

MINN: Learning the dynamics of differential-algebraic equations and application to battery modeling

Yicun Huang, Changfu Zou, *Senior Member, IEEE*, Yang Li, *Senior Member, IEEE*,
and Torsten Wik, *Member, IEEE*

Abstract—The concept of integrating physics-based and data-driven approaches has become popular for modeling sustainable energy systems. However, the existing literature mainly focuses on the data-driven surrogates generated to replace physics-based models. These models often trade accuracy for speed but lack the generalizability, adaptability, and interpretability inherent in physics-based models, which are often indispensable in modeling real-world dynamic systems for optimization and control purposes. We propose a novel machine learning architecture, termed model-integrated neural networks (MINN), that can learn the physics-based dynamics of general autonomous or non-autonomous systems consisting of partial differential-algebraic equations (PDAEs). The obtained architecture systematically solves an unsettled research problem in control-oriented modeling, i.e., how to obtain optimally simplified models that are physically insightful, numerically accurate, and computationally tractable simultaneously. We apply the proposed neural network architecture to model the electrochemical dynamics of lithium-ion batteries and show that MINN is extremely data-efficient to train while being sufficiently generalizable to previously unseen input data, owing to its underlying physical invariants. The MINN battery model has an accuracy comparable to the first principle-based model in predicting both the system outputs and any locally distributed electrochemical behaviors but achieves two orders of magnitude reduction in the solution time.

Index Terms—Lithium-ion batteries, battery management systems, battery modeling, model simplification, physics-informed machine learning, model-integrated neural networks.

1 INTRODUCTION

RAPID advances in electromobility have positioned battery as a key player in the transition towards a more sustainable future, with its impact on carbon neutrality continuing to gain momentum. While most battery research has been mainly focused on searching for novel materials [1, 2], the established battery chain and its circular economy have been dominated by lithium-ion batteries (LIBs) foreseen to prevail due to their proven long-term stability, cost-effective production and recycling. Consequently, the pressure of electromobility has been put on the optimization of LIBs in the foreseeable future, from cell-level chemistry, structure, and manufacturing process to system-level solutions for improved safety, reliability, performance, and lifetime. One key limiting factor in unleashing the full potential of LIBs for electromobility is the current battery management systems (BMS), which limit usage by imposing more or less fixed constraints on battery cell external measurements. The next-generation BMS should enable accurate monitoring and optimal control of dynamical local behaviors distributed inside each cell for real-time optimized utilization of the battery systems.

A battery is a compact, multiphysics system with multiple state variables, domains, material phases and physical parameters over disparate time- and length scales. The

current strategies for probing battery internal states involve battery modeling based on equivalent circuits, which at their best, are able to mimic the battery electric behaviors under specific conditions [3]. More sophisticated electrochemical models for locally distributed internal states, with their minimal assumptions and greater flexibility, offer a distinct advantage over other battery models by providing higher accuracy under a broader range of usage conditions. This makes them an ideal choice for all-purpose and comprehensive battery modeling. Although electrochemical models have been driving a wealth of LIB research in system identification [4, 5], state estimation [6, 7], fault and aging predictions [8, 9], and optimal control [10, 11], their typical end-user applications, such as smartphones and laptop computers, still require considerable computational power due to the highly nonlinear and stiff PDAEs, let alone the upscaling to pack-level and vehicle fleet-level battery applications. Despite numerous offline implementations employing state-of-the-art numerical techniques [12, 13, 14, 15, 16], it is still infeasible to consider advanced electrochemical models for on-board battery management with current hardware.

The fundamental challenges in solving PDAEs for control-oriented applications have been commonly addressed by reduced-order modeling. These reduced-order models (ROMs) attempt to lower the computational complexity by either exploring the mathematical structure of the governing equations or simplifying the physics of the original model. For example, the single particle model [18] features cell-interior variables derived from volume-averaged active materials and uniform molar flux, which result in sys-

This work was supported by Marie Skłodowska-Curie Actions Postdoctoral Fellowships under the Horizon Europe programme (Grant No. 101068764).

All the authors are with the Department of Electrical Engineering, Chalmers University of Technology, Gothenburg, 412 96, Sweden. Emails: yicun@chalmers.se; changfu.zou@chalmers.se; yang.li@chalmers.se; tw@chalmers.se. Corresponding authors: Changfu Zou and Torsten Wik.

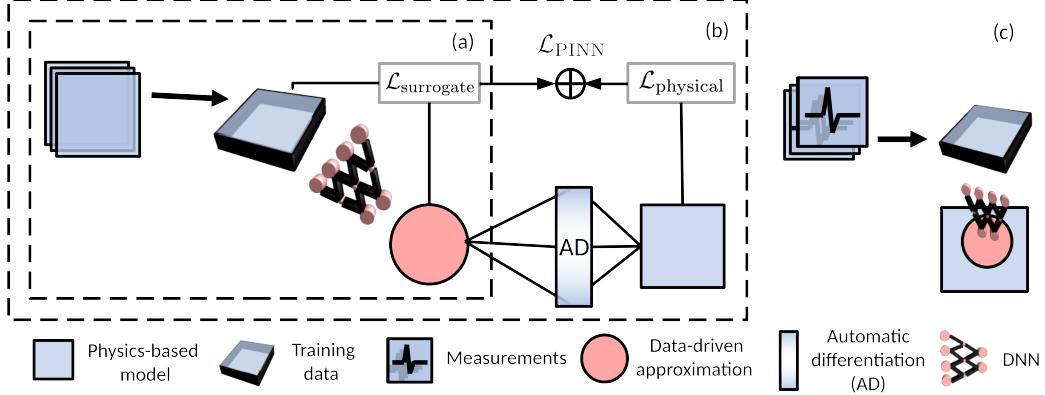


Fig. 1: Existing physics-based integration strategies for the blending of neural networks and physics-based models in order to retain their individual merits. (a) A data-driven surrogate model using supervised learning requires relevant and representative training data generated by snapshots of the physics-based model solutions. (b) A surrogate model regularised physical constraints within the PINN framework, of which the PINN loss, $\mathcal{L}_{\text{PINN}}$, is composed of the loss due to model-data inconsistency, $\mathcal{L}_{\text{surrogate}}$, and the loss owing to physical constraints, $\mathcal{L}_{\text{physical}}$. (c) The PINN workflow for inverse problems used to estimate physical parameters as part of parametric PDEs [17].

tems of partial differential equations (PDEs). Other ROMs, such as the physics-based equivalent circuit model [19, 20], also result in a simplified system with fewer assumptions. On a high level, these model reduction strategies can be seen as an attempt to replace the original PDAE formulation with a simplified system, resulting in less computational complexity and fewer parameters but compromising accuracy under performance-limiting conditions.

To circumvent the stiffness issues associated with the first-principle electrochemical models, data-driven approaches have emerged as powerful tools for identifying high-dimensional patterns in battery data. For example, the cycle life of batteries can be accurately predicted using machine learning methods given enough relevant data [21, 22, 23], and fast estimations of the terminal voltage and state of charge (SOC) can be achieved by using recurrent neural networks (RNN) [24]. Nevertheless, training a purely data-driven model to predict internal states of battery cells, such as the electrolyte concentration, local temperature and lithium plating potential, is obscured due to the lack of measurements. However, these internal states are highly important for battery safety, health and performance optimization purposes. Without tracking them, lithium dendrites may ultimately cause internal short-circuits to grow rapidly under conditions of electrolyte depletion [25], high-temperature gradient [26] and negative plating potential [27]. Notably, as illustrated in Fig. 1a, this class of data-driven surrogate models relies on generic neural networks that are agnostic to the underlying dynamics of the battery. Additionally, given a large number of trainable parameters in these neural network models, representative datasets are essential for training to minimize out-of-sample errors, which are often generated by physics-based models.

In contrast to reduced-order modeling and data-driven approximation, physics-constrained learning offers a distinctive approach to integrate neural networks with physics-based modeling. For instance, neural ordinary differential equations (neural ODEs) leverage the approximation capabilities of neural networks to model dynamic states de-

scribed by differential equations [28]. Neural ODEs enable a continuous-time formulation of deep learning models, enhancing their flexibility and interpretability in handling dynamic systems. Despite their success in predicting the state of health in batteries [29], the dynamic equations approximated by neural ODEs remain essentially a black box. This limitation prevents the direct integration of domain-specific knowledge, such as current and energy conservation laws or constitutive equations of battery systems, into the learning process. Consequently, it undermines their generalization capability and robustness in predicting the evolution of system-specific states.

To blend neural networks with governing physical laws directly, the physics-informed neural networks (PINNs) have been introduced to approximate solutions to PDEs by incorporating physical constraints into the loss function [30]. As illustrated in Fig. 1b, the PINN framework utilizes automatic differentiation (AD) to compute the residual of PDEs in an unsupervised manner. This process results in a physical constraint loss term, $\mathcal{L}_{\text{physical}}$, which is added to the supervised loss, $\mathcal{L}_{\text{surrogate}}$. Additionally, the PINN framework can be employed to estimate the parameters of a physics-based model as schematized in Fig. 1c, even with a limited experimental dataset. Furthermore, PINNs have proven effective even for stiff systems [31], and various software tools have been developed to automate PINN implementation, making it accessible for different physical systems formulated as initial value problems [32, 33, 34]. However, PINNs are known to encounter difficulties with complex problems [35] and are inherently unable to handle non-autonomous systems, or systems with different initial conditions. Specifically, the external control inputs or disturbances of a non-autonomous system, or different initializations of a system's states, will alter the system's dynamical behaviors. Thus, employing PINNs to approximate these classes of dynamic systems, such as battery management and control systems, is impractical.

To bridge the research gap, this work proposes a model-integrated deep learning framework, termed model-

integrated neural networks (MINN), designed to leverage the approximation power of neural networks, and the physical insight and numerical machinery, from those of a physics-based model. MINN is shown to be extremely data-efficient to train and can extrapolate beyond the operating conditions considered in the training data. Furthermore, it retains the physical significance of hidden states and model parameters that can be used directly for system identification, model adaptation, state estimation, and model-based control of LIB. The generality of the proposed framework allows for the easy adoption of other dynamic systems.

2 MODEL-INTEGRATED NEURAL NETWORKS

Different from the two ways of coupling data-driven and physics-based approaches introduced by the PINN framework, this work presents a new hybrid approach leveraging sequence-to-sequence learning to overcome the limitations of PINN. Our approach is designed for a class of generic dynamic systems with control inputs, or different initial conditions. Instead of learning the entire space-time solution using PINN, the main objective of MINN is to derive an explicit function that is implicitly embedded in physics-based models, in order to accelerate the solution process. To achieve this, physics-based equations are integrated into a neural network architecture directly. With the key idea schematized in Fig. 2, MINN is formulated and explained step-by-step in this section.

General multi-timescale dynamic systems can be formulated as coupled differential-algebraic equations (DAEs), resulting from the spatial discretization of the original PDAEs

$$\dot{h}_d(t) = f(t, h_d, h_z, u), \quad (1)$$

$$y(t) = Y(t, h_d, h_z, u), \quad (2)$$

$$0 = g(t, h_d, h_z, u). \quad (3)$$

The above DAE system features differential states h_d , algebraic variables h_z , and a time-varying control input u . y is the output of interest, and Y is the function to compute the output from the states and the input. The origin of the algebraic equation (3) is threefold, i.e., it can stem from the boundary conditions, the singular perturbation of the original PDAEs, or conservation laws naturally arising from the physical problem. In some special cases, explicit solutions for (3) exist for h_z , which makes it replaceable by a function $G(t, h_d, u)$ in (1) and (2). However, in most cases, h_z does not have a fixed-form solution in terms of t , h_d and u . In such cases, more computationally involved DAE solvers must be used.

Due to the high computational cost of solving DAE systems, the solution process, or the model itself, must be simplified to suit many model-based applications. If the end result of training PINNs is a fast solution and of numerical methods, a slow, coarse-grained solution, training MINN generates a simplified dynamic model, as shown in the schematics of MINN in Fig. 2a. To this end, we parameterize an explicit function G_{NN} within the recurrent unit by θ . This function can be any nonlinear approximator, e.g. neural networks, as illustrated by the “data-driven approximation” module in Fig. 2b. The colors pink and blue in Fig. 2

represent approximations and physics-based equations, respectively. For instance, the input of the neural network within the “data-driven approximation” module consists of functions of time t , differential states h_d , and the control input u , whereas the output h_z^* , which is the approximated algebraic variable, is fed to the g function of the “physics-based equations” module, yielding the approximated conserved quantities \bar{g} . In this manner, the underlying physical invariants and domain-specific priors are encoded in the recurrent unit.

Mathematically, the hidden states of the proposed MINN framework at time step $k+1$ are updated via the “numerical solver” module using

$$h_d^{k+1} = h_d^k + f(t^k, h_d^k, h_z^{*,k}, u^k) \cdot \delta t^k, \quad (4)$$

$$h_z^{*,k} = G_{\text{NN}}(t^k, h_d^k, u^k; \theta), \quad (5)$$

where (4) is a discretized form of the continuous-time dynamic equation (1), and the algebraic variables h_z^* in (5) are approximations to the roots of (3). The function G_{NN} offers a shortcut to solving the implicit algebraic equations of the DAE system. Additionally, the time step δt taken can be adaptively adjusted by the ODE solver, and the form of (4) can vary if implicit or multi-stage schemes are used. The system output y and the conserved quantities of the battery system \bar{g} are computed by

$$y^k = Y(t^k, h_d^k, h_z^{*,k}, u^k), \quad (6)$$

$$\bar{g}^k = g(t^k, h_d^k, h_z^{*,k}, u^k), \quad (7)$$

where \bar{g} is an approximation of g , computed at each time step k . \bar{g} does not strictly vanish due to the approximation error, determined by trainable parameters θ . In addition, the MINN framework integrates the differential equations of the DAE system into the neural network architecture through the physics-based hidden recurrent units, as shown in Fig. 2b.

The search for an optimally simplified model (4)–(6) is cast as the following nonlinear optimization problem

$$\arg \min_{\theta} \mathcal{L}_{\text{MINN}}(\theta; \lambda) = \mathcal{L}_y(\theta) + \lambda \mathcal{L}_g(\theta), \quad (8)$$

in which the physics-based recurrent units are used to formulate a physics-constrained loss function with \mathcal{L}_y and \mathcal{L}_g given by

$$\mathcal{L}_y(\theta) = \sum_{k=0}^K \left(Y(t^k, h_d^k, G_{\text{NN}}(t^k, h_d^k, u^k), u^k) - \hat{y}^k \right)^2 \delta t^k, \quad (9)$$

$$\mathcal{L}_g(\theta) = \sum_{k=0}^K \left| g(t^k, h_d^k, G_{\text{NN}}(t^k, h_d^k, u^k), u^k) \right| \delta t^k, \quad (10)$$

where \hat{y}^k are measured outputs, and K is the number of samples in the time series. In the training of MINN, we seek parameters θ by solving for h_d for a given profile u^k . The loss function comprises a term \mathcal{L}_g to quantify physical inconsistency, which due to conservation laws, is a function of the algebraic variables, plus a loss associated with the (measurable) output of the dynamic system. Unlike methods that require an accurate estimate of time derivatives of the states, such as SINDy [36], the loss function of MINN is a sum of model-measurement mismatch and \mathcal{L}_g . They are

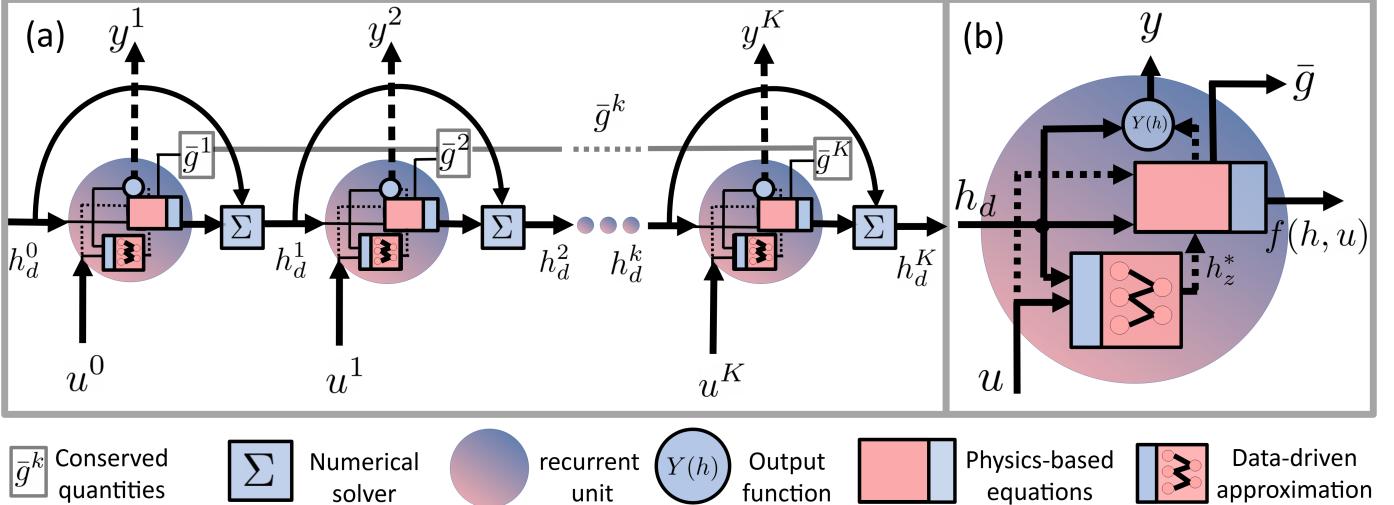


Fig. 2: The proposed MINN architecture for dynamic systems. (a) An iterative update of the hidden states h_d^k , output y^k and conserved quantities \bar{g}^k , is controlled by input $u^k = u(t^k)$ at time t^k through physics-based hidden units. This update is handled by the time integration via the numerical solver. (b) The design of a physics-based recurrent unit contains physics-based equations, a deep learning-enabled approximation (5) and an output function Y .

obtained by integrating the system via an ODE solver, and this approach is known to be more noise-tolerant [37]. During training, the parameterized loss function $\mathcal{L}_{\text{MINN}}(\theta)$ is minimized via a gradient-based optimizer, and the Lagrange multiplier λ is updated iteratively by the steepest ascent. Through (8)–(10), it can be seen that the training of G_{NN} is dependent on the time trajectory of the dynamic system (4)–(7). This enables the deep integration of neural networks and physics-based models. Once the MINN model is obtained, an array of ODE solvers can be readily employed to reproduce the system's state dynamics controlled by an arbitrary profile u^k .

The architecture of MINN integrates elements from RNN and residual neural networks. RNN, known for its modular and flexible design, is particularly suited for sequence-to-sequence applications. As a bi-directional network, RNN makes a good candidate for modeling dynamic systems with time-stepping of states governed by the system dynamics and control input. The design of RNN's recurrent units facilitates the integration of physics-based equations, allowing for modifications to neuron connectivity and functionality. Despite its advantages, the baseline RNN architecture is hindered by limitations such as short-term memory and the vanishing gradient problem. These issues can be partially mitigated by incorporating mechanisms like gated recurrent units [38], echo state networks [39] and the long short-term memory [40]. Nonetheless, these enhancements do not fully resolve the challenges related to data efficiency, and are still susceptible to overfitting and poor extrapolative capabilities [41]. Therefore, incorporating domain-specific knowledge derived from well-understood systems specified by differential equations is crucial for developing neural network architectures that embed physics-based prior knowledge effectively.

The MINN model is physically informed of the dynamics of the hidden states h_d by adding skip (residual) connections and input to model the dynamic system controlled by

u^k . This approach enables the integration of a numerical solver that optimizes the time stepping. During training, the learnable parameters θ are updated by backpropagating gradients through the solver. Physically, the input u^k and the hidden states h_d^k at time step k are fed into the recurrent unit, resulting in the approximation of the (algebraic) hidden variables h_z^k used in the calculation of a vector-valued function f representing the time derivative. This facilitates the explicit integration of physics-based equations and meaningful states into the MINN architecture. Different from neural ODE [28], the model-integrated recurrent units allow for the incorporation of control input, thereby enhancing interpretability and extrapolation capabilities. The recurrent units transform hidden states in a sequence-to-sequence manner, generating a time series of conserved quantities, \bar{g}^k , for the physics-constrained loss function.

3 APPLICATION TO BATTERY MODELING

As stated before, lithium-ion batteries represent a prevalent technology in electromobility and sustainable energy storage that are important forces in the fight against climate change. In this respect, BMS plays a crucial role in battery safety, reliability, sustainability, and dynamic performance. The central thesis for enabling advanced BMS is to develop a battery model that simultaneously preserves physical insights, accuracy and computational efficiency. To this end, the proposed MINN architecture is applied to the modeling of lithium-ion batteries, where each hidden state of the MINN model is assigned to an electrochemical state of the first principle battery model. Depending on the application, the control u can be the current I for a battery system. The output may include the terminal voltage, SOC and lithium plating potential if a reference electrode is used. The training data generation for MINN involves only the output that can be measured using, e.g., a three-electrode cell setup. Here, instead of learning blindly from the training data as illustrated by the hybrid approach shown in Fig. 1c,

we integrate prior knowledge, i.e., the equations from the PDAE system, into the neural network architecture. Fig. 3 shows the realization of the physics-based equations in the recurrent unit of MINN, for which the circuitry is based on Newman's P2D model [42, 43, 44].

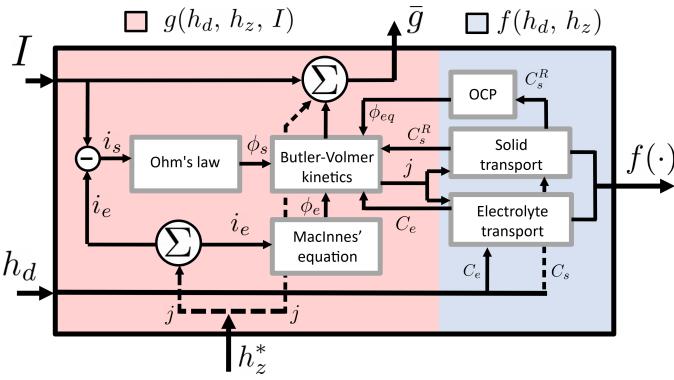


Fig. 3: The realization of the “physics-based equations” module in the physics-based recurrent unit of Fig. 2. For battery systems, the control input u is the applied current, i.e. $u^k = I(t^k)$, and the differential state and algebraic variables are represented by $h_d = [C_s, C_e]^T$ and $h_z = j$, respectively. The g -component evaluates the conservation laws at each time step k with the approximated algebraic variable h_z^* , while the f -component evaluates the time derivative of the differential states \dot{h}_d . The two components in the circuitry feature P2D equations, e.g., the open circuit potential (OCP) is a fitted function that takes in the solid concentration at the active material surface and outputs the equilibrium potential ϕ_{eq} .

3.1 Physics-based Battery Model

The most widely used model for Li-ion battery electrochemistry is the celebrated pseudo-two-dimensional (P2D) model, after the paradigm coined by Newman and co-workers [42, 43, 44]. The P2D model consists of a set of coupled PDAEs describing the lithium ion dynamics in solid and liquid phases based on porous electrode theory and concentrated solution theory. Although it is a macroscopic model, the model formulations span over multiple length scales. Starting from the pore-scale dynamics within the active particles, the lithium-ion concentration in the solid phase C_s is conserved given a thermodynamic driving force, i.e., the chemical potential μ , according to

$$\frac{\partial C_s}{\partial t} = \nabla \cdot \left(\frac{D_s C_s}{k_B T} \nabla \mu \right), \quad (11)$$

where D_s , k_B and T are the solid diffusion coefficient, Boltzmann constant and temperature, respectively. The driving force, also termed the chemical potential of the system, adopts the Nernst relation assuming a concentrated solution, i.e., with only entropic contribution $\mu = k_B T \ln C_s$. Consequently, (11) reduces to Fick's diffusion equation, which in spherical coordinates has the form

$$\frac{\partial C_s}{\partial t} = \frac{D_s}{r^s} \frac{\partial}{\partial r} \left(r^2 \frac{\partial C_s}{\partial r} \right), \quad (12)$$

where r represents the radial (pseudo) dimension. At the center of the particle ($r = 0$), there is a no-flux boundary condition. Imposed by charge transfer, the derivative at the particle surface ($r = R$) gives the interfacial flux j , i.e.,

$$j = D_s \frac{\partial C_s}{\partial r} \Big|_{r=R} = -\frac{2i_0}{\mathcal{F}} \sinh \left(\frac{\mathcal{F}}{2RT} \eta \right). \quad (13)$$

Assuming symmetric Butler-Volmer kinetics, (13) describes the local reaction molar flux j as a function of (symmetric) exchange current density

$$i_0 = \mathcal{F} k_0 \sqrt{(C_s^{\max} - C_s^R) \cdot C_s^R \cdot C_e}, \quad (14)$$

and the local overpotential η in the electrode thickness dimension (x), i.e.

$$\eta = \phi_s - \phi_e - \phi_{eq}(C_s^R) - R_{SEI} \cdot j. \quad (15)$$

In the above equations, k_0 , C_s^{\max} , \mathcal{F} , R and R_{SEI} stand for reaction rate constant, maximum solid concentration, universal gas constant, Faraday constant and resistance of the solid electrolyte interface (SEI). ϕ_{eq} denotes the equilibrium potential (relative to lithium metal Li^0), determined by the open circuit potential of the materials, which is a function of the concentration at the particle surface $C_s^R = C_s|_{r=R}$. ϕ_s , ϕ_e and C_e are the electrical potential, ionic potential and electrolyte concentration fields treated as superimposed continua, along with the currents in the solid and electrolyte (i_s and i_e). They are determined by ϕ_s , ϕ_e and C_e according to Ohm's law and the modified Ohm's law, known as MacInnes' equation

$$\frac{i_s}{\sigma_{\text{eff}}} = -\frac{\partial \phi_s}{\partial x}, \quad (16)$$

$$\frac{i_e}{\kappa_{\text{eff}}(C_e)} = -\frac{\partial \phi_e}{\partial x} + A \frac{\partial \ln C_e}{\partial x}, \quad (17)$$

where σ_{eff} and κ_{eff} are effective ionic conductivities in the solid and electrolyte, respectively, and $A = 2(1 - t_+^0)\mathcal{R}T/\mathcal{F}$ term in the (17) accounts for the diffusion overpotential induced by an electrolyte concentration gradient. The parallel currents i_s and i_e are constrained by Kirchhoff's law, i.e., $i_s + i_e = u(t)$ where $u(t)$ is the applied current. Fig. 4 illustrates the various fields, domains and boundaries characteristic of the electrochemistry of a Li-ion battery system as per the P2D model.

To complete the P2D formulation, the electrolyte transport is modeled by a diffusion-reaction equation with a source term that couples it to the lithium-ion diffusion given by (12) through the molar, interfacial flux j ,

$$\epsilon_e \frac{\partial C_e}{\partial t} = \frac{\partial}{\partial x} \left(D_e^{\text{eff}} \frac{\partial C_e}{\partial x} \right) + a_s (1 - t_+^0) j, \quad (18)$$

where ϵ_e , a_s , D_e^{eff} and t_+^0 are the volume fraction and specific interfacial surface area of the active materials, the effective diffusion coefficient of the electrolyte and the transference number, respectively. The interfacial flux j only exists in the anode and cathode domains and is zero otherwise.

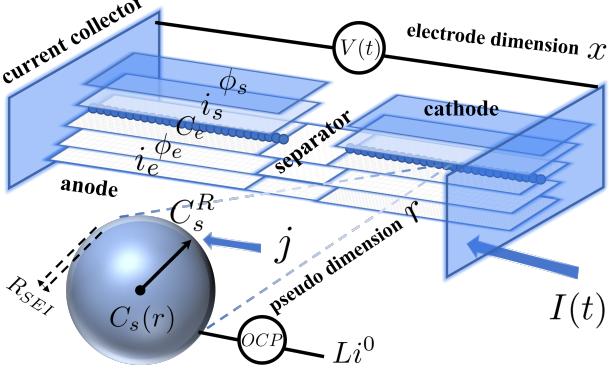


Fig. 4: P2D representation of a LIB cell with superimposed continua spanning over two phases and three domains. The nomenclature can be found in the Supplementary Information.

3.2 MINN Battery Model

Eqns. (12)–(18) form a system of PDAEs, characterized by the circular, nested loops of algebraic variables $h_z(t)$, which couple the dynamical equations of the differential (dynamic) states $h_d(t)$ through the molar flux terms. The origin of these algebraic variables lies in the enormously different characteristic time scales of ionic and electron transport [45]. The resulting system is highly nonlinear and stiff, which poses challenges to numerical techniques. In order to employ specialized solvers optimized for accurate and stable time integration, the system is normally discretized in space, which results in a DAE system in its semi-explicit form,

$$\dot{h}_d = f(h_d, h_z), \quad (19)$$

$$0 = g(h_d, h_z, I), \quad (20)$$

where (19) is an ODE system for ionic transport and (20) are conservation laws resulting from (simplified) electron transport. The so-called hidden states are composed of differential states $h_d = [C_s, C_e]^T$ and algebraic variables $h_z = j$. It is also required for the DAE solver to have a consistent initial condition $h(0) = [h_d^0, h_z^0]^T$ that satisfies (20). In the solution process of a DAE system, it must find roots of the algebraic system of equations $g(h_d, h_z, I)$ iteratively within solver tolerances because h_z cannot be explicitly derived. In addition, the system is unstable at an occurrence when the system deviates from $g(h_d, h_z, I) = 0$. In such an event, re-initialization is necessary through the discontinuous callback of the DAE solver whenever there is discontinuity detected in the input $I(t)$. Consequently, solving the DAE system originating from the P2D model becomes very expensive in the case of real-world driving cycles, which usually render the DAE solver prohibitively slow. To this end, the proposed MINN model circumvents the need for root finding as well as re-initialization.

The DAE system (12)–(18) is of index one [7], which means that for h_d at a given time t , (18) defines h_z uniquely. We can therefore find a locally unique solution h_z^* for (20). Accordingly, the DAE system can be written as one system of ODE,

$$\dot{h}_d = f(h_d, h_z^*). \quad (21)$$

The time integration of the ODE system (21) requires no re-initialization and computationally less workload. We then proceed to parameterize the function $G(h_d, I)$ by a neural network whose size is dependent on the number of dynamic states (input) and algebraic variables (output). Consider approximating G with a DNN of L layers, i.e.,

$$\begin{cases} a^{[1]} = [h_d, I]^T \in \mathbb{R}^m \\ a^{[l]} = \sigma(W_l a^{[l-1]} + b_l), \quad \text{for } l = 2, 3, \dots, L \\ h_z^* = W_L a^{[L]} + b_L \in \mathbb{R}^n \end{cases} \quad (22)$$

Incidentally, the number of trainable parameters θ of this approximation $G_{\text{NN}}(h_d, I; \theta)$ gets large when the order of the system $N = m + n$ is large, especially if the weight matrices W_l , biases b_l and L are also large. An orthogonal collocation method is used for the spatial discretization of the PDAE system (12)–(18) in this work to relieve the difficulty of training. For the same number of discretization points, this method is known to yield much smaller truncation errors than finite volume [12, 15], finite difference [46] and finite element [15, 47] commonly used in the battery modeling community, thanks to spectral accuracy.

There are 16 boundary conditions in the P2D formulation that are necessary to describe the current and potential fields in three domains. This results in 16 additional terms signifying the boundary loss in the loss function for a standard PINN setup, which is expensive and difficult to train. The convergence and regularization of these terms also require additional hyperparameters for tuning. Unlike the physically constrained loss function in the PINN framework [17, 30], MINN accounts for the boundary conditions without specifying them explicitly in the loss function. Instead, they are imposed on the integration constants by considering, for example, the ionic potential ϕ_e as a function of the electrolyte current i_e integrated over electrode dimension x , i.e.,

$$\phi_{e,i} = - \int \frac{i_{e,i}(x)}{\kappa_{\text{eff}}(x)} dx + A \ln C_{e,i}(x) + B_i. \quad (23)$$

By substituting (23) into the boundary conditions, three integration constants B_i are obtained for each domain i , i.e., anode, separator and cathode. By the same token, the interfacial flux $j_i = \frac{1}{a_{s,i}\mathcal{F}} \frac{\partial i_e}{\partial x}$ can be integrated as follows

$$i_{e,i} = a_{s,i}\mathcal{F} \int j_i(x) dx + E_i, \quad (24)$$

where the integration constant E_i is equal to $I(t)$ in the cathode and 0 in the anode such that $i_e = I(t)$ at the electrode-current collector interfaces. In this way, both $i_{e,i}$ and $\phi_{e,i}$ can be exclusively calculated from j_i , and so is $i_{s,i}$ because of Kirchhoff's current law, $i_s + i_e = I(t)$. Likewise, (16) can be integrated to yield two integration constants $\phi_{s,i}^{cc}$, which stand for the anode and cathode potentials at the current

collector. This gives

$$\phi_{s,i} = \int -\frac{i_s(x)}{\sigma_{\text{eff}}} dx + \phi_{s,i}^{\text{cc}} \quad (25)$$

$$\begin{aligned} \phi_{s,i}^{\text{cc}} &= \frac{2RT}{\mathcal{F}} \sinh^{-1} \left(\frac{\mathcal{F}j_i}{2i_0} \right) + \phi_{e,i} \\ &\quad + \phi_{eq,i} - \phi_{s,i} + j_i \cdot R_{\text{SEI}} \end{aligned} \quad (26)$$

$$Y_V = \phi_{s,a}^{\text{cc}} - \phi_{s,c}^{\text{cc}}. \quad (27)$$

Eqns. (23)–(25) reduce the algebraic variables to only the interfacial flux j_i , where j_i is only defined in the anode and cathode. In summary, the algebraic system of equations amounts to

$$g = \begin{bmatrix} a_{s,a} \mathcal{F} \int j_a(x) dx \\ a_{s,c} \mathcal{F} \int j_c(x) dx \end{bmatrix} + \begin{bmatrix} -1 \\ 1 \end{bmatrix} \cdot I(t). \quad (28)$$

During training, the first term \mathcal{L}_y in the loss function (8) measures the error in the model outputs, and they can be the terminal voltage Y_V , SOC or lithium plating potential for a battery system. The SOC of the battery system is defined by the average concentration of the anode particles over the electrode thickness δ , normalized by the electrode stoichiometry at 100% and 0% SOC, and the plating potential is the difference between the solid and liquid potentials at the anode-separator interface (ASI), which give

$$\begin{aligned} Y_{\text{SOC}}(h_d, h_z^*, t) &= \frac{3}{\delta \cdot R^3 (C_s^{100\%} - C_s^{0\%})} \\ &\cdot \left(\int_0^\delta \int_0^R \frac{r^2}{C_s^{\max}} C_s(x, r, t) dr dx - C_s^{0\%} \right), \end{aligned} \quad (29)$$

$$Y_{\text{plp}}(h_d, h_z^*, t) = \phi_s^{\text{ASI}}(t) - \phi_e^{\text{ASI}}(t). \quad (30)$$

4 BENCHMARKING AND TRAINING

This section introduces four state-of-the-art battery models for evaluating the performance of MINN, using physics-based, data-driven and hybrid approaches. Initially, ground truth solutions are obtained using the P2D model. Subsequently, two benchmarking scenarios are considered: 1) for predefined control input, three baseline designs are developed, including DNN, PINN, and neural ODE battery models; 2) for time-varying control input, which may be unknown *a priori* (e.g., derived from real-time optimization or control), a battery model termed data-driven-reduced order model (DD-ROM) is developed to benchmark MINN's performance under real-world driving cycles.

4.1 Data Generation

Newman's P2D model serves as the ground truth for benchmarking. The model's accuracy is dependent on the spatial discretization of the equations introduced in Section 3.1. To this end, we obtained a high-fidelity P2D model using the spectral collocation method consisting of 130 states and 14 algebraic variables. The resulting 144-order model is generated symbolically with *Symbolics.jl* [48], and the time integration is done using a legacy IDA solver [49].

4.2 Baseline Designs

4.2.1 DNN Battery Model

We developed a purely data-driven battery model using deep learning. This model is parameterized by a three-layer deep neural network with an input size of one, corresponding to the time coordinate, and an output size equal to the number of internal states. The DNN is trained to map the time coordinates to the corresponding internal states of the battery, $[h_d^k, h_z^k]^T$, for a predefined current rate. The loss function is defined as

$$\mathcal{L}_{\text{DNN}} = \sum_{k=0}^K \left([\mathcal{NN}_d^k, \mathcal{NN}_z^k]^T - [h_d^k, h_z^k]^T \right)^2 \delta t^k, \quad (31)$$

where \mathcal{NN}_d^k and \mathcal{NN}_z^k are the outputs of the DNN relating to the differential states and algebraic states at timestep k , and δt^k is the sampling intervals.

4.2.2 PINN Battery Model

The PINN hybrid battery model is developed using the schemes illustrated in Fig. 1b. Different from the original PINN formulation [30], which has a spatial coordinate and time as input, i.e. $[x, t]$, the inputs of our PINN battery model are the solution trajectories of all discretized differential states and algebraic variables. Although each of these solution trajectories can be approximated by a PINN mapping space-time coordinates $[x, t]$ to seven solution trajectories, i.e. C_e in three domains plus C_s and j in two domains, the training of these PINNs jointly results in a complex loss function that can undermine the performance of PINN, according to [35]. There are also 16 boundary conditions that further complicate the loss function. For these reasons, we formulate the PINN battery model using the discretized P2D model formulation, i.e., (19)–(20), to include only the time coordinate as the input. In addition to the data-driven loss \mathcal{L}_{DNN} , the loss function for the PINN battery model includes a physical loss due to physical inconsistency:

$$\mathcal{L}_{\text{PINN}} = \mathcal{L}_{\text{DNN}} + \mathcal{L}_{\text{physical}}, \quad (32)$$

$$\begin{aligned} \mathcal{L}_{\text{physical}} &= \sum_{k=0}^K \left(f(\mathcal{NN}_d^k, \mathcal{NN}_z^k) - \dot{h}_d^k \right)^2 \delta t^k \\ &\quad + \sum_{k=0}^K \left| g(\mathcal{NN}_d^k, \mathcal{NN}_z^k, I^k) \right| \delta t^k, \end{aligned} \quad (33)$$

where \dot{h}_d^k is the time derivative of the differential states in the training data generated by the P2D model. The functions f and g are the right-hand sides of (19) and (20), respectively, from the P2D formulation.

4.2.3 Neural ODE Battery Model

Given that both DNN and PINN battery models learn solution trajectories rather than dynamics, they lack adaptability to changes in initial conditions. To address this, a neural ODE (NODE) battery model is incorporated into the baseline design. Unlike the training of NODE [28] using single shooting, multiple shooting is used for training a NODE due to the battery's stiff nonlinear nature. This approach, akin to that detailed in a prior work [37], involves fitting multiple successive trajectory segments. To do this for a battery

system, we first approximate the dynamics of the battery by a neural network, i.e., $dh/dt \approx \mathcal{NN}(h)$. It should be noted that the hidden state h here does not differentiate between h_d and h_z . The loss function is augmented to accommodate shooting constraints using Lagrangian multipliers.

4.2.4 DD-ROM Battery Model

For benchmarking MINN under dynamic control input, an idealized battery model termed DD-ROM is developed, as the baseline designs above are limited to predefined control input. DD-ROM is a reduced-order model, as it eliminates algebraic variables, thereby reducing the model's complexity. Moreover, it is data-driven, requiring measurements of all internal states $[h_d^k, h_z^k]^T$. In practice, these measurements can only be obtained using next-generation embedded sensors in laboratory settings [50].

The training dataset for DD-ROM consists of all internal state data with the corresponding labels, i.e., a pair of input-label data $\mathcal{X} = [h_d^k, I^k]^T$ and $\mathcal{Y} = h_z^k$. The loss function for DD-ROM is defined by the mean square error (MSE) loss of the training data, in a supervised learning fashion. For 49 pairs of input-label data taken from the high-order P2D solution, a mapping function $\mathcal{X} \mapsto \mathcal{Y}$ is parameterized by a DNN, and the loss is minimized down to machine zero using an optimizer. The downside of this hybrid scheme is that it depends on a vast amount of training data and has to include all battery internal states, of which most are not measurable but can only be obtained by physics-based models. Nevertheless, we train and present such a model with a never-before-seen current profile.

4.3 Training Details

The MINN model offers a new path to generating appropriate battery models by seamlessly blending the features of first-principle-based models with neural network architecture, which retrains the individual advantages of physics-based and data-driven approaches. In practice, because certain neural network parameters θ may lead to unphysical states in (4)–(7), we rectify the dynamic, algebraic and output functions f , g and Y by introducing rectified exponentials during training, e.g. the rectified square root, $\sqrt{x}_{\text{ReLU}} = \sqrt{\max(0, x)}$. In addition, we make the electrolyte concentration strictly non-negative. This is done when calculating the model outputs of all four benchmarking models.

The DNN in the MINN architecture consists of three hidden layers with the same number of nodes as the other models. The gradient of the loss function is computed using forward mode AD as implemented in *ForwardDiff.jl* package in Julia [51]. We have noticed that although reverse mode approaches are more efficient in most cases, they are not compatible with the solvers used. Gaussian error linear unit activation is used to mitigate training issues such as vanishing gradients usually associated with RNN [52]. All models in the benchmarks are trained with an AdamW optimizer [53]. The learning rate is set to 0.001, and the training ends when the loss function flatlines. Generally, a wide-scale separation in the internal states and model output leads to the imbalance of loss function components. We scale the input-output data by characteristic time and

internal state scales in order to approximate the widely separated scales in a single DNN.

5 RESULTS AND DISCUSSION

This section evaluates the proposed MINN and compares it against the benchmarking models introduced in Section 4. The results of the comparative study are characterized by the model prediction accuracy, data efficiency, physical interpretability, and computational cost. It is worth noting that real-world battery applications could be complicated and in a wide range of usage conditions that may never be included in any training dataset. However, a reliable battery model must be able to generalize to out-of-sample usage profiles so that the levels of battery safety and degradation governed by physical states are not under- or overestimated. To test this critical ability of the referred models, we deliberately limit the range of conditions used to generate the training dataset.

5.1 Performance under Predefined Control Input

For a predefined control input, such as a constant charge current, the developed DNN, PINN, NODE, and MINN battery models are trained using the first 300 seconds of a P2D spatial-temporal solution under 1C charge starting from 30% SOC. Testing is conducted from 300 seconds to 1200 seconds. As shown in Fig. 5b–e, all four models capture the trend of the time evolution for spatially resolved electrolyte concentration. However, the MAPE of the electrolyte concentration in Fig. 5g indicates that the MINN model outperforms the other three models. In particular, the NODE predictions diverge significantly from the P2D reference for the testing data. This divergence may be due to the fact that, although NODE approximates a dynamic system, it does not incorporate domain-specific equations relevant to a battery system. It is well known that electrolyte depletion can lead to safety risks such as lithium dendrite formation and pathological pathways in the battery's aging trajectory [54]. Inaccurate predictions of electrolyte concentration will severely undermine the effectiveness of health-aware BMS.

For anode potential, the DNN model also shares the limitation of being model-agnostic. As seen in Fig. 5i, the anode potential predicted by the DNN model is unreliable across temporal and spatial coordinates. Both the baseline DNN and NODE models fail to accurately forecast the anode potential trajectories for the testing data. The error in plating potential, shown in Fig. 5f, is a critical metric for battery models, as large errors may lead to lithium plating during fast charging applications.

The PINN battery model is capable of following the trend of the spatial-temporal solution, as demonstrated in the electrolyte concentration and anode potential plots. This capability is due to the model's loss function being informed by the P2D equations. The testing error of PINN in Fig. 5f and Fig. 5g is due to the fact that the physical loss term (33) in the loss function is only a soft constraint [35]. It is also important to note that DNN and PINN cannot be used for solutions with different initializations. For example, if the initial condition is altered, both DNN and PINN yield

completely unreliable results. On the other hand, NODE and MINN benefit from sequence-to-sequence learning, enabling them to account for changes in initial conditions. To illustrate this, we designed an additional test where the initial condition of the testing data differed from that of the training data, the results of which are shown in the Supplementary Information.

The internal state trajectories shown in Fig. 5 have significant implications for battery health and safety diagnostics and must be accurately captured by the deployed model for next-generation BMS. The MINN battery model consistently captures all local states and model outputs, as illustrated by the spatio-temporal plots in Fig. 5e and 5l. Furthermore, it can adapt to different initial conditions because MINN learns the dynamics of the physical system rather than the solution trajectories of an autonomous system. Moreover, the MINN model features physically meaningful states and parameters that can be adapted to battery aging.

5.2 Performance under Dynamic Control Input

To evaluate the effectiveness of the MINN model for learning battery dynamics, e.g., excited by a prior unknown input profile, we use an arbitrary vehicle driving cycle to generate testing data. A challenging training dataset is purposefully chosen to evaluate MINN's generalizability on unseen testing data. As shown in Fig. 6, the training dataset consists of 49 snapshots generated by the P2D model with the initial SOC fixed at 30% and a sinusoidal input signal lasting only two seconds and bounded by 1C. By contrast, a highly dynamic current is used in the testing where the initial SOC is set to 90% and the maximum current reaches 5C.

For an ideal case where all internal battery states are measurable, the DD-ROM is developed, whereby part of the model states are obtained by a relationship learnt from sampling the state trajectories of the P2D model. In comparison, the training of the MINN model involves only the experimentally measurable output of the P2D model, including the terminal voltage, lithium plating potential and SOC. Thanks to the built-in, problem-specific recurrent unit, it does not require the acquisition of the internal state data.

Accurate prediction of the system outputs, including the terminal voltage, plating potential and SOC, is important to advance BMS functionalities, such as power capability prediction [56] and health-aware fast charging [10, 57]. Table 1 shows the computational complexity, training dataset and numerical accuracy of the above three battery models in predicting these outputs, while Fig. 7g displays the trajectory of the plating potential. Both the MINN model and DD-ROM model show high accuracy in the terminal voltage, achieving generalization errors of less than 12 mV. In Fig. 7g, the black dashed line at 0 V highlights the critical level below which the lithium plating is triggered. To predict such plating potential, the MINN model is as good as the DD-ROM which is developed under the hypothetically available information of all battery states. Regarding SOC prediction, the MINN model has greater accuracy than the DD-ROM, with a generalization error of only 0.06%. Other testing data generated at various initial conditions other than 85–90% SOC have also been considered, which has yielded

similar results in predicting the model outputs and further confirmed the superiority of MINN over DD-ROM. This achievement by MINN with only the measurement data for training is practical during real-world battery usage. In fact, the upper limit of the accuracy of MINN is not the first-principle model used in this benchmark but the battery system itself.

To evaluate MINN's capability of predicting the dynamics of internal battery states, the locally distributed electrolyte concentration and solid-phase surface concentration are examined in spatiotemporal plots. As shown in Fig. 7c, MINN is capable of reproducing the electrolyte evolution accurately along the challenging operating profile with a MAPE plotted in Fig. 7e of less than 2%. Although DD-ROM battery model gives less than 1% absolute percentage error (APE) for the terminal voltage as shown in Fig. 7d, it under- and overestimates the degree of electrolyte depletion. For example, at around 100 seconds, marked by a blue isosurface, the DD-ROM model underestimates the depletion in the anode and yet overestimates it in the anode, separator and cathode starting from 600 seconds (light orange). Fig. 7h–j depicts the comparative results for the solid-phase concentration. Here, the colour black represents the theoretical maximum surface concentration of graphite particles, near which the movement of lithium ions matches the intercalation threshold of lattice sites in the anode and the dendritic growth of lithium is inevitable [58]. The DD-ROM prediction in Fig. 7i yields larger black areas adjacent to the anode-separator interface, which means significantly more lithium plating if one deploys it in the BMS. In addition, DD-ROM overestimates the surface concentration in the anode at 500 seconds. When imposing a constraint on the surface concentration for vehicle battery control, such overestimation will curtail the energy recovered from regenerative braking [59]. However, the MINN model predicts the critical surface concentration in Fig. 7j as close as the P2D during the entire process, thereby allowing accurate monitoring and control of the solid-phase concentration.

5.3 Computational Complexity

In this section, the computational complexities of DD-ROM and MINN are compared to that of the P2D model. In general, the computational complexity is dependent on the system order. To this end, we define the order of the system as the total number of differential states and algebraic variables. Consequently, the system order is a result of spatial discretization. In our previous work [60], we evaluated different spatial discretization techniques, e.g. finite volume, finite difference, control volume and spectral methods, for the accuracy of pulse current responses of the solid-phase diffusion. It is found that while most of these techniques capture characteristics in the low-frequency region, some techniques outperform others with the same system order for a wider frequency response range. In particular, the spectral method of order three is more accurate than the 10th-order finite volume method. Although this result cannot be simply extrapolated to the full P2D equations, the assessment of the computational complexity should be based on the DAE or ODE systems derived using the same spatial discretization technique. We have in this regard chosen the

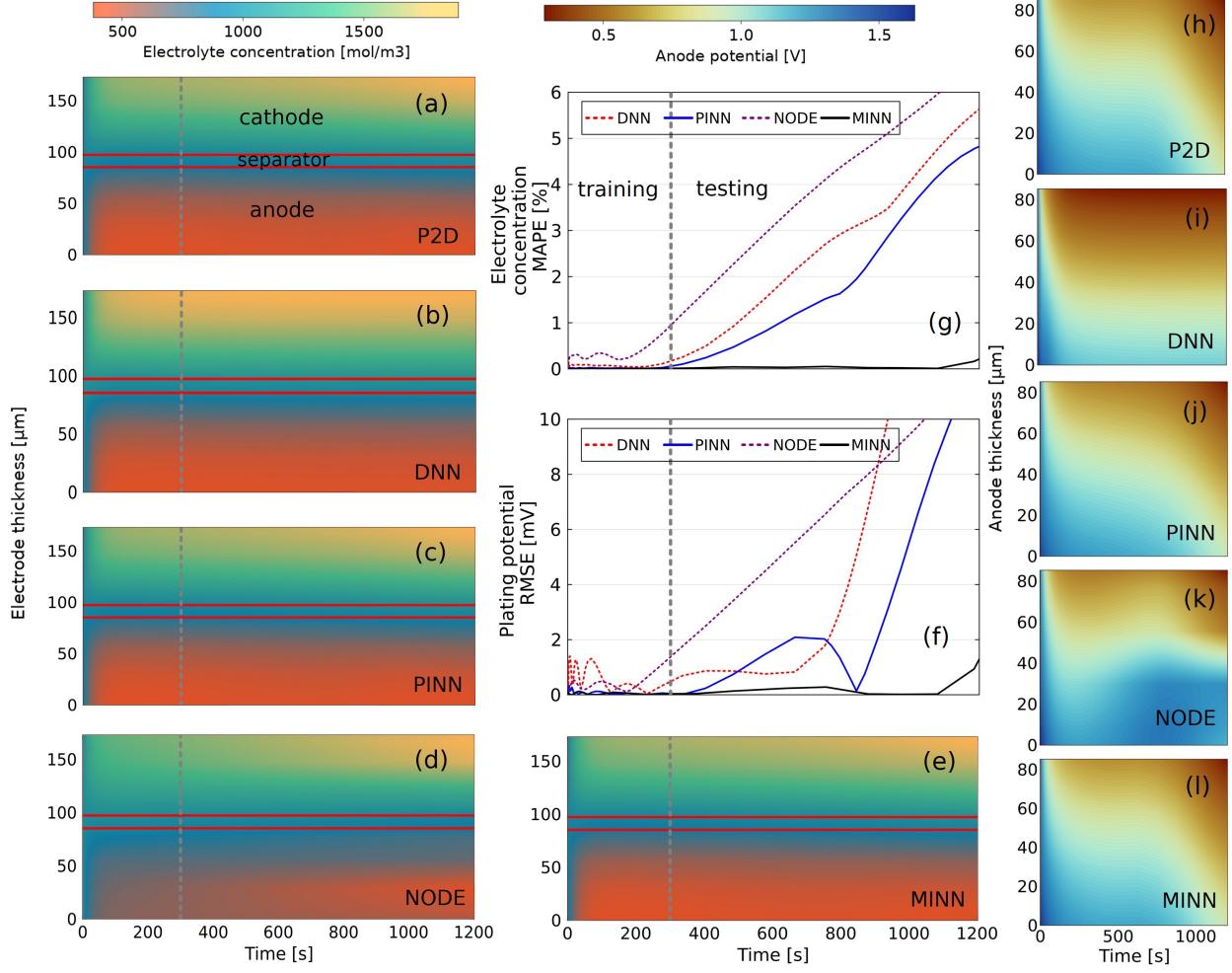


Fig. 5: Comparison of different data-driven and hybrid model predictions under 1C charge with P2D predictions as ground truth. All four models are trained using data sampled from the first 300 seconds of the 1C charge data starting from 30% SOC. (a)-(e) Spatiotemporal plots of electrolyte concentration. (f) The mean absolute percentage error of electrolyte concentration averaged over the thickness. (g) The root mean square error of the plating potential averaged over the thickness. (h)-(l) Spatiotemporal plots of anode potential $\phi_s - \phi_e$.

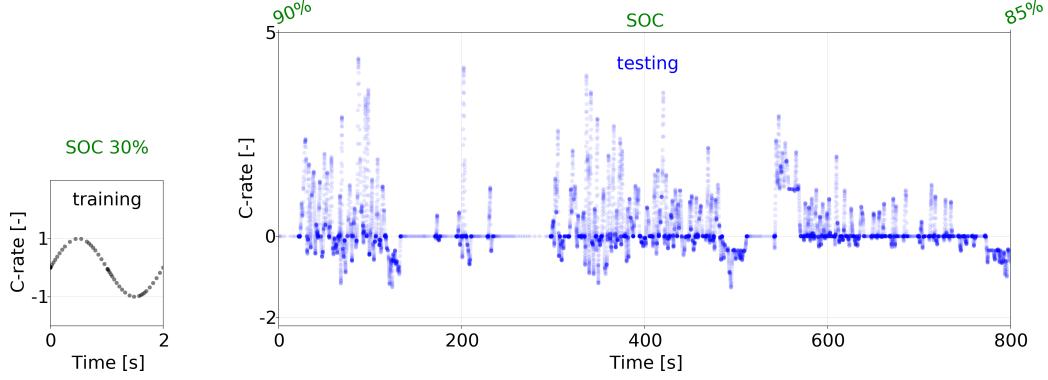


Fig. 6: Dynamic current input profile and SOC range for generating training and testing dataset.

spectral method for its superior performance. The resulting system orders of the three battery models in this case study are shown in Table 1.

Additionally, the computational complexity is also characterized by the Jacobian of these battery models. Since timescale separation is determined by the condition number

of the Jacobian matrix, we compute the condition numbers at different time steps of the training sinusoidal response. It is equal to the ratio between maximal and minimal eigenvalues of the Jacobian, which roughly measures the local timescale separation. We report in Table 1 the maximum condition number for the three models. By convention, a

Metrics Models	Complexity					Training	Generalization error			
	System	Model order	Solver	Condition number	Solution speedup (average)		Battery dataset	η_{plp} [mV]	Voltage [mV]	SOC [%]
P2D	DAE	130 + 14	IDA (SUNDIALS)	6.0×10^{19}	1X					
DD-ROM	ODE	130	Rodas4	6.7×10^9	91X	Internal states h	9.88	9.87	0.635	
MINN	ODE	82	Rodas4	3.0×10^8	182X	Measurement Y	6.28	11.6	0.059	

TABLE 1: Model complexity, data efficiency in training. The generalization error of DD-ROM and MINN battery models are compared with results of a dynamic driving profile obtained by a high-fidelity P2D model with the LG M50 parameterization (see *Supplementary Information: Model Parameterization*). The generalization error is measured in root mean square error (RMSE) against the P2D benchmark. The solver used for the DAE system representing the 144-order P2D model is a legacy SUNDIALS solver [49], and the stiff solver for DD-ROM and MINN employs a 4th-order A-stable stiffly stable Rosenbrock method (Rodas4). All models are implemented using the *DifferentialEquations.jl* package in Julia [55].

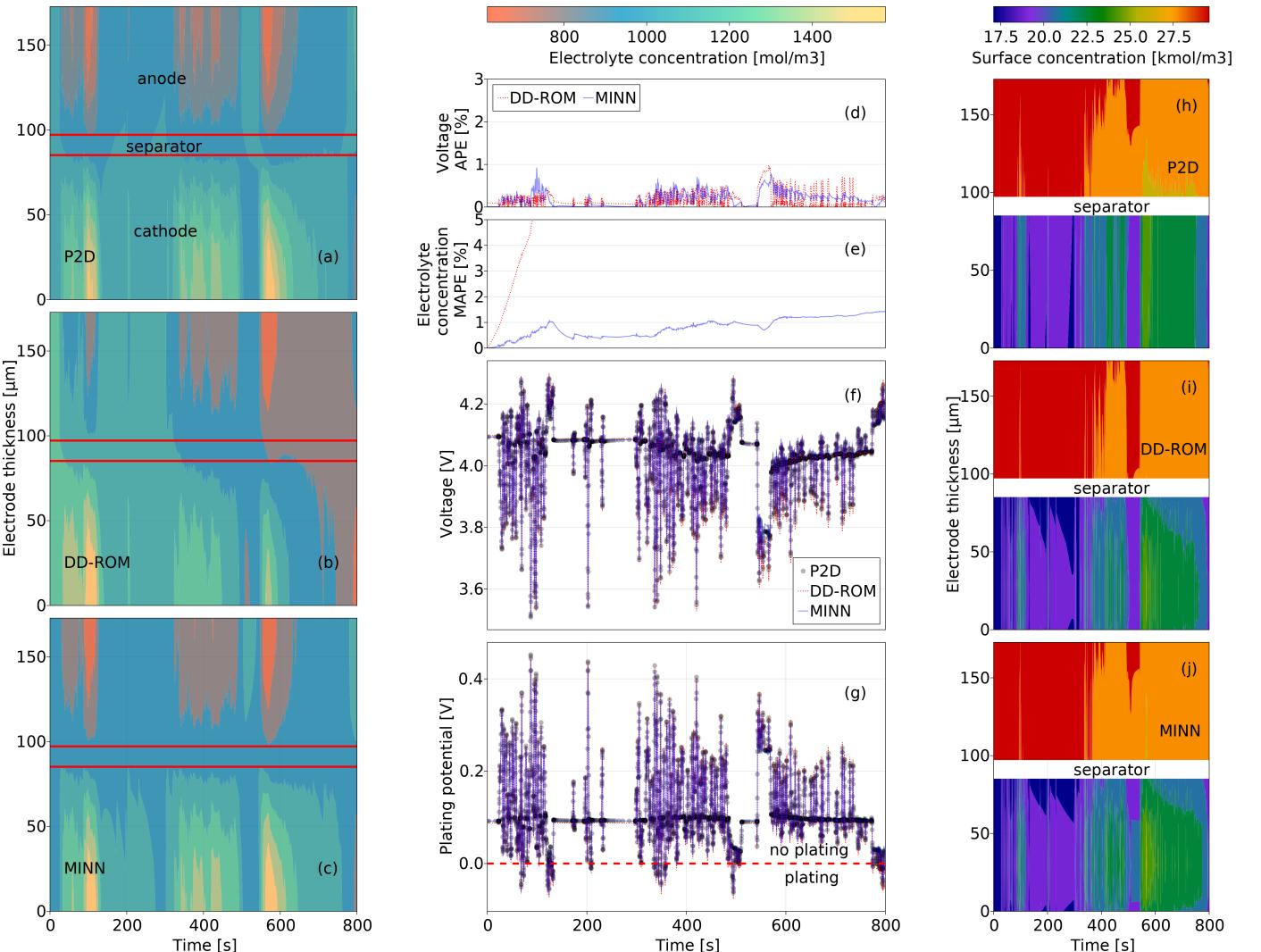


Fig. 7: Performance of the DD-ROM and MINN battery models in learning system dynamics, compared with the results of reference 144-order P2D model. (a)-(c) Liquid-phase ion concentration (electrolyte concentration). (d)-(e) Generalization errors of terminal voltage and electrolyte concentration. (f) Terminal voltage response to the testing current. (g) Anode plating potential at the anode-separator interface. (h)-(j) Solid-phase concentration at the surface of particles.

system is said to be stiff if this number is large, and it requires stabilized stepping, which is potentially computationally costly.

In practice, most P2D model implementations feature additional algebraic state variables that are converged at each time step by using, e.g. an iterative algorithm. While

these implementations may realize millisecond-scale simulations [14] for static charge and discharge, the solution time can be prohibitively slow for dynamic driving profiles due to increasing stiffness. As shown in Table 1, the two hybrid models, i.e., MINN and DD-ROM, achieve two orders of magnitude speedup in the solution time for an 800-second vehicle driving test, of which the MINN battery model has a slight edge over the DD-ROM model. The significant speed improvement of the MINN framework compared to DD-ROM is attributed to its high data efficiency, which allows for learning the complex dynamics of batteries without the need for a fixed number of internal states. This unique feature enables low-order approximations, as demonstrated by developing an 82-order model in Table 1, in contrast to DD-ROM, which requires 130 states for similar accuracy. The remarkable speedup in computational time will make onboard model-based applications possible, such as online parameter identification, state estimation and closed-loop control. Indeed, a vast majority of daily battery usage is driven by time-varying current profiles, under which the identifiability of battery models, including the P2D and MINN, will often be improved significantly compared to static excitations. Therefore, improving computational efficiency under dynamic operating conditions will help lift the computational burden of parameterization.

5.4 Discussion on Adaptive Battery Modeling

During battery lifetime, conventional physics-based modeling requires periodic re-parameterization because of the ever-changing nature of multi-physical battery parameters due to ageing [61, 62]. The need for computationally expensive re-parameterization undermines the applications of physics-based models, which are supposed to have minimal dependence on data acquisition and training. This is evidenced by the fact that no mass-produced BMS on the market today has claimed the usage of physics-based models. MINN allows for the simplification of DAE-based structures and may potentially improve the identifiability of parameters. Accordingly, MINN can then be used for aging adaptive models for a wide range of intelligent battery management applications, not only in the short term of several hours or days, but also over the battery's entire lifespan. Numerous examples of such applications include fast charging, lifetime optimization, thermal fault detection, and safety prognosis, which are the main challenges of BMS algorithm design.

6 CONCLUSION

The rapid upscaling of battery-powered electric vehicles makes it possible to collect big data. Based on the data, a wave of data-driven models under the hood of machine learning has recently been developed in the battery community. While data-driven surrogate models excel in learning complex battery characteristics, they inherently lack the ability to generalize beyond the training data and to provide a physical interpretation of the internal battery status. To fundamentally bridge the identified research gap, we proposed a conceptually novel physics-based deep learning architecture, MINN, to seamlessly combine the merits of physics-based and data-driven models.

MINN stands out through its remarkable accuracy and acceleration, surpassing state-of-the-art benchmarks across the full spectrum of system modeling. In comparison with physics-based simulations, MINN achieved two orders of magnitude speedup while maintaining comparable accuracy. Unlike purely data-driven models, MINN is data-efficient to train and generalizable to unseen operational conditions. Its incorporation of physical parameters and interpretable hidden states, by design, facilitates the learning of system dynamics rather than being limited to input-output relationships. In contrast to the existing practices of physics-based machine learning, MINN offers distinct advantages. Specifically, it can be trained without the need for internal state data, addressing the limitations of current sensing technologies in real-world battery applications. Furthermore, MINN's unique capability to model general non-autonomous PDAE systems under any control input empowers the implementation of advanced and real-time control strategies for internal states.

The substantial and practical advantages of MINN make it an exceptional choice for developing the next-generation BMS, including battery system identification, fault diagnostics, safety prognostics, and physics-based control. By integrating machine learning with physics-based modeling, the MINN framework offers a powerful tool for analyzing general dynamic systems commonly found in diverse fields, such as mechatronics, thermal fluid dynamics, electrical power systems, and energy storage systems.

REFERENCES

- [1] Y. Liang and Y. Yao, "Positioning organic electrode materials in the battery landscape," *Joule*, 2018.
- [2] Y. Jin, K. Liu, J. Lang, D. Zhuo, Z. Huang, C.-A. Wang, H. Wu, and Y. Cui, "An intermediate temperature garnet-type solid electrolyte-based molten lithium battery for grid energy storage," *Nature Energy*, vol. 3, pp. 732–738, 2018.
- [3] A. Farnmann and D. U. Sauer, "Comparative study of reduced order equivalent circuit models for on-board state-of-available-power prediction of lithium-ion batteries in electric vehicles," *Appl. Energy*, 2018.
- [4] W. Li, D. Cao, D. Jöst, F. Ringbeck, M. Kuipers, F. Frie, and D. U. Sauer, "Parameter sensitivity analysis of electrochemical model-based battery management systems for lithium-ion batteries," *Appl. Energy*, vol. 269, p. 115104, 2020.
- [5] Z. Khalik, M. Donkers, J. Sturm, and H. Bergveld, "Parameter estimation of the Doyle-Fuller-Newman model for lithium-ion batteries by parameter normalization, grouping, and sensitivity analysis," *J. Power Sources*, vol. 499, p. 229901, 2021.
- [6] C. Zou, C. Manzie, D. Nešić, and A. G. Kallapur, "Multi-time-scale observer design for state-of-charge and state-of-health of a lithium-ion battery," *J. Power Sources*, vol. 335, pp. 121–130, 2016.
- [7] A. Bizeray, S. Zhao, S. Duncan, and D. Howey, "Lithium-ion battery thermal-electrochemical model-based state estimation using orthogonal collocation and a modified extended Kalman filter," *J. Power Sources*, vol. 296, pp. 400–412, 2015.

- [8] X.-G. Yang, Y. Leng, G. Zhang, S. Ge, and C.-Y. Wang, "Modeling of lithium plating induced aging of lithium-ion batteries: Transition from linear to nonlinear aging," *J. Power Sources*, vol. 360, pp. 28–40, 2017.
- [9] P. K. Jones, U. Stimming, and A. A. Lee, "Impedance-based forecasting of lithium-ion battery performance amid uneven usage," *Nat. Commun.*, vol. 13, no. 1, Aug. 2022.
- [10] C. Zou, C. Manzie, and D. Nescic, "Model predictive control for lithium-ion battery optimal charging," *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 2, pp. 947–957, 2018.
- [11] S. Kolluri, S. V. Aduru, M. Pathak, R. D. Braatz, and V. R. Subramanian, "Real-time nonlinear model predictive control (NMPC) strategies using physics-based models for advanced lithium-ion battery management system (BMS)," *J. Electrochem. Soc.*, vol. 167, no. 6, p. 063505, 2020.
- [12] M. Torchio, L. Magni, R. B. Gopaluni, R. D. Braatz, and D. M. Raimondo, "LIONSIMBA: A Matlab framework based on a finite volume model suitable for Li-ion battery design, simulation, and control," *J. Electrochem. Soc.*, vol. 163, no. 7, pp. A1192–A1205, 2016.
- [13] S. Han, Y. Tang, and S. K. Rahimian, "A numerically efficient method of solving the full-order pseudo-2-dimensional (P2D) Li-ion cell model," *J. Power Sources*, vol. 490, p. 229571, 2021.
- [14] M. D. Berliner, H. Zhao, S. Das, M. Forsuelo, B. Jiang, W. H. Chueh, M. Z. Bazant, and R. D. Braatz, "Nonlinear identifiability analysis of the porous electrode theory model of lithium-ion batteries," *J. Electrochem. Soc.*, vol. 168, no. 9, p. 090546, 2021.
- [15] V. Sulzer, S. G. Marquis, R. Timms, M. Robinson, and S. J. Chapman, "Python Battery Mathematical Modelling (PyBaMM)," *J. Open Res. Softw.*, vol. 9, no. 1, p. 14, 2021.
- [16] I. Korotkin, S. Sahu, S. E. J. O'Kane, G. Richardson, and J. M. Foster, "DandeLiion v1: An extremely fast solver for the Newman model of lithium-ion battery (dis)charge," *J. Electrochem. Soc.*, 2021.
- [17] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics informed deep learning (Part II): Data-driven discovery of nonlinear partial differential equations," *ArXiv*, 2017.
- [18] N. A. Chaturvedi, R. Klein, J. Christensen, J. Ahmed, and A. Kojic, "Algorithms for advanced battery-management systems," *IEEE Control Syst.*, vol. 30, pp. 49–68, 2010.
- [19] Y. Li, M. Vilathgamuwa, T. W. Farrell, S. S. Choi, N. T. Tran, and J. Teague, "A physics-based distributed-parameter equivalent circuit model for lithium-ion batteries," *Electrochim. Acta*, 2019.
- [20] Y. Li, D. M. Vilathgamuwa, E. Wikner, Z. Wei, X. Zhang, T. Thiringer, T. Wik, and C. Zou, "Electrochemical model-based fast charging: Physical constraint-triggered PI control," *IEEE Trans. Energy Convers.*, vol. 36, no. 4, pp. 3208–3220, 2021.
- [21] A. Aitio and D. A. Howey, "Predicting battery end of life from solar off-grid system field data using machine learning," *Joule*, 2021.
- [22] X. Hu, L. Xu, X. Lin, and M. G. Pecht, "Battery lifetime prognostics," *Joule*, 2020.
- [23] B.-R. Chen, C. Walker, S. Kim, M. R. Kunz, T. R. Tanim, and E. J. Dufek, "Battery aging mode identification across nmc compositions and designs using machine learning," *Joule*, 2022.
- [24] Y. Liu, X. Shu, H. Yu, J. Shen, Y. Zhang, Y. Liu, and Z. Chen, "State of charge prediction framework for lithium-ion batteries incorporating long short-term memory network and transfer learning," *J. Energy Storage*, vol. 37, p. 102494, 2021.
- [25] S. J. Harris, A. T. Timmons, D. R. Baker, and C. W. Monroe, "Direct in situ measurements of Li transport in Li-ion battery negative electrodes," *Chem. Phys. Lett.*, vol. 485, pp. 265–274, 2009.
- [26] H. Wang, Y. Zhu, S. C. Kim, A. Pei, Y. Li, D. T. Boyle, H. Wang, Z. Zhang, Y. Ye, W. Huang, Y. Liu, J. Xu, J. Li, F. Liu, and Y. Cui, "Underpotential lithium plating on graphite anodes caused by temperature heterogeneity," *Proc. Natl. Acad. Sci.*, vol. 117, pp. 29 453 – 29 461, 2020.
- [27] T. Gao, Y. Han, D. Fragedakis, S. Das, T. Zhou, C.-N. Yeh, S. Xu, W. C. Chueh, J. Li, and M. Z. Bazant, "Interplay of lithium intercalation and plating on a single graphite particle," *Joule*, 2021.
- [28] T. Q. Chen, Y. Rubanova, J. Bettencourt, and D. K. Duvenaud, "Neural ordinary differential equations," in *NeurIPS*, 2018.
- [29] S. Pepe, J. Liu, E. Quattrochi, and F. Ciucci, "Neural ordinary differential equations and recurrent neural networks for predicting the state of health of batteries," *Journal of Energy Storage*, vol. 50, p. 104209, 2022.
- [30] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics informed deep learning (Part I): Data-driven solutions of nonlinear partial differential equations," *ArXiv*, 2017.
- [31] M. Raissi, P. Perdikaris, and G. Karniadakis, "Multistep neural networks for data-driven discovery of nonlinear dynamical systems," *ArXiv: Dynamical Systems*, 2018.
- [32] L. Lu, X. Meng, Z. Mao, and G. E. Karniadakis, "Deepxde: A deep learning library for solving differential equations," *SIAM Rev.*, vol. 63, pp. 208–228, 2020.
- [33] O. Hennigh, S. Narasimhan, M. A. Nabian, A. Subramaniam, K. M. Tangsali, M. Rietmann, J. del Águila Ferrandis, W. Byeon, Z. Fang, and S. Choudhry, "Nvidia simnet: an AI-accelerated multi-physics simulation framework," in *ICCS*, 2021.
- [34] K. Zubov, Z. McCarthy, Y. Ma, F. Calisto, V. Pagliarino, S. Azeglio, L. Bottero, E. Luján, V. Sulzer, A. Bharambe et al., "NeuralPDE: Automating physics-informed neural networks (PINNs) with error approximations," *arXiv*, 2021.
- [35] A. S. Krishnapriyan, A. Gholami, S. Zhe, R. M. Kirby, and M. W. Mahoney, "Characterizing possible failure modes in physics-informed neural networks," in *NeurIPS*, 2021.
- [36] S. L. Brunton, J. L. Proctor, and J. N. Kutz, "Discovering governing equations from data by sparse identification of nonlinear dynamical systems," *Proceedings of the National Academy of Sciences*, vol. 113, pp. 3932–3937, 2015.
- [37] P. Hu, W. Yang, Y. Zhu, and L. Hong, "Revealing hidden dynamics from time-series data by ODENet," *J. Comput. Phys.*, vol. 461, p. 111203, 2022.
- [38] K. Cho, B. van Merriënboer, Çaglar Gülcöhre, D. Bah-

- danau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *Conference on Empirical Methods in Natural Language Processing*, 2014.
- [39] H. Jaeger and H. Haas, "Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication," *Science*, vol. 304, pp. 78 – 80, 2004.
- [40] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, pp. 1735–1780, 1997.
- [41] D. P. Finegan, J. Zhu, X. Feng, M. Keyser, M. Ulmefors, W. Li, M. Z. Bazant, and S. J. Cooper, "The application of data-driven methods and physics-based learning for improving battery safety," *Joule*, vol. 5, pp. 316–329, 2021.
- [42] M. Doyle, T. F. Fuller, and J. Newman, "Modeling of galvanostatic charge and discharge of the lithium/polymer/insertion cell," *J. Electrochem. Soc.*, vol. 140, no. 6, pp. 1526–1533, 1993.
- [43] T. F. Fuller, M. L. Doyle, and J. Newman, "Simulation and optimization of the dual lithium ion insertion cell," *J. Electrochem. Soc.*, vol. 141, pp. 1–10, 1994.
- [44] J. Newman and K. E. Thomas-Alyea, *Electrochemical systems*. John Wiley & Sons, 2012.
- [45] C. Zou, C. Manzie, and D. Nescic, "A framework for simplification of PDE-Based lithium-ion battery models," *IEEE Trans. Contr. Syst. Technol.*, vol. 24, pp. 1594–1609, 2016.
- [46] J. Newman, "Fortran programs for simulation of electrochemical systems: Dualfoil," <http://www.cchem.berkeley.edu/jsngrp/fortran.html>, 1998.
- [47] L. Cai and R. E. White, "Mathematical modeling of a lithium ion battery with thermal effects in COMSOL Inc. Multiphysics (MP)," *J. Power Sources*, vol. 196, no. 14, pp. 5985–5989, 2011.
- [48] S. Gowda, Y. Ma, A. Cheli, M. Gwózdz, V. B. Shah, A. Edelman, and C. Rackauckas, "High-performance symbolic-numerics via multiple dispatch," *ACM Commun. Comput. Algebra*, vol. 55, no. 3, p. 92–96, 2022.
- [49] A. C. Hindmarsh, P. N. Brown, K. E. Grant, S. L. Lee, R. Serban, D. E. Shumaker, and C. S. Woodward, "Sundials: Suite of nonlinear and differential/algebraic equation solvers," *ACM Trans. Math. Softw.*, vol. 31, no. 3, p. 363–396, 2005.
- [50] D. P. Finegan, J. Zhu, X. Feng, M. Keyser, M. Ulmefors, W. Li, M. Z. Bazant, and S. J. Cooper, "The application of data-driven methods and physics-based learning for improving battery safety," *Joule*, vol. 5, pp. 316–329, 2021.
- [51] J. Revels, M. Lubin, and T. Papamarkou, "Forward-mode automatic differentiation in Julia," *arXiv*, 2016.
- [52] D. Hendrycks and K. Gimpel, "Gaussian error linear units (GELUs)," *arXiv: Learning*, 2016.
- [53] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *ICLR*, 2017.
- [54] P. M. Attia, A. Bills, F. B. Planella, P. Dechent, G. dos Reis, M. Dubarry, P. Gasper, R. Gilchrist, S. Greenbank, D. A. Howey, O. Liu, E. Khoo, Y. Preger, A. S. Soni, S. Sripad, A. G. Stefanopoulou, and V. Sulzer, "Review—"knees" in lithium-ion battery aging trajec-
- tories," *J. Electrochem. Soc.*, 2022.
- [55] C. Rackauckas and Q. Nie, "Differentialequations.jl – a performant and feature-rich ecosystem for solving differential equations in Julia," *J. Open Res. Softw.*, vol. 5, p. 15, 2017.
- [56] T. Wik, B. Fridholm, and H. Kuusisto, "Implementation and robustness of an analytically based battery state of power," *J. Power Sources*, vol. 287, pp. 448–457, 2015.
- [57] J. Sieg, J. Bandlow, T. Mitsch, D. Dragicevic, T. Materna, B. Spier, H. Witzenhausen, M. Ecker, and D. U. Sauer, "Fast charging of an electric vehicle lithium-ion battery at the limit of the lithium deposition process," *J. Power Sources*, vol. 427, pp. 260–270, 2019.
- [58] Z. Guo, J. Zhu, J. Feng, and S. Du, "Direct in situ observation and explanation of lithium dendrite of commercial graphite electrodes," *RSC Advances*, vol. 5, pp. 69 514–69 521, 2015.
- [59] L. Wikander, B. Fridholm, S. Gros, and T. Wik, "Ideal benefits of exceeding fixed voltage limits on lithium-ion batteries with increasing cycle age," *J. Power Sources*, 2019.
- [60] Y. Li, D. Karunathilake, D. M. Vilathgamuwa, Y. Mishra, T. W. Farrell, S. S. Choi, and C. Zou, "Model order reduction techniques for physics-based lithium-ion battery management: A survey," *IEEE Ind. Electron. Mag.*, vol. 16, pp. 36–51, 2022.
- [61] M. Streb, M. Ohrelius, M. Klett, and G. Lindbergh, "Improving Li-ion battery parameter estimation by global optimal experiment design," *J. Energy Storage*, 2022.
- [62] M. Streb, M. Andersson, V. L. Klass, M. Klett, M. Johansson, and G. Lindbergh, "Investigating reparametrization of electrochemical model-based battery management using real-world driving data," *eTransportation*, 2023.