus \boldsymbol{\Theta}_{\text{fixed}}$, followed by additional 50,000 evaluations on $\boldsymbol{\Theta}_{\text{uncertain}}$, c) 50,000 evaluations followed by a local IA, followed by additional 50,000 evaluations using a new, $\overline{D}$-optimal experimental design for $\boldsymbol{\Theta}_{\text{uncertain}}$, and d) 50,000 evaluations followed by a local IA, followed by additional 50,000 evaluations using a new $\overline{E}^*$-optimal experimental design. To reduce random effects due to the stochastic nature of the DE algorithm, each parameter estimation procedure is run three times using a different random seed and only the mean results are displayed. The five most informative experimental designs based on $\overline{D}_{\text{opt}}$ (red) and $\overline{E}^*_{\text{opt}}$ (green), cf. Fig. 7, are highlighted in order to visualize whether these experimental design choices would have led to a good parameter estimation result in each of the algorithmic scenarios a) – d).

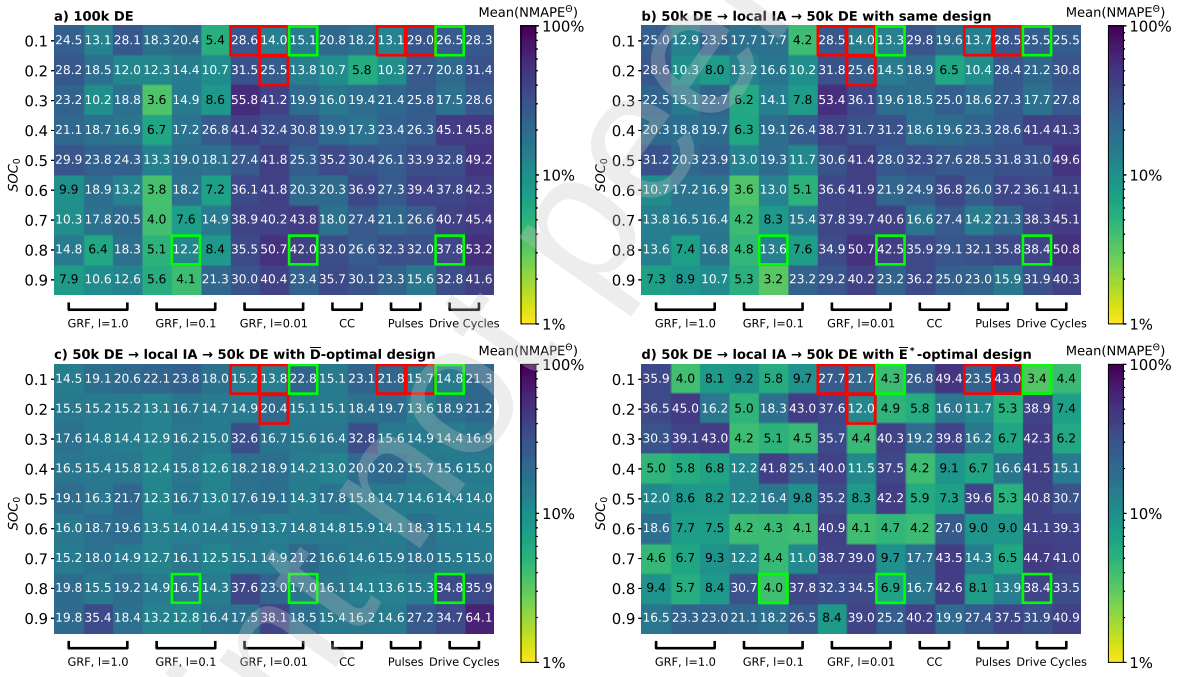

Figure 8: Mean parameter estimation accuracy using PI-DeepONet in four algorithmic approaches a) – d) with 135 initial experimental designs (I, SOC_0) at three different random seeds. Highlighted cells indicate the five most informative experimental designs based on global optimalities $\overline{D}_{\text{opt}}$ (red) and $\overline{E}^*_{\text{opt}}$ (green) as identified in Fig. 7.

Based on Fig. 8 a), we conclude that the default estimation approach with 100,000 DE evaluations can yield accurate results for some designs but neither $\overline{D}_{\text{opt}}$ nor $\overline{E}^*_{\text{opt}}$ from Fig. 7 can reliably identify these designs a-priori. Figure 8 b) shows that fixing the locally identifiable parameters after 50,000 evaluations without optimizing the experiment for the second 50,000 evaluations based on $\boldsymbol{\Theta}_{\text{uncertain}}$ improves some results slightly but does not show a clear path to a-priori identification of good designs via $\overline{D}_{\text{opt}}$ or $\overline{E}^*_{\text{opt}}$ either. Subplot c) highlights the aforementioned shortcomings of the $\overline{D}_{\text{opt}}$ metric, which overemphasizes

Table 1: Comparison of runtimes between *PyBaMM* and the trained PI-DeepONet within the parameter estimation algorithm in Fig. 2 assuming 135 experimental designs and $\boldsymbol{\Theta}_{\text{uncertain}} \in \mathbb{R}^5$ in the second iteration of the algorithm. The considered PI-DeepONet was trained once for around 80 hours and runtimes in the second column refer to the evaluation of a single voltage response for a fixed experimental design and parameter set.

	Hardware (runtime)	$\overline{E}^*$-optimal design (3^7 FIMs per design)	$2 \cdot 50k$ DE	$\overline{E}^*$-optimal re-design ($\approx 3^5$ FIMs per design)	$\sum$
PyBaMM	CPU* (319 ms)	$135 \cdot 2{,}187 \cdot 28$ SPMs ≈ 28 days	≈ 8.3h	$135 \cdot 234 \cdot 28$ SPMs ≈ 3 days	≈ 31 days
PI-DeepONet	CPU* (3.61 ms)	$135 \cdot 9$ AD batches à 7.2 s ≈ 2.4 h	≈ 361 s	135 AD batches à 7.2 s ≈ 972 s	≈ 2.8 h (265x)
	GPU** (0.67 ms)	135 AD batches à 111 ms ≈ 14.9 s	≈ 67s	135 AD batches à 13 ms ≈ 1.7 s	≈ 84 s (31,886x)

* Intel Xeon Gold 6244 (single core),

** NVIDIA H200 (single GPU)

the most sensitive parameters and therefore yields the same experimental re-design – profile #07 with $SOC_0 = 0.1$, cf. Fig. 7 – for almost all initial estimation results without properly estimating the less sensitive parameters in $\boldsymbol{\Theta}$. Finally, Fig. 8 d) uses the $\overline{E}^*_{\text{opt}}$ criterion and shows more robust results with many different initial design choices leading to final parameter estimation errors of 4–5 % NMAPE$^{\Theta}$. This robustness highlights the ability of the iterative estimation algorithm in Section 2.4 to recover most parameters reasonably accurate by performing local IA and selecting an optimal design for the reduced parameter set $\boldsymbol{\Theta}_{\text{uncertain}}$, cf. Fig. 2. More importantly, most of the $\overline{E}^*$-optimal designs, highlighted in green, yield good mean results in Fig. 8 d), with the $\overline{E}^*$-optimal design – profile #13 at $SOC_0 = 0.1$ – obtaining a mean NMAPE$^{\Theta}$ of 3.4 % over three estimation procedures. One instance of the underlying estimation procedures is visualized in more detail in the Appendix A.3. This result suggests that $\overline{E}^*_{\text{opt}}$ is a suitable criterion for identifying a good design for the first 50,000 DE iterations a-priori before performing local IA and identifying the $\overline{E}^*$-optimal design for $\boldsymbol{\Theta}_{\text{uncertain}}$ and the remaining 50,000 DE evaluations. Better results in terms of NMAPE$^{\Theta}$ are likely hindered by the prediction error of PI-DeepONet, cf. Fig. 3, which is above the RMSE < 0.01 mV that we observe for reference results with NMAPE$^{\Theta} < 1$ % using *PyBaMM* as error-free surrogate model, cf. Fig. A2 in the Appendix. However, obtaining such low errors in PI-DeepONet would not be beneficial in practical applications as the expected voltage measurement errors are likely above that order of magnitude. Instead, future work should tackle the current design limitation induced by relatively low C-rates, $t_{\text{max}} = 10$ min, and a limited budget of 100,000 DE evaluations. Adapting these would increase the RMSE, at which parameters $\boldsymbol{\Theta}$ can be estimated accurately, to a more realistic order of magnitude, but would also require an electrochemical model suitable for higher C-rates, i.e., the SPMe or DFN model, which is left for future work.

To quantify the efficiency of PI-DeepONet in the proposed parameter estimation algorithm, Tab. **??** compares the expected runtimes for finding a $\overline{E}^*$-optimal experimental design among 135 options, performing 2 times 50,000 DE iterations, and computing a second $\overline{E}^*$-optimal design on a reduced parameter set in between. For this comparison, we assume two fixed parameters after the local IA, i.e., $\boldsymbol{\Theta}_{\text{uncertain}} \in \mathbb{R}^5$ and 3^5 FIMs to be computed for each experimental design in the reduced parameter space. Runtimes were measured for inference based on the compiled and trained PI-DeepONet and for calling *model.solve()* in *PyBaMM*, respectively, i.e., any shared overhead for writing, reading or post-processing results as well as the one-time effort for compiling and training PI-DeepONet, cf. Fig. 3 is neglected. Based on this comparison, PI-DeepONet offers a speedup factor of 265 on the considered CPU, and more than 30,000 on a state-of-the-art GPU, which is likely to increase further with more advanced hardware, problems of higher dimensionality, or computationally more complex LIB models.

Note that Fig. 8 – considering four algorithmic designs, 135 experimental designs and three different random seeds – required a total of $162 \cdot 10^6$ model evaluations and close to 600,000 FIM computations, each requiring another 20–24 model evaluations in a numerical five-point stencil approach. These numbers indicate that even with multi-processing capabilities on HPC clusters, efforts like these are hardly tractable with classical simulations, e.g., based on *PyBaMM* [48]. Hence, PI-DeepONet or similar successors look

much more suited to further develop and improve such wholistic parameter estimation algorithms for LIB models, combining computationally heavy global experimental design and population-based optimization methods.

4 Conclusion and outlook

The proposed parametrized PI-DeepONet is able to approximate solutions of the SPM for various current profiles and parametric inputs within few milliseconds while achieving errors less than 1 % in surface concentration profiles and a RMSE < 1 mV in the resulting voltage response for current profiles with moderate dynamics. Slightly larger errors of around 3 % in surface concentration profiles and 2 mV in voltage RMSE are reported for the most dynamic profiles at a sampling rate of 1 Hz suggesting room for improvement in the current training setup. Nevertheless, the data-free, purely physics-informed training approach shows good extrapolation accuracy, e.g., for constant-current discharge, pulse profiles, and representative driving cycles for automotive applications (UDDS, US06) with a reported RMSE $<$ 1 mV at inference. Furthermore, it was shown that local Fisher-Information obtained via AD on the trained PI-DeepONet is sufficiently accurate and allows a global, yet computationally tractable, view on identifiability and optimal experimental design across the parameter space. An exception is presented for experimental designs with very low identifiability in some parameters, which causes FIMs to be ill-conditioned and results in corresponding optimalities to diverge between PI-DeepONet and the reference solver *PyBaMM*. Finally, an iterative parameter estimation scheme was proposed that exploits the high inference speed and automatic differentiability of PI-DeepONet to combine global optimal experimental design with local identifiability analyses while iteratively removing the parameter space dimensions that are most likely to be estimated correctly. This approach – coupled with a suitable optimality metric – improves robustness in the parameter estimation process and paves the way for parameter-based state estimation in realistic applications that require lightweight, but robust state estimation algorithms.

Further work is planned to improve the accuracy of PI-DeepONet for high-frequency current profiles as these will play a crucial role, e.g., for state estimation in automotive applications of LIBs. Extending the training objectives of PI-DeepONet to surrogate models of higher fidelity such as the SPMe or DFN model would expand the applicability of this approach to higher C-rate scenarios and is therefore another crucial next step. The presented parameter estimation algorithm should be viewed as a first proof-of-concept and can be refined in several ways. These include applying more than two iterations of the proposed algorithm, allowing a higher total model evaluation count, incorporating adaptive bayesian priors in the sampling of global optimalities, or adaptively reducing parameter confidence bounds instead of fixing locally identifiable parameters based on a rigorous threshold. The DE algorithm could be replaced by optimizers that are both gradient- and population-based to further exploit the automatic differentiability of PI-DeepONet and to increase the likelihood of finding a global minimum in parameter space. Finally, the proposed algorithm based on PI-DeepONet needs to be validated with real measurement data to prove its viability in realistic state estimation scenarios that include measurement errors and a less structured experimental design space. Within this process, obtaining a trustworthy ground truth for eSOH parameters and related degradation modes is probably the most difficult part, but, e.g., recent efforts by Kirkaldy et al., cf. [49], are targeting the same challenge and should be considered as a viable pathway for experimental validation.

Acknowledgements

The authors acknowledge support from the European Union under grant agreement No. 101103755 and by UKRI under grant agreement No. 10078013, respectively (FASTEST – "Fast-track hybrid testing platform for the development of battery systems").

References

[1] Maitane Berecibar, Iñigo Gandiaga, I Villarreal, Noshin Omar, Joeri Van Mierlo, and Peter Van den Bossche. Critical review of state of health estimation methods of li-ion batteries for real applications. *Renewable and Sustainable Energy Reviews*, 56:572–587, 2016. doi:10.1016/j.rser.2015.11.042.

[2] Zuolu Wang, Guojin Feng, Dong Zhen, Fengshou Gu, and Andrew Ball. A review on online state of charge and state of health estimation for lithium-ion batteries in electric vehicles. *Energy Reports*, 7:5141–5161, 2021. doi:10.1016/j.egyr.2021.08.113.

[3] Marc Doyle, Thomas F Fuller, and John Newman. Modeling of galvanostatic charge and discharge of the lithium/polymer/insertion cell. *Journal of the Electrochemical society*, 140(6):1526, 1993. doi:10.1149/1.2221597.

[4] Shriram Santhanagopalan, Qingzhi Guo, Premanand Ramadass, and Ralph E White. Review of models for predicting the cycling performance of lithium ion batteries. *Journal of power sources*, 156(2):620–628, 2006. doi:10.1016/j.jpowsour.2005.05.070.

[5] Scott G Marquis, Valentin Sulzer, Robert Timms, Colin P Please, and S Jon Chapman. An asymptotic derivation of a single particle model with electrolyte. *Journal of The Electrochemical Society*, 166(15):A3693, 2019. doi:10.1149/2.0341915jes.

[6] Guodong Fan, Dongliang Lu, M Scott Trimboli, Gregory L Plett, Chong Zhu, and Xi Zhang. Nondestructive diagnostics and quantification of battery aging under different degradation paths. *Journal of Power Sources*, 557:232555, 2023. doi:10.1016/j.jpowsour.2022.232555.

[7] Iker Lopetegi, Gregory L Plett, M Scott Trimboli, Laura Oca, Eduardo Miguel, and Unai Iraola. A new battery soc/soh/esoh estimation method using a pbm and interconnected spkfs: Part ii. soh and esoh estimation. *Journal of The Electrochemical Society*, 171(3):030518, 2024. doi:10.1149/1945-7111/ad30d5.

[8] Ruihe Li, Niall D Kirkaldy, Fabian F Oehler, Monica Marinescu, Gregory J Offer, and Simon EJ O'Kane. The importance of degradation mode analysis in parameterising lifetime prediction models of lithium-ion battery degradation. *Nature Communications*, 16(1):2776, 2025. doi:10.1038/s41467-025-57968-3.

[9] Tobias Hofmann, Jacob Hamar, Bastian Mager, Simon Erhard, and Jan Philipp Schmidt. Physics-constrained transfer learning: Open-circuit voltage curve reconstruction and degradation mode estimation of lithium-ion batteries. *Energy and AI*, 20:100493, 2025. doi:10.1016/j.egyai.2025.100493.

[10] Malin Andersson, Moritz Streb, Jing Ying Ko, Verena Löfqvist Klass, Matilda Klett, Henrik Ekström, Mikael Johansson, and Göran Lindbergh. Parametrization of physics-based battery models from input–output data: A review of methodology and current research. *Journal of Power Sources*, 521:230859, 2022. doi:10.1016/j.jpowsour.2021.230859.

[11] Moritz Streb, Mathilda Ohrelius, Matilda Klett, and Göran Lindbergh. Improving Li-ion battery parameter estimation by global optimal experiment design. *Journal of Energy Storage*, 56:105948, 2022. doi:10.1016/j.est.2022.105948.

[12] Elia Zonta, Ivana Jovanovic Buha, Michele Spinola, Christoph Weißinger, Hans-Joachim Bungartz, and Andreas Jossen. Time-dependent global sensitivity analysis of the Doyle-Fuller-Newman model. *arXiv preprint arXiv:2502.17275*, 2025. doi:10.48550/arXiv.2502.17275.

[13] Ratnak Sok and Jin Kusaka. Global sensitivity analysis on parameter identifications of electrochemical li-ion cell model using transient test data scaled from battery electric vehicle experiments. *Future Batteries*, page 100085, 2025. doi:10.1016/j.fub.2025.100085.

[14] Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 378:686–707, 2019. doi:10.1016/j.jcp.2018.10.045.

[15] Salvatore Cuomo, Vincenzo Schiano Di Cola, Fabio Giampaolo, Gianluigi Rozza, Maziar Raissi, and Francesco Piccialli. Scientific machine learning through physics–informed neural networks: Where we are and what's next. *Journal of Scientific Computing*, 92(3):88, 2022. doi:10.1007/s10915-022-01939-z.

[16] Woojin Cho, Minju Jo, Haksoo Lim, Kookjin Lee, Dongeun Lee, Sanghyun Hong, and Noseong Park. Parameterized physics-informed neural networks for parameterized pdes. *arXiv preprint arXiv:2408.09446*, 2024. doi:10.48550/arXiv.2408.09446.

[17] Sifan Wang, Hanwen Wang, and Paris Perdikaris. Learning the solution operator of parametric partial differential equations with physics-informed DeepONets. *Science advances*, 7(40):eabi8605, 2021. doi:10.1126/sciadv.abi8605.

[18] Zongyi Li, Hongkai Zheng, Nikola Kovachki, David Jin, Haoxuan Chen, Burigede Liu, Kamyar Azizzadenesheli, and Anima Anandkumar. Physics-informed neural operator for learning partial differential equations. *ACM/IMS Journal of Data Science*, 1(3):1–27, 2024. doi:10.1145/3648506.

[19] Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. Learning nonlinear operators via deeponet based on the universal approximation theorem of operators. *Nature machine intelligence*, 3(3):218–229, 2021. doi:10.1038/s42256-021-00302-5.

[20] Somdatta Goswami, Aniruddha Bora, Yue Yu, and George Em Karniadakis. Physics-informed deep neural operator networks. In *Machine Learning in Modeling and Simulation: Methods and Applications*, pages 219–254. Springer, 2023. doi:10.1007/978-3-031-36644-4_6.

[21] Shanling Ji, Jun Yuan, Bojing Zhang, Aleksei Sanin, Leon Merker, Zhisheng Zhang, Jianxiong Zhu, and Helge Sören Stein. Continuous physics-informed learning expedited battery mechanism decoupling. *Advanced Science*, page e06772, 2025. doi:10.1002/advs.202506772.

[22] Malik Hassanaly, Peter J Weddle, Ryan N King, Subhayan De, Alireza Doostan, Corey R Randall, Eric J Dufek, Andrew M Colclasure, and Kandler Smith. PINN surrogate of Li-ion battery models for parameter inference, Part I: Implementation and multi-fidelity hierarchies for the single-particle model. *Journal of Energy Storage*, 98:113103, 2024. doi:10.1016/j.est.2024.113103.

[23] Francisco J Méndez-Corbacho, Beñat Larrarte-Lizarralde, Rubén Parra, Javier Larrain, Diego del Olmo, Hans-Jürgen Grande, and Elixabete Ayerbe. Physics-informed neural networks for modeling li-ion batteries: Solving the single particle model without labeled data. *Journal of The Electrochemical Society*, 171(11):110534, 2024. doi:10.1149/1945-7111/ad940a.

[24] Qiang Zheng, Xiaoguang Yin, and Dongxiao Zhang. State-space modeling for electrochemical performance of Li-ion batteries with physics-informed deep operator networks. *Journal of Energy Storage*, 73:109244, 2023. doi:10.1016/j.est.2023.109244.

[25] Malik Hassanaly, Peter J Weddle, Ryan N King, Subhayan De, Alireza Doostan, Corey R Randall, Eric J Dufek, Andrew M Colclasure, and Kandler Smith. PINN surrogate of Li-ion battery models for parameter inference, Part II: Regularization and application of the pseudo-2D model. *Journal of Energy Storage*, 98:113104, 2024. doi:10.1016/j.est.2024.113104.

[26] Qiang Zheng, Xiaoguang Yin, and Dongxiao Zhang. Inferring electrochemical performance and parameters of Li-ion batteries based on deep operator networks. *Journal of Energy Storage*, 65:107176, 2023. doi:10.1016/j.est.2023.107176.

[27] Amir Ali Panahi, Daniel Luder, Billy Wu, Gregory Offer, Dirk Uwe Sauer, and Weihan Li. Fast and generalizable parameter-embedded neural operators for lithium-ion battery simulation. *Energy and AI*, page 100647, 2025. doi:10.1016/j.egyai.2025.100647.

[28] Philipp Brendel, Igor Mele, Andreas Rosskopf, Tomaž Katrašnik, and Vincent Lorentz. Parametrized physics-informed deep operator networks for design of experiments applied to lithium-ion-battery cells. *Journal of Energy Storage*, 128:117055, 2025. doi:10.1016/j.est.2025.117055.

[29] Josu Yeregui, Iker Lopetegi, Sergio Fernandez, Erik Garayalde, and Unai Iraola. On-site estimation of battery electrochemical parameters via transfer learning based physics-informed neural network approach. *arXiv preprint arXiv:2503.22396*, 2025. doi:10.48550/arXiv.2503.22396.

[30] Lorenzo Brancato, Alexander Gabriel Harej, Marco Giglio, and Francesco Cadini. Physics-informed operator learning for real-time battery state estimation. *Applied Energy*, 402:126987, 2026. doi:10.1016/j.apenergy.2025.126987.

[31] Chang-Hui Chen, Ferran Brosa Planella, Kieran O'Regan, Dominika Gastol, W Dhammika Widanage, and Emma Kendrick. Development of experimental techniques for parameterization of multi-scale lithium-ion battery models. *Journal of The Electrochemical Society*, 167(8):080534, 2020. doi:10.1149/1945-7111/ab9050.

[32] Loïc Lavigne, J Sabatier, J Mbala Francisco, F Guillemard, and A Noury. Lithium-ion open circuit voltage (ocv) curve modelling and its ageing adjustment. *Journal of Power Sources*, 324:694–703, 2016. doi:10.1016/j.jpowsour.2016.05.121.

[33] Sergio Fernandez, Iker Lopetegi, Laura Oca, Josu Yeregui, Erik Garayalde, and Unai Iraola. Intuitive degradation mode estimation tool: Modest. In *2024 Energy Conversion Congress & Expo Europe (ECCE Europe)*, pages 1–7. IEEE, 2024. doi:10.1109/ECCEEurope62508.2024.10751868.

[34] Niall Kirkaldy, Mohammad Amin Samieian, Gregory J Offer, Monica Marinescu, and Yatish Patel. Lithium-ion battery degradation: Measuring rapid loss of active silicon in silicon–graphite composite electrodes. *ACS applied energy materials*, 5(11):13367–13376, 2022. URL: https://doi.org/10.1021/acsaem.2c02047.

[35] Kieran O'Regan, Ferran Brosa Planella, W Dhammika Widanage, and Emma Kendrick. Thermal-electrochemical parameters of a high energy lithium-ion cylindrical battery. *Electrochimica Acta*, 425:140700, 2022. doi:10.1016/j.electacta.2022.140700.

[36] Sifan Wang, Hanwen Wang, and Paris Perdikaris. On the eigenvector bias of fourier feature networks: From regression to solving multi-scale PDEs with physics-informed neural networks. *Computer Methods in Applied Mechanics and Engineering*, 384:113938, 2021. doi:10.1016/j.cma.2021.113938.

[37] Lu Lu, Raphael Pestourie, Wenjie Yao, Zhicheng Wang, Francesc Verdugo, and Steven G Johnson. Physics-informed neural networks with hard constraints for inverse design. *SIAM Journal on Scientific Computing*, 43(6):B1105–B1132, 2021. doi:10.1137/21M1397908.

[38] Christopher Straub, Philipp Brendel, Vlad Medvedev, and Andreas Rosskopf. Hard-constraining neumann boundary conditions in physics-informed neural networks via fourier feature embeddings. *arXiv preprint arXiv:2504.01093*, 2025. doi:10.48550/arXiv.2504.01093.

[39] Saehong Park, Dylan Kato, Zach Gima, Reinhardt Klein, and Scott Moura. Optimal experimental design for parameterization of an electrochemical lithium-ion battery model. *Journal of The Electrochemical Society*, 165(7):A1309, 2018. doi:10.1149/2.0421807jes.

[40] Adrien M Bizeray, Jin-Ho Kim, Stephen R Duncan, and David A Howey. Identifiability and parameter estimation of the single particle lithium-ion battery model. *IEEE Transactions on Control Systems Technology*, 27(5):1862–1877, 2018. doi:10.1109/TCST.2018.2838097.

[41] René Schenkendorf, Xiangzhong Xie, Moritz Rehbein, Stephan Scholl, and Ulrike Krewer. The impact of global sensitivities and design measures in model-based optimal experimental design. *Processes*, 6(4):27, 2018. doi:10.3390/pr6040027.

[42] Andrea Pozzi, Xiangzhong Xie, Davide M Raimondo, and René Schenkendorf. Global sensitivity methods for design of experiments in lithium-ion battery context. *IFAC-PapersOnLine*, 53(2):7248–7255, 2020. doi:10.1016/j.ifacol.2020.12.558.

[43] Alen Alexanderian, Philip J Gloor, and Omar Ghattas. On bayesian A- and D-optimal experimental designs in infinite dimensions. 2016. doi:10.1214/15-BA969.

[44] Tom Rainforth, Adam Foster, Desi R Ivanova, and Freddie Bickford Smith. Modern bayesian experimental design. *Statistical Science*, 39(1):100–114, 2024. doi:10.1214/23-STS915.

[45] Holger Dette and H-M Neugebauer. Bayesian D-optimal designs for exponential regression models. *Journal of Statistical Planning and Inference*, 60(2):331–349, 1997. doi:10.1016/S0378-3758(96)00131-0.

[46] Calyampudi Radhakrishna Rao, Calyampudi Radhakrishna Rao, Mathematischer Statistiker, Calyampudi Radhakrishna Rao, and Calyampudi Radhakrishna Rao. *Linear statistical inference and its applications*, volume 2. Wiley New York, 1973.

[47] Valentin Sulzer, Scott G Marquis, Robert Timms, Martin Robinson, and S Jon Chapman. Python battery mathematical modelling (PyBaMM). *Journal of Open Research Software*, 9(1), 2021. doi:10.5334/jors.309.

[48] Ruihe Li, Simon O'Kane, Jianbo Huang, Monica Marinescu, and Gregory J Offer. A million cycles in a day: enabling high-throughput computing of lithium-ion battery degradation with physics-based models. *Journal of Power Sources*, 598:234184, 2024. doi:10.1016/j.jpowsour.2024.234184.

[49] Niall Kirkaldy, Mohammad A Samieian, Gregory J Offer, Monica Marinescu, and Yatish Patel. Lithium-ion battery degradation: Comprehensive cycle ageing data and analysis for commercial 21700 cells. *Journal of Power Sources*, 603:234185, 2024. doi:10.1016/j.jpowsour.2024.234185.

[50] Lu Lu, Xuhui Meng, Zhiping Mao, and George Em Karniadakis. DeepXDE: A deep learning library for solving differential equations. *SIAM Review*, 63(1):208–228, 2021. doi:10.1137/19M1274067.

A Appendix

A.1 SPM parameters

Table A1 lists the relevant parameters for the SPM as reported by Chen et al. [31]. Note that we use slightly modified nominal stoichiometries that were adapted and verified to satisfy Eq. (9) via *PyBaMM*'s internal electrode balancing algorithm.

Table A1: Cell specifications and parameters for the SPM as reported by Chen et al. [31].

Parameter	Unit	Description	Anode ($j = n$)	Cathode ($j = p$)
Q_{nom}	[Ah]	Nominal cell capacity	5	
A_j	[m^2]	Electrode surface area	0.1027	
L_j	[m]	Electrode thickness	8.52e-5	7.56e-5
R_j	[m]	Electrode particle radius	5.86e-6	5.22e-6
$\theta_j^{0\%}$	[−]	Stoichiometry at 0% SOC	0.026	0.854
$\theta_j^{100\%}$	[−]	Stoichiometry at 100% SOC	0.911	0.264
ε_j	[−]	Active material volume fraction	0.75	0.665
$c_j^{\max}$	$[\frac{\text{mol}}{\text{m}^3}]$	Maximum concentration	33,133	63,104
a_j	$[\frac{1}{\text{m}}]$	Specific interfacial area	$3\frac{\varepsilon_n}{R_n}$	$3\frac{\varepsilon_p}{R_p}$
c_j^0	$[\frac{\text{mol}}{\text{m}^3}]$	Initial solid phase concentration	cf. Eq. (4)	cf. Eq. (5)
D_j	$[\frac{\text{m}^2}{\text{s}}]$	Solid phase lithium diffusivity	$3.3 \cdot 10^{-14}$	$4.0 \cdot 10^{-15}$
$c_{e,j}$	$[\frac{\text{mol}}{\text{m}^3}]$	Electrolyte concentration (const.)	1000	
F	$[\frac{\text{C}}{\text{mol}}]$	Faraday's constant	96485.33	
R	$[\frac{\text{J}}{\text{mol K}}]$	Universal gas constant	8.31	
T	$[K]$	Temperature	298.15	
$t_{\max}$	[min]	Length of time interval	10	

Figure A1 shows the six different profile groups containing a total of 15 profiles that are used to evaluate PI-DeepONet with respect to surrogate model accuracy, accuracy of Fisher-Information, and eventually for its suitability in the proposed parameter estimation algorithm. The two drive cycles contain the Urban Dynamometer Driving Schedule (UDDS) representing light-duty driving conditions and the US06 protocol representing more aggressive conditions. Both cycles are provided by *PyBaMM*, but scaled to a maximum C-rate of 0.5C and limited to a duration of 10 minutes for the scope of this work.

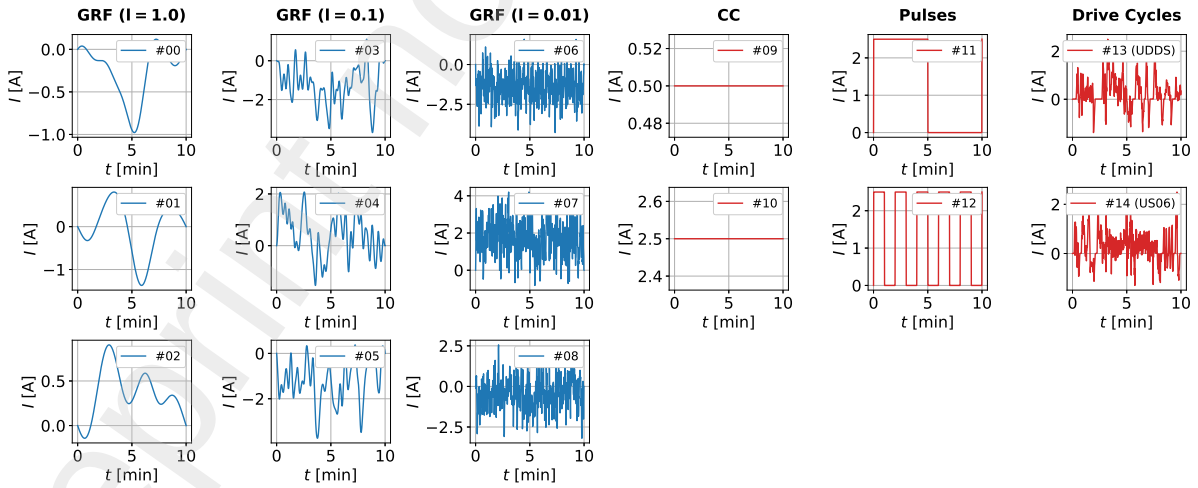

Figure A1: Current profile groups for evaluation of interpolation with GRF profiles of different length scales $l \in \{1.0, 0.1, 0.01\}$ (left) and extrapolation with battery-specific profiles represented by two constant-current profiles (CC), two pulse profiles and two drive cycles obtained via *PyBaMM* (right).

For the scaling of OCPs based on stoichiometric limits, we adapt the original functions from Chen et al., cf. [31], that are available in *PyBaMM* as

$$U_p^{\mathrm{OCP}}(\theta_p) = -0.8090 \cdot \tilde{\theta}_p + 4.4875 - 0.0428 \cdot \tanh\big(18.5138 \cdot (\tilde{\theta}_p - 0.5542)\big) \\ - 17.7326 \cdot \tanh\big(15.7890 \cdot (\tilde{\theta}_p - 0.3117)\big) + 17.5842 \cdot \tanh\big(15.9308 \cdot (\tilde{\theta}_p - 0.3120)\big), \quad \text{(A1)}$$

$$U_n^{\mathrm{OCP}}(\theta_n) = 1.9793 \cdot \exp(-39.3631 \cdot \tilde{\theta}_n) + 0.2482 - 0.0909 \cdot \tanh\big(29.8538 \cdot (\tilde{\theta}_n - 0.1234)\big) \\ - 0.04478 \cdot \tanh\big(14.9159 \cdot (\tilde{\theta}_n - 0.2769)\big) - 0.0205 \cdot \tanh\big(30.4444 \cdot (\tilde{\theta}_n - 0.6103)\big), \quad \text{(A2)}$$

and scale them for different stoichiometric inputs $\theta_j^{0\%}, \theta_j^{100\%}$ based on the electrode-balanced nominal values reported in Tab. A1, i.e.,

$$\tilde{\theta}_p = 0.263849 + \frac{0.853974 - 0.263849}{\theta_p^{0\%} - \theta_p^{100\%}} (\theta_p - \theta_p^{100\%}) \ , \quad \text{(A3)}$$

$$\tilde{\theta}_n = 0.026347 + \frac{0.910612 - 0.026347}{\theta_n^{100\%} - \theta_n^{0\%}} (\theta_n - \theta_n^{0\%}) \ . \quad \text{(A4)}$$

A.2 Implementation details

The PI-DeepONet introduced in Section 2.2 was implemented based on *DeepXDE* with the *TensorFlow 1* backend [50] using hyperparameters as described in Tab. A2. Training performance and runtimes were evaluated on a single NVIDIA H200 GPU with 144GB VRAM.

Parameter	Value
N_Ω	50,000
N_{BC}	10,946
N_I^{train}	10,000 (GRF, $l \in \{1.0, 0.1, 0.01\}$, cf. Eq. (10))
Epochs	1,000,000
Activation Function	SiLU
Initialization	Glorot uniform
Initial Learning Rate (η_0)	$5 \cdot 10^{-4}$
Exp. Learning Rate Decay	$\eta(\mathrm{epoch}) = \eta_0 \cdot 0.8^{\frac{\mathrm{epoch}}{50k}}$
Batch Size - Branch N_I	4
Branch Input Dimension N_t	601
Branch Size $[n_{\mathrm{layers}} \cdot n_{\mathrm{nodes}}]$	$5 \cdot 200$
Trunk Size $[n_{\mathrm{layers}} \cdot n_{\mathrm{nodes}}]$	$3 \cdot [7 \cdot 100]$
Latent Output Dimension m	300
Multi-Output-Strategy (cf. [50])	independent (separate sub-networks per output c_j)
Trainable Parameters	1,167,802
Loss Weights (λ_{PDE}, λ_{BC})	(0.1, 1)

Table A2: Hyperparameters used for training PI-DeepONet in Fig. 1.

Reference solutions were generated based on *PyBaMM 23.5* with default solver settings and time discretization. The mesh size was increased to 1000 points in radial dimension for each electrode, as this was found to be required for sufficient accuracy in the surface concentration profiles, especially for low diffusivity values that cause steep concentration gradients close to the particle surface. In Section 3.3, the DE implementation provided via the open-source *SciPy* package was used with default hyperparameters except for the slightly tuned mutation (0.8) and recombination constants (0.9). Note that due to the high computational costs of the parameter estimation algorithm when relying on the error-free *PyBaMM* surrogate, cf. Fig. A2, these and other DE hyperparameters like population size could not be tuned exhaustively in the scope of this work.

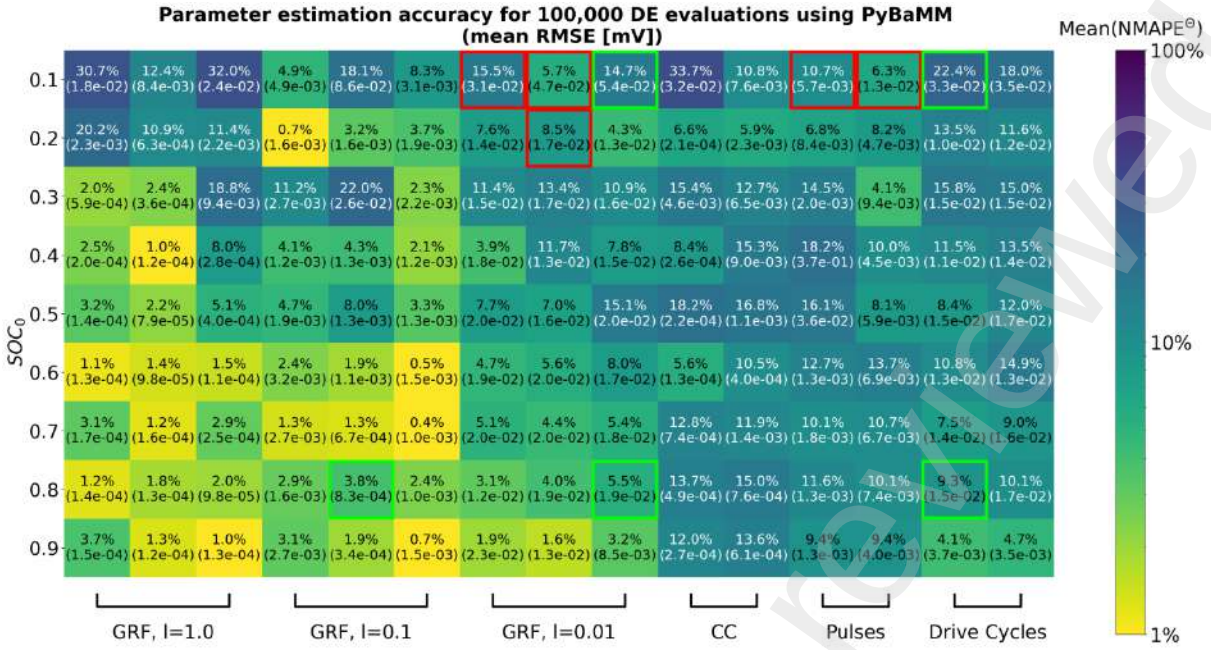

Figure A2: Parameter estimation results using 100,000 DE iterations and *PyBaMM*'s SPM implementation as error-free model. The mean is computed over three estimation runs with different random seeds and the number in brackets refers to the corresponding mean RMSE resulting from Eq. (34). Highlighted cells indicate the five most informative experimental designs based on global optimalities $\overline{D}_{\text{opt}}$ (red) and $\overline{E}^{*}_{\text{opt}}$ (green) as identified in Fig. 7.

A.3 Example of iterative parameter estimation

Figure A3 visualizes one instance of the parameter estimation procedure described in Section 2.4 and evaluated in Section 3.3 in more detail.

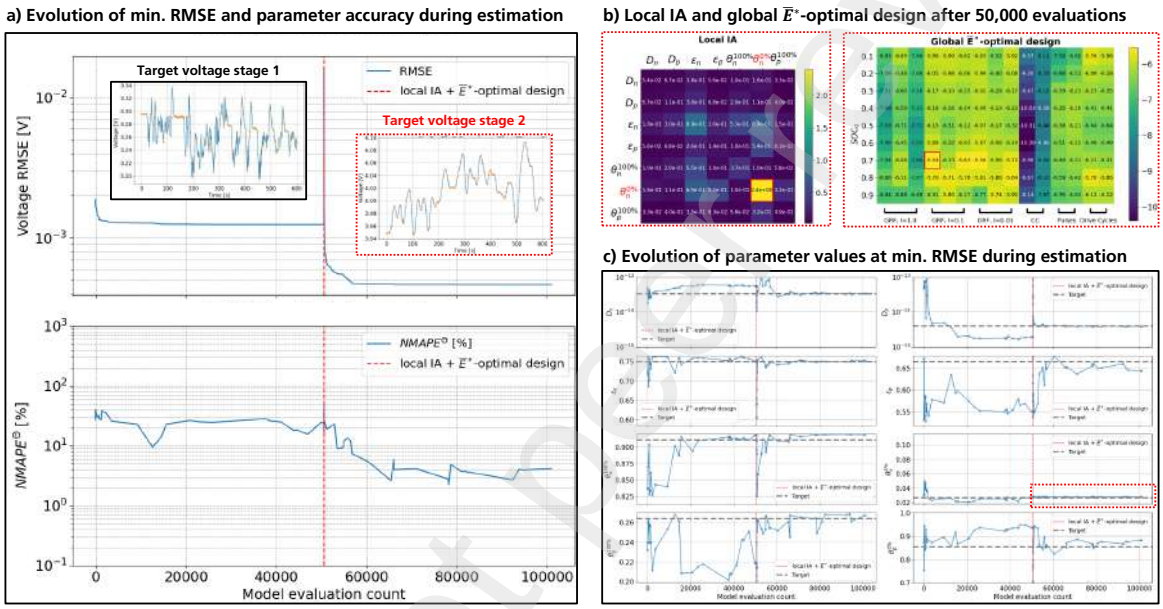

Figure A3: Example of the proposed parameter estimation scheme using profile #13 from Fig. A1 at $SOC_0 = 0.1$ as initial design. a) Evolution of the minimum voltage RMSE during DE optimization, cf. Eq. (34), and corresponding estimation accuracy in terms of NMAPE$^{\Theta}$. b) After 50,000 DE evaluations, parameter $\theta_n^{100\%}$ is fixed as locally identifiable, and the new $\overline{E}^*$-optimal design is chosen via profile #03 at $SOC_0 = 0.7$. c) Evolution of the parameter values corresponding to the best results in terms of voltage RMSE during DE optimization.