

RESEARCH ARTICLE OPEN ACCESS

Using Physics-Informed Neural Networks to Predict the Damage in Single Particle Models for Li-Ion Batteries Active Materials

Nikolai Erhardt¹  | Tim Laufer²  | Hermann Nirschl¹

¹Institute of Mechanical Process Engineering and Mechanics (MVM), Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany | ²Institute for Applied and Numerical Mathematics (IANM), Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany

Correspondence: Nikolai Erhardt (Nikolai.Erhardt@kit.edu)

Received: 4 December 2025 | **Revised:** 16 September 2026 | **Accepted:** 23 September 2026

Keywords: concentration | coupled differential equations | damage | displacement | machine learning | mechanical degradation | nickel manganese cobalt oxide (NMC) | physics-informed neural network (PINN) | single particle model

ABSTRACT

In order to ensure the longevity of active materials, microscale processes that occur during lithiation and delithiation must be better understood. These processes include chemical diffusion, the mechanical equilibrium of the host material, and fatigue phenomena in the material, which result in damage. Since experimental investigation of these individual but strongly intercoupled effects is hardly possible given the current state of research, this work presents the mathematical, theoretical equations for a 1D single-particle model with their mutual feedbacks and a phase field approach for the damage calculation. To solve the resulting system of equations, a Physics-Informed Neural Network (PINN) is used, which combines the advantages of Artificial Neural Networks (ANN) of fast execution and grid freedom with the advantages of conventional numerical methods such as interpretability and a flexible handling of boundary conditions. It is shown that a PINN is capable of learning the complex coupled governing differential equations. NMC622 is investigated as the active material on the cathode side. It was found that during lithiation with a stress-free surface, notably high currents lead to damage in the particle. This damage then occurs predominantly in the center of the particle and is mainly favored by a high level of lithiation.

1 | Introduction

In the quest for sustainable energy solutions, lithium-ion batteries have emerged as a cornerstone technology due to their high energy density, long cycle life, and lightweight design. These attributes have propelled their widespread adoption across a range of applications, from portable electronics to electric vehicles and large-scale energy storage systems. As the demand for efficient and reliable energy storage continues to grow, advancing the performance and longevity of lithium-ion batteries remains a critical challenge. Understanding the fundamental mechanisms that govern their operation and degradation is essential for the development of next-generation battery technologies capable of meeting the increasing energy demands of modern society [1].

Within this context, mechanical stresses in the cathode, resulting from ion insertion and extraction, pose significant hurdles to the longevity and durability of lithium-ion batteries. These stresses can lead to material degradation and structural failures, ultimately undermining the battery's overall performance and lifespan [2–4]. For layered materials such as lithium nickel manganese cobalt oxide (NMC), lithium nickel cobalt aluminum oxide (NCA), and nickel cobalt oxide (NCO), which often exhibit polycrystalline structures, cracking can occur intergranularly within the grain boundaries or intragranularly within the primary particles [5]. Various practical strategies exist to combat unintended micro-crack propagation, including the use of single-crystal particles, doping with foreign ions, applying protective coatings, or designing

This is an open access article under the terms of the [Creative Commons Attribution](https://creativecommons.org/licenses/by/4.0/) License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2026 The Author(s). *Energy Technology* published by Wiley-VCH GmbH.

concentration gradients. However, each of these approaches has distinct limitations [6].

The phenomenon driving the mechanical degradation of batteries is a complex interplay among chemical (de)intercalation, mechanical displacement within the host lattice, and the resulting fracture processes that generate microcracks within electrode particles. Given the high costs of experimental approaches and the lack of in-operando analysis, a realistic modeling framework that captures the holistic interplay of these processes is vital for advancing the understanding of mechanical degradation in lithium-ion batteries.

Frequently, the modeling of chemical and mechanical processes is conducted using first-principles approaches. Coupled models include here the mechanical stress contributions to the chemical potential. The result is a nonlinear partial differential equation to determine the concentration profile inside the electrode particle. The strain inside the electrode particles results from the spatial confinement of intercalated lithium atoms [7–9]. Strain levels vary between approximately 1%–3% in NMC, LCO, and NCA [10].

To model the fracture process, several approaches are available, including linear elastic fracture models (LEFM), phase-field models (PFM), and cohesive zone models (CZM) [11]. Some comprehensive models employ PFM, integrating it with the nonlinear mechanical and chemical process recoupling and earning a complex system of differential equations, which is expensive to solve [12, 13].

In our modern world, artificial intelligence (AI) is increasingly used to solve challenging tasks in engineering [14]. This encompasses a variety of computational techniques designed to mimic human cognitive functions, ranging from simple rule-based systems to complex, data-driven models. AI methods can be broadly categorized based on their dependency on domain knowledge and data: white-box, gray-box, and black-box models [15, 16]. White-box models are interpretable algorithms incorporating explicit domain knowledge, allowing for transparent decision-making processes. In contrast, black-box models, such as deep neural networks, excel at capturing complex patterns from large datasets but often lack interpretability. Gray-box models strike a balance, integrating both data-driven insights and domain knowledge to improve interpretability and reliability [17].

Amid these approaches, Physics-Informed Neural Networks (PINNs) have emerged as a powerful framework that incorporates known physical laws into the training of neural networks [18]. By embedding partial differential equations directly into the learning process, PINNs enable the modeling of complex systems with improved accuracy and generalization capabilities. This blend of data-driven learning and physical constraints offers a promising avenue for modeling intricate problems across various scientific and engineering domains, addressing some limitations of traditional black-box models by introducing transparency and robustness rooted in established physical principles. So, PINNs can be used as a fast, mesh-free, and implicit method to approximate the solution of differential equations. PINNs have found widespread application in solving forward problems, where they are used to predict system behavior by integrating known physical laws into the learning process [19–21]. In addition to forward

problems, PINNs are also adept at tackling inverse problems, where they help infer unknown parameters or system states from the observed data [22–24]. Therefore, by bridging data-driven learning with physical constraints, PINNs offer a versatile and powerful toolset for both forward and inverse problem solving across various disciplines.

Recently, the trend is moving from PINNs toward Deep Energy Models (DEMs), which do not use the strong formulation of the differential equation to be learned but rather a weak variational formulation [25]. This has the advantage of reducing the order of the differential equation and allowing some boundary conditions to be more naturally incorporated into the variational formulation. On the one hand, this facilitates the optimization problem. On the other hand, it accelerates and stabilizes practical training. In contrast to PINNs, numerical integration must be applied with DEMs, which makes training more computationally intensive and makes the solution dependent on the integration schemes used [26]. Since optimization is performed using integrals here, resolving fine-grained features may also pose a problem. In [27], such DEMs were applied in comparison to classical PINNs.

To tackle the partial differential equations, which govern the processes inside the battery, PINNs are already used to estimate the concentration profile inside the particle for a simple, single-phase diffusion equation [28–30]. Here, the chemical potential in the diffusion equation still lacks the contribution of the mechanical stress and therefore is not of a nonlinear nature. PINNs are designed to be an unsupervised machine learning technique that learns from unlabeled data. However, it is also possible to incorporate labeled data in the training process to increase the accuracy [31]. However, a general weakness of PINNs is that they tend to have difficulty learning functions with strong fluctuations, hence very high frequencies [32]. One important aspect of PINNs is the weighting of the different parts in training. For given boundary value problems in engineering, boundary conditions are essential to meeting the physically correct solution. So, the boundary points are of special interest in the training of PINNs and have a high influence on the quality of the PINN's approximation to the real solution [33]. A recent technique for handling the weighting of training for a neural network is the application of self-adaptive PINNs (SA-PINNs) [34, 35].

Regarding batteries modeling, for example, in [12], AI is used to construct the complex polycrystalline structure of the NMC particle. Another example of the use of AI in Batteries modeling is shown in [36], where the state of lithiation in the electrode is estimated. There is relatively little literature on modeling the governing equations for transport phenomena using AI. However, an example in which the chemical and mechanical processes are modeled using a PINN is described in [37]. Although this study determines the diffusion-induced mechanical stresses in 1D within a spherical electrode particle, it does not consider a further damage process.

In this work, the relevant system of equations for the chemical, mechanical, and fracture processes is first derived in order to calculate the concentration, displacement, and damage in a single-particle model. A PINN is then trained to solve the complex system of coupled equations. The solutions of PINN are compared with a numerical method and evaluated.

2 | Theoretical Framework

2.1 | Governing Equations

The system of equations for this work is set up below. It consists of partial differential equations for a chemical process, a mechanical process, and a fracture process. These equations are coupled with each other. A PFM was chosen for the fracture model.

A spherical, radially symmetrical particle is assumed. Furthermore, no depletion or blockage should occur in the electrolyte. Perfect conductivity is assumed in the electrode, which could be achieved by a correct mixture of conductivity additives and the active material [38]. The resulting single-particle model can therefore be set up radially symmetrical, 1D, in spherical coordinates according to Figure 1.

2.1.1 | Chemical Process

Assuming a constant charging rate in a cathode particle, the lithium concentration within the particle is described by the mass balance. This mass balance describes the diffusion process of lithium ions in the active material. The driving force for diffusion is the chemical potential. Because lithium ions have a spatial dimension, their incorporation into the active material causes mechanical strain and, as a result, mechanical stress. The chemical potential in an ideal solid solution under mechanical stress is given by

$$\mu = \mu_0 + RT \ln x - \bar{V} \sigma_h \quad (1)$$

where μ_0 is a constant, R is the universal gas constant, and T is the temperature. The molar fraction of lithium is x , and $\bar{V}$ is the partial molar volume of a lithium ion. The flux of a species can be expressed in terms of mobility M , concentration c and chemical potential μ [9]:

$$J = -Mc \nabla \mu \quad (2)$$

The damage phase field variable d introduced Section 2.1.3 should influence the diffusion coefficient. In the event of defects in the material, diffusion must bypass the defects in the material. Therefore, the diffusion coefficient should be degraded via a degradation function when damage occurs (see Figure 2a). In addition, mechanical tensile stresses and the resulting elastic

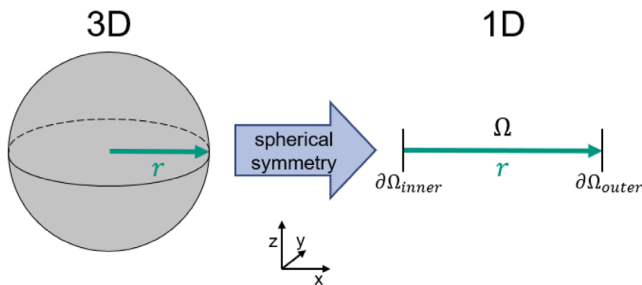


FIGURE 1 | Reduction of the model dimension from spatial 3D cartesian coordinates to 1D spherical coordinates for an active material particle inside the electrode.

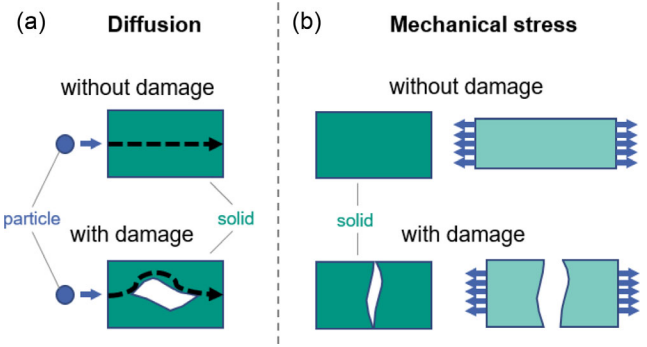


FIGURE 2 | Influence of damage on the effective diffusion coefficient of a material (a) and on the effective tensile stress inside the material (b).

strain energies can no longer be absorbed in damaged materials. Accordingly, the tensile stresses are also degraded (see Figure 2b).

For this reason, a degradation function $w(d)$ is introduced, which maps the effects of damage on diffusion and mechanical stresses. The requirements for the degradation function are summarized in [39] and cover several conditions:

$$w(0) = 1, \quad w(1) = 0, \quad w'(1) = 0 \quad (3)$$

The simplest function that meets these requirements is used here and is declared as

$$w(d) = (1 - d)^2 + k \quad (4)$$

with a small numeric value k added to the actual function in order to avoid division by 0 at all times and therefore increase the stability of the approach.

Substituting Equations (1) into (2) with the introduction of the diffusion coefficient ($D = MRT$) and the degradation of the diffusion coefficient and stresses via Equation (4), yields an expression for the flux

$$J = -w(d)D \left(\nabla c - \frac{\bar{V}c}{RT} \nabla \sigma_h^* \right) \quad (5)$$

in which σ_h^* is the degraded hydrostatic mechanical stress, which is described in Section 2.1.2.

The conservation equation for the species leads to the following balance law:

$$\frac{\partial c}{\partial t} + \nabla \cdot J = 0 \quad (6)$$

To obtain the diffusion equation for a radially symmetric problem, the flux J from Equation (5) is inserted into Equation (6):

$$\begin{aligned} \frac{\partial c}{\partial t} = & \nabla(w(d)D) \\ & \cdot \left(\nabla c - \frac{\bar{V}}{RT} c \nabla \sigma_h^* \right) + w(d)D \left(\nabla^2 c - \frac{\bar{V}}{RT} \nabla c \cdot \nabla \sigma_h^* - \frac{\bar{V}}{RT} c \nabla^2 \sigma_h^* \right) \end{aligned} \quad (7)$$

Since this investigation is limited to a 1D radially symmetric state, a coordinate transformation to spherical coordinates yields the final diffusion equation

$$\frac{\partial \tilde{c}}{\partial t} R_0^2 = \frac{\partial w(d(\tilde{r}))}{\partial \tilde{r}} D \left(\frac{\partial \tilde{c}}{\partial \tilde{r}} - \frac{\bar{V} \tilde{c}}{RT} \frac{\partial \sigma_h^*}{\partial \tilde{r}} \right) + w(d) D \left(\frac{1}{\tilde{r}^2} \frac{\partial}{\partial \tilde{r}} \left(\tilde{r}^2 \frac{\partial \tilde{c}}{\partial \tilde{r}} \right) - \frac{\bar{V}}{RT} \frac{\partial \tilde{c}}{\partial \tilde{r}} \frac{\partial \sigma_h^*}{\partial \tilde{r}} - \frac{\bar{V} \tilde{c}}{RT \tilde{r}^2} \frac{\partial}{\partial \tilde{r}} \left(\tilde{r}^2 \frac{\partial \sigma_h^*}{\partial \tilde{r}} \right) \right) \quad (8)$$

where the concentration and the spatial variable are normalized so that they run in a value range between 0 and 1. This is done by introducing the new variables $\tilde{c} c_{\max} = c$ and $\tilde{r} R_0 = r$. Here $c_{\max}$ is the maximum possible lithium concentration in the particle and R_0 is the radius of the particle. The resulting partial differential Equation (8) thus determines the concentration profile in an active material particle.

2.1.2 | Mechanical Process

Mechanical stresses in the active material particles are determined by the mechanical equilibrium of forces. Since no external forces are present here, the mechanical equilibrium is generally defined by the momentum equation:

$$\nabla \cdot \sigma = 0 \quad (9)$$

Despite the 1D diffusion equation and the assumed radial symmetry, a 3D stress state is used. The stress tensor then consists of a radial component (index r) and two tangential components (index t) that are equivalent due to the symmetry. The mechanical equilibrium can be expressed in spherical coordinates [9]:

$$\frac{d\sigma_r}{dr} + \frac{2}{r}(\sigma_r - \sigma_t) = 0 \quad (10)$$

Since it is assumed that radial symmetry is maintained before and after any deformation, the deformation must be distortion-free. Consequently, the entries in the stress and strain tensors are reduced to the diagonal entries. For the special case of isotropic media assumed here, Hooke's law is generalized by the following relationship between the strains of the strain tensor ϵ and the stresses of the stress tensor σ and extended by a contribution from the stored Li-ions [9, 13]

$$\epsilon_{ij}^{\text{tot}} = \frac{1}{E} ((1 + \nu_p) \sigma_{ij} - \nu_p \sigma_{kk} \delta_{ij}) + \frac{\bar{V}(c - c_{\text{ref}})}{3} \delta_{ij} = \epsilon_{ij}^{\text{ela}} + \epsilon_{ij}^{\text{chem}} \quad (11)$$

where the first term on the right-hand side describes elastic strain. The second term on the right-hand side represents chemical strain due to the stretch of the stored lithium ions. This approach is analogous to the modeling of thermal stresses from [40].

The Kronecker delta δ_{ij} distinguishes the diagonal entries of the tensors from the nondiagonal entries. Young's modulus E and the poisson ratio ν_p are primarily responsible for the elastic strains in the material. The chemical strains are determined by the difference between the concentration and a reference concentration c_{ref} , in which the particle is assumed to be stress-free. Adding both strain contributions together gives the total strain in the material.

Furthermore, for small strains the relationships between the total strains and the displacements u in the material still apply:

$$\epsilon^{\text{tot}} = \frac{1}{2} (\nabla \mathbf{u} + \nabla^T \mathbf{u}) = : \nabla_s \mathbf{u} \quad (12)$$

In 1D spherical coordinates, this relationship is expressed component-wise [9]:

$$\epsilon_r^{\text{tot}} = \frac{du}{dr}, \quad \epsilon_t^{\text{tot}} = \frac{u}{r} \quad (13)$$

The elastic strain energy absorbed by the material during lithiation is calculated

$$\psi = \frac{1}{2} \sigma : \epsilon^{\text{ela}} \quad (14)$$

and if Hooke's law is applied to the stresses σ_{ij} with the Lamé constants G and λ , the elastic strain energy can be obtained as

$$\psi = G((\epsilon^{\text{ela}} : \epsilon^{\text{ela}})) + \frac{\lambda}{2} ((\epsilon^{\text{ela}})_{kk})^2 \quad (15)$$

in which the derivative of the elastic strain energy with respect to strain results in the mechanical stress:

$$\sigma : = \frac{d\psi}{d\epsilon^{\text{ela}}} \quad (16)$$

Figure 2b illustrates that a completely intact material can still absorb all tensile stresses. In extreme cases, if the material is completely damaged, it can no longer absorb tensile stresses except for the stability correction k which leaves the artificial elastic rest energy density. Since the particle can only absorb less elastic strain energy from the tensile strain in a damaged material, in our model, only the positive portion of the elastic strain energy is degraded by the degradation function (4). Accordingly, the elastic strain energy must be decomposed into positive and negative strain components:

$$\psi(\epsilon) = w(d) \psi(\epsilon^{\text{ela},+}) + \psi(\epsilon^{\text{ela},-}) = w(d) \psi^+ + \psi^- \quad (17)$$

As mechanical stress is defined by Equation (16), it is also influenced by the degradation function (4):

$$\sigma = w(d) (2G\epsilon^{\text{ela},+} + \lambda \text{tr}(\epsilon^{\text{ela},+}) I) + (2G\epsilon^{\text{ela},-} + \lambda \text{tr}(\epsilon^{\text{ela},-}) I) \quad (18)$$

For the diffusion equation from Section 2.1.1, the degraded hydrostatic stress $\sigma_h^* = \frac{1}{3} \text{tr}(\sigma)$ in the material is required. This is calculated from the degraded stresses from Equation (18). The distribution into a positive and a negative part of the strains is performed using ramp functions. The small constants k_1, k_2 are added to ensure numerical stability.

$$\epsilon_{ij}^+ = \frac{\epsilon_{ij} + \sqrt{\epsilon_{ij}^2 + k_1}}{2}, \quad \epsilon_{ij}^- = \frac{\epsilon_{ij} - \sqrt{\epsilon_{ij}^2 + k_2}}{2} \quad (19)$$

The aim of the mechanical model is to calculate a displacement u that satisfies mechanical equilibrium. To do this, a relationship between elastic strain and displacement is established from Equations (13) and (11). The elastic strain is then decomposed into positive and negative parts using Equation (19) and inserted into Equation (18). This allows formulations for the radial and

tangential stress components to be determined. Inserting these formulations into the mechanical equilibrium from Equation (10), a nonlinear differential equation for the displacement u is obtained. To obtain more manageable values, u is normalized using the particle radius resulting in $\tilde{u} R_0 = u$.

2.1.3 | Damage Process

The damage parameter d is determined using a phase field model. For a detailed derivation of the used PFM, please refer to [39]. The principle of the derivation is that a functional of the total energy is solved for a stationary point. The functional consists of temporal derivatives of the stored elastic energy, the dissipation energy, and external energies.

In the following, the dot above a variable denotes the time derivative. Assuming sufficient smoothness, the functional of the rate of elastic strain energy $E(\psi(\varepsilon(u(t)), d(t)))$ from Equation (14) is obtained using the chain rule:

$$\dot{E} = \int_{\Omega} \partial_t \psi \, dV = \int_{\Omega} \partial_{\varepsilon} \psi : \dot{\varepsilon} + \partial_d \psi \, \partial_t d \, dV \quad (20)$$

in which the known quantities can be substituted:

$$\dot{E} = \int_{\Omega} \sigma : \nabla_s \dot{\mathbf{u}} - 2(1-d)\psi^+ \dot{d} \, dV \quad (21)$$

Introducing a damage surface density function that applies per unit volume reveals the functionality of the dissipation rate. The damage surface density function is given by [39]:

$$\gamma = \frac{1}{2l} d^2 + \frac{l}{2} |\nabla d|^2 \quad (22)$$

where l describes a characteristic length scale of the damage. The functional of the rate of dissipated energy is also given by [39]:

$$\dot{D} = \int_{\Omega} g_c \, \partial_t \gamma \, dV = \int_{\Omega} g_c \, \partial_d \gamma \, \partial_t d \, dV \quad (23)$$

$$\dot{D} = \int_{\Omega} g_c \left(\frac{1}{l} (d - l^2 \Delta d) \right) \dot{d} \, dV \quad (24)$$

The rate of energy dissipated is therefore proportional to the change in damage $\dot{d}$. For thermodynamic reasons, both γ must increase continuously with increasing damage variable and d must increase continuously with increasing time. Accordingly, $\partial_d \gamma$ and $\dot{d}$ are always greater than or equal to zero. The parameter g_c refers to the energy required to create new surface area [39].

Since no external power influences damage formation, Equations (21) and (24) can be used to establish the total functional, which is the sum of the power that can be stored in the material with the strain energy per unit of time and the sum of the power dissipated by the damage that occurs:

$$\Pi(\dot{\mathbf{u}}, \dot{d}) = \dot{E}(\dot{\mathbf{u}}, \dot{d}) + \dot{D}(\dot{d}) \quad (25)$$

Variational calculus is then used to find the stationary point, which is the equilibrium for the total functional:

$$\{\dot{\mathbf{u}}, \dot{d}\} = \text{Arg} \left\{ \text{stat}_{\dot{\mathbf{u}}, \dot{d} \geq 0} \Pi(\dot{\mathbf{u}}, \dot{d}) \right\} \quad (26)$$

This results in a system of equations:

$$\nabla \cdot \sigma = 0 \quad (27)$$

$$\frac{g_c}{l} (d - l^2 \Delta d) = 2(1-d)\psi^+ \quad (28)$$

Equation (27) is the condition for mechanical equilibrium, which was already introduced as Equation (9). Equation (28) determines the development of the phase field damage parameter d and corresponds to the damage equation of the system. Equation (28) shows that positive elastic strain energy is the source of damage formation.

To ensure that the damage is not reduced again, a history field is introduced, which displays the maximum positive elastic strain energy that has occurred so far [39]:

$$\mathcal{H}(\mathbf{u}, t) = \max_{s \in [0, t]} \psi(\varepsilon^{\text{ela},+}(\mathbf{u}, s)) \quad (29)$$

This then results in the final damage equation for the 1D radially symmetric case:

$$\frac{g_c}{l} \left(d - \frac{l^2}{R_0^2} \left(\frac{\partial^2 d}{\partial \tilde{r}^2} + \frac{2}{\tilde{r}} \frac{\partial d}{\partial \tilde{r}} \right) \right) = 2(1-d)\mathcal{H} \quad (30)$$

2.2 | Boundary and Initial Conditions

At the start of lithiation, the initial concentration is set to the discharged reference concentration in the particle. Because it is assumed that the reference concentration is stress-free, the displacement in the particle is set to zero at the start. It is possible that damage is already present in the particle. Accordingly, the initial condition for the damage-phase field can be freely selected:

$$\tilde{c}(r, t=0) = \frac{c_{\text{ref}}}{c_{\text{max}}}, \quad \tilde{u}(r, t=0) = 0, \quad d(r, t=0) = d_{\text{ini}}(r) \quad (31)$$

Since the spherical particle is simulated in 1D and radially symmetrical, the only spatial axis runs along the radius from the center of the particle to the surface edge of the particle. At the center of the particle, the displacement is set to zero. For the other fields, a symmetrical boundary condition was chosen:

$$\frac{\partial \tilde{c}}{\partial \tilde{r}} = 0, \quad \tilde{u} = 0, \quad \frac{\partial d}{\partial \tilde{r}} = 0 \quad \text{on } \partial \Omega_{\text{inner}} \quad (32)$$

Equation (28) shows that damage in the particle only occurs under tensile stress. Tensile load in the particle is determined by the concentration and displacement outer boundary conditions. Figure 3 illustrates the various combinations.

Due to galvanostatic operation, for delithiation and lithiation a constant current is applied at the surface of the particle to the chemical flux where i_n is the surface-specific current:

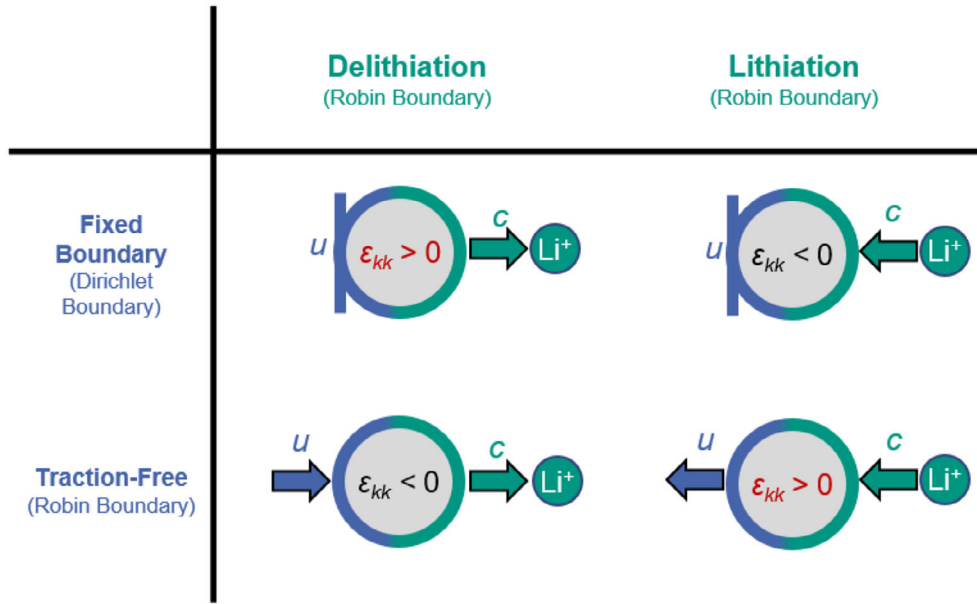


FIGURE 3 | Choice of boundary conditions for tensile or compressive load in the particle. The trace of the strain tensor ϵ_{kk} is a measure for expansion or compression. Delithiation and lithiation are referring to boundary conditions for the concentration profile. Fixed boundary means $u = 0$ and traction-free means the radial stress equals zero at the boundary. Tensile loads occur for delithiation and fixed boundary or lithiation and traction-free boundary. Compressive loads occur for delithiation and traction-free or lithiation and fixed boundary.

$$w(d)D \left(\nabla c - \frac{\Omega c}{RT} \nabla \sigma_h \right) = \frac{i_n}{F} \quad \text{on } \partial \bar{V}_{\text{outer}} \quad (33)$$

This results in

$$\frac{\partial \tilde{c}}{\partial \tilde{r}} - \frac{\Omega}{RT} \tilde{c} \frac{\partial \sigma_h}{\partial \tilde{r}} = \frac{R_0^2 (c_{\max} - c_{\text{ref}}) \dot{C}}{10800 w(d) D c_{\max}} \quad \text{on } \partial \bar{V}_{\text{outer}} \quad (34)$$

with the introduction of the C-rate $\dot{C} = 3600I/Q_{\max}$ using the current I and the maximum capacity $Q_{\max} = V(c_{\max} - c_{\text{ref}})F$. Lithiation and delithiation are distinguished solely by the sign of the applied current I .

For the displacement, according to Figure 3, either a Dirichlet boundary condition ($u = 0$), which does not allow displacement at the boundary, or a Robin boundary condition in the displacement ($\sigma_r = 0$), which guarantees radially stress-free displacement at the boundary, can be applied at the outer boundary. In this study, the case of lithiation with a traction-free boundary is investigated. The damage field at the outer boundary condition is set to have a zero gradient in the radial direction.

3 | PINN

A common representative of machine learning methods for solving differential equations is the PINN. In comparison with conventional numerical methods, approaches derived from the class of neural networks have certain advantages, including independence from a computing grid and the requirement for recursive calculations. In order to exploit these advantages, especially for complex problems and to better understand PINNs, this study uses a PINN to solve the proposed nonlinear coupled differential equation.

3.1 | General PINN

In a standard Artificial Neural Network (ANN), the loss function consists of the error between the expected and observed output, which is calculated using given labeled training data. The trainable parameters of the network were optimized to minimize this loss function. Using PINNs, the error is primarily determined from the residue of a given differential equation with its boundary conditions and initial conditions. For this purpose, the necessary (partial) derivatives of the output with respect to the input are determined with an algorithm called backpropagation. A data-based standard error can be added to the loss function as desired. For a general boundary value problem for a general function $f: \mathbb{R}^n \rightarrow \mathbb{R}^m$, Equation (35) applies, in which $\mathcal{L}[\cdot]$ is a differential operator and $\mathcal{B}[\cdot]$ is a boundary operator:

$$\begin{cases} \mathcal{L}[f](x) = g(x) & \forall x \in \Omega \\ \mathcal{B}[f](x) = h(x) & \forall x \in \partial\Omega \end{cases} \quad (35)$$

As an advantage over ANNs, PINNs learn not only the values at the data points but also the slopes and curvatures of these values by specifying the differential equation. This ensures that areas between the data points can be learned better and more smoothly. The function f_θ is the approximation of the PINN of the function f with the trainable parameters θ . Figure 4 shows the basic architecture of a PINN.

Since the level of the solution to the differential equation is usually only described by the boundary conditions or initial conditions, it is essential to include this information in PINN as well. In Equation (36), the loss function from which the PINN learns has different components:

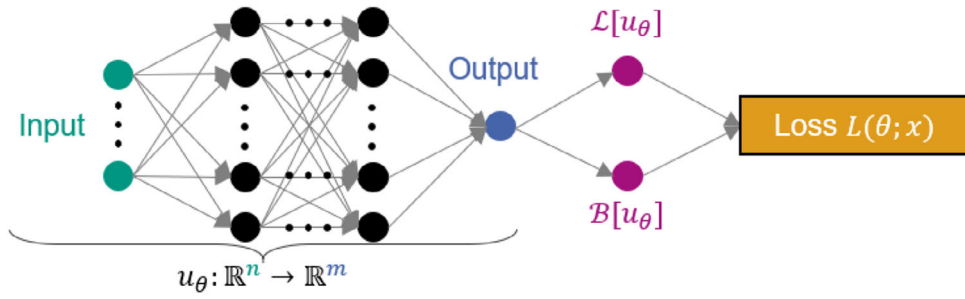


FIGURE 4 | Scheme of a PINN. The differential and boundary operators can be calculated via backpropagation.

$$L = \frac{\lambda_{ip}}{N_{ip}} \sum_{i=1}^{N_{ip}} L_{ip} + \frac{\lambda_{bc}}{N_{bc}} \sum_{i=1}^{N_{bc}} L_{bc} + \frac{\lambda_{ic}}{N_{ic}} \sum_{i=1}^{N_{ic}} L_{ic} \left(+ \frac{\lambda_{data}}{N_{data}} \sum_{i=1}^{N_{data}} L_{data} \right) \quad (36)$$

The total loss consists of the errors of the interior points (ip), errors due to the boundary points (bc), and errors due to the initial conditions (ic). Even though the classic PINN method does not require labeled data points, if applicable, labeled data points (data) may also be used to simplify the training process. The data sets of the individual components have the sizes N_{ip} , N_{bc} , N_{ic} or N_{data} , respectively. The different errors are weighted using the factors λ_i . This weighting is crucial for the accuracy of PINN [33]. If the boundaries are weighted too heavily, especially in the case of Dirichlet boundaries, it is possible that only this low-frequency solution will be learned for the entire domain. In order to avoid focusing exclusively on the weighting between boundary points, initial points, and interior points, data points are helpful here as they can determine the level of the differential equation without learning a low-frequency pseudo-solution to the problem. The weighting between equation training and data training can be determined here by λ_{data} .

The error for the differential equation or boundary condition is calculated from the residual of Equation (35). To make the error independent of the sign, it is squared. An error for the initial conditions works in the same way. To quantify the error at each data point, the squared difference between the observed and expected outputs is used.

As with ANN, the actual training of a PINN involves an update step for the trainable parameters θ . This involves gradient-based optimization, usually using a gradient descent method or a modification thereof.

3.2 | Self-Adaptive Weighting

To address the difficulty of learning high-frequency solutions for PINN and determine a reasonable weighting of the error contributions, this paper uses the SA-PINN presented by [34] instead of the standard PINN. This approach counteracts those two problems of conventional PINN during training. Each data point here gets its own weight. So, compared to Equation (36), the new loss function differs from Equation (36):

$$L = \frac{1}{N_{ip}} \sum_{i=1}^{N_{ip}} \lambda_{ip,i} L_{ip,i} + \frac{1}{N_{bc}} \sum_{i=1}^{N_{bc}} \lambda_{bc,i} L_{bc,i} + \frac{1}{N_{ic}} \sum_{i=1}^{N_{ic}} \lambda_{ic,i} L_{ic,i} + \frac{1}{N_{data}} \sum_{i=1}^{N_{data}} \lambda_{data,i} L_{data,i} \quad (37)$$

This means that training points that have not been learned well will be given greater consideration in the next training step. The idea is to improve the balancing of the individual terms in the loss function and thus focus more on the error contributions that are poorly trained. Using the SA-PINN also creates an update step for the weights besides the standard update step for the trainable parameters. The weights of the data points that cause a large error should be maximized via gradient ascent.

4 | Numerical Procedure

4.1 | Generation of Training Data

Since the arguments in favor of labeled data-driven training are obvious, in this work, labeled data is used for the training process. It is experimentally unfeasible to obtain data on the concentration, displacement, and damage to single particles. So, numerical methods are used to generate labeled data, some of which was incorporated into the actual training process as a term for a labeled data loss in the total loss function. However, the majority of these labeled data points were ultimately used to validate the trained model. Since no complex unstructured grids are used in this 1D simulation in spherical coordinates, the finite difference method (FDM) was chosen as the numerical method. The fields were solved sequentially. A visual summary of the numerical process is shown in Figure 5.

The concentration Equation (8) was implicitly discretized using a Crank–Nicholson method, and the resulting tridiagonal system of equations was solved using a Thomas algorithm. Since the displacement equation from Section 2.1.2 exhibits a nonlinear behavior, a multidimensional Newton–Raphson method is used here to iteratively minimize the residuals of the displacement equation. The elastic strain energy was calculated at each grid point from the converged displacement profile, and finally, the damage Equation (30) was solved using the efficient Thomas algorithm again. The data in Table 1 for NMC622 deliver the used material parameters.

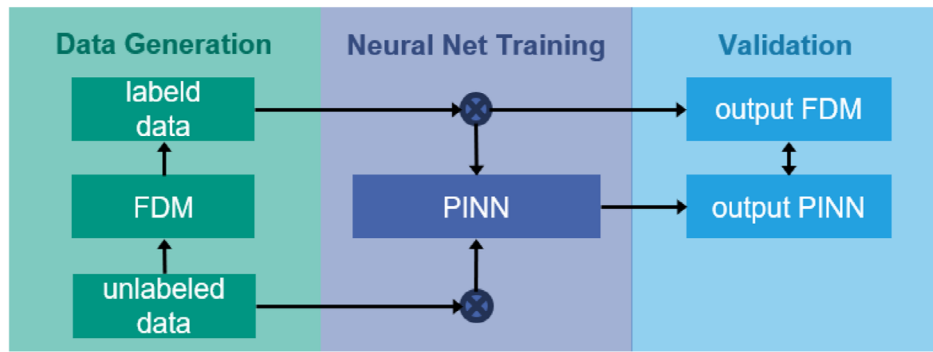


FIGURE 5 | Visualization of the generation of training data, the learning process and the validation of the PINN. The term “unlabeled data” here refers to a dataset of input parameters without the corresponding results. In contrast, “labeled data” includes these results in the dataset.

TABLE 1 | Material parameters of NMC622.

Parameter	Value	Unit	Source
$c_{\max}$	50 451	molm^{-3}	[41]
c_{ref}	18 409.57	molm^{-3}	[41]
D	$4.3032\text{e}-14$	m^2s^{-1}	[41]
R_0	$500\text{e}-9$	m	Chosen
$\bar{V}$	$1.8\text{e}-6$	$\text{m}^3\text{mol}^{-1}$	[13]
E	$10\text{e}9$	$\text{kgm}^{-1}\text{s}^{-2}$	[42]
v_p	0.3	—	[42]
T	293.15	K	Chosen
l	$5\text{e}-9$	m	Chosen

For the FDM method, the grid and time step were refined until the solution no longer changed. Data for training the neural network was then generated using the optimized grid widths and time steps.

4.2 | PINN Configuration

The PINN used is shown in Figure 6 with its input and output parameters. Since the concentration, displacement, and damage

fields are based on the previous damage field due to the feedback, the previous damage field must be specified in the form of the absolute value and in the form of the local gradient at every point under consideration as an input parameter. In addition, the C-rate, which already appears in Equation (34), is specified as an input parameter. This parameter controls how much lithium flows into the particle from outside via the Robin boundary condition. It is therefore decisive for the slope in the concentration profile and, accordingly, also decisive for displacements and damage in the particle. Another essential input parameter is the radial coordinate of the 1D fields, which is specified in the parameter $\tilde{r}$.

Looking at the equations for the concentration, displacement and damage fields, the last input parameter, which is missing from the concentration Equation (8), is the time t . Due to galvanostatic operation, which assumes a constant charging or discharging rate, the time dependence is replaced in this work by a dependence of the state of charge (SoC) via Equation (38):

$$-\Delta\text{SoC} = \Delta\text{DoD} = \frac{\Delta t \cdot I}{Q_{\max}} = \frac{\Delta t \cdot \dot{C}}{3600} \quad (38)$$

Since cathode active material particles are considered here, the depth of discharge (DoD) is a more convenient measure than the

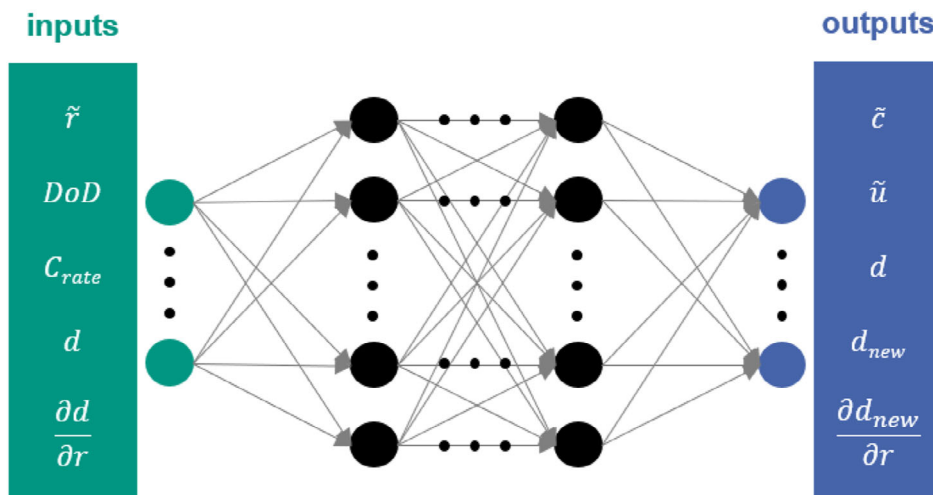


FIGURE 6 | Input and Output Parameters of the proposed PINN. The PINN consists of three hidden layers with 30 neurons each. Therefore, it is considered to be a shallow neural network.

SoC. Accordingly, time derivatives can be substituted with derivatives based on the *DoD* using the chain rule. The *DoD* is zero for a completely delithiated cathode active material particle, one for a completely lithiated one, and is defined by Equation (39):

$$DoD = \frac{\int_{\Omega} (c(r, t) - c_{Li,ref}) dV}{\int_{\Omega} (c_{Li,max} - c_{Li,ref}) dV} \quad (39)$$

The output of the PINN is the normalized concentration field $\tilde{c}$ and the normalized displacement field $\tilde{u}$. In addition, the currently calculated damage field d is returned. Since the introduction of the history field $\mathcal{H}$ in Equation (29) is intended to ensure that d cannot become smaller, the PINN compares the input d with the output d and returns the larger value in d_{new} . Since the derivative of d_{new} is also relevant as an input for a possible next time step, it is also returned as output.

Three hidden layers, each with 30 neurons, were used to configure the PINN. This results in a total of 2313 trainable parameters. A $\tanh$ was used as the activation function in this work [43].

4.3 | PINN Training

In addition to the generated data points, the equations of the differential equations, boundary conditions, and initial conditions were used for training. To avoid having to determine optimal individual weights between the individual parts, the self-adaptive weighting discussed in Section 3.2 was used. The self-adaptive weights were optimized by a standard stochastic gradient ascent optimizer. A large part of the generated data was used exclusively for validation. The data sets for the individual training data are shown in Table 2. Mini-batches were used to train more efficiently. The mini-batch size was chosen so that all different types of training data were included for each training step. At each training step, the trainable parameters of the PINN are updated.

Since the gradients of the various training sets vary greatly in magnitude, the gradients were clipped and rescaled before being added together linearly. In doing so, those gradients that have a greater influence on the quality of the solution, such as the gradients of the data points and the boundary conditions, were given a greater weight than the gradients of the differential equation or those of the initial conditions. Stochastic gradient descent was used as the optimizer in combination with a line search. This allows the learning rate to be chosen flexibly so that the residuals of the individual training parts are minimized as much as possible per update step.

TABLE 2 | Number of data points and mini-batch sizes for the PINN training.

Parameter	Value	Mini-batch size
N_{data}	6000	100
N_{ip}	1500	25
N_{bc}	1500	25
N_{ic}	1500	25
$N_{validation}$	10 000	—

5 | Results and Discussion

5.1 | PINN Training Error

The training of the PINN used is shown in Figure 7. The learning epochs were plotted on the horizontal axis. One epoch is complete when training with all training data points has been completed once. The used PINN was trained for 7500 epochs, meaning that all training data was shown to PINN 7500 times. The vertical axis shows the validation mean absolute error (MAE). The MAE describes the error between the output of the PINN and the desired output and is defined in Equation (40):

$$MAE_x = |x_{PINN} - x_{real}| \quad (40)$$

Since the PINN in the untrained state returns a value in the order of magnitude 10^{-1} due to the $\tanh$ activation functions, one initially sees a larger MAE for the smaller fields $\tilde{u}$ and d . For all sizes, one can see that the MAE decreases most significantly during the first epoch. The remaining training after the first epoch until the last epoch then reduces each field by about one order of magnitude.

It is apparent that the MAE for $\tilde{c}$ is the highest. This is because $\tilde{c}$ also has the highest values in the order of magnitude 10^{-1} compared to d and $\tilde{u}$ (both approx. 10^{-2}). Presumably, the PINN learns the field $\tilde{u}$ better than d because d is explicitly coupled to u via $\mathcal{H}$. Therefore, the field u must also be learned for the field d , which increases the error. In general, the frequencies of the MAE indicate that the error is even with the used line search not improved in every step due to the differential equation residues. Despite the fluctuations in training, the general trend of PINN training is toward error minimization. Figure 8 further shows the relative L1 residuals of the training points, plotted between the initial residual at the start of training and the final residual after training. It can be seen that not all terms of the loss function could be minimized to the same extent during training. This is primarily because the loss function consists of many different components, making the minimization problem highly sensitive to the weighting of the individual loss terms. Nevertheless, all terms were minimized, which is primarily attributable to a suitable initial weighting as well as the use of an SA-PINN. To obtain exact concentration, displacement, and damage fields from the PINN, the terms in the loss function that must be given greater weight

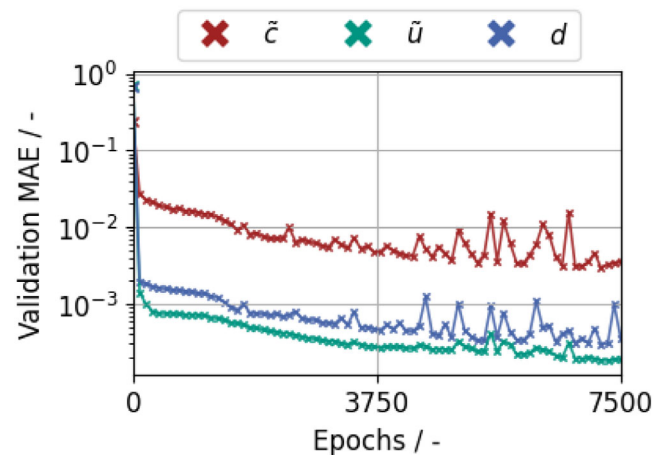


FIGURE 7 | Training process of the PINN.

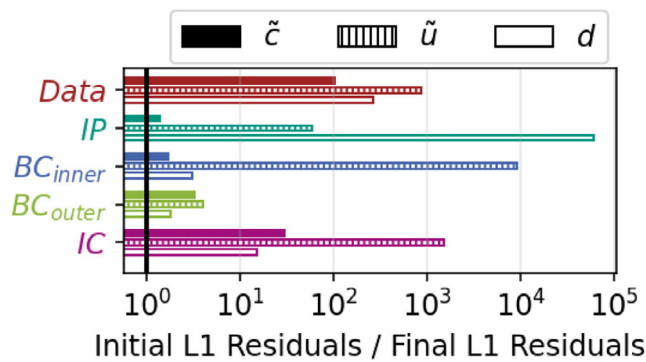


FIGURE 8 | Ratio of the L1 residuals between the initial values and the final values of the training process for all terms of the loss function. All values greater than one indicate an improvement over the initial residuals.

vary depending on the field. For example, in the damage field, the differential equation is more profile-determining than the natural boundary conditions. Another example is the concentration field, where the trivial solution $\tilde{c}=0$ would satisfy the differential equation everywhere, but the boundary conditions prevent this trivial solution, which is also favored by spectral bias. Accordingly, the boundary conditions must be weighted more heavily here. Since the differential equation of the concentration field alone can be satisfied by various profiles, the residual here decreased less during training compared to that in the other fields. The initial estimate from the PINNs Xavier-Uniform initializer already satisfied the differential equation rather well, despite its physical nonsensicality. This means that while the final state of the parameters is not reflected in the residual to the same extent. Even with the optimization of the initial weighting and the use of a SA-PINN, it is clear how challenging the nonlinear and coupled minimization problem is, and there is still further potential for optimization.

When comparing the PINN and ANN training in Table 3, it is noticeable that the values only differ in terms of stochastic

TABLE 3 | Comparison of the validation MAE's of the different fields after the 7500th epoch of training.

Field	PINN MAE	ANN MAE
$\tilde{c}$	$3.14 \cdot 10^{-3}$	$3.34 \cdot 10^{-3}$
$\tilde{u}$	$1.87 \cdot 10^{-4}$	$1.71 \cdot 10^{-4}$
d	$2.87 \cdot 10^{-4}$	$3.08 \cdot 10^{-4}$

probability distribution. Thus, no better-performing network can be identified on the validation dataset. However, the validation dataset was created with the FDM on the grid points that were also used for training. By learning the slopes and curvatures of the profiles using PINN, the interpolated data points should be better represented by PINN. The same applies to extrapolation as ANNs fail to extrapolate beyond the learned boundaries [44]. By specifying and teaching the differential equation in PINNs, extrapolation can be less of a problem here [45].

5.2 | PINN and ANN

Figure 9 illustrates the advantages of a PINN over an ANN. Three different concentration profiles can be seen, which are labeled in the legend according to their respective calculation methods. The FDM solution, as a numerical solution, represents the validation solution for the two different concepts of neural networks.

When comparing the concentration profiles, the PINN solution shows a significantly more realistic slope and curvature than the ANN solution. This is due to the incorporation of the boundary value problem during training. Therefore, it results in a more realistic progression of the solutions. This is particularly important when the gradients of the solution are of interest, for example, to infer the mechanical stresses from the learned displacement in the cathode-active material particle.

5.3 | PINN Accuracy

The following heat maps in Figure 10 show lithiation with a C-rate of three. As validation for the PINN, the solution of the numerical FDM method is also plotted in the second column for comparison. In addition, the MAE between the PINN and FDM solutions according to Equation (40) is plotted in the third column of the plot. The first row of Figure 10 considers the $\tilde{c}$ field, the second row the $\tilde{u}$ field, and the third row the d field.

The heat maps show the complete lithiation state of the one-dimensional model for 3C. The fields determined by PINN show good agreement with the numerically calculated fields of the FDM method. The respective errors of the three fields are all two orders of magnitude below the orders of magnitude of the respective fields, which is considered accurate enough. Looking at the error plot, there is no preferred error spot visible, neither at the boundary nor inside the domain. This indicates a good

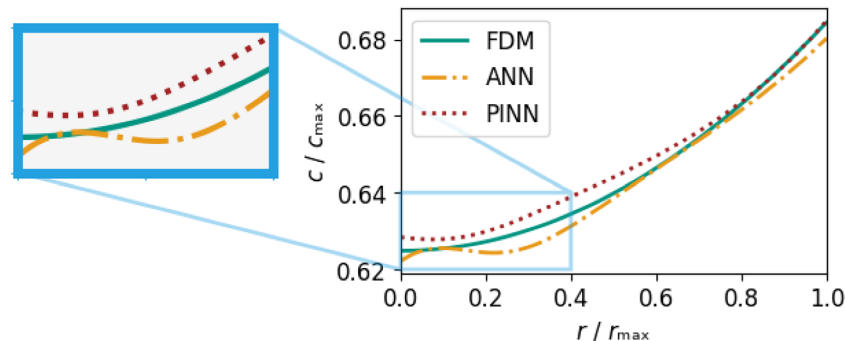


FIGURE 9 | Comparison between PINN and ANN with a numerical FDM solution. The plot corresponds to a 1C current.

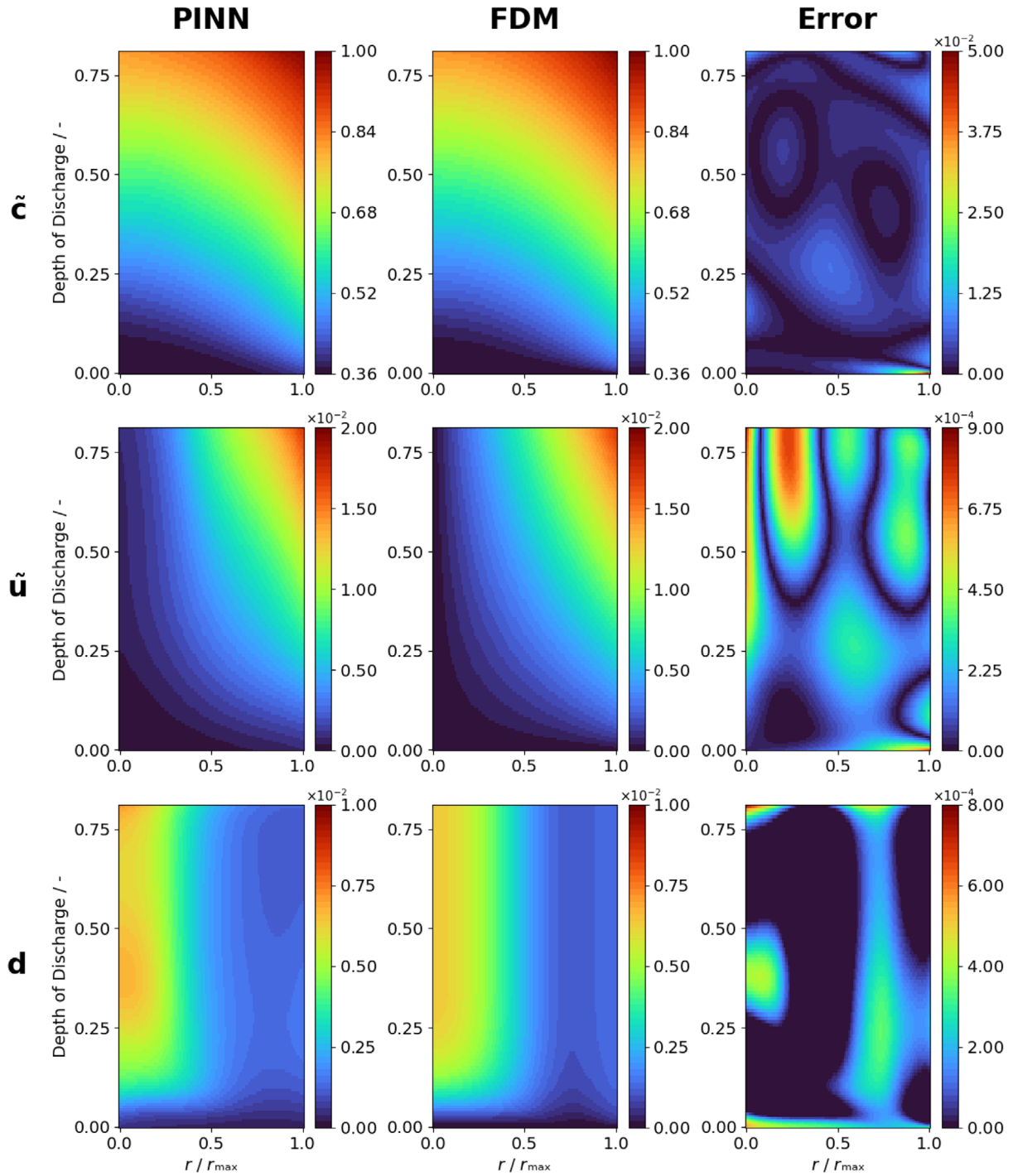


FIGURE 10 | Validation of the proposed PINN with a FDM simulation. The particle is lithiated with a 3C current. The first row is related to the normised concentration field, the second row to the normalized displacement field and the third row to the damage-phase field. In the first column are the solutions of the PINN model, in the second column the solution of a numerical FDM solution and the absolute error between the PINN solution and the FDM solution is displayed on the right column.

balance between learned boundary points and interior points and is based on the use of SA-PINN.

5.4 | Investigation of Lithiation States

One major advantage of substituting the numerical model with a PINN is certainly the explicit nature of the PINN. This means

that it is no longer necessary to recursively run through all previous states in order to examine certain parameter states. Furthermore, due to its independence from any calculation grids, any parameter state can be called up by PINN. These two properties facilitate in the investigation of the maximum damage occurring depending on the applied C-rate and the DoD in Figure 11a and depending on the applied C-rate and the normalized radial coordinate in Figure 11b, respectively.

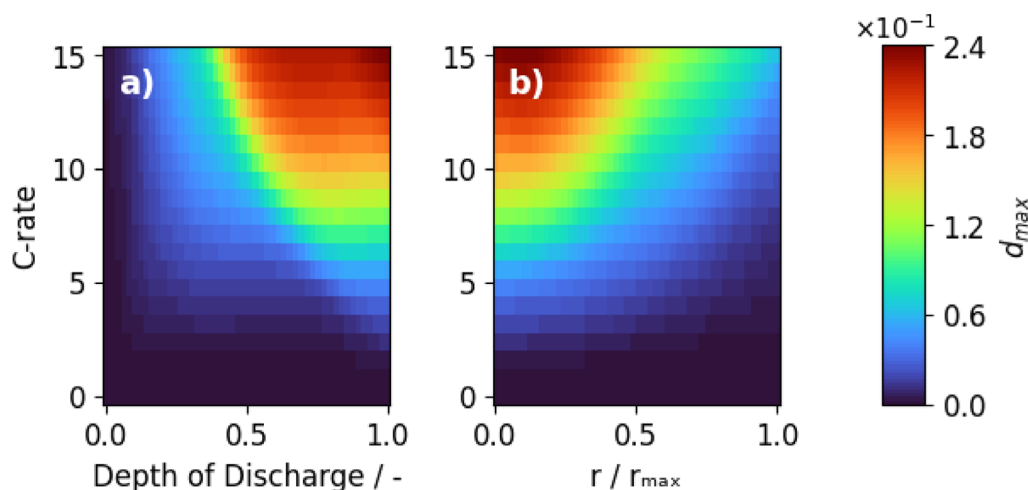


FIGURE 11 | Maximum damage which occurs during lithiation depending on the C-rate and the DoD in (a) or on the C-rate and the radial coordinate in (b). It can be seen that, with a higher C-rate and a higher state of lithiation, the damage increases and is dominant at the center of the particle.

Figure 11a shows the maximum damage value along the radial component plotted against the DoD and the C-rate. It can be seen that the maximum damage in the particle increases with increasing C-rate and increasing DoD. The areas below approximately 5C are relatively uncritical. This is consistent with the literature, which also claims that lithiations below 5C do not cause significant damage to NMC [46]. Otherwise, higher DoD ranges and higher currents result in greater damage to the particle. In [47], high DoD ranges and high currents are also identified as risk factors for batteries. However, in this case, the entire battery was considered, not just the cathode-active material particles. At higher C-rates, the damage can be limited by restricting the state of lithiation (here DoD).

In Figure 11b, the dependence of DoD and the radial component is reversed. Here, the radial coordinate is shown on the horizontal axis, and the coloring indicates the maximum damage for all DoD states at the respective point in the particle above the C-rate. Here, too, it can be seen that currents below 5C are relatively harmless for the model's damage prediction. At higher currents, the model predicts generally high damage with maximum values inside the particle, which has also been shown in the literature for lithiations [48]. In order to minimize the predicted damage, constructive methods that can structurally strengthen the particle inside would be promising.

The trend that high currents lead to high damage can also be seen from the model equations in Section 2. A higher C-rate during lithiation inevitably increases the concentration gradient in the particle. A higher concentration gradient leads to greater strains and ultimately to higher strain energies, which are the driving force behind the damage.

5.5 | General Notes Regarding the Model

In the following, the model will be discussed in general terms and with regard to its informativeness. In our three-dimensional world, damage actually occurs in the form of defects or cracks. Nevertheless, the model presented calculates damage in a simplified, one-dimensional manner. This leads to problems, primarily

because NMC particles are commonly found as secondary particles, and cracks often occur at grain boundaries [5]. The distinction between grain boundaries and primary particles is not yet taken into account in this model. In addition, there is no possibility of experimental validation as in-operando insights into the lithiation process are hard to obtain. Furthermore, individual parameters such as Young's modulus of NMC vary over several orders of magnitude in the literature studies. Such material parameters are difficult to determine, and due to its complexity, the model is very sensitive to changes in parameters, which is why quantitative statements from the model should be treated with caution.

Nevertheless, modeling purely theoretical equations helps to better understand the process of intercalation-induced damage in single particles. The simulation approach allows for a better understanding of the complex coupling of individual material and process parameters. As shown in the results section, the qualitative statements of the model correspond well with the results presented in the literature. The model can therefore be used to provide an initial assessment of where and to what extent damage may occur.

The use of PINNs as a solution method demonstrates the potential of data-driven models to solve even complex and highly coupled differential equations. This behavior becomes particularly interesting when conservative numerical solution methods reach the limits of computing power due to an increase in the dimensionality and complexity of the model.

6 | Conclusion

A model was presented that estimates the intercalation-induced damage to single particles. The model consists of a chemical, mechanical, and damage process. For each process, a differential equation with a respective target field variable was derived. The chemical process results in a concentration profile along the particle, the mechanical process in a displacement profile, and the damage process in a damage profile. Stresses and strains can be

further calculated from the displacement. The damage has the character of a phase field and therefore takes values between 0 and 1, where 0 indicates a completely intact material, and 1 indicates a completely damaged material. For simplicity, the model was kept rotationally symmetric in 1D. Degradation feedback on solid diffusion and absorbable mechanical stresses is obtained from the damage model in the form of a degradation function. Only the lithiation of the particle with a freely movable surface was investigated, which, in combination, leads to a tensile stress case that is susceptible to damage. A PINN was used to solve the three complex coupled differential equations, whereby the standard neural network was extended with the residual of the differential equations in order to expand the model to include physical interpretability. For training the PINN, in addition to the points for the differential equation, the boundary conditions, the initial conditions, and previously calculated data points were also used. Despite the trained PINN showing similarly good validation results as a comparable trained ANN, the physics in the PINN provide better interpolation and for some PDEs, even extrapolation capabilities. It can be seen that the solution of the PINN is smoother than the solution of the ANN due to the specification of the differential equation. The errors between the PINN and the numerical comparison method show that the deviations between the two quantities are approximately two orders of magnitude smaller than the respective fields. Even though the model can only make qualitative rather than quantitative statements due to its sensitivity to inaccurately determinable parameters and its simplifications, it is consistent with the literature. A lithiation current of less than 5C causes hardly any damage to the particle. In accordance with the literature, damage during lithiation occurs primarily at high currents and high DoD ranges inside the particle.

This work has shown that PINN is indeed capable of learning a complex problem consisting of several coupled differential equations in order to provide a fast-executable and grid-free alternative to conventional numerical methods for solving such problems. This becomes particularly relevant when numerical solvers reach the limits of their computing power due to the addition of further dimensions.

Acknowledgments

The authors gratefully acknowledge the funding and support by the German Research Foundation (DFG Deutsche Forschungsgemeinschaft) within the research training group SiMET under the project number DFG GRK 2218/3.

Open Access funding enabled and organized by Projekt DEAL.

Funding

This study was supported by Deutsche Forschungsgemeinschaft (GRK 2218/3).

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

1. European Commission. Joint Research Centre, *Clean Energy Technology Observatory, Batteries for Energy Storage in the European Union: Status Report on Technology Development, Trends, Value Chains and Markets: 2022* (Publications Office, 2023).
2. F. Wu, J. Maier, and Y. Yu, "Guidelines and Trends for nExt-generation Rechargeable Lithium and Lithium-ion Batteries," *Chemical Society Reviews* 49, no. 5 (2020): 1569.
3. J. S. Edge, S. O'Kane, R. Prosser, et al., "Lithium Ion Battery Degradation: What You Need to Know," *Physical Chemistry Chemical Physics: PCCP* 23, no. 14 (2021): 8200.
4. B. Xu, D. Qian, Z. Wang, and Y. S. Meng, "Recent Progress in Cathode Materials Research for Advanced Lithium Ion Batteries," *Materials Science and Engineering: R: Reports* 73, no. 5-6 (2012): 51.
5. K. Mao, Y. Yao, Y. Chen, et al., "Fracture Mechanisms of NCM Polycrystalline Particles in Lithium-Ion Batteries: A Review," *Journal of Energy Storage* 84 (2024): 110807.
6. M. Jiang, D. L. Danilov, R. Eichel, and P. H. L. Notten, "A Review of Degradation Mechanisms and Recent Achievements for Ni-Rich Cathode-based Li-ion Batteries," *Advanced Energy Materials* 11, no. 48 (2021): 2103005.
7. B. Wu and W. Lu, "Modeling the Effect of Monomer Conformational Change on the Early Stage of Protein Self-Assembly into Fibrils," *The Journal of Physical Chemistry C* 121, no. 35 (2017): 19022.
8. P. Stein and B. Xu, "3D Isogeometric Analysis of Intercalation-Induced Stresses in Li-ion Battery Electrode Particles," *Computer Methods in Applied Mechanics and Engineering* 268 (2014): 225.
9. X. Zhang, W. Shyy, and A. Marie Sastry, "Numerical Simulation of Intercalation-Induced Stress in Li-Ion Battery Electrode Particles," *Journal of the Electrochemical Society* 154, no. 10 (2007): 910.
10. X. Cheng and M. Pecht, "In Situ Stress Measurement Techniques on Li-ion Battery Electrodes: A Review," *Energies* 10, no. 5 (2017): 591.
11. F. Pistorio, D. Clerici, F. Mocera, and A. Somà, "Review on the Numerical Modeling of Fracture in Active Materials for Lithium Ion Batteries," *Journal of Power Sources* 566 (2023): 232875.
12. W.-X. Chen, J. M. Allen, S. Rezaei, et al., "Cohesive Phase-Field Chemo-Mechanical Simulations of Inter- and Trans-Granular Fractures in Polycrystalline NMC Cathodes via Image-Based 3D Reconstruction," *Journal of Power Sources* 596 (2024): 234054.
13. A. M. Boyce, E. Martínez-Pañeda, A. Wade, et al., "Cracking Predictions of Lithium-Ion Battery Electrodes by X-Ray Computed Tomography and Modelling," *Journal of Power Sources* 526 (2022): 231119.
14. V. Venkatasubramanian, "The Promise of Artificial Intelligence in Chemical Engineering: Is It here, Finally?," *AIChE Journal* 65, no. 2 (2019): 466.
15. F. Rhein, L. Hibbe, and H. Nirschl, "Hybrid Modeling of Hetero-Agglomeration Processes: A Framework for Model Selection and Arrangement," *Engineering With Computers* 40, no. 1 (2024): 583.
16. O. Zhai, N. Ehret, F. Rhein, and M. Gleiss, "Enhancing Decanter Centrifuge Process Design with Data-driven Material Parameters in Multi-compartment Modeling," *Journal of Advanced Manufacturing and Processing* 6, no. 3 (2024): e10179.
17. M. L. Thompson and M. A. Kramer, "Modeling Chemical Processes Using Prior Knowledge and Neural Networks," *AIChE Journal* 40, no. 8 (1994): 1328.
18. M. Raissi, P. Perdikaris, and G. Karniadakis, "Physics-Informed Neural Networks: A Deep Learning Framework for Solving Forward and Inverse Problems Involving Nonlinear Partial Differential Equations," *Journal of Computational Physics* 378 (2019): 686.

19. M. Raissi, A. Yazdani, and G. E. Karniadakis, "Hidden Fluid Mechanics: Learning Velocity and Pressure Fields from Flow Visualizations," *Science* 367, no. 6481 (2020): 1026.
20. F. Sahli Costabal, Y. Yang, P. Perdikaris, D. E. Hurtado, and E. Kuhl, "Physics-Informed Neural Networks for Cardiac Activation Mapping," *Frontiers in Physics* 8 (2020): 42.
21. Z. Mao, A. D. Jagtap, and G. E. Karniadakis, "Physics-Informed Neural Networks for High-Speed Flows," *Computer Methods in Applied Mechanics and Engineering* 360 (2020): 112789.
22. I. Depina, S. Jain, S. Mar Valsson, and H. Gotovac, "Application of Physics-Informed Neural Networks to Inverse Problems in Unsaturated Groundwater Flow," *Georisk: Assessment and Management of Risk for Engineered Systems and Geohazards* 16, no. 1 (2022): 21.
23. Y. Chen, L. Lu, G. E. Karniadakis, and L. Dal Negro, "Physics-Informed Neural Networks for Inverse Problems in Nano-Optics and Metamaterials," *Optics Express* 28, no. 8 (2020): 11618.
24. A. D. Jagtap, Z. Mao, N. Adams, and G. E. Karniadakis, "Physics-Informed Neural Networks for Inverse Problems in Supersonic Flows," *Journal of Computational Physics* 466 (2022): 111402.
25. E. Samaniego, C. Anitescu, S. Goswami, et al., "An Energy Approach to the Solution of Partial Differential Equations in Computational Mechanics via Machine Learning: Concepts, Implementation and Applications," *Computer Methods in Applied Mechanics and Engineering* 362 (2020): 112790.
26. X. Wang, J. Zhao, Z.-Y. Yin, and X. Zhuang, "Failure Mechanisms and Resolution in Deep Energy Method," *International Journal of Mechanical Sciences* 313 (2026): 111278.
27. Y. Wu, X. Zhao, Y. Li, K. Zhang, and F. Yang, "Domain Decomposed Physics-Informed Neural Networks for Coupled Diffusion and Deformation in Bilayer Electrode Materials," *Journal of Energy Storage* 141 (2026): 119199.
28. S. Singh, Y. E. Ebongue, S. Rezaei, and K. P. Birke, "Hybrid Modeling of Lithium-Ion Battery: Physics-Informed Neural Network for Battery State Estimation," *Batteries* 9, no. 6 (2023): 301.
29. F. J. Méndez-Corbacho, B. Larrarte-Lizarralde, R. Parra, et al., "Physics-Informed Neural Networks for Modeling Li-Ion Batteries: Solving the Single Particle Model Without Labeled Data," *Journal of the Electrochemical Society* 171, no. 11 (2024): 110534.
30. M. Hassanal, P. J. Weddle, R. N. King, et al., "PINN Surrogate of Li-Ion Battery Models for Parameter Inference, Part I: Implementation and Multi-Fidelity Hierarchies for the Single-Particle Model," *Journal of Energy Storage* 98 (2024): 113103.
31. F. M. Rohrhofer, S. Posch, C. Gößnitzer, and B. C. Geiger, "Data vs. Physics: The Apparent Pareto Front of Physics-Informed Neural Networks," *IEEE Access* 11 (2023): 86252.
32. N. Rahaman, A. Baratin, D. Arpit, et al., "Proceedings of the 36th International Conference on Machine Learning," in *Proceedings of Machine Learning Research* (PMLR, 2019), Vol. 97, 5301–5310, <https://proceedings.mlr.press/v97/rahaman19a.html>.
33. R. Van Der Meer, C. W. Oosterlee, and A. Borovykh, "Optimally Weighted Loss Functions for Solving Pdes With Neural Networks," *Journal of Computational and Applied Mathematics* 405 (2022): 113887.
34. L. D. McClenney and U. M. Braga-Neto, "Self-Adaptive Physics-informed Neural Networks," *Journal of Computational Physics* 474 (2023): 111722.
35. W. Dörfler, M. Elasm, and T. Laufer, "Taper-Based Scattering Formulation of the Helmholtz Equation to Improve the Training Process of Physics-Informed Neural Networks," (2024), <https://arxiv.org/abs/2404.09794>, Version Number: 1.
36. S. Ly, M. A. Sadeghi, N. Misaghian, H. Fathiannasab, and J. Gostick, "Rapid Prediction of Particle-Scale State-of-Lithiation in Li-ion Battery Microstructures Using Convolutional Neural Networks," *Applied Energy* 360 (2024): 122803.
37. Y. Xue, Y. Li, K. Zhang, and F. Yang, "A Physics-Inspired Neural Network to Solve Partial Differential Equations—Application in Diffusion-induced Stress," *Physical Chemistry Chemical Physics* 24, no. 13 (2022): 7937.
38. A. Chauhan, E. Asylbekov, S. Kespe, and H. Nirschl, "Influence of Carbon Binder Domain on the Performance of Lithium-ion Batteries: Impact of Size and Fractal Dimension," *Electrochemical Science Advances* 3, no. 1 (2023): e2100151.
39. C. Miehe, M. Hofacker, and F. Welschinger, "A Phase Field Model for Rate-Independent Crack Propagation: Robust Algorithmic Implementation Based on Operator Splits," *Computer Methods in Applied Mechanics and Engineering* 199, no. 45 (2010): 2765.
40. S. P. Timoshenko and J. N. Goodier, *Theory of Elasticity* (McGraw-Hill Book Company, inc, 1951).
41. L. S. Kremer, A. Hoffmann, T. Danner, et al., "Manufacturing Process for Improved Ultra-thick Cathodes in High-Energy Lithium-ion Batteries," *Energy Technology* 8, no. 2 (2020): 1900167.
42. A. Paolone, R. Cantelli, G. Rousse, and C. Masquelier, "The Charge Order Transition and Elastic/Anelastic Properties of LiMn_2O_4 ," *Journal of Physics: Condensed Matter* 15, no. 3 (2003): 457.
43. J. Lederer, "Activation Functions in Artificial Neural Networks: A Systematic Overview," (2021), <https://arxiv.org/abs/2101.09957>, Version Number: 1.
44. G. Martius and C. H. Lampert, "Extrapolation and Learning Equations," (2016), <https://arxiv.org/abs/1610.02995>, Version Number: 1.
45. L. Fesser, L. D'Amico-Wong, and R. Qiu, "Understanding and Mitigating Extrapolation Failures in Physics-Informed Neural Networks," (2023), <http://arxiv.org/abs/2306.09478>, ArXiv:2306.09478 [cs].
46. Y. Zhang, C. Zhao, and Z. Guo, "Simulation of Crack Behavior of Secondary Particles in Li-ion Battery Electrodes During Lithiation/de-Lithiation Cycles," *International Journal of Mechanical Sciences* 155 (2019): 178.
47. G. Yükek and A. Alkaya, "2023 14th International Conference on Electrical and Electronics Engineering (ELECO)," in *Effect of the Depth of Discharge and C-Rate on Battery Degradation and Cycle Life* (IEEE, 2023), 1–5, <https://ieeexplore.ieee.org/document/10415967/>.
48. H. C. W. Parks, A. M. Boyce, A. Wade, et al., "Direct Observations of Electrochemically Induced Intergranular Cracking in Polycrystalline NMC811 Particles," *Journal of Materials Chemistry A* 11, no. 39 (2023): 21322.