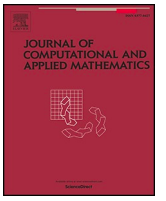
Journal of Computational and Applied Mathematics 483 (2026) 117365

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam



Shape classification of battery cycling profiles via K-Means clustering based on a Sobolev distance

Maria Grazia Quarta [a], Ivonne Sgura [a,*], Massimo Frittelli [a], Raquel Barreira [b,c], Benedetto Bozzini [d]

[a] Università del Salento, Dipartimento di Matematica e Fisica "E. De Giorgi", via per Arnesano, 73100 Lecce, Italy
[b] Instituto Politécnico de Setúbal, Escola Superior de Tecnologia de Setúbal, Campus do IPS Estefanilha, 2914-508 Setúbal, Portugal
[c] CEMS.UL, Center for Mathematical Studies, University of Lisbon, Lisbon, Portugal
[d] Politecnico di Milano, Dipartimento di Energia, via Lambruschini, 2, 20156 Milano, Italy

ARTICLE INFO

Keywords:
Clustering
K-means
Discrete sobolev distance
Finite differences
Lithium metal
Symmetric cell
Galvanostatic cycling

ABSTRACT

Battery cycling, both in application and Research and Development (R&D) environments, generates a wealth of information that often remains underexploited. Thus, potentially valuable information contained in electrical transients is frequently overlooked. In this framework, battery response modelling and model-based data analysis provide powerful tools to extract valuable information on battery status, its evolution and its correlation with functional performance. In this scenario, recently, we have developed a PDE model of battery potential response controlled by electrode shape changes in the BCs for high energy-density metal electrodes. In this work, on the basis of this model, we carry out a classification of the potential transient types, to enable a systematic comparison between model solution and experimental time-series. For transient shape classification purposes, we found that cluster analysis can play a key role in discovering hidden structures within the data. Specifically, in this paper, we apply the K-Means clustering algorithm to classify voltage profiles obtained as numerical solutions of the PDE model for the case of symmetric Li/Li cells. We introduce a *weighted discrete Sobolev distance* that allows us to spot changes in the shape of the voltage profiles, such as formation of peaks, valleys and concavities, that standard metrics such as the L^2 norm fail to capture. As an application, we consider a selection of experimental galvanostatic discharge-charge potential time-series to classify their shape in terms of cluster centroids. Moreover, we show that the new clustering algorithm can provide a segmentation of the parameter space of the PDE model. This partitioning is useful to link the experimental profiles to specific parameter ranges. In particular, we report an example to validate the fitting results of a recent publication of ours obtained via a Deep Learning approach for the same measured profiles.

1. Introduction

Mathematical modelling of electrochemical processes has strongly contributed to the advancement of battery science and technology in the last two decades (see, e.g.: [1–3]). Nevertheless, the advancement of R&D initiatives in the field of post-lithium ion

* Corresponding author.
E-mail addresses: mariagrazia.quarta@unisalento.it (M.G. Quarta), ivonne.sgura@unisalento.it (I. Sgura), massimo.frittelli@unisalento.it (M. Frittelli), raquel.barreira@estsetubal.ips.pt (R. Barreira), benedetto.bozzini@polimi.it (B. Bozzini).

https://doi.org/10.1016/j.cam.2026.117365
Received 9 May 2025; Received in revised form 27 December 2025;
Available online 7 January 2026

chemistries appears to be lagging behind initial projection. One reason is that current mathematical modelling does not fully support the rationalization of dynamic electrode and electrolyte processes towards single-cell scale-up, particularly in the case of systems implementing conversion chemistries. This limited effectiveness is probably due to two main reasons: leading academic focus on specific rather than cell-scale processes (e.g. [4–7]) and only very recent attention to the systematic comparison of mathematical models with extensive corpora of experimental data [8–12].

The present work is meant to contribute to both aspects. In fact, we discuss a PDE model linking electrode shape-changes to electrical cell response and we propose an approach – based on cluster analysis – for the interpretation of large battery response dataset in terms of this model. Cluster analysis is a widely used tool in a variety of application fields, from engineering [13,14] to biology [15], psychology [16], econometry [17], medical [18] and social sciences [19].

It is especially useful when dealing with large datasets, as it helps uncover hidden patterns that may not be immediately evident. One of the most famous algorithms for clustering is K-Means, whose characteristics we shall briefly outline below. In the field of battery-related engineering applications, K-Means has often been used as a tool for model-agnostic data interpretation. For example, in [13] and [14] authors apply K-Means for early detection of battery faults. In [20] the K-means algorithm has been used to analyze battery voltage and current time-series to estimate the battery state and predict its lifespan. Instead [21] uses discharge curve clustering for the evaluation of battery performance.

The Euclidean distance is the dissimilarity measure originally used in the K-means algorithm [22,23]. However, other classical distance metrics have been studied to evaluate and possibly improve the clustering performance. Among the most commonly used measures are the Euclidean and the Manhattan measures [24–27], but other measures have also been explored, such as Canberra [28], Minkowski [29], Chebychev [30] and cross-correlation [31]. Nevertheless, all the distances mentioned do not consider derivative information in calculating the similarity between two data, even though this is crucial for shape and trend identification, as highlighted in a series of publications. For example, a similarity distance using derivative information has been used in [32–34]. More in general, several authors agree in recommending the choice of norms stronger than the L^2 norm to capture the similarity in the (higher-order) derivatives. For example, [35–37] propose to use a Sobolev space of order $k \in \{1,\ 2\}$ of a function f. Furthermore, in [38] a Sobolev-type distance is proposed to measure the dissimilarity from image data. Similarly [39], shows that the Sobolev measure outperforms others considered distances for biomedical data.

Developing on this approach, in this paper we use the K-Means algorithm to study the shape of battery cycling voltage profiles, obtained as numerical solutions of the PDE model, referred as Metal Anode Charging (MAC) model and introduced in [40,41]. In fact, the MAC model simulates voltage profiles for symmetric cells with metal electrodes subjected to constant current discharge/charge cycles. In particular, we use a weighted discrete Sobolev distance to calculate the similarity among data. Here we show how our distance is able to capture changes in profile shapes, taking into account peaks, valleys and concavity. We also show that the L^2 distance is unable to capture the global shape of some data. Differently from previous approaches using Sobolev-type norms in clustering, our formulation introduces a weighted discrete Sobolev distance, specifically designed to compare numerical profiles pointwise. The inclusion of weights allows us to assign higher relevance to regions where the curve slope differs significantly, thus enhancing the sensitivity of the distance to local shape variations.

Specifically, we have applied the K-Means method to potential profiles obtained by numerically solving the MAC model presented in the next section, systematically varying the two physically most relevant parameters (D, r_{max}). As we will detail further in Section 2, D is the diffusion coefficient of the electroactive species and r_{max} represents the maximum radius of the hemispheres arranged on the electrode surface, modelling 3D phase formation. We show that our approach enables the extraction of information from experimental data, by assigning an experimental profile to a specific shape obtained from the clustering of simulated data. In fact, through the K-Means algorithm, a given experimental profile can be associated with a cluster of simulated data by evaluating its distance from the identified cluster centroid.

Moreover, we apply the new Sobolev-based clustering algorithm to the original dataset of voltage profiles used in [10], where the PDE model parameters (D, r_{max}) have been identified by a CNN-LSTM network for a selection of experimental data with different characteristic features. Other recent works on the application of Neural Networks to PDE-related problems are for example [11,42–49].

As a further result, in this paper we show that the outcomes of the new proposed clustering algorithm allow to partition the parameter space of the MAC-PDE model in different regions, thus establishing a link between cluster assignment and specific parameter ranges (Fig. 8). For the experimental profiles studied here, we report the coordinates of the actual calculated (D, r_{max}) values from [10] in the segmented parameter space. As a cross-validation, we also predict, using the Deep Learning approach of [10], the parameter pairs for the centroid profiles found by the K-means algorithm, and we report their coordinates in the segmented parameter space.We confirm that all the parameters identified indeed fall within the regions predicted by the Sobolev-based clustering. Thus, this result provides a powerful and direct validation of the clustering-based segmentation.

Briefly, the key outcomes of this paper are the following.

(i) Using the K-Means method based on the weighted Sobolev distance, applied to a dataset X of simulated data, we determine the number of clusters required via the *Elbow Method* and identify the cluster centroids corresponding to the profile shapes in the dataset.
(ii) Computing the weighted Sobolev distance between experimental data and the nearest centroid allows to assign experimental profiles to specific clusters and shapes.
(iii) By representing the clusters in the space of the parameters used to generate the simulated dataset X, we can segment the parameter space itself. This partition links the profiles shapes to specific parameter ranges and to the corresponding physical meaning.

(iv) This procedure is also used to validate our approach against the parameter identification results of [10].

This work is organized as follows. In Section 2 we describe the MAC model for the Galvanostatic Discharge-Charge (GDC) process of a Li/Li symmetrical cell in presence of nonlinear BCs. In Section 3, we explain how the simulated dataset was built, describing the choice of model parameters, detailing the numerical method based on finite differences for the approximation of the MAC model, and the data preprocessing employed before applying the K-Means algorithm. Section 4 contains the details of the K-Means algorithm with the definition of the weighted discrete Sobolev distance we propose. In Section 5, we expound the results detailed in the above points (i)-(ii) for cluster/centroid identification and for shape classification of experimental data. In Section 5.3, we present the results of point (iii) and the segmentation of the parameter space for the dataset and fitting results in [10], linking profile shapes to specific parameter ranges. In Appendix A, we provide a pseudocode for the Sobolev-based K-means algorithm. In Appendix B, we report the sensitivity analysis for the weighting parameter in the Sobolev distance and in Appendix C we discuss the results on the segmentation of the parameter space.

2. The mathematical model

In the battery science and technology framework sketched in the Introduction, the study of battery voltage profiles is crucial for understanding and optimizing battery performance and lifetime. Recently, we have proposed a PDE model which describes the behavior of voltage profiles for symmetric cells with metal electrodes subjected to galvanostatic (constant current) discharge/charge cycles, as a function of electrode state changes, including 3D outgrowth and passivation, called Metal Anode Charging (MAC) model [40,41]. For the Readers' perusal, in this Section we briefly outline it.

The MAC model describes the electrochemical mass transport of the electroactive species u $\left[\text{mol cm}^{-3}\right]$, Li^+ ions in the present case, and the distribution of the electrostatic potential ϕ [V] across the cell. Our model is inspired by [50], but it has been modified to introduce crucial features of real-life electrodes, such as irreversible shape changes and passivation. These features were not considered in the original model and led to unrealistic, ideally periodic, exclusively cathodic shape changes, regardless of the extent of cycling. The MAC model is:

$$\begin{cases} \dfrac{\partial u}{\partial t} = -\dfrac{\partial J}{\partial x}, \\ \dfrac{\partial^2 \phi}{\partial x^2} = 0, \end{cases} \tag{1}$$

where $0 \le x \le L$ [cm], $0 \le t \le t_{exp}$ [s], with L the length of the cell, considered as 1D, and t_{exp} is the duration of the experiment and

$$J = -D\frac{\partial u}{\partial x} - \frac{zDF}{RT}u\frac{\partial \phi}{\partial x} \tag{2}$$

is the equation for the flux J driven by diffusion and migration. In Eq. (2), D is the diffusion coefficient of the electroactive species $\left[\text{cm}^2\ \text{s}^{-1}\right]$, z $\left[\text{eq mol}^{-1}\right]$ is the valence thereof, F is the Faraday constant, R is the gas constant and T the absolute temperature ($z = 1$ $\left[\text{eq mol}^{-1}\right]$, $F = 96480$ $\left[\text{C eq}^{-1}\right]$, $R = 8.31$ $\left[\text{C V K}^{-1}\ \text{mol}^{-1}\right]$, $T = 300$ [K]). As shown in [41] the BCs for the voltage are

$$\begin{cases} \phi(x=0,t) = \dfrac{RT}{zF}\ln\left[\dfrac{u(x=0,t)}{u_0}\right], \\ \phi(x=L,t) = \dfrac{RT}{zF}\ln\left[\dfrac{u(x=L,t)}{u_0}\right], \end{cases} \tag{3}$$

where u_0 $\left[\text{mol m}^{-3}\right]$ is the constant initial condition for $[Li^+]$ and the ICs of the MAC model are

$$\begin{cases} u(x,t=0) = u_0, \\ \phi(x,t=0) = 0, \end{cases} \quad x \in [0,L]. \tag{4}$$

Instead, the BCs for the flux are

$$\begin{cases} -D\left.\dfrac{\partial u}{\partial x}\right|_{x=0,t} - \dfrac{zDF}{RT}u\Big|_{x=0,t}\left.\dfrac{\partial \phi}{\partial x}\right|_{x=0,t} = \left.\dfrac{i(x,t)}{zF}\right|_{x=0}, \\ -D\left.\dfrac{\partial u}{\partial x}\right|_{x=L,t} - \dfrac{zDF}{RT}u\Big|_{x=L,t}\left.\dfrac{\partial \phi}{\partial x}\right|_{x=L,t} = \left.\dfrac{i(x,t)}{zF}\right|_{x=L}, \end{cases} \tag{5}$$

where $i(x,t)$ is the current density

$$i(x,t) = \frac{I(t)k_{0,hs}}{Ak_0(x,t)}\ x = 0, L\ . \tag{6}$$

In Eq. (6), $I(t)$ is the current intensity program imposed to the cell, a square wave in the present context. Here, $k_{0,hs}$ [cm s^{-1}] is the kinetic constant for the charge-transfer reaction, A [cm^2] is the nominal electrode area, corresponding to the initial flat geometry, and $k_0(x,t)$ [cm s^{-1}] at $x = 0, L$ represents the actual reaction rate corresponding to the instantaneous electrode state, depending on r_{max} [cm], k_{SEI} [1] and $k_{0,flat}$ [cm s^{-1}], that are defined below in Eqs. (8) and (9) [40,41]. The value of $r_{\max}$ represents the maximum radius of the hemispheres arranged on the electrode surface, modelling 3D phase formation, while k_{SEI} [1] accounts for the reaction rate inhibition (passivation) caused by SEI (solid electrolyte interphase) formation. Finally, $k_{0,flat}$ [cm s^{-1}] is the electrodeposition kinetic constant in the areas of the electrode where outgrowth features are not present. The 3D phase formation rate depends on the parameter F_C [1], modelling irreversible electrode shape changes in terms of the mass of the features grown during the charge cycle, that cannot be etched away during the discharge cycle. Thus, the nonlinear BCs contain the parameters bearing information on morphology development and SEI formation. Here, for simplicity and with moderate loss of generality, we shall consider the case of $F_C = 0$ and $k_{SEI} = 0$, corresponding to stable battery cycling in non-pathological operating conditions.

After having solved for $\Delta\phi(t) := \phi(L,t) - \phi(0,t)$, the potential difference between $x = 0$ and $x = L$, the potential drop between the electrodes $\Delta\Phi(t)$, also including electrokinetics and ohmic drop, is computed as:

$$\Delta\Phi(t) = \Delta\phi(t) + \frac{A}{\kappa} L \cdot I(t) + B\frac{I(t)}{|I(t)|} \cdot \ln\left[\frac{1}{2}\left(\frac{i(x=0,t)}{i_0} + \sqrt{\left(\frac{i(x=0,t)}{i_0}\right)^2 + 4}\right)\right] + B\frac{I(t)}{|I(t)|} \cdot \ln\left[\frac{1}{2}\left(\frac{i(x=L,t)}{i_0} + \sqrt{\left(\frac{i(x=L,t)}{i_0}\right)^2 + 4}\right)\right], \qquad t \in [0,\ t_{\exp}] \tag{7}$$

where, κ [Ω^{-1} cm] is the electrolyte specific conductivity, corresponding to an initially homogeneous distribution of electroactive species and flat electrode geometry, L has been already introduced in Eq. (4), the nominal electrode surface area A was introduced in Eq. (6), $i(\cdot,\ t)$ is the current density in Eq. (6), B [V] and i_0 $\left[\text{A cm}^{-2}\right]$ are, respectively, the Tafel slope and the exchange current density: the classical Butler-Volmer electrokinetic parameters (for physico-chemical details, see e.g. [51]). The numerical values of these constants are shown in Eqs. (8) and (9).

In [41] the MAC model Eqs. (1)-(6) has been solved by the `MATLAB` solver `pdepe`, using non uniform time integration steps. For some parameter choices the solver was unable to converge, especially when the profiles present rapid increasing layers during the charge-discharge dynamics. Here we propose a custom finite differences method based on the semi-implicit Euler scheme in time that, in particular, is able to deal in more controlled way with the nonlinearities involved in the BCs (3) and (5). The proposed method is described in the next Section 3.2.

The solution of the MAC model Eqs. (1)-(6) depends on a rather wide set of parameters, that play conceptually different roles. The first subset of parameters is:

$$L = 7.5 \cdot 10^{-2}\ [\text{m}],\quad t_{\exp} = 7200\ [\text{s}],\quad u_0 = 0.5 \cdot 10^{-3}\left[\text{mol m}^{-3}\right],\quad \kappa = 2\ \left[\Omega^{-1}\ \text{cm}\right],\quad k_{0,hs} = 3.5 \cdot 10^{-7}\ \left[\text{s}^{-1}\right],$$
$$k_{0,flat} = 9 \cdot 10^{-7}\ \left[\text{s}^{-1}\right],\quad \kappa = 2\ \left[\Omega^{-1}\ \text{cm}\right],\quad B = 50 \cdot 10^{-3}[\text{V}],\quad i_0 = 15 \cdot 10^{-6}\ \left[\text{A} \cdot \text{cm}^{-2}\right]. \tag{8}$$

They account for cell geometry, materials choice and experiment design in specific experimental conditions. These parameters can – in principle – be varied in a physically meaningful range, but we chose to keep them constant by sticking to the values proposed in [41], since our focus is on tracking the evolution of the anodic cell material.

The second subset of parameters consists of physical constants whose values are intrinsically fixed: $z = 1\ \left[\text{eq mol}^{-1}\right]$, $F = 96480\ \left[\text{C eq}^{-1}\right]$, $R = 8.31\ \left[\text{C V K}^{-1}\ \text{mol}^{-1}\right]$, $T = 300\ [\text{K}]$.

The third set of parameters describes the evolution of the materials brought about by cell cycling, and is formed by the following quantities:

$$A = 1\ \left[\text{cm}^2\right],\quad F_C = 0\ [1],\quad k_{SEI} = 0\ [1]. \tag{9}$$

In this work, along the lines of [40,41], we have varied only the two most physically relevant parameters: the diffusion coefficient D, present in the flux equation (first equation in Eq. (1)), and the maximum radius r_{max} of 3D growth features, present in the nonlinear BCs Eq. (5) through $k_0(x,t)$. In our previous studies, we have shown that, on the one hand, these parameters can capture the key evolutions of the electrolyte chemistry (D) and of electrode shape change (r_{max}) and, on the other hand, that model solutions obtained with appropriate choices of these parameter values can reproduce the potential profiles observed experimentally so far.

3. Dataset definition

In this Section, we discuss the choice of the dataset used for clustering. In particular, in Section 3.1 we motivate the selection of D and r_{max} values. In Section 3.2 we detail the original numerical method we developed for the present study. Finally, in Section 3.3 we describe data preprocessing preliminary to the use of the K-Means algorithm.

3.1. Parameter selection for the solution of the PDE system

The dataset for K-Means analysis is composed of numerical solutions of the MAC model Eqs. (1)-(6). The computed quantity

corresponding to the experimental potential time-series is $\Delta\Phi(t)$ in Eq. (7) modelling the battery voltage as a function of time. The dataset is obtained by varying the values of two parameters in the ranges:

$$D \in \Omega_D = \left[3.208 \cdot 10^{-7}, 3.208 \cdot 10^{-5}\right] \text{ and } r_{\max} \in \Omega_{r_{\max}} = [0.001,\ 0.1],$$

measured in $cm^2 s^{-1}$ and cm, respectively. In the following, we will refer to $\Omega_D \times \Omega_{r_{max}}$ as the parameter space.

As hinted at in the previous Section, this choice is based on the outcomes of our previous sensitivity analysis of the model published in [40,41], which pinpointed that the voltage transient shape is mainly controlled by these two parameters. In particular, D is the parameter that chiefly governs the amplitude of the profiles, while r_{max} controls the shape of the potential transients. In fact, as discussed in [40] the transient shapes vary from a continuous growth to the formation of a maximum, followed by a relaxation to a voltage asymptote and to a maximum followed by a minimum. To build the dataset, we consider a grid of $d \times d$, $d = 45$, logarithmically equally spaced values of D and r_{max} on the parameter space $\Omega_D \times \Omega_{r_{max}}$. We then solve the MAC model Eqs. (1)-(6) with the semi-implicit numerical method detailed in Section 3.2. Hence our dataset contains 2025 simulated voltage profiles. We solve the PDE model for $N_{cyc} = 2$ number of cycles and final time of integration $t_{exp} = 7200$ [s]. We remark that, in the construction of the dataset, the solver `pdepe` failed to solve the PDE system for some choices of the parameters (see for instance Fig. S4(b) in the Supplementary Information of [10]), while our custom solver converged correctly. For instance, this occurs when the parameters are close to $r_{max} = 2.3101 \cdot 10^{-3}$ and $D = 3.562 \cdot 10^{-7}$. This further motivates the usage of a custom numerical solver.

In Fig. 1, we report a representative selection of simulated voltage profiles from the dataset. It is easy to observe that, by varying D and r_{max} the shape and the amplitude of the potential profiles can be profoundly affected.

3.2. Numerical method

In this section we describe the numerical method we devised in this paper for solving the PDE model (1) with BCs and ICs: (3), (5) and (4). By setting $\Omega = [0, L]$, the problem can be restated as:

$$\begin{cases} \dfrac{\partial u}{\partial t} = D\dfrac{\partial^2 u}{\partial x^2} + \dfrac{zDF}{RT}\dfrac{\partial}{\partial x}\left(u\dfrac{\partial \phi}{\partial x}\right) & x \in \Omega, \quad t \in \left[0, t_{\text{exp}}\right], \\ \dfrac{\partial^2 \phi}{\partial x^2} = 0 & x \in \Omega, \quad t \in \left[0, t_{\text{exp}}\right], \\ \phi(x,t) = \dfrac{RT}{zF}\ln\left(\dfrac{u(x,t)}{u_0}\right) & x \in \partial\Omega, \quad t \in \left[0, t_{\text{exp}}\right], \\ -D\dfrac{\partial u}{\partial x}(x,t) - \dfrac{zDF}{RT}u(x,t)\dfrac{\partial \phi}{\partial x}(x,t) = \dfrac{i(t)}{zF} & x \in \partial\Omega, \quad t \in \left[0, t_{\text{exp}}\right]. \end{cases} \tag{10}$$

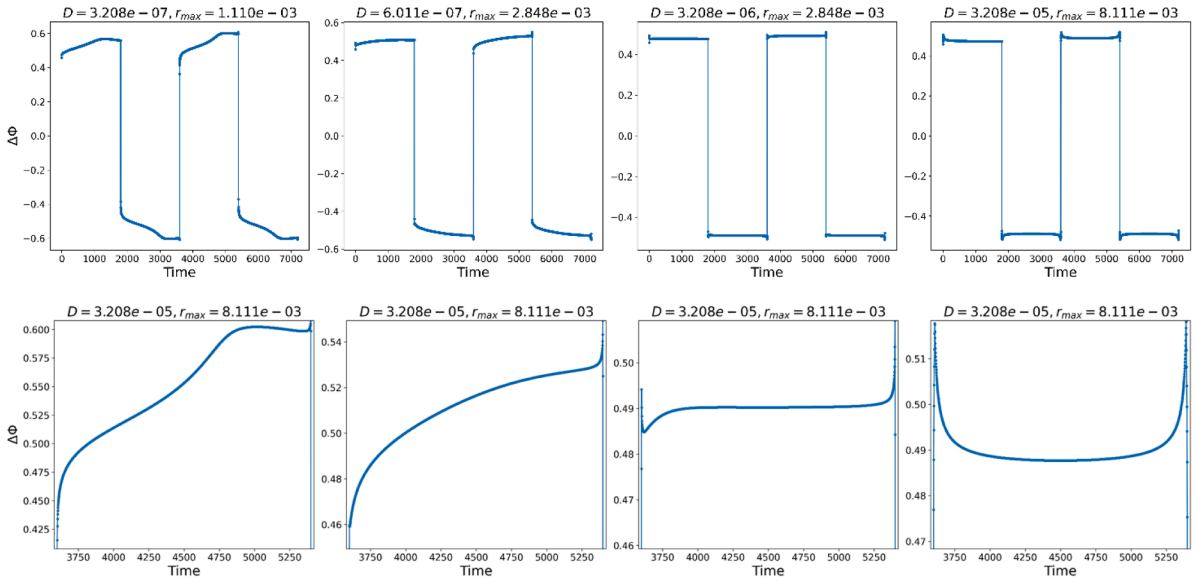


Fig. 1. Dataset samples. Top row: Numerical solutions of the PDE system Eq. (1) with fixed parameters Eqs. (8) and (9) and different pairs $(D, r_{\max})$. Bottom row: Corresponding zoom expansion of the potential transient region showing a richer structure.

It is possible to rewrite the system (10) in an equivalent form by eliminating one equation. In fact, the analytical solution of the second equation is:

$$\phi(x,t) = \beta(t)x + \phi(0,t), \tag{11}$$

where $\beta(t) := \frac{\phi(L,t)-\phi(0,t)}{L} = \frac{\Delta\phi(t)}{L}$ from which it follows that $\frac{\partial\phi}{\partial x} = \beta(t)$. Substituting (11) into (10), we obtain the following reduced system:

$$\begin{cases} \dfrac{\partial u}{\partial t}(x,t) = D\dfrac{\partial^2 u}{\partial x^2}(x,t) + \dfrac{zDF}{RT}\dfrac{\partial}{\partial x}(u(x,t)\beta(t)) & x \in \Omega, \qquad t \in [0, t_{\exp}], \\ \beta(t) = \dfrac{RT}{LzF}\ln\left(\dfrac{u(L,t)}{u(0,t)}\right) & t \in [0, t_{\exp}], \\ -D\dfrac{\partial u}{\partial x}(x,t) - \dfrac{zDF}{RT}u(x,t)\beta(t) = \dfrac{i(x,t)}{zF} & x \in \partial\Omega, \qquad t \in [0, t_{\exp}]. \end{cases} \tag{12}$$

For convenience, we first carry out an appropriate discretization in time that allows us to simplify the numerical solution of the problem. In this case, we use a semi-implicit Euler method, because we treat explicitly the function $\beta(t)$. Let us consider a time step $\Delta t > 0$ and the discrete times $t_k = k\Delta t$, where $k = 0, \dots, N_t - 1$ and $N_t = \frac{t_{exp}}{\Delta t} \in \mathbb{N}$. The first equation of the system (12) becomes

$$\frac{u^{(k+1)}(x) - u^{(k)}(x)}{\Delta t} = D\frac{\partial^2 u^{(k+1)}}{\partial x^2}(x) + \frac{zDF}{RT}\beta^{(k)}\frac{\partial u^{(k+1)}}{\partial x}(x), \qquad x \in \Omega. \tag{13}$$

where $u^{(k+1)}$ is the time-discrete solution $u(x,t)$ at time t_k, $i^{(k+1)} := i(t_{k+1})$ and

$$\beta^{(k)} := \frac{RT}{LzF}\ln\left(\frac{u^{(k)}(L)}{u^{(k)}(0)}\right). \tag{14}$$

The nonlinear boundary conditions (12)-(3) for u can be discretized in time as follows:

$$\frac{\partial u^{(k+1)}}{\partial x}(x) = -\frac{1}{D}\left(\frac{i^{(k+1)}(x)}{zF} + \frac{zDF}{RT}u^{(k)}(x)\beta^{(k)}\right) =: g^{(k)}(x), \qquad x \in \partial\Omega. \tag{15}$$

We can observe that, since $g^{(k)}$ is already known as the result of the $(k-1)$-th iteration in time, the boundary conditions (15) are nonhomogeneous *linear* Neumann conditions for the unknown function $u^{(k+1)}$, which allow for more accurate numerical solutions compared to the original nonlinear boundary conditions. Hence, we have obtained an original linearization of the nonlinear coupled boundary conditions in (12). If the system (12) was discretized with a classical full discretization method in space and time, we would obtain a sequence of nonlinear algebraic systems that would require a nonlinear solver such as Newton's method.

For the spatial discretization, we consider $N+1$ equally spaced nodes $\{x_j\}_{j=0}^{N}$ in Ω with step $h = \frac{L}{N}$ and the two ghost nodes $x_{-1} = 0 - h$ and $x_{N+1} = L + h$. Using centered finite differences for the approximation of spatial derivatives, we can write

$$\begin{cases} \dfrac{u_j^{(k+1)} - u_j^{(k)}}{\Delta t} = D\dfrac{u_{j+1}^{(k+1)} - 2u_j^{(k+1)} + u_{j-1}^{(k+1)}}{h^2} + \dfrac{zDF}{RT}\beta^{(k)}\dfrac{u_{j+1}^{(k+1)} - u_{j-1}^{(k+1)}}{2h}, & j = 0, \cdots, N \\ u_j^{(0)} = u_0(x_j), & j = 0, \cdots, N \end{cases} \tag{16}$$

and for the approximation of the boundary conditions (15) we have

$$\frac{\partial u_0^{(k+1)}}{\partial x} \approx \frac{u_1^{(k+1)} - u_{-1}^{(k+1)}}{2h} = g^{(k)}(0), \qquad \frac{\partial u_N^{(k+1)}}{\partial x} \approx \frac{u_{N+1}^{(k+1)} - u_{N-1}^{(k+1)}}{2h} = g^{(k)}(L), \tag{17}$$

from which we get

$$u_{-1}^{(k+1)} = -2hg^{(k)}(0) + u_1^{(k+1)}, \qquad u_{N+1}^{(k+1)} = 2hg^{(k)}(L) + u_{N-1}^{(k+1)}. \tag{18}$$

Therefore, the full discretization of (12) is

$$\begin{cases} \dfrac{u_0^{(k+1)}-u_0^{(k)}}{\Delta t} = D\dfrac{2u_1^{(k+1)}-2u_0^{(k+1)}-2hg^{(k)}(0)}{h^2}+\dfrac{zDF}{RT}\beta^{(k)}g^{(k)}(0), & j=0 \\ \dfrac{u_j^{(k+1)}-u_j^{(k)}}{\Delta t} = D\dfrac{u_{j+1}^{(k+1)}-2u_j^{(k+1)}+u_{j-1}^{(k+1)}}{h^2}+\dfrac{zDF}{RT}\beta^{(k)}\dfrac{u_{j+1}^{(k+1)}-u_{j-1}^{(k+1)}}{2h}, & j=1,\cdots,N-1 \\ \dfrac{u_N^{(k+1)}-u_N^{(k)}}{\Delta t} = D\dfrac{2u_{N-1}^{(k+1)}-2u_N^{(k+1)}+2hg^{(k)}(L)}{h^2}+\dfrac{zDF}{RT}\beta^{(k)}g^{(k)}(L), & j=N \\ u_j(0)=u_0(x_j), & j=1,\cdots,N \end{cases} \tag{19}$$

In matrix form (19) can be restated as:

$$\left(I+D\Delta t\left(\frac{1}{h^2}A+\frac{zF}{2hRT}\beta^{(k)}B\right)\right)\boldsymbol{u}^{(k+1)} = \boldsymbol{u}^{(k)}+D\Delta t\left(-\frac{2}{h}\boldsymbol{B}_1{}^{(k)}+\frac{zF}{RT}\beta^{(k)}\boldsymbol{B}_2{}^{(k)}\right) \tag{20}$$

where $\boldsymbol{u}^{(k+1)} := \left(u_0^{(k+1)},\ldots,u_N^{(k+1)}\right)^T$ is the nodal vector of the fully discrete solution at the time step $k+1$, $\boldsymbol{u}_0 = (u_0(x_0),\ldots,u_0(x_N))^T$ is the nodal vector of the initial condition u_0, A is the matrix of size $(N+1)\times(N+1)$ of the discrete Laplacian with Neumann boundary conditions, B is the matrix of size $(N+1)\times(N+1)$ of the convection term:

$$A=\begin{bmatrix} 2 & -2 & 0 & \cdots & 0 \\ -1 & 2 & -1 & \ddots & \vdots \\ 0 & \ddots & \ddots & -1 & 0 \\ \vdots & & -1 & 2 & -1 \\ 0 & \cdots & 0 & -2 & 2 \end{bmatrix}, \qquad B=\begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ 1 & 0 & -1 & \ddots & \vdots \\ 0 & \ddots & \ddots & -1 & 0 \\ \vdots & & 1 & 0 & -1 \\ 0 & \cdots & 0 & 0 & 0 \end{bmatrix}. \tag{21}$$

$\boldsymbol{B}_1{}^{(k)}$ and $\boldsymbol{B}_2{}^{(k)}$ are column vectors:

$$\boldsymbol{B}_1{}^{(k)}=\begin{pmatrix} g^{(k)}(0) \\ 0 \\ \vdots \\ 0 \\ -g^{(k)}(L) \end{pmatrix}, \quad \boldsymbol{B}_2{}^{(k)}=\begin{pmatrix} g^{(k)}(0) \\ 0 \\ \vdots \\ 0 \\ g^{(k)}(L) \end{pmatrix}. \tag{22}$$

For simplicity let us set:

$$C^{(k)} := D\left(\frac{1}{h^2}A+\frac{zF}{2hRT}\beta^{(k)}B\right), \qquad \boldsymbol{\gamma}^{(k)} := D\Delta t\left(-\frac{2}{h}\boldsymbol{B}_1{}^{(k)}+\frac{zF}{RT}\beta^{(k)}\boldsymbol{B}_2{}^{(k)}\right). \tag{23}$$

Therefore Eq. (20) can be written as:

$$\left(I+\Delta t\ C^{(k)}\right)\boldsymbol{u}^{(k+1)} = \boldsymbol{u}^{(k)}+\boldsymbol{\gamma}^{(k)}, \qquad k=0,\cdots,N_t. \tag{24}$$

For the spatial and the temporal discretization, we choose h and Δt appropriately so that the solutions converged throughout the parameter space $\Omega_D\times\Omega_{r_{max}}$. For each step k we solve the linear system (24) by using the LU direct solver (backslash) in MATLAB. For the computations presented in the remainder of the paper, we will need the voltage $\Delta\phi^{(k)} = L\,\beta^{(k)}$ obtained from the first and last components of the fully discrete solution $\boldsymbol{u}^{(k)}$ according to Eq. (14) for $k=1,\ldots,N_t$. Hence, the discrete values for the variable of interest $\Delta\Phi^{(k)}$ for $k=1,\ldots,N_t$ can be recovered by evaluating the expression in Eq. (7).

3.3. Data preprocessing

In this Section, we describe the preprocessing applied to the dataset of numerical solutions of the MAC model, as a preliminary step to the application of the K-Means clustering algorithm [22]. Since, in the present work, we are setting $F_C = 0$ and $k_{SEI} = 0$, the charge and discharge half-cycles are symmetrical. Hence the cycles exhibit constant amplitude over time, except for the first one, in which the transition from the initial state to the steady cycle occurs. In this paper we restrict our analysis to a half cycle in the periodic regime, that is fully representative of practical battery performance. This is done to avoid the initial transient period and focus on the stable, representative electrochemical behavior.

Considering that our objective is to classify profiles according to their shape (without taking amplitude into account) we normalize data between 0 and 1. This is also useful when dealing with experimental profiles because the amplitude could depend on the units of measurement used during the experiment.

The key steps are then listed below:

(i) for each $\Delta\mathbf{\Phi}_n,\ n = 1,\dots,N,\ N = 2025,$ evaluated according to Eq. (7), we select its last charge half-cycle: $\overline{\Delta\mathbf{\Phi}}_n$ of elements $\Delta\mathbf{\Phi}_{i,\,n}$ where $i = 3600,\dots,5400$ and $n = 1,\dots\dots,N$;

(ii) we normalize the resulting profiles between 0 and 1 to obtain our dataset $X = \{\boldsymbol{x}_n\}_{n=1}^{N}$, where

$$\boldsymbol{x}_n = \frac{\overline{\Delta\mathbf{\Phi}}_n - \min(\overline{\Delta\mathbf{\Phi}}_n)}{\max(\overline{\Delta\mathbf{\Phi}}_n) - \min(\overline{\Delta\mathbf{\Phi}}_n)},\ n = 1,\dots,N. \tag{25}$$

Therefore, $\boldsymbol{x}_n \in \mathbb{R}^{1800}$ and X contains $N = 2025$ data of length $s = 1800$.

In Fig. 2, we show an example of the post-processed numerical solution of the MAC mode according to Eq. (7), the charge half-cycle extracted for the clustering study (left plot,) and its normalized version (right plot).

4. K-Means clustering analysis of voltage profiles

To gain practically useful information for design and interpretation of experiments, based on the model considered in this work, it is essential to extract a clearcut physical message from the wealth of subtle details concealed in the population of numerical solutions. For this purpose, in this Section, we apply K-Means clustering to classify simulated voltage profiles, with the aim of identifying recurring shapes and distinct classes of behavior present in the data. Instead of using the classical Euclidean distance typically adopted in clustering algorithms, we introduce a new distance function based on *a weighted discrete Sobolev metric*, as first proposed in [10]. This choice is motivated by the fact that in voltage profiles both the data values and their derivatives contain essential information.

In addition to traditional parameter estimation approaches, based on least-squares cost functions and PDE-constrained optimization (see [52–54]), in [10] we applied a Deep Learning approach for parameter estimation using a hybrid Neural Network with convolutional and Long Short-Term Memory layers (CNN-LSTM). In [10], we have shown that the Sobolev distance allows better detection of shape differences, such as peaks, valleys and saddle points present in the data.

In view of developing a comprehensive toolbox for a systematic comparison between model solutions and experimental data, in combination with punctual parameter identification, it would also be useful to carry out a higher-level comparison of classes of voltage profiles on the basis of their shapes. Clustering precisely allows us to identify different groups of voltage profiles, each representing a specific voltage behavior, as informally put forward in [40]. Thus, by using the K-Means clustering algorithm on a population of charge-discharge voltage profiles, we aim to uncover hidden patterns in the data and to associate them with regions of the parameter space and of operating conditions.

4.1. Standard K-Means algorithm

This section outlines the key features of K-Means, a classical unsupervised learning algorithm [22,55], sometimes also referred to as the Lloyd–Forgy algorithm [23]. According to the unsupervised learning strategy, K-Means aims to group data points into clusters based on similarity, without requiring labels.

The K-Means strategy considers a dataset $X = \{\boldsymbol{x}_n\}$, $n = 1,\dots,N$ and each of the N observation is a s-dimensional real vector. The goal is to partition the dataset X into K clusters, where $K < N$ is given. Each cluster can be regarded as a collection of data points where the distances between points are relatively small compared to the distances to points outside the cluster itself. We can formalize this notion by first introducing a set of s-dimensional vectors $\boldsymbol{\mu}_k$, $k = 1,\dots,K$, where $\boldsymbol{\mu}_k$ is the prototype associated with the k-th cluster. In other words, we can think of $\boldsymbol{\mu}_k$ as representing the centers of the clusters and refer to them as *centroids*. The goal of the clustering task is thus to find: (i) a set of vectors $\{\boldsymbol{\mu}_k\}$ and (ii) an assignment of data points to clusters, such that the sum of the squares of the distances

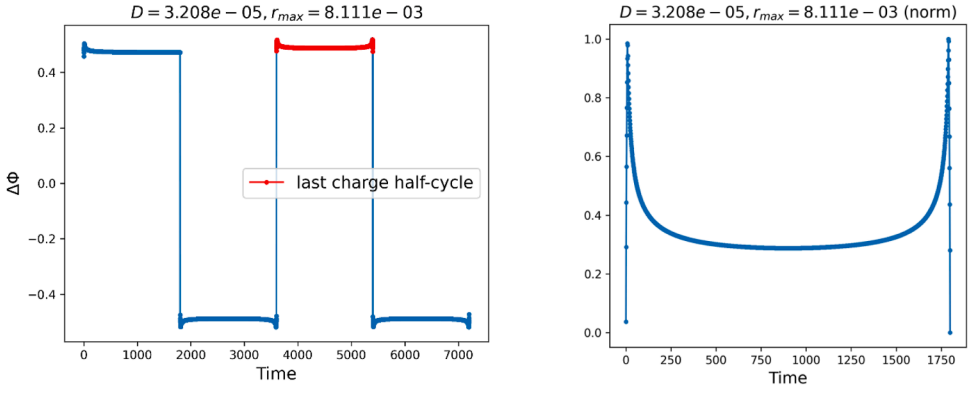


Fig. 2. Simulated Data sample. Left plot: Example of the postprocessed numerical solution according to Eq. (7) (blue) and extraction of its last charge half-cycle. (red). Right plot: the normalized half cycle used to build the corresponding dataset element.

of each data point from its assigned $\boldsymbol{\mu}_k$, is minimized [56]. It is convenient to introduce, for each data point $\boldsymbol{x}_n$, a corresponding set of binary variables $r_{n_k} \in \{0,1\}$ where $k = 1,\ldots,K$, indicating which of the K clusters the data point $\boldsymbol{x}_n$ is assigned to. Thus, if $\boldsymbol{x}_n$ is assigned to cluster k then $r_{n_k} = 1$ and $r_{n_j} = 0$ for $j \neq k$. This device is referred to as the 1-of-K coding scheme [56]. We can thus define a cost function given by:

$$Cost(\boldsymbol{x}_n, \boldsymbol{\mu}_k) = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{n_k} ||\boldsymbol{x}_n - \boldsymbol{\mu}_k||^2, \tag{26}$$

which represents the sum of the squares of the distances of each data point to its assigned centroid $\boldsymbol{\mu}_k$. Our goal is then to find values for $\{r_{n_k}\}$ and $\{\boldsymbol{\mu}_k\}$ such that the cost function *Cost* in Eq. (26) is minimized.

This task can be performed through an iterative procedure consisting of two successive steps: optimization with respect to r_{n_k} and $\boldsymbol{\mu}_k$. (i) In the first phase the K-Means algorithm randomly selects K centroids from the dataset. Then *Cost* in (26) is minimized with respect to the r_{n_k}, keeping $\boldsymbol{\mu}_k$ fixed. During this phase the Euclidean distance of each data point $\boldsymbol{x}_n$ from all the chosen centroids is evaluated, and each point is then assigned to the nearest centroid, becoming a member of that cluster. (ii) In the second phase, we minimize *Cost* with respect to the $\boldsymbol{\mu}_k$, keeping r_{n_k} fixed and re-evaluate the center of each cluster [57].

Let us first focus on step (i): determining r_{n_k}. Since the terms that involve different n are independent, we can optimize for each n separately by choosing r_{n_k} to be 1 for the value of k that gives the minimum value of $||\boldsymbol{x}_n - \boldsymbol{\mu}_k||^2$. This can be expressed as

$$r_{n_k} = \begin{cases} 1 & \text{if } \; k = \underset{n}{arg\min} ||\boldsymbol{x}_n - \boldsymbol{\mu}_k||^2, \\ 0 & \text{otherwise.} \end{cases} \tag{27}$$

Considering then step (ii), for the optimization of $\boldsymbol{\mu}_k$, since the objective function *Cost* is a quadratic function of $\boldsymbol{\mu}_k$, it can be minimized by setting its derivative with respect to $\boldsymbol{\mu}_k$ equal to zero:

$$2 \sum_{n=1}^{N} r_{n_k} (\boldsymbol{x}_n - \boldsymbol{\mu}_k) = 0, \tag{28}$$

and by solving for $\boldsymbol{\mu}_k$ we obtain

$$\boldsymbol{\mu}_k = \frac{\sum_{n=1}^{N} r_{n_k} \boldsymbol{x}_n}{\sum_{n=1}^{N} r_{i_k}}. \tag{29}$$

The denominator in (29) is equal to the number of data assigned to the cluster k, therefore $\boldsymbol{\mu}_k$ is the mean of all the data points $\boldsymbol{x}_n$ assigned to the cluster k. This is why this procedure is known as K-Means algorithm.

The two phases of: - (i) data point re-assignment to clusters and (ii) centroid re-computation for each cluster - are repeated until the assignments no longer change, or the maximum number of iterations is reached. Since each phase reduces the value of the objective function *Cost*, the algorithm is guaranteed to converge. However, this algorithm may converge to a local minimum [55], suggesting that a larger number of clusters K results in a lower square error [58]. Selecting the correct value for K is crucial and remains a significant challenge for the standard K-Means algorithm, as different values of K produce different results [58]. A well-known strategy to determine K is the Elbow Method [59]. See next Section 5.1 for further details.

4.2. K-Means algorithm with a custom cost function

In this Section, we introduce a novel cost function, devised to customize the K-Means algorithm for our purposes, which replaces the standard Euclidean cost function (26) described in Section 4.1. This cost function is based on a *weighted discrete Sobolev distance*, which extends existing Sobolev-based formulations by operating on discrete data points and considering weights. The weights strongly penalize regions where the first and the second derivatives of the profiles differ significantly, thereby avoiding local slope and concavity discrepancies and improving shape discrimination. We then motivate our choice by presenting an illustrative example and comparison with the classical Euclidean distance, highlighting the advantages of our approach in the context of analyzing experimental voltage cycle data.

The distance we propose is based on the H^2 Sobolev norm [60,61]. By definition, this distance captures functional properties that the standard L^2 distance overlooks. Specifically, the H^2 Sobolev-based distance incorporates information from both first and second derivatives of a function, allowing to account for features such as local maxima and minima, saddle points, and convexity. Therefore, we define here the H^2 discrete Sobolev distance between $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^s$ as follows:

$$H^2(\boldsymbol{u}, \boldsymbol{v}) = \left(||\boldsymbol{u} - \boldsymbol{v}||_{L^2}^2 + ||\boldsymbol{u}' - \boldsymbol{v}'||_{L^2}^2 + ||\boldsymbol{u}'' - \boldsymbol{v}''||_{L^2}^2 \right)^{\frac{1}{2}} \forall \boldsymbol{u}, \boldsymbol{v}, \tag{30}$$

where $\boldsymbol{u}', \boldsymbol{v}'$ and $\boldsymbol{u}'', \boldsymbol{v}''$ are the approximations of the first and second derivatives of $\boldsymbol{u}$ and $\boldsymbol{v}$, respectively, computed by finite differences. Specifically, we consider the weighted discrete H^2 distance given by:

$$H^2_W(\boldsymbol{u},\boldsymbol{v};a) = \left(||\boldsymbol{u}-\boldsymbol{v}||^2_{L^2_W} + ||\boldsymbol{u}'-\boldsymbol{v}'||^2_{L^2_W} + ||\boldsymbol{u}''-\boldsymbol{v}''||^2_{L^2_W}\right)^{\frac{1}{2}} \forall \boldsymbol{u},\boldsymbol{v}, \tag{31}$$

where $||\boldsymbol{u}-\boldsymbol{v}||_{L^2_W} = \left((\boldsymbol{u}-\boldsymbol{v})^T W(a)(\boldsymbol{u}-\boldsymbol{v})\right)^{\frac{1}{2}}$ is a weighted L^2 norm, with $W(a)$ the diagonal matrix $W = diag\left(\boldsymbol{w}_{d_1} \circ \boldsymbol{w}_{d_2}\right)$, where $\circ$ is the Hadamard product between the vectors $\boldsymbol{w}_{d_1}$ and $\boldsymbol{w}_{d_2}$. The weights include information about the first and the second (discrete) derivatives as follows:

$$\begin{aligned} (\boldsymbol{w}_{d_1})_i &= \begin{cases} 1, & \text{if } sign(\boldsymbol{u}_i') \neq sign(\boldsymbol{v}_i') \\ a, & \text{otherwise} \end{cases}, \\ (\boldsymbol{w}_{d_2})_i &= \begin{cases} 1, & \text{if } sign(\boldsymbol{u}_i'') \neq sign(\boldsymbol{v}_i'') \\ a, & \text{otherwise} \end{cases}, \end{aligned} \qquad \text{for } i = 1,\cdots,s, \tag{32}$$

where $a > 0$ is a positive small number. Specifically, for $a < 1$ the weighted norm emphasizes the contribution of the derivatives more than the function values themselves.

It can be easily shown that $H^2_W(a)$ is still a distance metric. This definition of $W(a)$ in terms of Eq. (32) penalizes discrepancies between the first and second derivatives of $\boldsymbol{u}$ and $\boldsymbol{v}$. In fact, the $H^2_W(a)$ distance in Eq. (31) increases whenever the profiles $\boldsymbol{u}$ and $\boldsymbol{v}$ differ in the sign of their derivatives at corresponding points.

With this definition of $H^2_W(a)$, following the notations introduced in Section 4.1, we can specify the custom cost function used in our K-Means algorithm as:

$$\overline{Cost}(\boldsymbol{x}_n,\boldsymbol{\mu}_k) = \sum_{n=1}^{N}\sum_{k=1}^{K} r_{n_k} H^2_W(\boldsymbol{x}_n,\boldsymbol{\mu}_k;a). \tag{33}$$

The code for the K-Means clustering using the weighted discrete Sobolev distance has been implemented in Python 3.11.0. In Appendix A we report the pseudocode to implement this technique, that we call K-Means-$H^2_W(a)$ algorithm. Moreover, it is worth noting that the proposed Sobolev-based K-means algorithm depends on the choice of the parameter a in the weighted Sobolev norm of Eqs. (31)-(32). In fact, different values of a can lead to different profile shapes corresponding to minimal distance solutions of (33). To guide the selection of an appropriate value, Appendix B provides a *sensitivity analysis* for the parameter a using an "*elbow-like*" technique, showing how different choices affect the minimal-distance profiles. For all the experimental data considered in this paper this analysis shows that $a = 0.1$ is the optimal value (See Fig. B–Appendix B): this value is thus used in the analyses presented in this work.

4.3. Comparing L^2 and H^2_W metrics on experimental data

In this Section, we show, on the basis of experimental data, that while the classical L^2 distance is unable to capture the global shape of profiles, the weighted Sobolev distance $H^2_W(a)$ we propose in (31) successfully accomplishes this task. This occurs because the Sobolev distance exploits information about the first and second derivatives. Specifically, to show this, we consider the experimental data of galvanostatic Li/Li symmetric cell cycling that we have employed for parameter identification in [10] -to which we refer for full experimental details- (see next Fig. 6).

For our goal, in practice we have computed both the L^2 and $H^2_W(a=0.1)$ distances between a chosen experimental profile and all

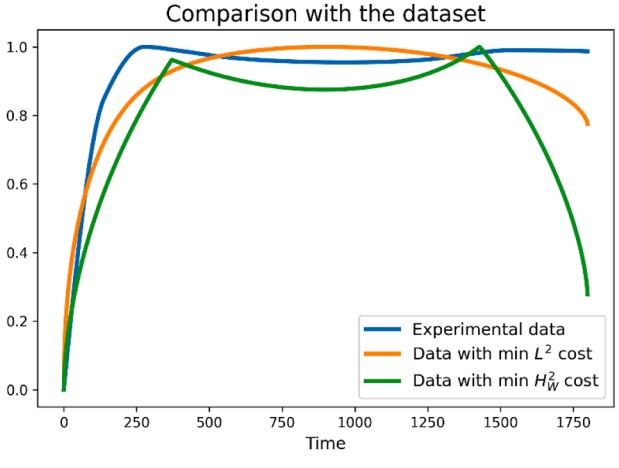


Fig. 3. Comparison between weighted Sobolev metric and L^2 metric. Comparison between the experimental data (blue) and the simulated data from the dataset $X = \{\boldsymbol{x}_n\}_{n=1}^N$ which minimize the L^2 metric and the weighted Sobolev metric $H^2_W(a=0.1)$, orange and green curves, respectively.

simulated profiles $\{\boldsymbol{x}_n\}_{n=1}^N$ in the dataset, and then we have extracted the two profiles $\boldsymbol{x}_{LS}$ and $\boldsymbol{x}_H$ minimizing these distances, respectively. Fig. 3 shows the considered experimental profile (blue curve, in the following called $\boldsymbol{\varepsilon}_2$) and the two optimal simulated profiles $\boldsymbol{x}_{LS}$ and $\boldsymbol{x}_H$ for this example. We can observe that this experimental data has an increasing trend, followed by a maximum, then decreases to a minimum and then rises again to a second maximum: therefore, overall, the curve has a minimum and two peaks. The identified Least Square (LS) profile is significantly different from the experimental profile and does not show a minimum and two peaks, in fact, LS increases to a single maximum and then decreases. Therefore, this example demonstrates that the traditional L^2 distance is not effective for comparing experimental and simulated profiles, because it is likely to miss important morphological features, thus failing to capture the main characteristics of the cycles' shapes. Instead, the profile of the dataset that minimizes the distance $H^2_W(a=0.1)$ presents a shape that reflects the characteristics details of the experimental data. In particular, it has a maximum, then a minimum, and another maximum, and respects the same concavity pattern as the experimental data.

5. Clustering results

In this Section, we expound the results of the clustering analysis performed on the dataset $X=\{\boldsymbol{x}_n\}_{n=1}^N$, generated in (25) of Section 3 and on the experimental data presented in Section 4.3. The analysis is carried out using the K-Means algorithm with the weighted discrete Sobolev distance (31) introduced in Sections 4.1 and 4.2. Specifically, in Section 5.1, we apply the *Elbow Method* to determine the optimal number of clusters and identify the corresponding centroids in the simulated dataset X. Section 5.2 focuses on the assignment of the four analyzed experimental potential profiles to the closest cluster centroids, which enables their shape classification (see Fig. 6). Finally, in Section 5.3, we apply the K-Means algorithm with the weighted Sobolev distance to the dataset of [10], to obtain a segmentation of the parameter space (D,r_{max}). We show that the parameters estimated in [10] through the CNN-LSTM approach for the same experimental data correctly fall within the corresponding subregions identified by the clustering analysis. Moreover, as cross-validation, we apply the fitting NN approach in [10] to predict the parameters of the cluster centroids and represent them in the partitioned parameter space (see Figs. 7 and 8).

5.1. Elbow method and centroids

To determine the number K of clusters into which the dataset should be divided, we used the Elbow Method [59], consisting in iterating the K-Means algorithm for $K=1,\cdots,9$. For each value of K, we calculate the Sum of Squared Distances (SSD) in the Sobolev norm, measuring the distance of each data sample from its nearest centroid, given by

$$SSD(K;a)=\sum_{n=1}^{N}\sum_{k=1}^{K}\left(H^2_W(\boldsymbol{x}_n,r_{n_k}\cdot\boldsymbol{\mu}_{k,};a)\right)^2,\ K=1,\dots,9 \tag{34}$$

where r_{n_k} has been defined in Eq. (27). As discussed in the previous section, we set $a=0.1$. By plotting the $SSD(K;0.1)$, as in Fig. 4, we can determine the optimal value of K by observing the "*elbow*" point. Concretely, the best K corresponds to the point where the difference of the computed SSD between successive K values falls below a small threshold. For small K, the SSD decreases rapidly, but after the optimal K is reached, SSD decreases more slowly and finally tends to an asymptote [59]. According to this Elbow Method, we identified $K=4$ as the optimal value.

Henceforth we refer to the heuristic clusters $\mathscr{C}_0$: "*Double-Peak Shallow Minimum*", $\mathscr{C}_1$: "*Single-Peak*", $\mathscr{C}_2$: "*Double-Peak Deep Minimum*" and $\mathscr{C}_3$: "*Monotonic*", each containing m_k elements for $k=0,\dots,3$. The corresponding centroids will be indicated as $\boldsymbol{\mu}_k\in\mathbb{R}^s$

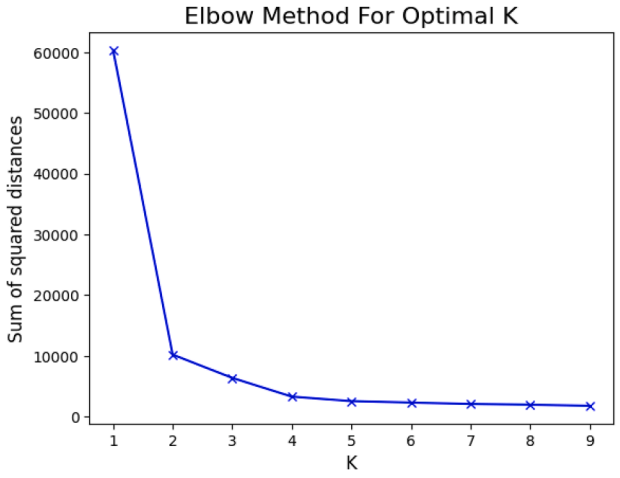


Fig. 4. Elbow method for optimal K. $SSD(K;a)$ variation as function of K, for optimal K estimation with the Elbow Method, for $a=0.1$. See main text for details.

for $k = 0, \ldots, 3$.

Furthermore, a visual inspection of the centroids for $K = 1, \ldots\ldots, 9$ shows that up to $K = 4$ the newly computed centroids exhibit clearly different shapes from each other, whereas for $K = 5, \ldots\ldots, 9$ some centroids can be assimilated to the same type of shape.

Our data clustering yields: $m_0 = 311$ profiles in $\mathscr{C}_0$: "*Double-Peak Shallow Minimum*", $m_1 = 554$ in $\mathscr{C}_1$: "*Single-Peak*", $m_2 = 600$ in $\mathscr{C}_2$: "*Double-Peak Deep Minimum*" and $m_3 = 560$ in $\mathscr{C}_3$: "*Monotonic*".

In analogy with the heuristic approach of [40], here we carry out a quantitative classification of the profiles in the dataset generated in Section 3, according to their shape. This classification is quantitative because it relies on the weighted discrete Sobolev distance as a numerical measure, but it is aimed at capturing qualitative aspects of the profiles, such as the presence of maxima, minima, and concavity features. By inspecting the centroid profiles of each cluster in Fig. 5, we can straightforwardly establish the respective key properties. Centroid $\boldsymbol{\mu}_0$ ("*Double-Peak Shallow Minimum*") presents a maximum, followed by a minimum and then by a maximum. Centroid $\boldsymbol{\mu}_1$("*Single-Peak*") increases, reaches a maximum and then decreases. Centroid $\boldsymbol{\mu}_2$ ("*Double-Peak Deep Minimum*") has two maxima reached with very high derivatives, and its minimum is deeper than that of $\boldsymbol{\mu}_0$. Finally, centroid $\boldsymbol{\mu}_3$ ("*Monotonic*") is monotonically increasing (except for a small neighborhood of the right endpoint).

5.2. Classification of experimental data, based on cluster analysis

In this Section we apply the outcomes of the clustering-based approach for the shape classification of the four experimental data of galvanostatic Li/Li symmetric cell cycling that we have considered for parameter identification in [10]. Specifically, we calculate the Sobolev distance (31) between the experimental data and all cluster centroids and assign a given experimental dataset to the cluster yielding the smallest distance.

Fig. 6 shows the 4 representative experimental profiles $\boldsymbol{\varepsilon}_j$, $j = 1, \cdots, 4$ employed in [10]. $\boldsymbol{\varepsilon}_1$ and $\boldsymbol{\varepsilon}_2$ exhibit two peaks and a maximum, while $\boldsymbol{\varepsilon}_3$ and $\boldsymbol{\varepsilon}_4$ have an increasing monotonic trend. For each $\boldsymbol{\varepsilon}_j$, $j = 1, \cdots, 4$, we calculated the distances with respect to the 4 centroids $\boldsymbol{\mu}_k$, $k = 0, \ldots, 3$, as explained above.

We found that:

- The experimental data $\boldsymbol{\varepsilon}_1$ belongs to cluster $\mathscr{C}_2$ "*Double-Peak Deep Minimum*" as both $\boldsymbol{\varepsilon}_1$ and centroid $\boldsymbol{\mu}_2$ have two maxima, with steep first derivatives, and a deep minimum;
- The experimental data $\boldsymbol{\varepsilon}_2$ belongs to cluster $\mathscr{C}_0$ "*Double-Peak Shallow Minimum*" since its centroid $\boldsymbol{\mu}_0$ has two maxima and a shallow minimum. Experimental data $\boldsymbol{\varepsilon}_2$ also has two maxima and a minimum, but exhibits a different slope before the first maximum, leading to its assignment to cluster $\mathscr{C}_0$;
- The experimental data $\boldsymbol{\varepsilon}_3$ and $\boldsymbol{\varepsilon}_4$ belong to cluster $\mathscr{C}_3$ "*Monotonic*", since both experimental data and the nearest centroid $\boldsymbol{\mu}_3$ exhibit a clear-cut increasing monotonic trend.

Therefore, we can conclude that using the weighted Sobolev distance (31) as cost function enables a clustering that captures the global shape of the experimental data and to assign it to a specific centroid.

5.3. Partitioning of the parameter space, based on clustering of potential profiles

In this Section, we show that our Sobolev-based clustering algorithm can be employed to obtain a partition of the parameter space $\Omega_D \times \Omega_{r_{max}}$ of D and r_{max}. Of course, the K-Means algorithm operates exclusively on the simulated profiles chosen in the dataset and no direct information regarding parameters (D, r_{max}) is employed for clustering. However, due to the obvious correlation between the parameters and the profile shapes, as expected, the profiles assigned to the same cluster are found to be generated with parameters lying in the same region of the parameter space. This reflects the fact that variations in (D, r_{max}) produce smooth and continuous changes in the profile shapes, leading to a natural clustering of similar parameter values.

In this Section, we have the additional goal of using our clustering approach to validate the fitting results obtained via Deep Learning in [10] for the experimental data used therein. To this aim, we take the following steps.

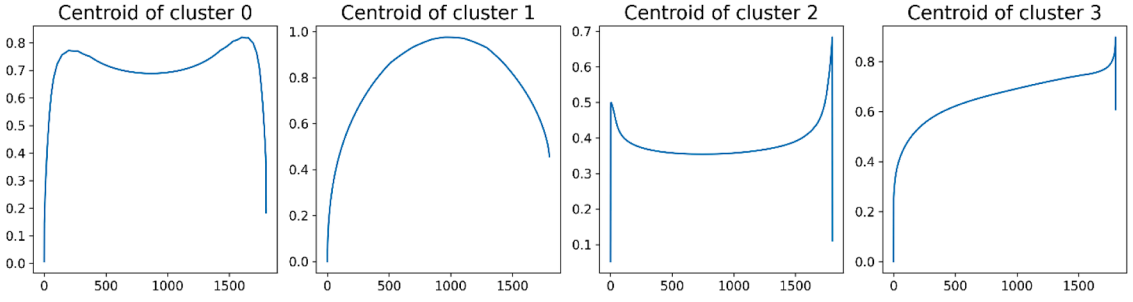


Fig. 5. Centroids profiles. From left to right, centroids $\boldsymbol{\mu}_0$, $\boldsymbol{\mu}_1$, $\boldsymbol{\mu}_2$, $\boldsymbol{\mu}_3$ of clusters $\mathscr{C}_0$ ("Double-Peak Shallow Minimum"), $\mathscr{C}_1$ ("Single-Peak"), $\mathscr{C}_2$ ("Double-Peak Deep Minimum"), $\mathscr{C}_3$ ("Monotonic"), respectively, obtained for $a = 0.1$.

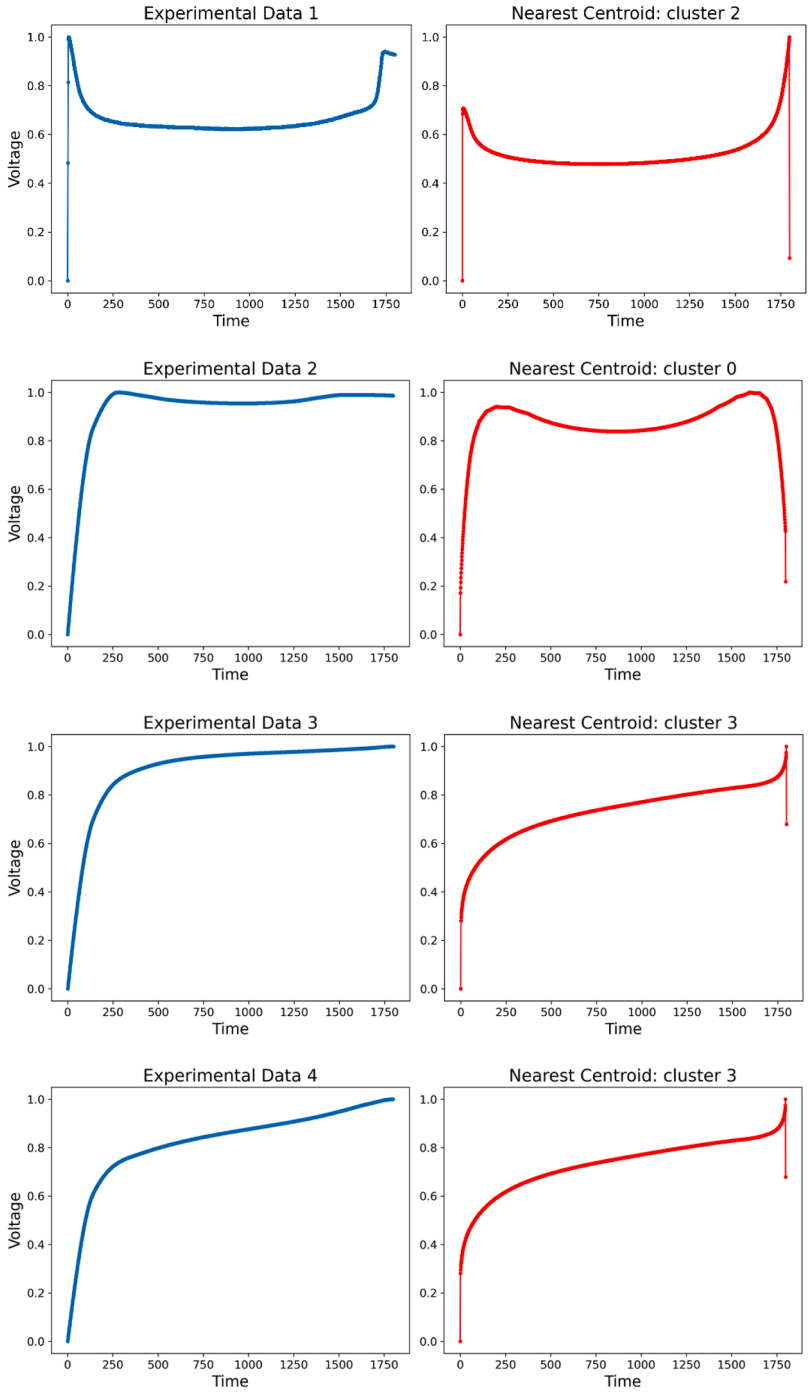


Fig. 6. Experimental data and nearest cluster's centroid. Left column: representative experimental data $\boldsymbol{\varepsilon}_j,\ j = 1, \cdots, 4$ (details in the main text). Right column: centroids $\boldsymbol{\mu}_k$ with minimum Sobolev distance Eq. (31) from the experimental data on the left, for $a = 0.1$.

First, we recompute clusters and centroids on the dataset of simulated potential $\Delta\phi$ (see Section 2, in Eq. (7)), used in [10] as training set for the parameter identification by the CNN-LSTM neural network. The new centroids found in this case, with the Sobolev-based K-Means algorithm with $a = 0.1$, are shown in Fig. 7 and exhibit evident characteristic properties. Specifically, $\boldsymbol{\mu}_0$, $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_3$ have shapes similar to those analyzed above in Section 5.2: for ease of reference, we assign to them the following denominations. $\mathscr{C}_0$: "*Double-Peak*", $\mathscr{C}_1$: "*Single-Peak*", $\mathscr{C}_3$: "*Monotonic*". Centroid $\boldsymbol{\mu}_2$ features a peak on the right-hand side and then decreases, therefore we call it $\mathscr{C}_2$: "*Right Peak*".

As a second step, we calculate the segmentation of the parameter space based on the four obtained clusters. Each of the four regions

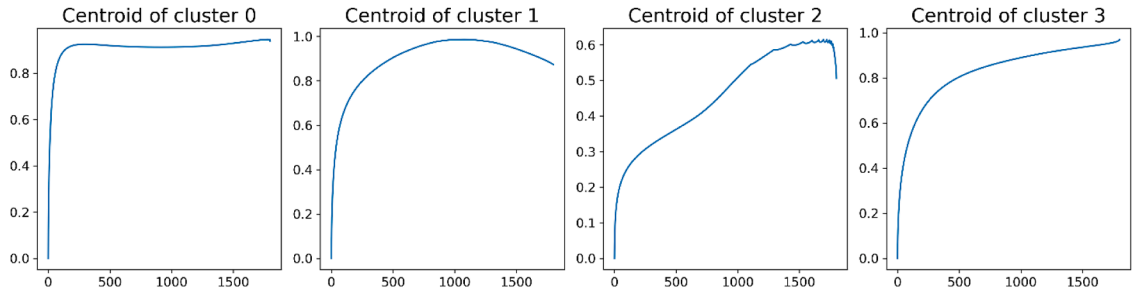


Fig. 7. Centroids profiles for dataset in Section 5.3. Centroids $\boldsymbol{\mu}_0$, $\boldsymbol{\mu}_1$, $\boldsymbol{\mu}_2$, $\boldsymbol{\mu}_3$ of clusters $\mathscr{C}_0$ ("Double-Peak"), $\mathscr{C}_1$ ("Single-Peak"), $\mathscr{C}_2$ ("Right Peak"), $\mathscr{C}_3$ ("Monotonic"), obtained by the Sobolev-based K-means for $a = 0.1$ as discussed in Section 5.3 for the dataset studied in [10] of $\boldsymbol{\Delta\phi}$ simulated profiles (see Eq. (7)).

represents the parameters corresponding to profiles within the same cluster. Given the dataset of simulated profiles $X = \{\boldsymbol{x}_j\}_{j=1}^{N}$ obtained with parameters $\boldsymbol{p}_j = (D_j, r_{max_j})$, the specific subregion $\mathscr{C}_j \subset \Omega_D \times \Omega_{r_{max}}$ to which each $\boldsymbol{p}_j$ belongs is computed by minimizing the distance from the centroids, that is

$$arg\min_{i=0,\cdots,3} H_W^2(\boldsymbol{\mu}_i, \boldsymbol{x}_j; a), \qquad j = 1, \cdots, N, \quad a = 0.1. \tag{35}$$

In Fig. 8, the four subregions we found are highlighted with different colors: cluster $\mathscr{C}_0$ ("*Double-Peak* ") is in dark blue, cluster $\mathscr{C}_1$ ("*Single-Peak*") in light blue, cluster $\mathscr{C}_2$ ("*Right Peak*") in cyan and cluster $\mathscr{C}_3$ ("*Monotonic*") is green.

As a final step, we represent in the segmented space the pairs of model parameters identified in [10] by the CNN-LSTM approach for the experimental data $\boldsymbol{\varepsilon}_j$, $j = 1, \ldots, 4$ (the same shown here in Fig. 6) which are:

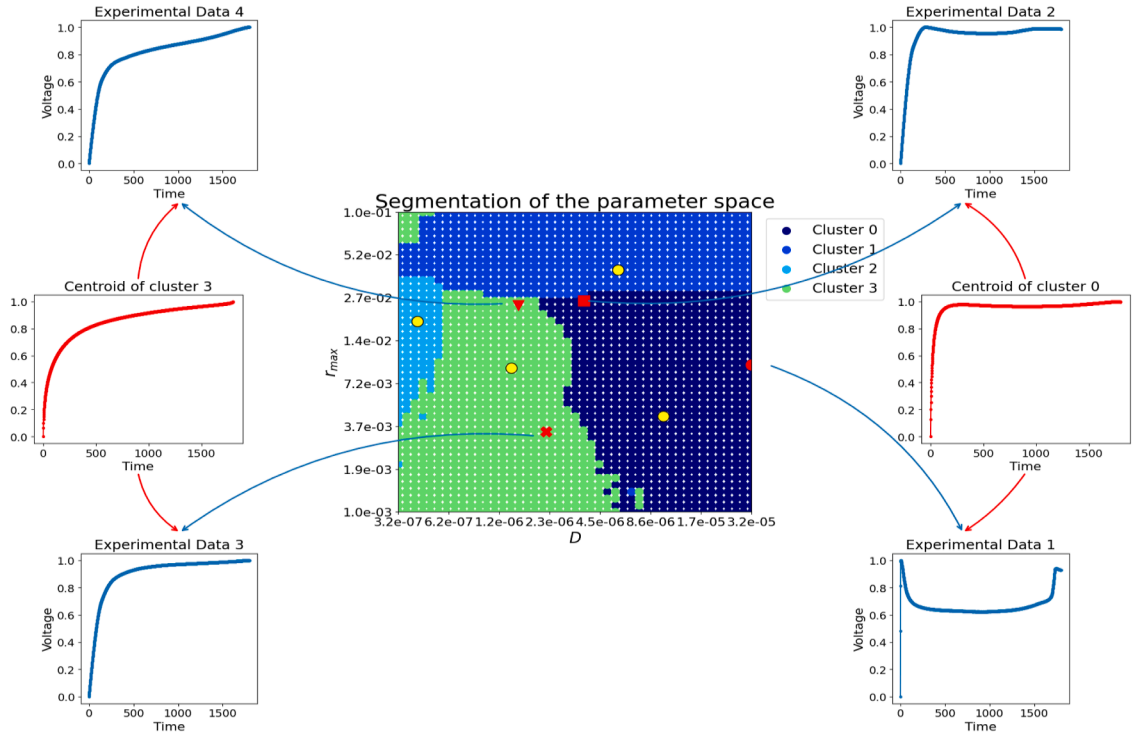


Fig. 8. Segmentation of the parameter space $\boldsymbol{\Omega_D \times \Omega_{r_{max}}}$, including parameter predictions for experimental data. Partitioning of the parameter space $(D, r_{\max})$, with axes in logarithmic scale, obtained via the weighted discrete Sobolev-distance in Eq. (35) for simulated data as described in Section 5.3. The subregions of $\Omega_D \times \Omega_{r_{\max}}$ corresponding to profiles in a given cluster are highlighted in different colors according to the legend. The red markers indicate the parameter pairs $\boldsymbol{p}_1$ (circle), $\boldsymbol{p}_2$ (square), $\boldsymbol{p}_3$ (cross), $\boldsymbol{p}_4$ (triangle) in Eq. (36) estimated by the CNN-LSTM neural network in [10] for the experimental data $\boldsymbol{\varepsilon}_j$, $j = 1, \cdots, 4$ shown in blue in the surrounding subplots. The yellow circles correspond to the parameters in Eq. (37) estimated for the cluster centroids. For completeness, we also report two subplots for the cluster centroids $\boldsymbol{\mu}_0$ of $\mathscr{C}_0$ ("Double-Peak") and $\boldsymbol{\mu}_3$ of $\mathscr{C}_3$ ("Monotonic") (in red) to highlight the right correspondence of shapes between simulated and experimental profiles.

$$\begin{aligned} \boldsymbol{p}_1 &= \left(3.2066 \cdot 10^{-5}, 9.5683 \cdot 10^{-3}\right), \ \boldsymbol{p}_2 = \left(3.5938 \cdot 10^{-6}, 2.5655 \cdot 10^{-2}\right), \\ \boldsymbol{p}_3 &= \left(2.2004 \cdot 10^{-6}, 3.4355 \cdot 10^{-3}\right), \ \ \boldsymbol{p}_4 = \left(1.5361 \cdot 10^{-6}, 2.3763 \cdot 10^{-2}\right), \end{aligned} \tag{36}$$

(See Table 4 in [10]). These coordinates are reported in Fig. 8 as points with different markers, as detailed in the caption. Moreover, for completeness, we employed the same Neural Network to estimate the parameters (D, r_{max}) associated with the four centroids $\boldsymbol{\mu}_i$, $i = 0, \ldots, 3$ shown in Fig. 7. The obtained estimated parameters are respectively:

$$\begin{aligned} \boldsymbol{p}_{\mu_0} &= \left(1.0164 \cdot 10^{-5},\ 4.3349 \cdot 10^{-3}\right), \ \ \boldsymbol{p}_{\mu_1} = \left(5.6466 \cdot 10^{-6},\ 4.1115 \cdot 10^{-2}\right) \\ \boldsymbol{p}_{\mu_2} &= \left(4.1082 \cdot 10^{-7},\ 1.8672 \cdot 10^{-2}\right), \ \boldsymbol{p}_{\mu_3} = \left(1.3997 \cdot 10^{-6},\ 9.1086 \cdot 10^{-3}\right). \end{aligned} \tag{37}$$

In Fig. 8, these values are displayed as yellow circles in the segmentation of the parameter space. We observe that each pair $\boldsymbol{p}_{\mu_i}$ lies within the corresponding cluster/subregion $\mathscr{C}_i$, as expected. This result is a cross-validation of our Deep Learning approach for parameters identification.

We note that in the segmentation of Fig. 8, three clusters meet at one triple point. A similar behavior is known in the clustering analysis literature, e.g. [62,63]. Appendix C provides a detailed discussion of the segmentation of the parameter space. In particular, we discuss the behavior of profiles near the triple point, their assignment to different clusters, and the physical interpretation of the parameter space partitioning in terms of electrochemical parameters.

As a final remark, plotting predicted fitting values for experimental data in the segmented parameter space provides a powerful and direct validation of the clustering-based segmentation, as it confirms that the parameters identified via Deep Learning in (36) indeed fall within the regions predicted by the shape-based clustering. This connection between clusters and regions of the parameter space allows us to draw physically valuable, but non-intuitive, conclusions on the impact of the parameter values on the electrical response of the cell. Moreover, parameter space partitioning can be employed for high-throughput physics-based classification of experimental data that are generated continuously during battery operation. Specifically, on the basis of the analysis proposed in this work, it is possible to estimate a plausible range of values for D and r_{max} associated with a given experimental time-series.

6. Conclusions

In this paper, we explored the use of the K-Means clustering algorithm to analyze battery voltage profiles derived from a PDE model for symmetric Li/Li cells. We showed that using a weighted discrete Sobolev-based distance, which incorporates derivatives information, is more effective than other traditional metrics, such as the Euclidean distance, in capturing the shape, peaks, valleys, and other physically diagnostic features of the data.

The clustering results on the numerically generated data demonstrated how this approach can classify experimental data, assigning each profile to a cluster. These results thus suggest an operative method for high-throughput physics-based classification of experimental data that are continuously generated during battery operation. Moreover, the clustering results revealed the possibility of segmenting the parameter space of the PDE model, pinpointing that variations in the parameters (D, r_{max}) lead to distinct potential profiles that can be clustered according to their shape. The proposed Sobolev-based K-Means approach provides a flexible framework, that can, of course, be applied to models beyond the specific electrochemical context considered here, offering a valuable tool for parameter space segmentation and parameter identification.

Author statement

M. G. Quarta: Conceptualization, Data curation, Software, Methodology, Investigation, Visualization, Formal analysis, Validation, Writing – original draft, Writing-review & editing. **I. Sgura**: Conceptualization, Methodology, Investigation, Validation, Writing–original draft, Formal analysis, Data curation, Writing-review & editing, Supervision. **B. Bozzini**: Conceptualization, Methodology, Investigation, Validation, Writing–original draft, Formal analysis, Data curation, Writing-review & editing, Supervision. **M. Frittelli**: Methodology, Software, Writing-review & editing. **R. Barreira**: Conceptualization, Software, Methodology, Investigation, Supervision.

Funding sources

M.G.Q. was supported by a scholarship financed by the Italian Ministerial Decree no. 351 of 9 April 2022, based on the NRRP—funded by the European Union – Next Generation EU – Mission 4, part of her work was carried out during the visiting period at Barreiro School of Technology of the Polytechnic Institute of Setúbal (Portugal).

I.S., M.F. and M.G.Q. acknowledge financial support under the National Recovery and Resilience Plan (PNRR), Mission 4, Component 2, Investment 1.1, by the Italian Ministry of University and Research (MIUR) and by the European Union – NextGeneration EU– PRIN2022 PNRR Project Title "BAT-MEN" (BATtery Modeling, Experiments & Numerics), Project code P20228C2PP_001, CUP F53D23010020001, Grant Assignment Decree No. 1379 01/09/2023.

For B.B. co-funding was received from ZnOrgBat project (no. 23034) under the EIT RawMaterials, part of the Horizon Europe funding scheme. This study was carried out within the MOST – Sustainable Mobility Center activities funded from the European Union Next-Generation EU (PIANO NAZIONALE DI RIPRESA E RESILIENZA (PNRR)—MISSIONE 4 COMPONENTE 2, INVESTIMENTO 1.4—D.D. 1033 17/06/2022, CN00000023).

R.B. is partially supported by funded by FCT - Fundação para a Ciência e a Tecnologia, I.P., through national funds, under the project UID/04561/2025.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

I.S., M.F. and M.G.Q. are members of the INdAM-GNCS activity group. The authors wish to thank the anonymous reviewers for helpful comments that helped improve the presentation of the paper.

Appendix A. Pseudocode

Below we report a pseudocode for the implementation of the K-Means algorithm applying the weighted Sobolev distance introduced in Section 4.2 as similarity distance.

K-Means-$H^2_W(a)$ Sobolev algorithm - pseudocode:

Input:

- $X = \{x_i\}_{i=1}^N$: dataset
- K: number of clusters
- a: weight parameter in $H^2_W(a)$
- max_iter: maximum number of iterations (default: 300)
- tol: tolerance for convergence (default: 10^{-5})
- $random\_seed$: seed for random initialization

Output:

- $\mu_k^{(j^*)}, k = 0, \dots, K-1$ centroids at the final step j^*
- $labels_i \in \{0, \dots, K-1\}, i = 1, \dots, N$ labels associating x_i to a cluster

Main:

Randomly select K curves from X as the initial centroids: $\mu_k^{(0)}, k = 0, \dots, K-1$
$j = 0$
while $(incr[k] > tol, \forall k)$ & $(iter < max\_iter)$:
    for $i = 1:N$
        for $k = 1:K$
            $distances[i,k] = H^2_W(\mu_{k-1}^{(j)}, x_i; a)$
        end
        $labels[i] = arg \min_{k=1,\dots,K} distances[i,k]$
    end
    for $k = 1:K$
        $\mu_{k-1}^{(j+1)} = \text{mean}(x_i \mid labels[i] = k-1)$
        $incr[k] = H^2_W(\mu_{k-1}^{(j+1)}, \mu_{k-1}^{(j)}; a)$
    end
$j = j + 1$
end
return $j^*, \mu_k^{(j^*)}, labels$

(continued on next page)

(*continued*)

Sobolev distance $H^2_W(\boldsymbol{u}, \boldsymbol{v}, a)$:

Compute first and second derivatives of $\boldsymbol{u}$: $\boldsymbol{u}', \boldsymbol{u}''$
Compute first and second derivatives of $\boldsymbol{v}$: $\boldsymbol{v}', \boldsymbol{v}''$
for $i = 1$: length$(\boldsymbol{u})$:
    if sign$(u'[i]) \neq$ sign$(v'[i])$
        $w_{d_1}[i] = 1$
    else $w_{d_1}[i] = a$
    end
    if sign$(u''[i]) \neq$ sign$(v''[i])$:
        $w_{d_2}[i] = 1$
    else:
        $w_{d_2}[i] = a$
    end
end
$W = diag(w_{d_1} \circ w_{d_2})$
$norm_0 = \sqrt{(\boldsymbol{u} - \boldsymbol{v})^T W (\boldsymbol{u} - \boldsymbol{v})}$
$norm_1 = \sqrt{(\boldsymbol{u}' - \boldsymbol{v}')^T W (\boldsymbol{u}' - \boldsymbol{v}')}$
$norm_2 = \sqrt{(\boldsymbol{u}'' - \boldsymbol{v}'')^T W (\boldsymbol{u}'' - \boldsymbol{v}'')}$
return $\sqrt{norm_0^2 + norm_1^2 + norm_2^2}$

Appendix B. Sensitivity analysis for the parameter a in the weighted Sobolev distance

The choice of the parameter a in the weighting matrix $W(a)$ in Eq. (32) is crucial to ensure that the discrete Sobolev distance introduced in Eq. (31) correctly identifies as minimal-distance profiles those that actually share the same global shape. In this Appendix, to this aim we present a sensitivity analysis using an *"elbow-like"* technique. The Elbow Method has been used and fully described in Section 5.1 to select the optimal number of clusters K in the K-Means technique (see e.g. [59]).

Here, we test 7 different values for the parameter a, drawn from the interval $[0.01,\ 1]$, namely $0.01,\ 0.033,\ 0.067,\ 0.1,\ 0.33,\ 0.67,\ 1$.

It is interesting to note that for $a = 1$, the weighted discrete Sobolev distance reduces to the discrete Sobolev distance H^2 in Eq. (30), since the weight matrix $W(a)$ becomes the identity matrix. For decreasing values of a, we increasingly penalize points where the slope and concavity of the curves differ. Consequently, the minimal-distance profiles tend to have similar slopes and concavities. Therefore, for each experimental dataset $\boldsymbol{\epsilon}_j,\ j = 1, \cdots, 4$ (see Section 5.2 for details on the experimental data), we have computed the profile that minimizes the weighted discrete Sobolev distance among all N simulated profiles contained in our dataset $X = \{\boldsymbol{x}_n\}_{n=1}^N$. We denote this profile as $\boldsymbol{x}_{\text{j,a}} = \arg\min_{i=1,\ldots,N} H^2_W(\boldsymbol{x}_i,\ \boldsymbol{\epsilon}_j; a)$.

In the left panel of Fig. B, we report, for each experimental profile $\boldsymbol{\epsilon}_j,\ j = 1, \cdots, 4$, the value of the weighted discrete Sobolev distance between the experimental data $\boldsymbol{\epsilon}_j$ and the corresponding minimal-distance profile $\boldsymbol{x}_{(j,a)}$, i.e., $H^2_W(\boldsymbol{x}_{(j,a)}, \boldsymbol{\epsilon}_j; a)$. We observe that as the parameter a decreases, the values of $H^2_W(a)$ also decrease. However, for all four experimental datasets considered, a clear "elbow" appears at $a = 0.1$. From left panel of Fig. B, we observe that, for values of a below 0.1, there is no significant further decrease in the $H^2_W(a)$ distance. This implies that the profiles $\boldsymbol{x}_{(j,0.1)}$ and $\boldsymbol{x}_{(j,0.01)}$ are very similar to each other. In other words, $a = 0.1$ represents an optimal trade-off: it sufficiently emphasizes the contribution of the derivatives while preserving the overall shape of the profiles. In conclusion, choosing smaller values of a would not significantly improve the shape matching.

In the right panel of Fig. B, each experimental profile $\boldsymbol{\epsilon}_j,\ j = 1, \ldots, 4$, is shown in black in each row. The simulated profiles $\boldsymbol{x}_{(j,1)}$, $\boldsymbol{x}_{(j,0.1)}$, $\boldsymbol{x}_{(j,0.01)}$ are plotted with colored dashed lines in red, blue, and green, respectively. We observe that as a varies, the shape of the minimal-distance profile changes. For $a = 1$, the profile obtained is similar to the one found using the L^2 distance (for the case of the experimental data $\boldsymbol{\epsilon}_2$, see Section 4.3 and Fig. 3). This behavior is due to the fact that in H^2 in Eq. (30), the contribution of $||\boldsymbol{u} - \boldsymbol{v}||^2_{L^2}$ dominates over the terms involving the first and second derivatives. In some cases, these derivative terms are smaller by up to two orders of magnitude, which prevents them from contributing significantly to the overall distance.

Looking in more detail at the right panel of Fig. B, one can notice that for the experimental data $\boldsymbol{\epsilon}_2$, the profile $\boldsymbol{x}_{(2,1)}$ has a maximum at the midpoint of the period, whereas the experimental data exhibit a maximum, followed by a minimum, and then by another maximum. This overall pattern is captured by the profiles $\boldsymbol{x}_{(2,0.1)}$ and $\boldsymbol{x}_{(2,0.01)}$. Similarly, for $\boldsymbol{\epsilon}_3$, which shows a monotonically increasing trend, the profile $\boldsymbol{x}_{(3,1)}$ does not reproduce the same global shape and instead increases to a maximum and then decreases. In contrast, the profiles $\boldsymbol{x}_{(3,0.1)}$ and $\boldsymbol{x}_{(3,0.01)}$ follow the same global shape of $\boldsymbol{\epsilon}_3$, except for a small neighborhood near the right endpoint.

Therefore, the choice of the parameter a is significant, as varying a can lead to profiles with different shapes. In this study, we selected the value of a indicated by the *"elbow-like"* method. This is also the value for which the minimal-distance profile achieves the best qualitative agreement in terms of distance, slope, and concavity of the voltage profiles.

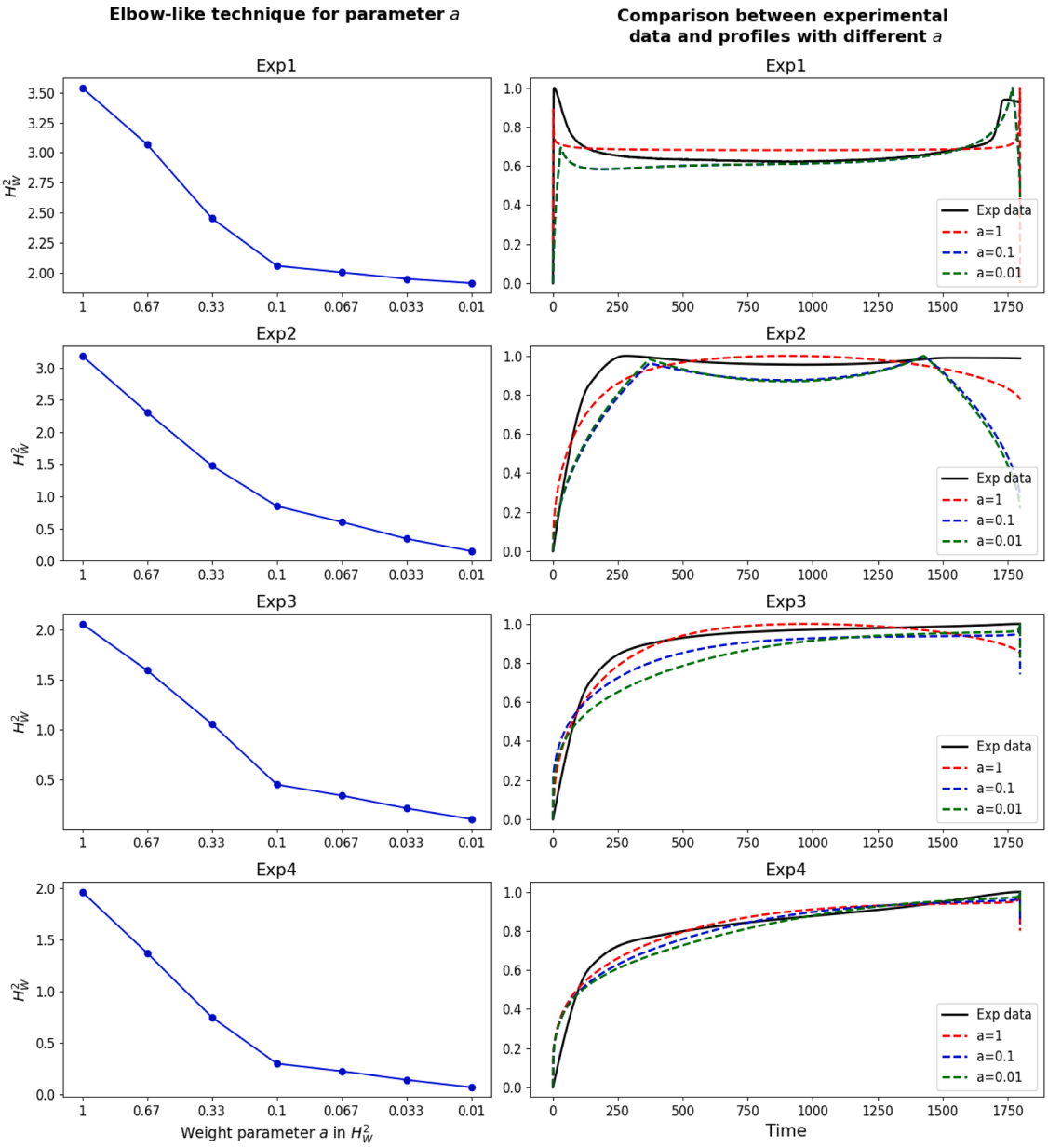


Fig. B. Elbow-like analysis of the weight parameter a in H^2_W and comparison with experimental profiles. Left plots: "elbow-like" analysis of the weight parameter a in H^2_W for the four experiments $\boldsymbol{\varepsilon}_j,\ j = 1, \cdots, 4$ for $a \in \{0.01,\ 0.033,\ 0.067,\ 0.1,\ 0.33,\ 0.67,\ 1\}$. Right plots: the normalized experimental data $\boldsymbol{\varepsilon}_j,\ j = 1, \cdots, 4$ (black solid lines) are compared with the simulated profiles (colored dashed lines) obtained for $a = 0.01,\ 0.1,\ 1$.

Appendix C. Discussion on the segmentation of the parameter space

Fig. 8, showing the segmentation of parameter space of the PDE model, and the related discussion in Section 5.3, allow us to provide a direct comparison of the clustering results with the parameters estimated in [10], using a CNN-LSTM neural network, for the experimental data considered, thereby validating the clustering-based segmentation. Nevertheless, in some subregions, the parameter space passes through 3 clusters fairly rapidly. For example, a rapid transition occurs near the triple point located around. $\boldsymbol{p}^* = (D, r_{max})$ given by $D = 2.2 \cdot 10^{-6}$ and $r_{max} = 2.6 \cdot 10^{-2}$. This transition through clusters $\mathscr{C}_0$ ("*Double-Peak*"), $\mathscr{C}_1$ ("*Single-Peak*"), and $\mathscr{C}_3$ ("*Monotonic*") is explained by the fact that, in this region, the corresponding profiles exhibit intermediate behavior, sharing characteristics of multiple clusters simultaneously. This phenomenon is well known in clustering analysis (see e.g. [62,63]) and is justified by the fact that data points near the boundaries between clusters often share features of neighboring groups, thus exhibiting gradual transitions that are not sharply separable in the feature space.

In Fig. C, we show an example of profiles belonging to clusters $\mathscr{C}_0$ ("*Double-Peak*"), $\mathscr{C}_1$ ("*Single-Peak*"), and $\mathscr{C}_3$ ("*Monotonic*") in

the area around $\boldsymbol{p}^*$. Although these profiles are assigned to three different clusters, they are quite similar to each other. However, similarities can be observed with each corresponding centroid, which lead the algorithm to assign each profile to its own specific cluster rather than to one of the others.

In particular, the panel on the left shows a profile belonging to cluster $\mathscr{C}_0$ ("*Double-Peak*"). This profile was assigned to the centroid $\boldsymbol{\mu}_0$ because it exhibits two peaks and a minimum. The central panel displays a profile assigned to cluster $\mathscr{C}_1$ ("*Single-Peak*"), whose centroid $\boldsymbol{\mu}_1$ is characterized by two peaks close to the center, which are about to converge into a maximum at the center, making it more similar to $\boldsymbol{\mu}_1$ than to the other centroids. Finally, the right panel shows an example of profile belonging to cluster $\mathscr{C}_3$ ("*Monotonic*"). This profile has a global increasing trend and a small peak on the right, indicating that the trend is becoming monotonically increasing, which results in the smallest distance from the centroid $\boldsymbol{\mu}_3$. For clarity, the title of each panel reports the corresponding values of the parameters D and r_{max} around the triple point $\boldsymbol{p}^*$.

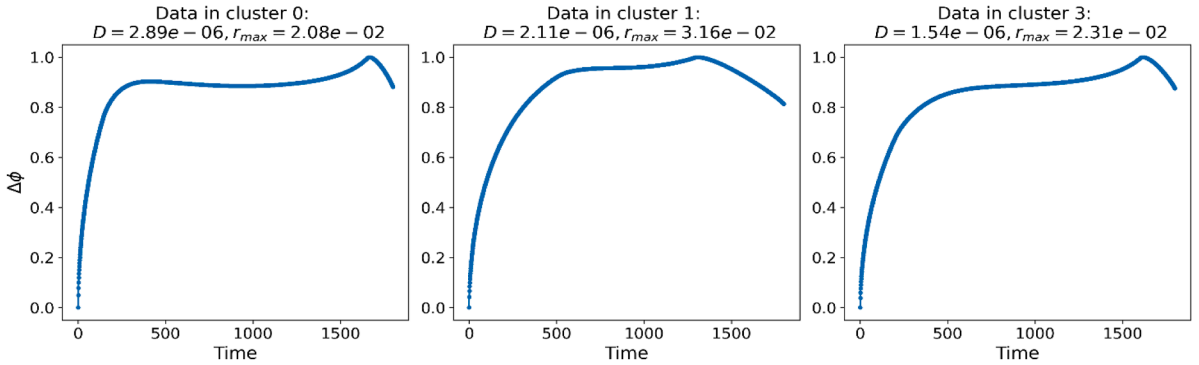


Fig. C. Profiles near the triple point $\boldsymbol{p}^* = (\mathbf{D}, \mathbf{r_{max}})$. (Left) Profile with parameters $\boldsymbol{p}_0^* = (2.9 \cdot 10^{-6}, 0.021)$ assigned to cluster $\mathscr{C}_0$ ("*Double-Peak*"), showing two peaks and a minimum. (Center) Profile with parameters $\boldsymbol{p}_1^* = (2.1 \cdot 10^{-6}, 0.032)$ assigned to cluster $\mathscr{C}_1$ ("*Single-Peak*"), characterized by two peaks close to the center, more similar to a profile with a maximum at the center. (Right) Profile with parameters $\boldsymbol{p}_3^* = (1.5 \cdot 10^{-6}, 0.023)$ assigned to cluster $\mathscr{C}_3$ ("*Monotonic*"), showing a global increasing trend and a small peak on the right. The titles in each panel reports the corresponding values of D and r_{max} around the triple point $\boldsymbol{p}^* = (2.2 \cdot 10^{-6}, 0.026)$.

The segmentation of the parameter space and the cluster positions obtained in the present work, as well as the points identified by the DL-based method of [10], have a clear physical meaning. Full details of the physical meaning of the model parameters are expounded in the seminal papers [40] and [41]. Increases in D account for enhanced mass transport in the bulk, due to changes in electrolyte chemistry accompanying cycling, such as evolution of the solid-electrolyte interphase or the build-up of reaction products that exhibit a stronger Li^+ solvation. Increases in r_{max}, instead, correspond to decreases in the surface density of nuclei developing as a result of Li plating during the charging period. A decrease in nuclei density can be due either to the nucleation-exclusion effect brought about by pre-existing nuclei (for the physico-chemical background, see, e.g. [64]) or, again, to changes in electrolyte chemistry that lead to adsorption of reaction products that inhibit Li electrodeposition.

On the basis of these electrochemical factors, the changes in potential transient shapes can be straightforwardly interpreted. Keeping D constant and increasing r_{max} (changing from cluster $\mathscr{C}_3$ *"monotonic"* to cluster $\mathscr{C}_1$ ("*Single-Peak*") the slope of the galvanostatic profile grows in correspondence with the localization of electroplating over the nuclei tips. Conversely, an increase in D occurring at constant r_{max} (moving from cluster $\mathscr{C}_3$ *"monotonic"* to cluster $\mathscr{C}_0$ *"Double-Peak"*) gives rise to a faster increase of the potential and to the formation of a minimum in correspondence with favored mass transport, compensating for process localization at the cathode tips, and increasing the electroplating rate in the electrode profile valleys. This mass-transport enhancement is effective only in the initial stages of plating (negative slope), because reagent consumption in the cathode profile valleys restores electrodeposition rate control by the cathode tips, resulting in a positive slope. Coherently, jointly increasing D and decreasing r_{max} (experimental data $\boldsymbol{\varepsilon}_1$), emphasizes minimum development. Instead, decreasing D and increasing r_{max} (from centroid $\boldsymbol{\mu}_0$ to experimental data $\boldsymbol{\varepsilon}_2$) retards the initial potential growth and puts forward the potential growth after the minimum.

Data availability

Data will be made available on request.

References

[1] J. Newman, N.P. Balsara, Electrochemical Systems, 4th ed., Wiley, 2021.
[2] A.A. Franco, M.L. Doublet, W.G. Bessler, Physical Multiscale Modeling and Numerical Simulation of Electrochemical Devices For Energy Conversion and Storage, Springer, 2016.
[3] Y. Tian, S. Zhang, Y. Wang, Mathematical Modeling for Enhanced Electrochemical Properties, 1st Edition, CRC Press, 2023.

[4] K. Chayambuka, G. Mulder, D.L. Danilov, P.H.L. Notten, Physics-based modeling of sodium-ion batteries part II. Model and validation, Electrochim. Acta 404 (2022) 139764, https://doi.org/10.1016/j.electacta.2021.139764.

[5] M. Neumann, T. Philipp, M. Häringer, G. Neusser, J.R. Binder, C. Kranz, Stochastic 3D modeling of nanostructured NVP/C active material particles for sodium-ion batteries, Batter. Supercaps 7 (4) (2024) e202300409, https://doi.org/10.1002/batt.202300409.

[6] W. Xu, D. Snihirova, M. Deng, L. Wang, B. Vaghefinazari, C. Wang, S.V. Lamaka, M.L. Zheludkevich, D. Höche, A mathematical model describing the surface evolution of Mg anode during discharge of aqueous Mg-air battery, J. Power. Sources. 542 (2022) 231745, https://doi.org/10.1016/j.jpowsour.2022.231745.

[7] S. Paruvayakode, A. OV, T. KA, F. Fasmin, Modeling and simulation of discharge behavior of Mg-H2O seawater battery, J. Power. Sources. 578 (2023) 233175, https://doi.org/10.1016/j.jpowsour.2023.233175.

[8] F. Conte, M. Giallongo, D. Kaza, G. Natrella, R. Tachibana, S. Tsuji, F. Silvestro, G. Vichi, Experimental validation of electrothermal and aging parameter identification for lithium-ion batteries, Energies 17 (10) (2024) 2269, https://doi.org/10.3390/en17102269.

[9] F. Tavola, A. Casalegno, G. Sordi, C. Rabissi, B. Bozzini, Deriving the numerical value of LIB mathematical model parameters from experiments: case of as-formed and aged NMC/LMO cathodes, J. Energy Storage 17 (2025) 116180, https://doi.org/10.1016/j.est.2025.116180.

[10] M.G. Quarta, I. Sgura, E. Emanuele, J. Strada, R. Barreira, B. Bozzini, A deep-learning approach to parameter fitting for a lithium metal battery cycling model, validated with experimental cell cycling time series, Sci. Rep. 15 (2025) 4620, https://doi.org/10.1038/s41598-025-87830-x.

[11] I. Sgura, L. Mainetti, F. Negro, M.G. Quarta, B. Bozzini, Deep-learning based parameter identification enables rationalization of battery material evolution in complex electrochemical systems, J. Comput. Sci. 66 (2023) 101900, https://doi.org/10.1016/j.jocs.2022.101900.

[12] B. Bozzini, A. Monti, I. Sgura, Model-reduction techniques for PDE models with turing type electrochemical phase formation dynamics, Appl. Eng. Sci. 8 (2021) 100074, https://doi.org/10.1016/j.apples.2021.100074.

[13] M. Wu, W. Du, F. Zhang, N. Zhao, J. Wang, L. Wang, W. Huang, Fault diagnosis method for lithium-ion battery packs in real-world electric vehicles based on K-means and the fréchet algorithm, ACS. Omega 7 (44) (2022) 40145–40162, https://doi.org/10.1021/acsomega.2c04991.

[14] W. Fang, H. Chen, F. Zhou, Fault diagnosis for cell voltage inconsistency of a battery pack in electric vehicles based on real-world driving data, Comput. Electr. Eng. 102 (2022) 208095, https://doi.org/10.1016/j.compeleceng.2022.108095.

[15] J.A. Botía, J. Vandrovcova, P. Forabosco, et al., An additional k-means clustering step improves the biological features of WGCNA gene co-expression networks, BMC. Syst. Biol. 11 (2017) 47, https://doi.org/10.1186/s12918-017-0420-6.

[16] A. Talasbek, A. Serek, M. Zhaparov, S.-M. Yoo, Y.-K. Kim, G.-H. Jeong, Personality classification experiment by applying k-means clustering, Int. J. Emerg. Technol. Learn. (iJET) 15 (16) (2020) 162–177, https://doi.org/10.3991/ijet.v15i16.15049.

[17] H.H. Yildirim, A. Akusta, Key drivers of volatility in BIST100 firms using machine learning segmentation, Int. J. Optim. Control: Theor. Appl. 15 (1) (2025) 183–201, https://doi.org/10.36922/ijocta.1707.

[18] A.N. Mat, O. İnan, M. Karakoyun, An application of the whale optimization algorithm with Levy flight strategy for clustering of medical datasets, Int. J. Optim. Control: Theor. Appl. (IJOCTA) 11 (2) (2021) 216–226, https://doi.org/10.11121/ijocta.01.2021.001091.

[19] A.Y. Clark, N. Blumenfeld, E. Lal, S. Darbari, S. Northwood, A. Wadpey, Using K-means cluster analysis and decision trees to highlight significant factors leading to homelessness, Mathematics 9 (17) (2021) 2045, https://doi.org/10.3390/math9172045.

[20] B. Gu, Health State estimation of electric vehicle power battery based on K-means algorithm, in: International Conference on Applied Intelligence and Sustainable Computing (ICAISC), 2023, pp. 1–6, https://doi.org/10.1109/ICAISC58445.2023.10199948.

[21] J. Huang, D. Chen, Y. Yang, M. Gao, Battery grouping based on improved K-means with curve fitting, in: 13th IEEE Conference on Industrial Electronics and Applications (ICIEA), 2018, pp. 1966–1971, https://doi.org/10.1109/ICIEA.2018.8398031.

[22] S.P. Lloyd, Least square quantization in PCM, IEEE Trans. Inf. Theory 28 (1957) 129–137, https://doi.org/10.1109/TIT.1982.1056489.

[23] E. Forgy, Cluster analysis of multivariate data: efficiency versus interpretability of classifications, Biometrics 21 (1965) 768–780. https://api.semanticscholar.org/CorpusID:118110564.

[24] R. Loohach, K. Garg, Effect of distance functions on K-means clustering algorithm, Int. J. Comput. Appl. 49 (6) (2012), https://doi.org/10.5120/7629-0698 (0975 –8887).

[25] R. Suwanda, Z. Syahputra, E.M. Zamzami, Analysis of euclidean distance and Manhattan distance in the K-means algorithm for variations number of centroid K, J. Phys.: Conf. Ser. 1566 (2020) 012058, https://doi.org/10.1088/1742-6596/1566/1/012058.

[26] S. Kapil, M. Chawla, Performance evaluation of K-means clustering algorithm with various distance metrics, in: IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES), 2016, pp. 1–4, https://doi.org/10.1109/ICPEICES.2016.7853264.

[27] A.M. Elsawah, B. Laala, G.K. Vishwakarma, Input-output space-filling representative points for clustering, modeling, and estimation, J. Comput. Appl. Math. 474 (1) (2026) 116928, https://doi.org/10.1016/j.cam.2025.116928.

[28] Y.S. Thakare, S.B. Bagal, Performance evaluation of K-means clustering algorithm with various distance metrics, Int. J. Comput. Appl. 110 (2015) 11, https://doi.org/10.5120/19360-0929 (0975 –8887).

[29] T.M. Ghazal, M.Z. Hussain, R.A. Said, A. Nadeem, M.K. Hasan, Ahmad M, M.A. Khan, M.T. Naseem, Performances of K-means clustering algorithm with different distance metrics, Intell. Autom. Soft Comput. 0 (2) (2021) 735–742, https://doi.org/10.32604/iasc.2021.019067.

[30] A. Singh, A. Yadav, A. Rana, K-means with three different distance metrics, Int. J. Comput. Appl. 67 (10) (2013), https://doi.org/10.5120/11430-6785 (0975 –8887).

[31] J. Paparrizos, L. Gravano, k-shape: efficient and accurate clustering of time series, in: Proceedings of the 2015 ACM SIGMOD international conference on management of data, 1855-1870, 2015, https://doi.org/10.1145/2723372.2737793.

[32] Y. Meng, J. Liang, F. Cao, Y. He, A new distance with derivative information for functional k -means, Inf. Sci. 463-464 (2018) 166–185, https://doi.org/10.1016/j.ins.2018.06.035.

[33] X. Chen, Y. Yang, Hanson–Wright inequality in Hilbert spaces with application to K-means clustering for non-euclidean data, Bernoulli 27 (1) (2021) 586–614, https://doi.org/10.3150/20-BEJ1251.

[34] D.L. Hanson, F.T. Wright, A bound on tail probabilities for quadratic forms in independent random variables, Ann. Math. Stat. 42 (1971) 1079–1083, https://doi.org/10.1214/aoms/1177693335.

[35] F. Ferraty, P. Vieu, Nonparametric Functional Data Analysis: Theory and Practice, Springer Series, 2006.

[36] F. Ieva, A.M. Paganoni, D. Pigoli, V. Vitelli, Multivariate functional clustering for the morpho logical analysis, J. R. Stat. Soc. C. Appl. Stat. 62 (2013) 401–418, https://doi.org/10.1111/j.1467-9876.2012.01062.x.

[37] T. Tarpey, K.K.J. Kinateder, Clustering functional data, J. Classif. 20 (2003) 93–114, https://doi.org/10.1007/s00357-003-0007-3.

[38] E. Brunenberg, R. Duits, Bt. Romeny, B. Platel, A Sobolev norm based distance measure for HARDI clustering: a feasibility study on phantom and real data, Med. Image Comput. Assist. Interv. 13 (2010) 175–182, https://doi.org/10.1007/978-3-642-15705-9_22.

[39] R. Ehsani, F. Drabløs, Robust distance measures for kNN classification of cancer data, Cancer Inf. 19 (2020) 1–9, https://doi.org/10.1177/1176935120965542.

[40] F. Rossi, L. Mancini, I. Sgura, M. Boniardi, A. Casaroli, A.P. Kao, B. Bozzini, Insight into the cycling behaviour of metal anodes, enabled by X-ray tomography and mathematical modelling, ChemElectroChem 9 (9) (2022) e202101537, https://doi.org/10.1002/celc.202101537.

[41] B. Bozzini, E. Emanuele, J. Strada, I. Sgura, Mathematical modelling and parameter classification enable understanding of dynamic shape-change issues adversely affecting high energy-density battery metal anodes, Appl. Eng. Sci. 13 (2023) 100125, https://doi.org/10.1016/j.apples.2022.100125.

[42] M. Raissi, P. Perdikaris, G.E. Karniadakis, Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, J. Comput. Phys. 378 (2019) 686–707, https://doi.org/10.1016/j.jcp.2018.10.045.

[43] F. Colace, D. Conte, G. Pagano, B. Paternoster, C. Valentino, Physics-informed neural networks for a Lithium-ion batteries model: a case of study, Adv. Comput. Sci. Eng. (ACSE) 2 (4) (2024) p1, https://doi.org/10.3934/acse.2024018.

[44] C. Valentino, G. Pagano, D. Conte, B. Paternoster, F. Colace, M. Casillo, Step-by-step time discrete physics-informed Neural networks with application to a sustainability PDE model, Math. Comput. Simul. 230 (2025) 541–558, https://doi.org/10.1016/j.matcom.2024.10.043.

[45] H. Qureshi, AI-driven analysis of buoyancy-convective flow of ternary-hybrid nanofluid in a porous medium over stretching cylinder, Nonlinear. Dyn. 113 (2025) 28907–28924, https://doi.org/10.1007/s11071-025-11620-3.
[46] H. Qureshi, A.I. Alakhras, Ai-driven multilayer modeling of Tetra-Hybrid Casson nanofluid flow with thermal radiation: implications for solar energy and energy conversion, Results Phys. 79 (2025) 108505, https://doi.org/10.1016/j.rinp.2025.108505.
[47] H. Qureshi, M. Zubair, S.A. Altmeyer, Machine learning investigation of ternary-hybrid radiative nanofluid over stretching and porous sheet, Nanomaterials 15 (2025) 1525, https://doi.org/10.3390/nano15191525.
[48] H. Qureshi, S. Altmeyer, M. Zubair, Machine learning investigation of Marangoni convection in hybrid nanofluids with Darcy-Forchheimer, Sci. Rep. 15 (1) (2025) 39657, https://doi.org/10.1038/s41598-025-23362-8.
[49] H. Qureshi, A. Latif, T. Athar, A. Raheem, T. Muhammad, Artificial neural network driven investigation of thermal exchange through hybrid nanofluid of polymer/CNT across parallel sheets, Case Stud. Therm. Eng. 74 (2025) 106903, https://doi.org/10.1016/j.csite.2025.106903.
[50] K.N. Wood, E. Kazyak, A.F. Chadwick, K.-H. Chen, J.-G. Zhang, K. Thornton, N.P. Dasgupta, Dendrites and pits: untangling the complex behavior of lithium metal anodes through operando video microscopy, ACS Cent. Sci. 2 (2016), https://doi.org/10.1021/acscentsci.6b00260.
[51] B. Bozzini, A simple numerical procedure for the simulation of "lifelike" linear-sweep voltammograms, J. Chem. Educ. 77 (1) (2000) 100, https://doi.org/10.1021/ed077p100.
[52] F. Evirgen, S. Uçar, N. Özdemir, Mathematical analysis and optimal control of a Caputo fractional diabetes system with parameter identification, J. Comput. Appl. Math. 477 (2026) 117151, https://doi.org/10.1016/j.cam.2025.117151.
[53] I. Sgura, A.S. Lawless, B. Bozzini, Parameter estimation for a morphochemical reaction-diffusion model of electrochemical pattern formation, Inverse Probl. Sci. Eng. 27 (5) (2018) 618–647, https://doi.org/10.1080/17415977.2018.1490278.
[54] J.L. Mead, Discontinuous parameter estimates with least squares estimators, Appl. Math. Comput. 219 (10) (2013) 5210–5223, https://doi.org/10.1016/j.amc.2012.11.067.
[55] J. MacQueen, Some methods for classification and analysis of multivariate observations, Berkeley Symp. Math. Stat. Prob. (1967) 281–297. http://projecteuclid.org/euclid.bsmsp/1200512992.
[56] C.M. Bishop, Pattern Recognition and Machine Learning, Information Science and Statistics, Springer, 2006.
[57] A.M. Ikotun, A.E. Ezugwu, L. Abualigah, B. Abuhaija, J. Heming, K-means clustering algorithms: a comprehensive review, variants analysis, and advances in the era of big data, Inf. Sci. 622 (2023) 178–210, https://doi.org/10.1016/j.ins.2022.11.139.
[58] M. Ahmed, R. Seraj, S.M.S. Islam, The K-means algorithm: a comprehensive survey and performance evaluation, Electron. (Switz.) 9 (8) (2020) 1–12, https://doi.org/10.3390/electronics9081295, 1295.
[59] M.A. Syakur, B.K. Khotimah, E.M.S. Rochman, B.D. Satoto, Integration K-means clustering method and elbow method for identification of the best customer profile cluster, IOP Conf. Ser.: Mater. Sci. Eng. 336 (2018) 0120017, https://doi.org/10.1088/1757-899X/336/1/012017.
[60] H. Brezis, Functional Analysis, Sobolev Spaces and Partial Differential Equations, Springer, 2011.
[61] L.C. Evans, Partial Differential Equations, 2nd Edition, 19, American Mathematical Society, 2010.
[62] J.C. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms, Springer, 1981.
[63] S.E. Hashemi, F. Gholian-Jouybari, M. Hajiaghaei-Keshteli, A fuzzy C-means algorithm for optimizing data clustering, Expert. Syst. Appl. 227 (2023) 120377, https://doi.org/10.1016/j.eswa.2023.120377.
[64] B. Bozzini, D. Lacitignola, C. Mele, I. Sgura, Morphogenesis in metal electrodeposition, Note Mat. 32 (2012) 7–46, https://doi.org/10.1285/i15900932v32n1p7.