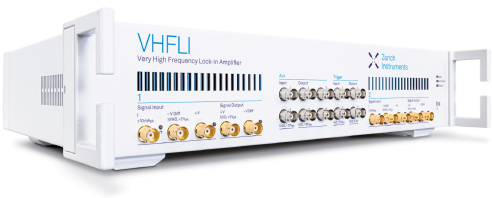
Journal of Applied Physics

RESEARCH ARTICLE | JULY 28 2025

An immersed interface Adaptive Mesh Refinement algorithm for Li-ion battery simulations. II. Multi-dimensional extension and separator modeling

Jiawei Lu ; Nandan Gokhale ; Nikolaos Nikiforakis

J. Appl. Phys. 138, 045002 (2025)
https://doi.org/10.1063/5.0281626



Journal of Applied Physics METHOD pubs.aip.org/aip/jap

An immersed interface Adaptive Mesh Refinement algorithm for Li-ion battery simulations. II. Multi-dimensional extension and separator modeling

Cite as: J. Appl. Phys. **138**, 045002 (2025); doi: 10.1063/5.0281626
Submitted: 20 May 2025 · Accepted: 2 July 2025 ·
Published Online: 28 July 2025

Jiawei Lu,[a] Nandan Gokhale, and Nikolaos Nikiforakis

AFFILIATIONS

Laboratory for Scientific Computing, Cavendish Laboratory, Department of Physics, University of Cambridge, J. J. Thomson Avenue, Cambridge, CB3 0HE, United Kingdom

[a]**Author to whom correspondence should be addressed:** jl2212@cam.ac.uk

ABSTRACT

We present a parallelizable, Adaptive Mesh Refinement (AMR)-compatible solver for computing solutions to multi-dimensional battery models, with added capability to resolve the complex geometries of battery components. Development is carried out within the finite volume framework, with diffuse and sharp (cut-cell) Cartesian immersed interface methods used to model interactions at material interfaces, allowing mesh generation to be carried out rapidly. The solver integrates seamlessly with hierarchical AMR, achieving accelerated computational efficiency while preserving solution accuracy. The parallelizable nature of the solver means that it can be run on massively parallel supercomputers to further reduce computational time. The performance and capabilities of the solver are demonstrated using the pseudo-three-dimensional model, which allows us to present for the first time in the literature a numerical study that directly investigates the effects of separator membrane microstructure on battery electrochemical performance, where the separator microstructures are resolved within the model. The solver was carefully validated under various operating conditions, with grid-aligned and non-grid-aligned battery boundary shapes, on uniform and AMR grids. The use of AMR was shown to significantly reduce computational time for multi-dimensional problems. The solver was also shown to demonstrate good "strong scaling" parallel performance. When using the solver to investigate the effects of separator microstructure, the influences of pore size and constrictivity on electrochemical performance were examined. Through a showcase study performed using realistic separator microstructures, the potential of the solver to be used as an effective tool for design and optimization of next-generation batteries was also demonstrated.

I. INTRODUCTION

With growing concerns over energy depletion and increasing global carbon dioxide emissions, many national governments have announced commitments to phase out internal combustion engine vehicles by 2040.[1,2] This transition highlights the critical and urgent need for advancements in battery technology to promote sustainable energy solutions. Lithium-ion batteries (LIBs) are favored in the battery sector because they possess a number of useful properties, such as high energy and power density, low self-discharge rate, and a long cycle life, to name a few.[3,4] However, several challenges still remain that hinder their widespread adoption, including achieving even higher power and energy densities, enabling ultra-fast charging, and addressing safety concerns.[5] Numerical modeling can serve as a powerful means to gain insights into the physical and electrochemical mechanisms involved, potentially enabling the optimization of battery performance in an efficient and economic manner. While there are some noteworthy examples of numerical investigations that have examined the active materials within battery cells,[5–7] the role of other components, such as separators, in battery performance remains largely unexplored. One of the major challenges in extracting more value from simulation lies in developing numerical solvers with the capability of accurately simulating the complex electrochemical processes while also resolving the

intricate microstructure of battery components, all in a computationally efficient manner.

Most existing solvers use the finite element method (FEM) to solve models of battery physics.[8,9] This is the case, for example, with the commercial software COMSOL Multiphysics, which has been used by many researchers to develop a number of useful and interesting battery models.[6,10–12] However, when dealing with problems that involve resolving complex microstructures of battery components, the computational time demanded by these solvers, both for mesh generation and solution evolution, tends to become daunting.[13,14] To improve computational efficiency, various efforts have been undertaken within the battery community to introduce adaptive mesh refinement or non-uniform mesh techniques.[6,15–22] However, only a limited amount of work has looked into adopting these techniques for multi-dimensional battery simulations that resolve complex microstructures. Mai *et al.*[6] utilized "conforming to interface structured adaptive mesh refinement" (CISAMR) for meshing virtually reconstructed microstructures. The model was implemented in COMSOL using the FEM, requiring a mesh generation process prior to solving the model. When modeling crack growth in storage particles during Li insertion, Klinsmann *et al.*[17] employed an energy-based mesh adaptation algorithm to generate a finite element mesh that is refined near the crack tip. To bypass the cumbersome mesh generation process, Malik and Yu[15] implemented adaptive mesh refinement (AMR) with a smoothed boundary method to simulate complex electrode microstructures.

Hierarchical adaptive mesh refinement is an alternative mesh refinement technique that is well-established in a number of different fields[23–28] but remains unexplored in the context of battery modeling. With this technique, the coarsest mesh is used to cover the entire domain, with finer meshes successively superimposed in regions of interest, offering the potential to realize significant savings in computational expense. In Part I of this paper,[29] we presented an efficient Cartesian immersed interface solver in conjunction with hierarchical Adaptive Mesh Refinement (AMR) within the finite volume method framework to solve the P2D model. Its performance in computing fast, accurate simulations was demonstrated through a wide range of tests under low to high C-rates. In this work, we extend the solver to enable it to numerically solve multi-dimensional battery models that resolve the complex geometries of battery components. Interactions at material interfaces are modeled using Cartesian immersed interface techniques. For example, at the electrode–separator interface when solving the electrolyte phase equations, we use the diffuse immersed interface technique introduced in our previous work. In this work, we also introduce the use of the sharp interface "cut-cell" method. This can be used to model interactions at interfaces created by arbitrarily complex geometries embedded in the Cartesian grid, such as at the interface between the separator membrane and its surroundings. The mesh generation process for the immersed interface techniques is carried out on the Cartesian grid using a signed distance function-based technique, which allows the process to be conducted rapidly, and bypasses the usual mesh generation bottleneck associated with the use of conventional body-fitted meshes.[30]

The use of Cartesian mesh technology in this work allows our solver to seamlessly integrate with the hierarchical AMR, which provides a marked improvement in computational efficiency while preserving solution accuracy when compared to solutions computed on uniform meshes. Nevertheless, the computational time required for multi-dimensional simulations can still be daunting, especially for tests that require a large number of cells to resolve the complex geometries of the battery components. Parallel computing offers a viable solution to accelerate these simulations. However, its adoption within the battery simulation community remains limited. The solver presented in this work utilizes the Message Passing Interface (MPI) for parallelization, allowing it to leverage contemporary supercomputer architectures to further decrease simulation time. In doing so, we provide a solver that is capable of simulating computationally intensive physics-based electrochemical models efficiently enough to investigate multi-dimensional battery systems with geometrically complex battery components.

Among the various physics-based battery models, we choose to solve the pseudo-three-dimensional (P3D) model in this work to demonstrate the performance of the aforementioned numerical techniques used by our solver. Building upon the P2D model, several pseudo-three-dimensional (P3D) models have been developed to explore variations across additional dimensions.[31–34] These models perform a two-dimensional simulation at the homogenized electrode-pair scale and a one-dimensional simulation at the particle scale, effectively capturing both in-plane and through-plane heterogeneities within a battery cell. Rather than focusing on individual electrochemical cell units, some researchers characterize P3D models as networks of interconnected P2D models, designed to simulate the entire battery.[35–39] These models are categorized within the multi-scale, multi-domain (MSMD) framework. We recognize the diversity of P3D models designed for various classes of problems, but our work specifically focuses on the particular type of P3D model that incorporates a two-dimensional simulation at the electrode-pair scale and a one-dimensional simulation at the particle scale, similar to the model adopted by Gao *et al.*[33] For clarity, the term "P3D model" in this paper will henceforth refer to this specific model configuration. Additionally, we augment this base P3D model, which uses homogenized material parameters to model the porous anode, cathode, and separator materials, with the capability to resolve some regions inside the battery cell with explicit representations of their geometries, making it possible to study the effects of local geometric heterogeneities on battery performance.

The augmented P3D model allows for focused investigations on specific battery components within a "complete" model of battery physics, which includes all battery components (anode, separator, and cathode) and accounts for the electrochemical processes taking place within the battery cell. In this work, we focus specifically on resolving complex separator microstructure, using our solver to investigate the effects of separator microstructure on battery cell performance. Research on separators has received relatively lesser attention in the literature compared to research on electrode materials. However, it is well-known that separators significantly impact the performance, safety, cycle life, and cost of battery cells.[40,41] Ideally, a desirable separator possesses an extremely thin thickness, suitable porosity with a homogeneous pore structure, good electrolyte wettability, and robust mechanical strength.[41–46] Designing such separators for high-performance batteries requires balancing different properties, including mechanical strength, ionic transport, electrical

resistance, thermal resistance, and their interactions, highlighting the necessity of investigating separator material, morphology, and performance.

Research to-date on separators has primarily been conducted experimentally.[43] Numerical simulations can serve as a useful addition, allowing the examination of properties that are challenging to measure experimentally, such as the internal states of the battery *in operando*. The simulations in this work specifically focus on the morphology of separators, a key factor in determining electrochemical performance,[47] structural stability,[48] and dendrite growth.[45] Currently, modeling of separator components with a consideration for varying geometric parameters, such as pore size, porosity, and tortuosity, remains rather immature.[9] In existing numerical models, the porous microstructure of separators is conventionally quantified by a set of homogenized parameters, such as porosity and the tortuosity factor. These parameters are then used to determine the effective material coefficients of the electrolyte in the separator region.[49–52] However, the real morphology of separators is not homogeneous,[53] and it is not necessarily sufficient to use a single set of commonly used homogenized parameters to represent the microstructural information of the entire separator.[41,54] Even with identical bulk physical characteristics, battery performance can vary significantly due to the degree of heterogeneity introduced by the separator microstructure.[46] The direct correlation between the separator membrane microstructure and battery performance has yet to be thoroughly investigated. The solver developed in this work is well-suited to filling this research gap, overcoming the limitations imposed by previous simplifications. Inspired by suggestions in the existing literature,[46,54] we first investigate the relationship between separator morphological characteristics and battery performance using a series of idealized geometries. This is followed by simulations where the separator region is explicitly resolved using the realistic microstructure of a commercial separator. To the best of our knowledge, this work is the first instance in the literature where a complete model of battery physics is used to find a direct correlation between separator membrane microstructure and cell electrochemical performance. The simulation methodology enables the identification and subsequent optimization of regions within the separator that most restrict ionic transport, which is critical for batteries operating at high rates,[55,56] where ionic transport becomes a limiting factor governing cell performance. The insights gained from the numerical investigations are, therefore, of significant scientific and practical values for the design and optimization of the separator membrane.

The rest of this paper is organized as follows. In Sec. II, we provide an outline of the governing equations of the base P3D model. Details of the numerical methods used for augmenting and solving the model are described in Sec. III. Note that the methods are not restricted in applicability to solving the P3D model and could be adapted to solve other multi-dimensional battery models if required. In Sec. IV, we present and discuss the results obtained with the P3D solver, highlighting the efficiency gains achieved with the use of AMR and parallelization. The results of the numerical investigation into how separator microstructure affects battery performance, including results with the use of complex, realistic separator microstructures, are shown in Sec. IV D. Finally, conclusions and areas for future work are provided in Sec. V.

II. THE P3D MODEL

The P3D model is a natural extension of the well-known P2D model, building upon it with an additional spatial dimension in the electrode-pair domain, allowing for the study of both the in-plane and through-plane heterogeneities within a battery cell. Figure 1 shows a visual representation of the P3D model, which comprises a two-dimensional domain at the homogenized electrode-pair scale (xy-domain), and a pseudo-dimension for the solid spherical particles at the microscale (r-coordinate). The xy-domain represents a two-dimensional homogenized electrode-pair domain that consists of the anode, the separator, and the cathode. The electrolyte phase Li^+ concentration (c_e), the electrolyte phase potential (ϕ_e), and the solid phase potential (ϕ_s) are modeled in the xy-domain. For the particle domain, the diffusion of the solid phase concentration

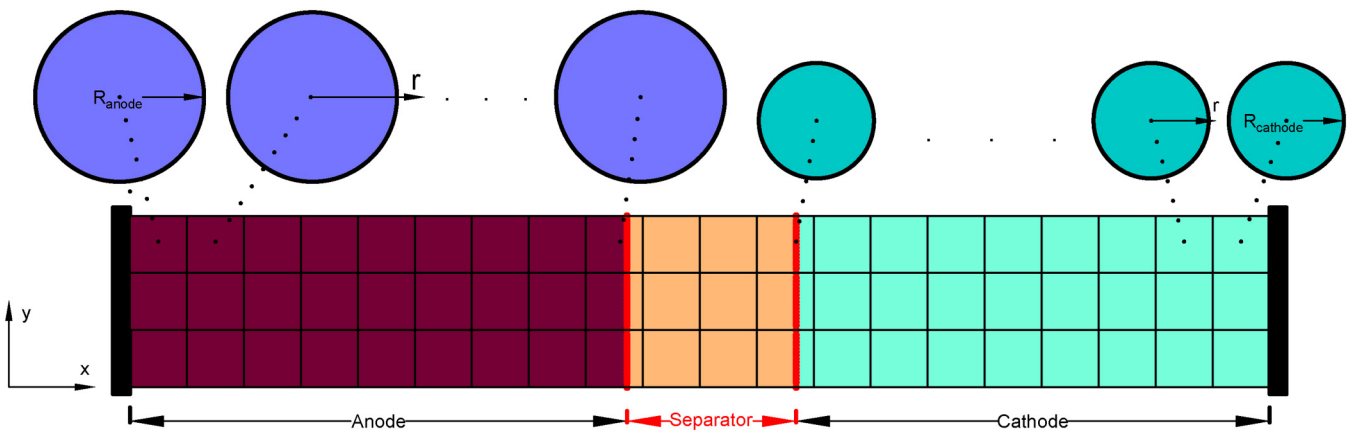


FIG. 1. Schematic of the pseudo-three-dimensional model. R_{anode} and $R_{cathode}$ represent the radii of the electroactive particles in the anode and cathode regions, respectively.

TABLE I. Governing equations for the P3D model.

Governing equation	Boundary condition	Equation no.
Diffusive transport of neutral Li within the solid electrode particles		
$\frac{\partial c_s}{\partial t} = \frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2 D_s \frac{\partial c_s}{\partial r}\right)$	$-D_s \frac{\partial c_s}{\partial r}\Big\|_{r=R_s} = r_{Li}, \quad \frac{\partial c_s}{\partial r}\Big\|_{r=0} = 0$	(1)
Diffusion of Li^+ in the electrolyte phase		
$\varepsilon_e \frac{\partial c_e}{\partial t} + \nabla \cdot (-D_e \nabla c_e) = (1 - t_+) a_s r_{Li}$	$\frac{\partial c_e}{\partial x}\Big\|_{x=0,\,L} = 0, \quad \frac{\partial c_e}{\partial y}\Big\|_{y=0,\,H} = 0$	(2)
Solid phase potential governed by Ohm's law		
$-\nabla \cdot (\sigma_s \nabla \phi_s) = -a_s F r_{Li}$	$\frac{\partial \phi_{s,a}}{\partial x}\Big\|_{x=L_a} = 0, \frac{\partial \phi_{s,c}}{\partial x}\Big\|_{x=L_a+L_s} = 0, \; -\sigma_{s,m}\frac{\partial \phi_{s,m}}{\partial x}\Big\|_{x=0,L} = I_{applied}, \frac{\partial \phi_{s,m}}{\partial y}\Big\|_{y=0,\,H} = 0$	(3)
Electrolyte phase potential governed by Ohm's law		
$\nabla \cdot \left\{-\kappa_e \left[\nabla \phi_e - \frac{2RT}{F}\left(1 + \frac{\partial \ln f_{\pm}}{\partial \ln c_e}\right)(1 - t_+)\nabla \ln c_e\right]\right\} = a_s F r_{Li}$	$\frac{\partial \phi_e}{\partial x}\Big\|_{x=0,L} = 0, \frac{\partial \phi_e}{\partial y}\Big\|_{y=0,\,H} = 0$	(4)
Electrochemical reaction kinetics governed by the Butler–Volmer equation		
$r_{Li} = r_0\left\{\exp\left(\frac{\alpha_a F}{RT}\eta\right) - \exp\left(-\frac{\alpha_c F}{RT}\eta\right)\right\}$		(5)
$\eta_{s,m} = \phi_{s,m} - \phi_e - U_{s,m}$ (m = anode, cathode)		
$r_0 = k_0 c_e^{\alpha_a}(1 - \theta_s)^{\alpha_a}\theta_s^{\alpha_c}, \quad \theta_s = c_{s,surf}/c_{s,max}$		

(c_s) inside the spherical active particles is modeled in the radial r-coordinate.

The governing equations of the P3D model remain similar to those of the P2D model, with the modeling of variation along the y axis added for the electrode-pair domain. A summary of these equations is provided in Table I.

When conducting the simulations using the homogenized parameters, the effective material coefficients of the electrolyte phase are determined by

$$D_{e,m} = \varepsilon_{e,m}^{brugg_m} D_{0,m} \quad \kappa_{e,m} = \varepsilon_{e,m}^{brugg_m} \kappa_{0,m} \quad (m = \text{anode, separator, cathode}), \tag{6}$$

where D_0 and κ_0 are the intrinsic diffusivity and conductivity of the electrolyte, respectively, both of which vary with concentration and temperature. "brugg" that appears in the exponent of Eq. (6) is the Bruggeman coefficient, which correlates the tortuosity with the porosity via the empirical Bruggeman relation.

When resolving separator microstructures within the model as is done in this work, the governing equations [Eqs. (1)–(5)] still apply, with adjustments made based on the explicit representation of the geometry. The regions occupied by the solid materials that constitute the separator (i.e., the separator membrane) are treated as inactive, making it unnecessary to solve for the properties in the computational cells inside these regions, while the electrolyte phase equations are solved in the regions of the electrolyte-filled pores. The interaction between the solid material and the surrounding electrolyte is incorporated into the model through internal boundary conditions imposed at the material interfaces,

$$\partial_n \phi_e = 0, \quad \partial_n c_e = 0, \tag{7}$$

where **n** denotes the normal direction at the surface of the inactive material. For the pores in the separator region, the electrolyte phase equations describe the behavior of the pure electrolyte; hence, the effective transport coefficients [see Eqs. (2) and (4)] are replaced by the bulk transport coefficients, and the electrolyte phase volume fraction in this region [see Eq. (2)] is set to one.

III. NUMERICAL METHOD

In Part I of this paper,[29] we presented an efficient Cartesian immersed interface solver that integrates with hierarchical AMR to solve the full-order P2D model. In this work, we extend the solver to simulate the multi-dimensional P3D model, augmented with the capability to explicitly resolve complex geometries embedded in the Cartesian mesh. Note that the numerical techniques presented may be applied to solve other multi-dimensional battery models as well; we use the P3D model with explicit resolution of geometries in the separator region as an illustrative application that is also of scientific interest. Methods for resolving other battery components, such as electrode-level cracks and voids, are also briefly commented upon in Sec. III A 4.

The governing equations for both the electrode-pair and particle domains are discretized on their respective Cartesian grids using the finite volume method. Hierarchical AMR is used to locally refine the mesh in targeted regions, such as in the vicinity of

the material interfaces, in order to improve solution accuracy while ameliorating the resulting increase in the computational cost. Complex geometries, such as those of the separator membrane, are treated as being embedded in the background Cartesian mesh. An immersed interface technique known as the "cut-cell method" is used to treat the resulting irregular interfacial cells and to model their interaction with the surroundings. Further details of the mesh generation, AMR, and interface treatment techniques employed are provided in Sec. III A. The numerical methods that we use to solve the highly coupled system of governing equations are the same as those described in Part I of this paper,[29] and we provide a brief recap in Sec. III B. Linear systems arising from the discretized form of the governing equations are solved on the AMR mesh using a multigrid linear solver, as elaborated in Sec. III C. The solver is parallelized using MPI, which allows computations to be distributed across multiple CPUs on modern supercomputing architectures to further reduce simulation time. The parallel implementation is non-trivial because of the presence of the pseudo-dimension in the model, and we provide implementation details in Sec. III D.

A. Mesh generation

We discretize equations on the electrode-pair domain using a single Cartesian mesh, with arbitrarily complex geometries, such as that of the separator membrane considered embedded within the Cartesian grid. To address the issue of material interfaces not necessarily being perfectly aligned with cell boundaries, we introduce immersed interface techniques to treat the "mixed" and "cut" cells created by material interfaces. Consider Fig. 2, which illustrates the two types of material interfaces that need to be considered. The interfaces between the electrode and separator regions are treated as "diffuse" interfaces when solving the electrolyte phase equations to model the continuous nature of the electrolyte as it diffuses between the electrode and separator regions. The cells at the interfaces are classified as "mixed cells." The material interfaces between the solid parts of the separator (i.e., the separator membrane) and the surroundings, on the other hand, are treated as "sharp"; the cells at these interfaces are classified as "cut cells" and are modeled using a cut cell sharp-interface method.

The use of the Cartesian immersed interface techniques allows the mesh generation bottleneck associated with conventional body-fitted meshes to be bypassed.[30] Specifically, we use the well-established signed distance function-based immersed interface mesh generation technique that enables rapid, automatic mesh generation to be carried out for arbitrarily complex geometries.[23,57] The process begins by calculating for all cells in the computational domain the signed distance function,[58] which determines the minimum distance between the geometry and the corresponding cell centers. In this signed distance function, a positive value designates the region inside the geometry, while a negative value designates the region outside the geometry. The geometry interface is treated implicitly as the zero level set of the signed distance function. Once the signed distance function has been determined, its values are used to compute the geometric properties required by

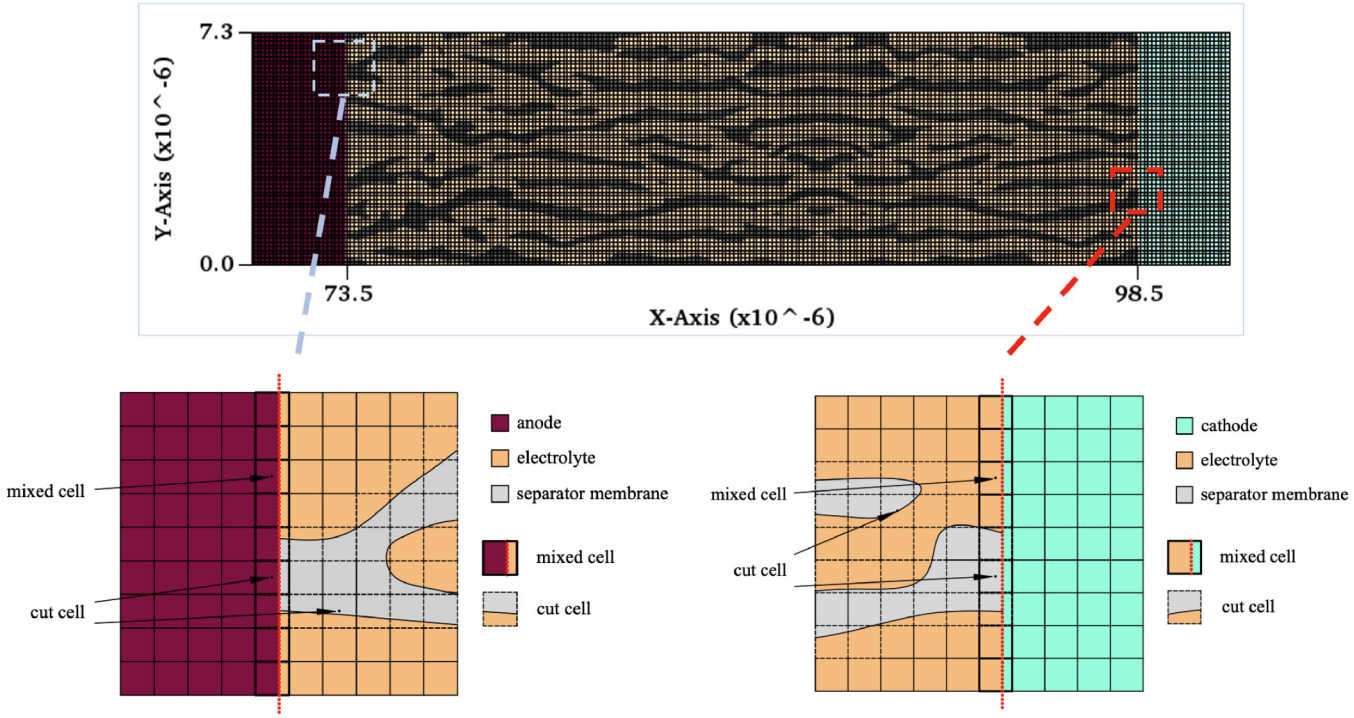


FIG. 2. Illustration of mixed cells and cut cells on the electrode-pair domain. The geometries colored in gray represent the separator membrane.

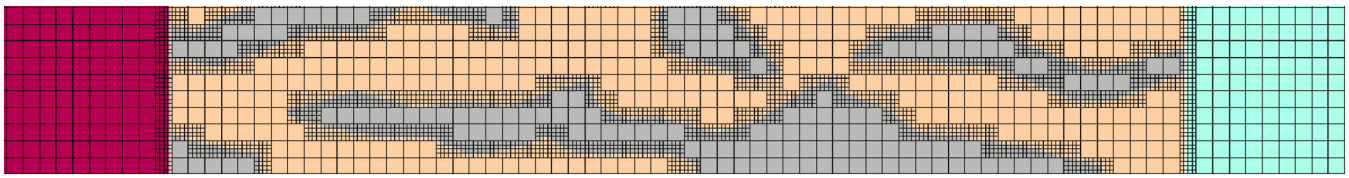


FIG. 3. Zoomed-in view of an example mesh on the electrode-pair domain. The refined mesh is generated in the vicinity of the interfaces between the separator surface and electrode materials, as well as near the material boundaries of the inactive solid materials that constitute the separator membrane.

the immersed interface methods. At the diffuse interfaces, the only required geometric parameters are the cell volume fractions. At the cut cell interfaces, other geometric parameters, such as the cell area fractions and interface normals, are also required (see Sec. III A 3 for more details). Details of how information from the signed distance function can be used to linearly reconstruct the interface and calculate the required geometric parameters in all cells can be found in Gokhale *et al.*[23] The geometric parameters are then used by their respective immersed interface methods as part of the system discretization to model the interactions at material interfaces.

As in Part I of this paper,[29] the use of a single Cartesian mesh on the entire electrode-pair domain also allows us to make use of hierarchical Adaptive Mesh Refinement (AMR)[59] to improve solution accuracy while mitigating the resulting increase in computational expense by only refining the mesh in target regions, such as near material interfaces. In Sec. III A 1, an overview is provided of the hierarchical AMR approach that we use on the electrode-pair domain. As concluded in the investigation carried out in Part I of the paper,[29] AMR on the particle domain is not required unless significant variations appear in the solid phase concentration. Hence, for the simulations in this work, the model on the particle domain is discretized on a uniform grid. In Secs. III A 2 and III A 3, respectively, we provide an overview of the diffuse and sharp (i.e., cut-cell) immersed interface approaches that we use. Specific details of how the cut cell treatment is applied when resolving the separator membrane, and when solving the solid phase potential equations, are provided in Secs. III A 4 and III A 5, respectively.

1. Adaptive Mesh Refinement

Adaptive Mesh Refinement (AMR) is an approach that allows the mesh to be locally refined in regions of interest in the domain where greater accuracy is required, such as at material interfaces. When compared to using a uniform mesh (also known as a unigrid mesh), an appropriate use of AMR can provide a resulting mesh that has the same "effective" or "equivalent" resolution as the unigrid case. By the "same effective resolution," we mean that the AMR mesh can have the same cell size as the unigrid mesh in refined regions and can produce a solution of comparable accuracy, while reducing both the computational expense and memory required when compared to the unigrid case.[60] To the best of our knowledge, Part I of this paper[29] and this work are the first presentations of the use of AMR for pseudo-dimensional battery models. In Part I, we demonstrated the significant savings in computational cost that the use of AMR provides. In this work, we demonstrate further how AMR reduces the computational cost in multi-dimensional simulations and when resolving complex geometries therein. There are several ways to implement Adaptive Mesh Refinement; popular choices include the octree[61] and hierarchical AMR types.[62,63] We adopt the hierarchical AMR approach, in which the computational domain is meshed with a set of hierarchical grids organized by "levels," with the grids at a particular level having the same resolution. At the bottom level, the coarsest grid covers the entire domain. In regions of interest that are to be resolved, such as near material interfaces, additional levels of successively refined meshes can be superimposed on the coarsest mesh. The discretized system of equations on the AMR hierarchy is solved concurrently, making use of data from all mesh levels.

We build our solver by making use of the C++-based AMReX library,[59,64] which provides an efficient, parallelized implementation of block-structured hierarchical AMR, embedded boundary (EB) descriptions of geometries on Cartesian meshes, and compatible multi-level linear solvers suitable for use in this work. The AMR functionality can be used to refine regions either dynamically based on appropriately defined criteria or statically by specifying the regions of interest. When solving the P3D model, we use the approach of placing fixed layers of refinement near the material interfaces to model the interfaces across different materials and to resolve the complex geometries accurately, as illustrated by the zoomed-in view of an example mesh in Fig. 3.

2. Diffuse interface method

We use the diffuse interface method as detailed in Part I of this paper[29] to treat the "mixed" cells created by the interfaces between the separator surface and electrode materials when solving the electrolyte phase equations. Consider Fig. 2, which illustrates the computational mesh in the vicinity of the electrode–separator interfaces. The diffuse interface technique requires the definition, in each cell, of a volume fraction variable, $\beta \in [0, 1]$, that is given by the ratio of electrode material volume to the regular cell volume, irrespective of whether the separator membrane occupies part of the volume. The volume fraction β is, therefore, between 0 and 1 in the "mixed" cells in which the interfaces are located. It is used to calculate material properties at the mixed cells and is also introduced naturally into the governing equations to reformulate them locally near the interface to allow the calculation of the evolution of the mixed cell states. Equations (2) and (4) for the electrolyte phase are reformulated in a straightforward manner as follows to take into consideration the part of the cell involving the reaction:

$$\varepsilon_{e,m}\frac{\partial c_e}{\partial t} + \nabla\cdot(-D_e\nabla c_e) = \beta_{s,m}(1-t_+)a_{s,m}r_{Li} \quad (m = \text{anode, cathode}), \tag{8}$$

$$\nabla\cdot\left\{-\kappa_e\left[\nabla\phi_e - \frac{2RT}{F}\left(1+\frac{\partial \ln f_\pm}{\partial \ln c_e}\right)(1-t_+)\nabla \ln c_e\right]\right\} = \beta_{s,m}a_{s,m}Fr_{Li} \quad (m = \text{anode, cathode}). \tag{9}$$

For determining cell-centered material properties in the mixed cells, we use the harmonic mean approach, weighted using the cell volume fraction $\beta_{i,j}$.

3. Cut-cell method

The cut-cell method, which belongs to the class of Cartesian "immersed boundary methods,"[30] is a conservative sharp-interface method where material interfaces are represented within computational cells using piecewise-linear reconstruction. We use this method to model the interface between the separator membrane and its surroundings, resulting in the creation of "cut cells" at the interface, as illustrated earlier in Fig. 2.

Figure 4 illustrates an example of this geometry generation process for a simple circular geometry embedded in a background Cartesian grid. The computational cells on the grid can be categorized into three types: regular cells, which are completely outside the geometry [indicated by the red cells in Fig. 4(b)]; covered cells, which are completely inside the geometry [indicated by the blue cells in Fig. 4(b)]; and cut cells, which intersect with the geometry outline. Figure 4(b) illustrates the piecewise-linear reconstruction of the circular geometry in the cut cells for a coarse, unigrid mesh. The accuracy with which the geometry is resolved can be readily increased by refining the mesh near the material interfaces locally with the use of AMR, as illustrated in Fig. 4(c).

In our implementation, we make use of the cut cells/embedded boundary functionality provided in the AMReX library[65] to calculate the required cut cell geometric properties. These include the volume fraction of the cut cell, $\psi_{i,j} \in [0, 1]$, evaluated as the ratio of the volume of the "valid" portion of the cell (i.e., the part outside the geometry of the solid material) to the total volume of a regular cell; the area of the reconstructed interface front $A^f_{i,j}$, and the "apertures," or "area fractions" of the cell edges $a \in [0, 1]$. Figure 5 illustrates the definitions of these quantities with examples for both regular and cut cells.

The cut cell geometric parameters are used as part of the discretization of the system on the cut cell grid, using the cut-cell method of Johansen and Colella,[66] which is designed to ensure that the condition number of the resulting matrix remains relatively unaffected by small cut cell volumes. The resulting discrete linear system is solved using a multigrid linear solver, as detailed in Sec. III C. Here, we briefly outline the second-order accurate method devised by Johansen and Colella[66] that can be used within the finite volume framework to discretize systems governed by both elliptic and parabolic equations, on Cartesian grids containing embedded complex geometries. In this method, the solutions for regular and cut cells are considered to be cell-centered, under the assumption that the solution extends smoothly beyond the domain. The discretized elliptic/parabolic operators at the cut cell, however, are centered at the centroid of the valid portion of the cell. The method is first-order accurate at the boundary, but has been shown to achieve higher than first order, and in some applications, second-order convergence in the global solution error.[64,66–68] Consider the Poisson equation in Eq. (10), which we use as an example to describe the discretization procedure for the cut cell with index (i, j) on the grids shown in Figs. 6(a) and 6(b),

$$\nabla\cdot\gamma\nabla\zeta = g \quad \text{on } \Omega. \tag{10}$$

For the cell with index (i, j) and cell center located at (x_i, y_j), let $\zeta_{i,j}$ represent the volume-averaged solution in that cell. $x_{i\pm1/2}$, $y_{j\pm1/2}$ are then the coordinates of the cell interfaces at which the fluxes F are applied. The governing equation of Eq. (10) is discretized as follows:

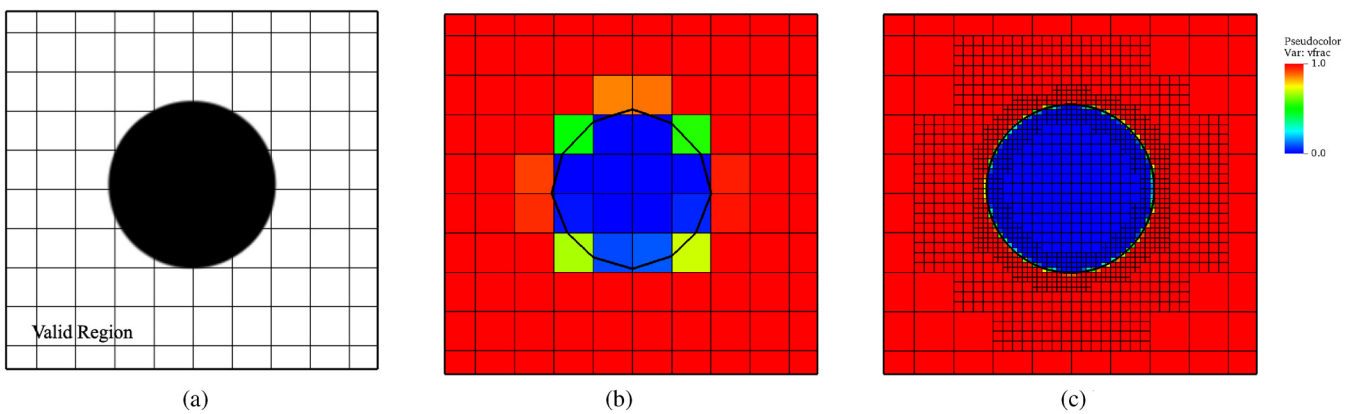


FIG. 4. Illustration of the cut cell grid generation process. (a) An example of resolving a circular geometry on a Cartesian mesh. Only the regions not covered by the geometry are considered valid and are involved in the system evolution. (b) Volume fraction plot where the black line illustrates the piecewise-linear reconstruction of the geometry in the cut cells. (c) Volume fraction plot illustrating the use of AMR to capture the geometry interface more accurately.

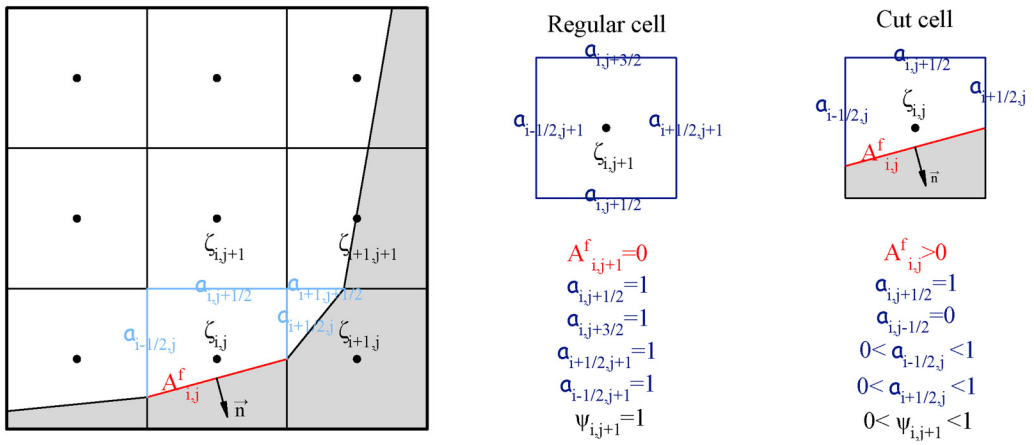


FIG. 5. Illustration of the area of the reconstructed interface A^f_{ij} (in red), the apertures (area fractions of the cell edges) $a \in [0, 1]$ (in blue), and the volume fraction of the cut cell $\psi_{ij} \in [0, 1]$. Two examples are provided: one for a regular cell and one for a cut cell.

$$(\nabla \cdot \gamma \nabla \zeta)_{i,j} = \frac{1}{\Delta x \Delta y \psi_{i,j}} \left(F_{i+1/2,j} - F_{i-1/2,j} + F_{i,j+1/2} - F_{i,j-1/2} - F^f_{i,j} \right) = \bar{g}_{i,j}. \quad (11)$$

F^f is included as part of the difference operator to account for the flux normal to the cut cell interface within the cell. As illustrated in Fig. 6(a), all fluxes are centered at the midpoint of their corresponding edge. For the Poisson equation, the flux is proportional to the gradient estimated using the cell-centered solutions. Note that for a regular cell, $\psi_{i,j}$ equals one and $F^f_{i,j}$ drops to zero, resulting in Eq. (11) reverting to a regular second-order spatial difference operator. The fluxes through cell edges are computed using the cell-centered solutions. For instance, the flux in the x direction at $(i + 1/2, j)$ is determined by

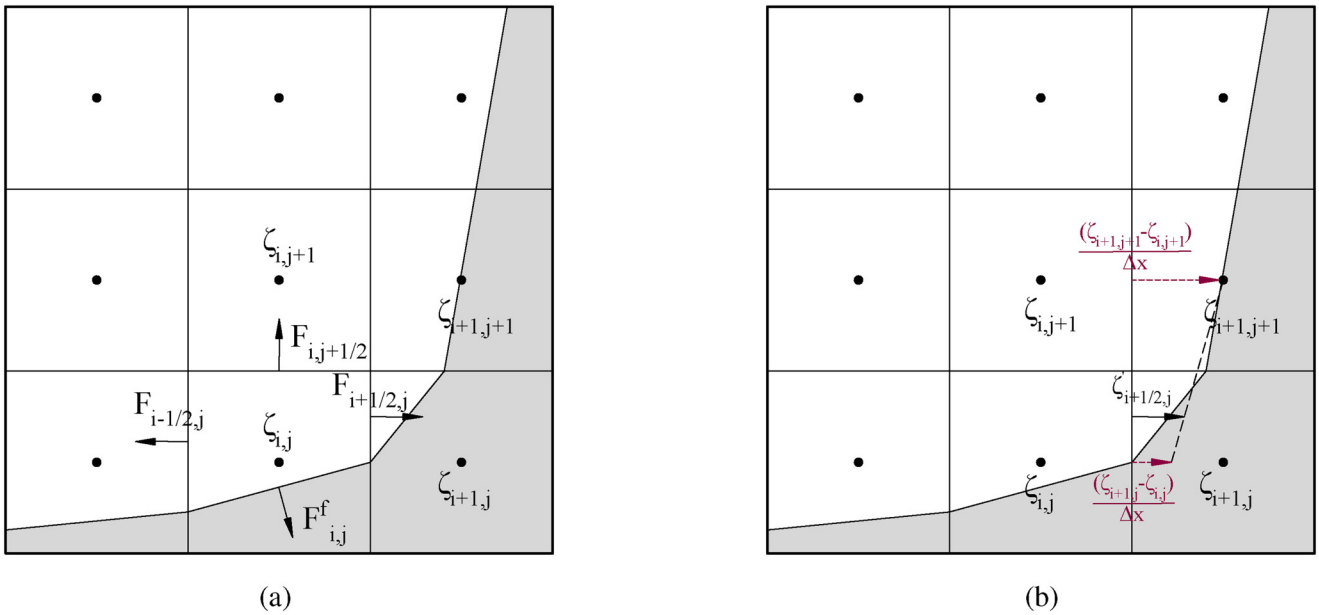


FIG. 6. Illustrations of the discretization procedure for the cut cell with index (i,j). (a) Schematic illustrating that all fluxes, including those of the partial edge, are centered at the midpoint of the edge. (b) Schematic demonstrating that the flux of a partial edge is evaluated by linearly interpolating two fluxes (edge gradients), which are determined using the cell-centered values.

$$F_{i+1/2,j} = \Delta y \gamma_{i+1/2,j} \frac{\left(\zeta_{i+1,j} - \zeta_{i,j}\right)}{\Delta x}. \quad (12)$$

For a cell edge that is partially covered by the geometry, its flux is computed by linearly interpolating two fluxes, which are determined from the cell-centered values. As illustrated in Fig. 6(b), the flux at $(i + 1/2, j)$ for the partial edge, for example, is evaluated using

$$F_{i+1/2,j} = a\Delta y \gamma_m \left[\frac{(1+a)}{2} \frac{\left(\zeta_{i+1,j} - \zeta_{i,j}\right)}{\Delta x} + \frac{(1-a)}{2} \frac{\left(\zeta_{i+1,j+1} - \zeta_{i,j+1}\right)}{\Delta x} \right], \quad (13)$$

where γ_m is at the midpoint of the partial edge.

For the problem of interest in this work, homogeneous Neumann boundary conditions are applied to the surface of the embedded geometry (see Sec. II). The flux F^f, which models the gradient across the material interface, therefore, drops to zero. If a Dirichlet boundary condition is to be applied instead, a three-point gradient stencil is employed to determine the gradient at the midpoint of the surface using values from neighboring cells. Interested readers can refer to the paper of Johansen and Colella[66] for further details.

4. Interface treatment around the electrochemically inactive solid materials

When resolving separator geometries on a Cartesian grid, as in this work, we use the cut-cell method (see Sec. III A 3) to model the interactions between the inactive solid materials (i.e., separator membranes) and the surrounding electrolyte. For problems involving inactive material embedded in electrode materials, such as when modeling the effects of electrode-level voids, the cut-cell method can be used at the material boundaries of the voids. With the focus of this paper specifically on resolving the separator geometry, we will not delve further into the methods used to model flaws in the electrode materials. This will be the focus of a future publication.

5. Interface treatment for solving the solid phase potential

When solving for the solid phase potential in the anode and cathode regions, the boundary conditions across the electrode and separator interfaces are set based on the constraint that no current flows into the separator region. Specifically,

$$\left.\frac{\partial \phi_{s,a}}{\partial x}\right|_{x=L_a} = 0, \quad \left.\frac{\partial \phi_{s,c}}{\partial x}\right|_{x=L_a+L_s} = 0.$$

Homogeneous Neumann boundary conditions are applied at the interfaces between the separator and electrode materials using the cut-cell method detailed in Sec. III A 3.

When solving for the solid phase potential of an electrode, the rest of the domain is treated as inactive, with the inactive geometry represented as a cut cell embedded boundary (EB). Figure 7 shows schematics of the inactive EB geometry used when solving for the solid phase potential for the anode and cathode. The discretized form of Eq. (3) is then solved using EB-compatible multigrid linear solvers (see Sec. III C). Two linear solves are carried out, one for the anode and one for the cathode.

B. Solution algorithm

The solution approach that we use to numerically solve the coupled system of the P3D model is the same as that used when solving the P2D model in Part I of this paper.[29] Here, we provide a brief recap of the solution algorithm used. The governing equations are discretized using the finite volume method on both the electrode-pair and particle domains and are solved implicitly following the sequential full-order method proposed by Han *et al.*[69] In this sequential method, only the potential fields and the volumetric current density are solved iteratively, while the concentration equations are moved outside the iteration loop and are solved only once per time step. The flow chart in Fig. 8 summarizes the numerical method used to solve the governing equations at each time step. The system evolves with a fixed time step size until it reaches the specified stop time, or the terminal voltage falls below the cut-off voltage. Further details on our C++-based implementation can be found in Part I of our paper.

C. Multigrid solvers

The linear systems arising in the solution of the governing equations are discretized on each AMR level and solved in a composite manner using an AMR-compatible multigrid solver. The

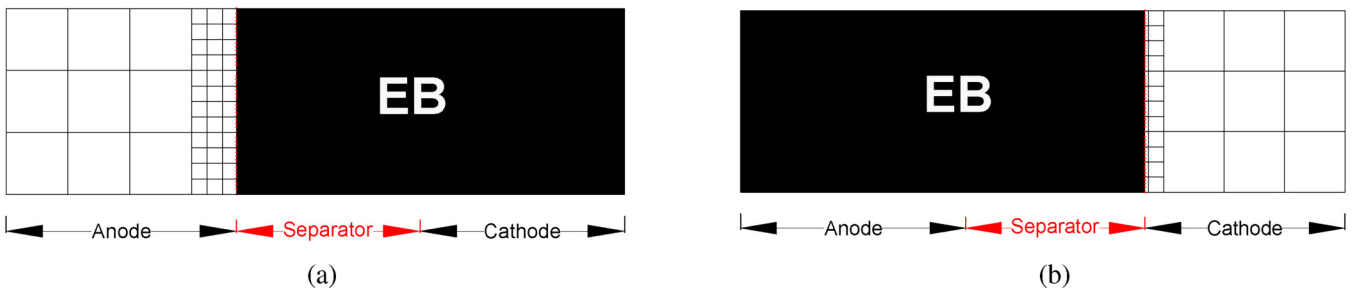


FIG. 7. EB geometry used for solving the solid phase potential in the electrode materials. (a) EB geometry used for solving the solid phase potential in the anode. (b) EB geometry used for solving the solid phase potential in the cathode.

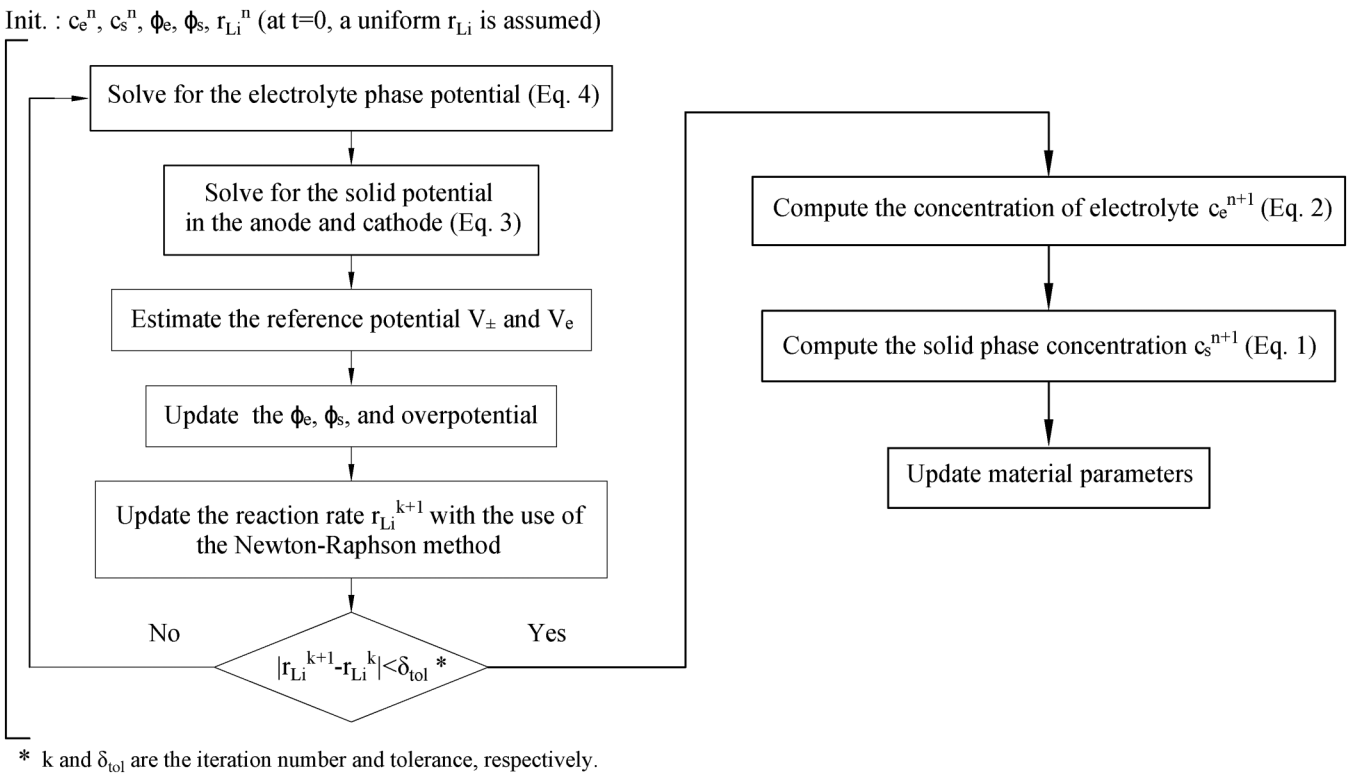


FIG. 8. A summary of the numerical method used for solving the governing equations at each time step.

AMReX library contains an implementation of the solver that sets boundary conditions across the multiple levels and solves the linear system iteratively using the geometric multigrid method.[59,66] Here, we provide an overview of how the solver works.

When solving a system that is discretized with the use of AMR, it becomes important to properly communicate information across the coarse-fine boundary. This process allows data from the refined mesh to be used to improve the accuracy of the solution at the coarser level in regions not overlaid by the refined mesh. To transfer data bi-directionally between the coarse and fine meshes (i.e., from coarse to fine and from fine to coarse), a composite Laplacian operator is defined across the multiple AMR levels at the interfaces between coarse and fine mesh patches.[70] Interested readers can refer to the work by Martin and Cartwright[70] for further details. The solution process for the linear system proceeds iteratively using a V-cycle multigrid approach.[70,71] The iterative process stops once the residuals of all the AMR levels fall below a user-defined tolerance. At the finest AMR level, the residual is evaluated using all cells at that level, whereas at coarser levels, it is computed only in regions not covered by any finer AMR levels.

D. Parallel computing

Parallel computing has the potential to significantly reduce the wall time required for computationally intensive simulations, such as those involving the use of multi-dimensional battery models. These models are not only numerically challenging to solve due to the highly coupled, non-linear nature of their governing equations, but they also require the use of a large number of computational cells to accurately resolve the battery components. Despite its widespread application in computational fluid dynamics,[72] the adoption of parallel computing within the battery simulation community remains limited.

The C++-based AMReX library that we employ provides support for MPI for parallelization, where computations are distributed across MPI ranks, with each rank processing its own data.[62] Each rank is typically allocated to a separate processor core. While the excellent strong and weak scaling performance of the AMReX library has been demonstrated in various applications,[62,73,74] applying its parallel computing capability to pseudo-dimensional models is non-trivial. The inclusion of the additional radial/pseudo-dimension, in particular, adds complexity to the data structures and requires special consideration. Here, we briefly discuss the implementation details used to enable parallelization of the full-order P3D model via MPI, enabling the use of contemporary supercomputing architectures to minimize simulation times.

The use of the AMReX library significantly reduces the effort required to make our implementation parallel-compatible. In particular, AMReX provides an implementation of a flexible, multi-component, array data structure called a "MultiFab," which can be distributed and load balanced across processors during solution evolution. For more details of the MPI implementation of AMReX, the reader is referred

to Zhang *et al.*[59,62] While the parallelization of the system on the electrode-pair domain is handled by the AMReX library once data are stored in MultiFabs, special consideration needs to be given to parallelization in the pseudo-dimension. Several options are possible for achieving the parallelization; here, we provide details of the specific approach that we use. Consider Fig. 9, which illustrates the data stored in the MultiFab for a single cell on the electrode-pair domain. Various properties on the electrode-pair domain, such as the electrolyte phase concentration c_e and potential ϕ_e, are stored as separate variables in the MultiFab. As shown in the figure, the solid phase concentrations c_s for all cells on the corresponding particle/radial domain are also stored as consecutive components of the MultiFab.

For each cell on the electrode-pair domain, the solid phase diffusion equation on the corresponding radial domain is solved locally on the same processor that owns the data for that cell. Specifically, a linear system in the form of "Ax = b" is assembled by discretizing the solid phase diffusion equation according to the finite volume scheme, with boundary conditions at both ends imposed. The solution of the linear system is computed using a QR decomposition algorithm provided in the Eigen library,[75] and the updated data on the radial domain are stored back into the "MultiFab" as consecutive components for subsequent operations, as illustrated in Fig. 9. This implementation ensures that operations on properties at the pseudo-dimension happen on the same processor that owns the corresponding data on the electrode-pair domain. The MPI parallelization of the linear solves on the electrode-pair domain, for which the data are stored in MultiFabs, is handled by the AMReX implementation of the multigrid linear solver (see Sec. III C).

Our solver can, therefore, be run on a single-core processor as well as on massively parallel supercomputers, while also allowing the use of multiple AMR levels for mesh refinement. These features significantly accelerate simulation times, making the solver well-suited for computationally intensive simulations involving realistic microstructures, and highlighting its potential as an effective tool for the design and optimization of next-generation batteries. While this study focuses on two-dimensional simulations for the electrode-pair domain, these capabilities are expected to be even more beneficial for three-dimensions or when incorporating additional physics. We investigate the parallel scalability of our implementation in Sec. IV C.

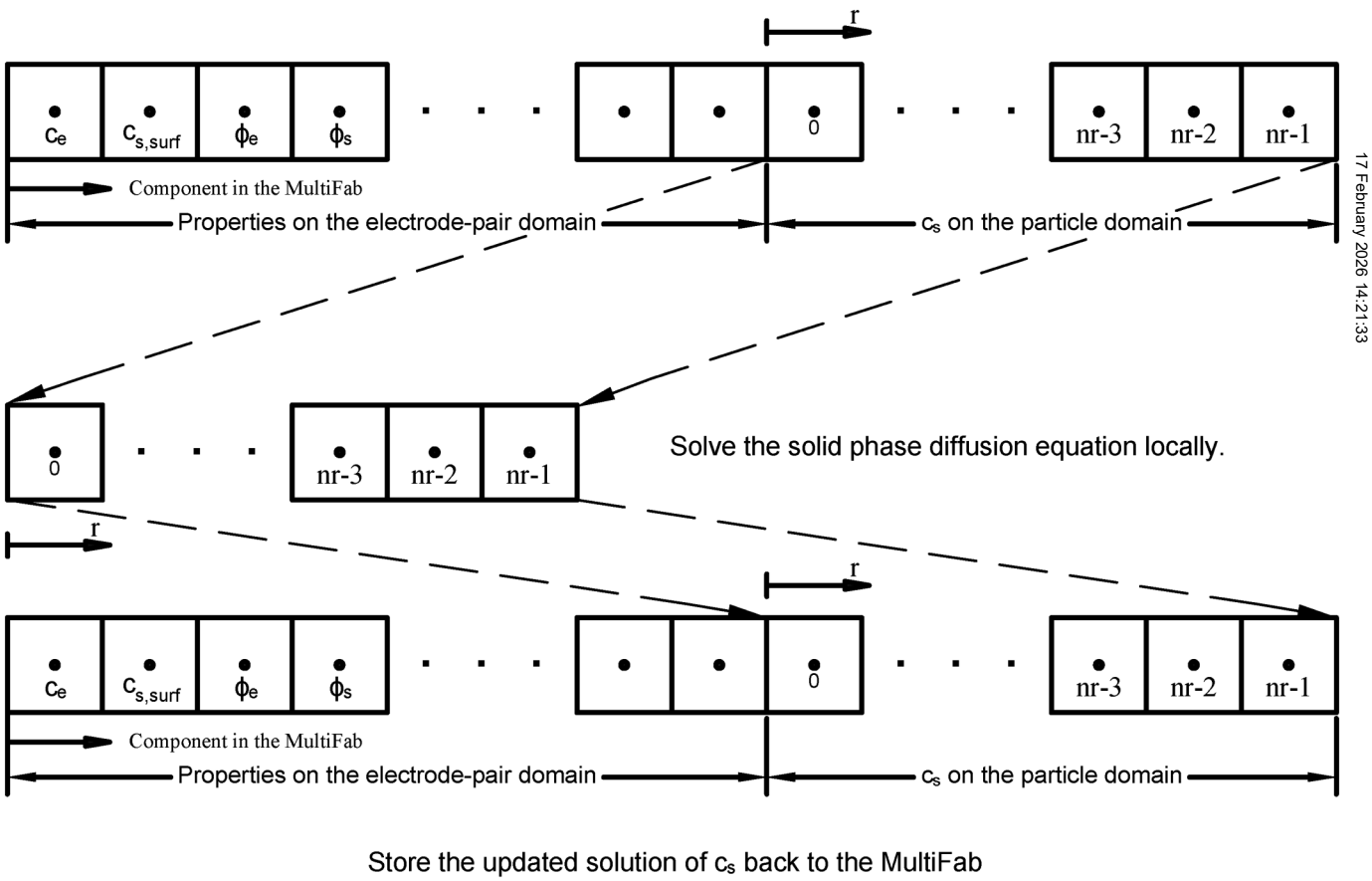


FIG. 9. Illustration of the data structure and the process of calculating and storing the solid phase concentration across the radial dimension for a computational cell on the electrode-pair domain.

IV. RESULTS AND DISCUSSION

In this section, we present and discuss the results obtained with the solver developed in this work, for problems that use the P3D model to predict battery electrochemical performance. Results for the validation of the solver and for demonstrating the efficiency gains achieved with the use of AMR and parallelization are presented in Secs. IV A–IV C. A numerical investigation of separator microstructure, made possible by the model and solver development of this work, is presented in Sec. IV D. All simulations were performed on Intel "Ice Lake" nodes of the Cambridge Service for Data-Driven Discovery (CSD3)[76] cluster, where each node is equipped with up to 76 Intel Xeon Platinum 8368Q 2.60 GHz CPUs.

A. Solver validation and investigation of AMR efficiency

We validate the P3D solver implementation using the well-known constant discharge rate, isothermal, tests of Rahimian *et al.*,[77] which have been previously considered in various works.[77–79] As in Part I of this paper,[29] these tests are typically simulated using the P2D model, where the electrode-pair domain is one-dimensional. To validate our P3D solver, where the electrode-pair domain is two-dimensional, we calculate solutions in two configurations: one with the electrode-pair direction oriented horizontally (grid aligned) and the other oriented diagonally at 45° to the horizontal (non-grid-aligned), with the solution in both cases compared with the P2D solution of Rahimian *et al.* The simulations are carried out under various constant discharge rates ranging from 1C to 5C. Table II summarizes the material parameters used for this study, which are comparable to those of an LCO/graphite lithium-ion cell.

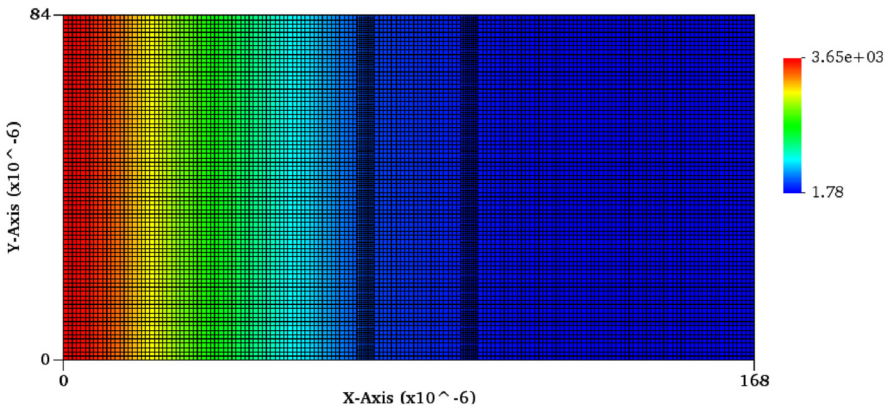
AMR computational meshes for the grid-aligned and non-grid-aligned configurations are shown in Fig. 10, in which the meshes are plotted along with computed solutions for the electrolyte concentration. Note that for the diagonal case, we embed the rotated electrode-pair domain into a background Cartesian mesh, illustrated by the gray region in Fig. 10(b). This configuration requires additional modeling consideration to deal with the interfaces between the battery cell and its surrounding background mesh, which is treated as a region of inactive material to replicate the effects provided by domain boundary conditions in the grid-aligned cases. We use the cut-cell method (see Sec. III A 3) to treat these boundaries, with AMR used near the material boundaries to improve the solution accuracy. Note that this technique may also be used for solving problems that involve more complex, non-grid-aligned battery boundary shapes, such as those in 3D batteries.[83] While studying those problems is beyond the scope of this work, the results presented here serve to illustrate that our solver has the capability to be used to study such problems as well.

Both non-AMR and AMR grids are used to investigate the accuracy and efficiency of the solver. For the horizontal configuration, an initial reference solution is computed on a uniform, unigrid mesh of 320 × 160 cells on the electrode-pair domain and 16 cells on the particle domain. To investigate the effect on solution accuracy and efficiency of AMR on the electrode-pair domain, we also compute solutions on a 160 × 80 cell grid with one level of AMR of refinement factor 2 and on a 80 × 40 cell grid with two levels of AMR with refinement factor 2 on each level. The AMR grids, therefore, have the same "effective resolution" as the unigrid setup.

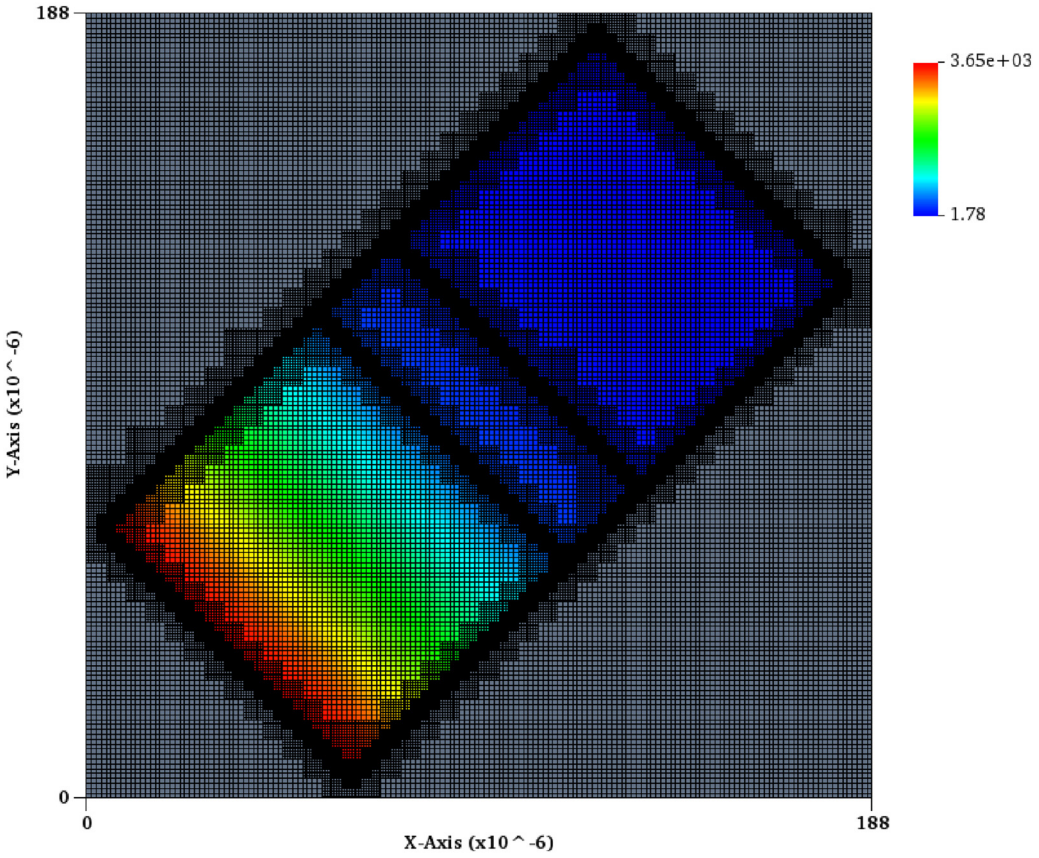
The results presented in Fig. 10 show that the solutions from the two configurations are consistent with each other. To investigate this more closely, we plot in Fig. 11(a) the computed cell

TABLE II. Material parameters used in the constant discharge solver validation test.

Parameter	Cathode	Separator	Anode	Reference
Thickness L (μm)	70	25	73.5	77
Radius of solid particle R_s (μm)	8.5	…	12.5	77
Specific interfacial area of electrode particles a_s (m^{-1})	1.941×10^6	…	1.212×10^5	80
Porosity ε_e	0.4382	0.43	0.3	77
Bruggeman coefficient brugg	1.5	2.3	4.1	77
Diffusivity in solid particle D_s(m^2 s^{-1})	1×10^{-11}	…	5.5×10^{-14}	77
Maximum solid phase concentration $c_{s,max}$(mol m^{-3})	51 555	…	30 555	77
Initial state of charge for discharge $c_{s,max}$(mol m^{-3})	0.465	…	0.756	77
Reaction rate constant k_0($m^{2.5}$ $mol^{-0.5}$ s^{-1})	3.437×10^{-6}	…	5.390×10^{-7}	80
Effective solid phase electronic conductivity σ_s (S m^{-1})	10	…	100	77
Open circuit potential $U_{s,m}$ (V)	Ramadass *et al.*	…	Ramadass *et al.*	81
Li^+ transference number in the electrolyte t_+	0.435			77
Thermodynamic factor $\partial \ln f_{\pm}/\partial \ln c_e$	0			77
Initial electrolyte concentration $c_{e,0}$(mol m^{-3})	1000			77
Intrinsic electrolyte diffusivity $D_{0,m}$ (m^2 s^{-1})	Valøen and Reimers			82
Intrinsic electrolyte conductivity $\kappa_{0,m}$(S m^{-1})	Valøen and Reimers			82
Temperature T (K)	298.15			77
Ideal gas constant R (J mol^{-1} K^{-1})	8.314			
Faraday's constant F (C mol^{-1})	96 487			
Cell capacity per active area (Ah m^{-2})	19.667			79



(a)



(b)

FIG. 10. Computed electrolyte concentrations (unit: mol m^{-3}) at T = 576 s for the 5C discharge rate test, with computational meshes overlaid. (a) Electrode-pair direction oriented horizontally (grid-aligned). The simulation is performed using a base resolution of 160 × 80 cells refined with one level of AMR and a refinement ratio of 2. (b) Electrode-pair direction oriented at 45° to the horizontal (non-grid-aligned). The simulation is performed using a base resolution of 160 × 160 cells refined with two levels of AMR of refinement ratio 2 each.

voltage for the grid-aligned and non-grid-aligned configurations. As may be seen, the calculated solution is the same in all cases, confirming that the accuracy of the solver is not dependent on the alignment of the grid. Figure 11(b) shows a comparison of the non-AMR and AMR results for the horizontal configuration, with results from the literature. As may be seen, the computed results are consistent with each other and with those from the literature. Taken together, the results of Figs. 10 and 11 serve as validation of the solver and show that the computed results from all configurations, both with and without the use of AMR, are in good agreement with the solutions reported in the literature.

The results from the horizontal configuration are used to further investigate, in a quantitative manner, the effect of the use of AMR. The computed 320 × 160 cell unigrid solution is used as a reference to compare the accuracy and efficiency of the AMR solutions against, as shown in Table III. The computational times reported in the table are all from single CPU core simulations (i.e., without any parallelization), using a fixed simulation time step Δt of 1 s. The accuracy of the AMR solutions is quantitatively measured by calculating the values of the root mean square percentage error (RMSPE) for the computed cell voltage profile, compared to the reference unigrid simulation results. As may be seen, the RMSPE values for all the AMR simulations are small and less than 0.004% in all cases. To quantitatively compare computational efficiency, we measure the factor by which the AMR simulation wall-time is reduced when compared to the reference unigrid case; these values are shown in the fifth column titled "Speed-up" of Table III. It may be seen that the AMR simulations offer a significant saving in computational cost over the unigrid simulations in all cases. Furthermore, AMR simulations with the use of a lower base-level resolution achieve greater efficiency gains. These efficiency gains increase with the operating rate, with the highest reduction in wall-time (speed-up of approximately 6.86 times) achieved by the use of AMR occurring in the 5C rate test case. This is due to the increasingly non-linear nature of the solutions at higher operating rates, where more iterations are required to determine the potential fields and the volumetric current density,[78] and consequently, a greater number of linear systems have to be solved within each time step. This makes the 5C rate test the most numerically challenging among all test cases. The linear systems arising from the system discretization in the AMR simulations are solved more efficiently by the multigrid solver compared to those discretized on the uniform grids, as a result of which a greater reduction in wall-time can be achieved through the use of AMR. The results clearly demonstrate the benefits of using AMR for multi-dimensional problems.

B. Investigating the use of AMR for resolving separator geometries

To demonstrate the capability of the solver to tackle problems involving separator geometries, we devise a test problem involving an idealized separator geometry, as shown in Fig. 12. The initial conditions and material parameters used are the same as those used in Sec. IV A. Note that results for more realistic and complex separator geometries are demonstrated later in Sec. IV D.

As discussed earlier, higher operation rate tests are more numerically challenging to solve for; we, therefore, focus on presenting results only for the most challenging 5C discharge rate test. Simulations are carried out with and without the use of AMR on the electrode-pair domain, using the same effective resolution of 512 × 256 cells in all cases, as summarized in Table IV. All other settings remain the same as in the benchmark validation test described in Sec. IV A. The simulations for this section are all performed on a single CPU core and are run to a time of 500 s.

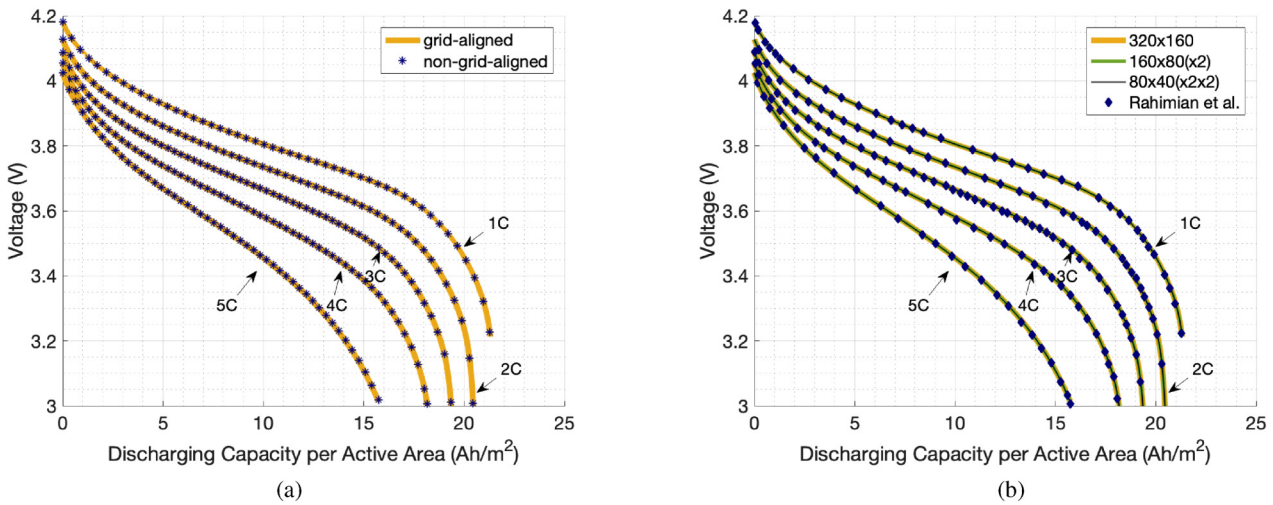


FIG. 11. Computed results for the constant discharge rate tests. (a) Comparison of computed cell voltages from simulations with the electrode-pair direction oriented horizontally (grid-aligned) and at 45° to the horizontal (non-grid-aligned). (b) Comparison against the results of Rahimian *et al.*,[77] of computed cell voltages from simulations with and without the use of AMR.

TABLE III. Quantitative assessment of solution accuracy and computational efficiency for the solver validation tests, simulated with and without the use of AMR. The speed-up is measured against the computational time of the unigrid case. The "RMSPE in V(%)" represents the root mean square percentage error calculated using the voltage profile, estimated with respect to the unigrid results.

Discharge rate (C)	Mesh	Equivalent resolution	Computational time (s)	Speed-up	RMSPE in V (%)
1	320 × 160	320 × 160	7829.7	…	…
1	160 × 80 (×2)	320 × 160	2783.2	2.81	2.29×10^{-5}
1	80 × 40 (×2×2)	320 × 160	2049.2	3.82	1.44×10^{-4}
2	320 × 160	320 × 160	3665.2	…	…
2	160 × 80 (×2)	320 × 160	1235.2	2.97	8.07×10^{-5}
2	80 × 40 (×2×2)	320 × 160	918.8	3.99	4.40×10^{-4}
3	320 × 160	320 × 160	3198.5	…	…
3	160 × 80 (×2)	320 × 160	755.0	4.24	2.91×10^{-4}
3	80 × 40 (×2×2)	320 × 160	560.0	5.71	0.001
4	320 × 160	320 × 160	2915.5	…	…
4	160 × 80 (×2)	320 × 160	648.2	4.50	6.07×10^{-4}
4	80 × 40 (×2×2)	320 × 160	475.7	6.13	0.003
5	320 × 160	320 × 160	3295.2	…	…
5	160 × 80 (×2)	320 × 160	588.9	5.60	6.75×10^{-4}
5	80 × 40 (×2×2)	320 × 160	480.4	6.86	0.004

Table IV summarizes the computational efficiency and solution accuracy of the tests conducted, with the Speed-up and RMSPE values for the AMR simulations measured against the unigrid case. It is evident that not only are the RMSPE values in the voltage profiles low for the AMR simulations, they also offer a significant reduction in computational time over the unigrid simulation. The simulation using 4 levels of AMR, for example, is able to achieve a speed-up of almost 25 times over the unigrid case, with the RMSPE value remaining low at approximately 0.023%.

The various AMR results reported in Table IV demonstrate that our solver evidently supports the use of multiple levels of AMR, providing users with flexibility in selecting the number of levels of AMR and the refinement ratio used at each grid level, depending on the problem at hand. For the same effective resolution, the results suggest that computational time generally decreases when a coarser base-level mesh is used for the simulation, at the cost of reduced solution accuracy, as expected.

The results clearly demonstrate the importance of using AMR for problems involving complex geometries. The geometry selected

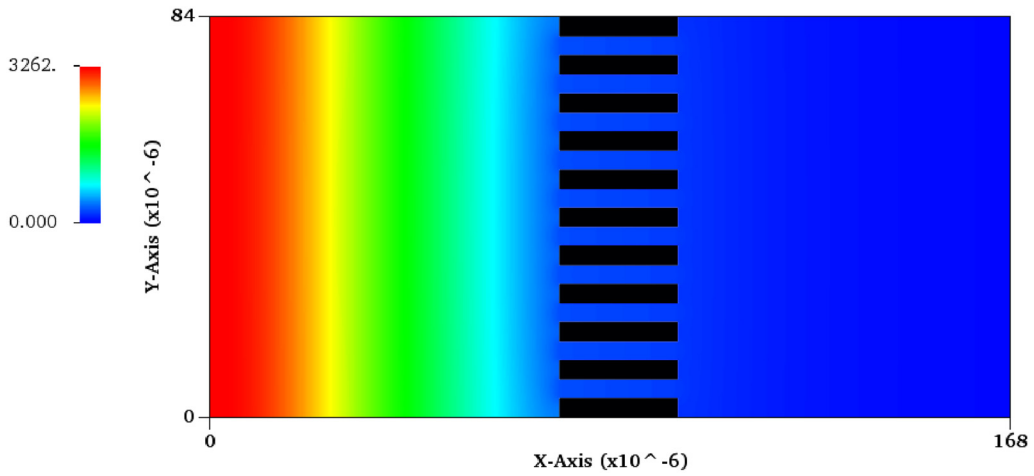


FIG. 12. Computed electrolyte concentration distribution at T = 500 s (unit: mol m^{-3}) for the 5C discharge rate test, in which an idealized separator geometry is used. The geometries colored in black represent the idealized geometry of the separator membrane.

TABLE IV. Quantitative assessment of solution accuracy and computational efficiency for the 5C discharge rate test involving the separator geometry modeled by an idealized geometry, simulated with and without the use of AMR. The speed-up is measured against the computational time of the unigrid case. "RMSPE in V(%)" represents the root mean square percentage error calculated using the voltage profile, relative to the unigrid result.

Mesh	Equivalent resolution	Computational time (s)	Speed-up	RMSPE in V (%)
512 × 256	512 × 256	34 167.2	…	…
256 × 128 (×2)	512 × 256	12 607.8	2.71	0.002
128 × 64 (×4)	512 × 256	8737.4	3.91	0.004
128 × 64 (×2×2)	512 × 256	2984.5	11.45	0.006
64 × 32 (×4×2)	512 × 256	2983.4	11.45	0.008
64 × 32 (×2×4)	512 × 256	3295.0	10.37	0.009
64 × 32 (×2×2×2)	512 × 256	1429.4	23.90	0.010
32 × 16 (×2×2×2×2)	512 × 256	1373.7	24.87	0.023

for this study was relatively simple and was chosen primarily to demonstrate solver capability. On the other hand, for scenarios involving realistic complex geometries that require a fine mesh to resolve material interfaces (see Sec. IV D 3 for an example), a simulation using a unigrid mesh is likely to have a prohibitive computational expense. The use of AMR can make such simulations practical, by allowing the fine resolution to be used in a targeted manner only where needed, such as at material interfaces.

C. Parallel computing

As described in Sec. III D, our solver has the capability to be run in parallel across multiple processors. To assess the parallel performance of the code, we repeat the test described in Sec. IV B, where the separator geometry is resolved. The solver is run using multiple CPU cores to examine the strong scaling performance of the code, i.e., to examine the speed-up in computational time achieved as more cores are employed. Tests using four different mesh resolutions are used in this study: a coarse grid test using 256 × 128 unigrid cells, a moderate grid test using 512 × 256 unigrid cells (i.e., the same resolution used in Sec. IV B), a test using 256 × 128 base-level cells with two levels of AMR of refinement ratio 2 each, and a fine grid test using 512 × 256 base-level cells with one level of AMR and a refinement ratio of 2. Note that the AMR tests use an effective resolution of 1024 × 512 cells. All the tests were run to a time of 100 s.

Table V shows a comparison of the "Time per time step" and the "Speed-up" (compared to the single-core result) for the tests performed. As may be seen, the runtime per time step decreases with increasing numbers of cores at all resolutions. However, for a fixed problem size, the speed-up obtained from the use of additional cores decreases as more cores are used. For the low-resolution case, for example, the runtime per time step levels off at 32 cores and increases slightly when using 64 cores. This occurs when a core limit is reached, at which point the MPI communication overhead begins to outweigh any reduction in compute time, and becomes too large for further parallelization.[73] The core limit rises with problem size; for the 512 × 256 base resolution simulations, the runtime per time step for the unigrid and AMR simulations continues to decrease when going from 32 to 64 cores, which is the maximum number of cores used in this study.

The other evidence of better strong scaling performance for larger problem sizes may be observed in the speed-up results. For the 512 × 256 base resolution AMR simulation, the speed-up of approximately 3.96 achieved with 4 cores represents almost ideal strong scaling performance. For the subsequent doubling of cores from 4 to 8, 8 to 16, 16 to 32, and 32 to 64 cores, the compute time decreases by approximately 1.34, 1.64, 1.62, and 1.24 times, respectively, which represents good strong scaling performance.

The results reveal that a relatively small number of CPU cores are sufficient to obtain good computational performance for problems of this size. The value of the parallelization is more significant for more complicated problems, such as that of the realistic separator geometry presented later in Sec. IV D 3, where the fine resolution required to resolve the geometry results in a much greater problem size. Furthermore, it is expected that the parallelization will be essential for practical modeling of even larger, three-dimensional problems. Studying these problems is beyond the scope of this work, although it may be noted that the techniques we have introduced are readily extendable to three-dimensions.

TABLE V. Strong scaling results for the 5C discharge rate test, involving a separator geometry modeled by an idealized geometry, run on multiple CPU cores. All tests were run to a time of 100 s.

	Number of cores	1	2	4	8	16	32	64
256 × 128 cells	Time per time step (s)	6.48	4.56	2.57	1.53	1.18	0.89	0.95
	Speed-up	…	1.42	2.52	4.23	5.49	7.25	6.86
512 × 256 cells	Time per time step (s)	51.11	29.82	14.49	10.26	6.10	3.93	3.18
	Speed-up	…	1.71	3.53	4.98	8.38	13.01	16.07
256 × 128 (×2×2) cells	Time per time step (s)	28.00	15.39	9.11	5.73	4.09	3.29	3.30
	Speed-up	…	1.82	3.07	4.89	6.85	8.52	8.47
512 × 256 (×2) cells	Time per time step (s)	111.72	58.58	28.19	21.07	12.86	7.96	6.41
	Speed-up	…	1.91	3.96	5.30	8.69	14.04	17.44

D. Investigation of separator microstructure

To represent separator microstructure, the usual practice with existing models is to use the Bruggeman factor, an empirical relation, to describe the microstructural characteristics of the separator. The capabilities of our solver, on the other hand, allow the separator microstructure to be directly resolved. This allows an investigation to be performed into the effects of separator microstructure on battery cell electrochemical performance, an aspect that has not been studied numerically in the literature. In this study, we use the same initial conditions and material parameters as those used in the reference test of Sec. IV A. Tests are conducted under a high operating rate (5C discharge rate), where ionic transport becomes a limiting process, and the effects of separator microstructure become more evident.[84] A series of idealized separator microstructures are investigated to quantitatively assess how the morphological characteristics of the separator membrane affect the electrochemical performance of a battery cell. We use parallel, straight channels to study the effect of pore size in Sec. IV D 1 and tapered channels to investigate constrictivity in Sec. IV D 2. In these investigations, the porosity and thickness of the separator region are kept constant, with variations made solely to the separator geometry to examine these effects.

The simulations are conducted on 2D representations of the separator membrane geometries. While this does not take into account 3D effects, such as connectivity in the 3D pore network,[56] our work serves as a first step toward understanding how separator microstructure affects battery performance, an effect that has received little attention in the literature. Furthermore, the use of 2D representations of the separator geometries is compatible with the aim of our study, which is to investigate the influence of specific morphological characteristics on cell performance. In addition to serving as a demonstration of the capability of our solver, we believe that insights gained from these numerical investigations can be of significant value in guiding the design of next-generation separators. We conclude the study by presenting in Sec. IV D 3 the results of a showcase study of realistic separator geometries modeled based on the real microstructure of a commercial polypropylene (PP) separator. Such a problem has not previously been numerically studied within a complete model of battery physics that includes all battery components (anode, separator, and cathode) and accounts for the electrochemical processes taking place within a lithium-ion cell. For the investigations in this section, we assess battery performance quantitatively using the cell voltage profile, while the concentration and the potential distribution of the electrolyte are presented to describe ionic transport performance.

1. Idealized separator microstructure composed of straight parallel channels

For this study, we use a series of idealized separator microstructures composed of parallel, straight channels, with constant bulk porosity (i.e., the total porosity of the separator region remains constant), to investigate the effect of pore size on battery performance. The use of parallel, straight channels also ensures that tortuosity remains constant across all configurations considered, thereby isolating the effect of pore size from that of tortuosity. Figure 13 shows the computed cell voltage and line plots for electrolyte phase concentration and potential for the various cases performed. Computed two-dimensional electrolyte concentration distributions are plotted in Fig. 14, where the separator geometries plotted in black also serve to illustrate the configurations studied.

Figure 13(a) shows a comparison of the computed cell voltage for the various cases simulated, with the benchmark result of Sec. IV A that was obtained with the use of homogenized parameters to model the separator region. It is evident that the delivered capacity increases with the number of channels, suggesting that cell performance improves with a reduction in the separator pore size. As depicted by the electrolyte concentration distributions in Fig. 14, the ionic transport performance is affected by the pore size of the separator. This effect is not limited to the separator region; a larger pore size can also lead to a less uniform electrolyte concentration distribution in the electrode material regions. Such nonuniform diffusion may result in uneven lithium plating, which, in turn, can provoke dendrite growth.[46] A closer look at the electrolyte concentration distributions inside the separator region reveals that a more uniform concentration distribution can be achieved by reducing the pore size of the separator. This finding is further supported by the results in Fig. 13(b), which present line plots of the electrolyte concentration and potential profiles at mid-height. The lowest ohmic drop and the smallest concentration gradient across the separator thickness occur in the case with multiple channels, corresponding to a separator pore size of 0.86 μm. Given that smaller variations in the electrolyte phase concentration and potential distributions correspond to higher ionic mobility and ionic conductivity, respectively, these results further confirm that a smaller pore size is preferable.

While the results indicate that battery performance gets improved as separator pore size decreases (for a given porosity in the separator region), the effect appears to weaken when the pore size is relatively small. This is likely because of the effectiveness of the low tortuosity parallel straight channels in facilitating ionic transport in the separator region such that below a certain pore size, ionic transport is no longer the limiting factor for battery performance. It should be noted, however, that a larger average pore size may be preferred in some cases to ensure that the paths for ion transport stay connected under loads.[85] Note that the conclusions from this investigation apply to cases where the nominal pore size of separators ranges from hundreds of nanometers to several micrometres[86,87] and do not account for limitations arising from electrolyte wetting in cases with ultra fine pores.[88] In such cases, incomplete electrolyte filling can become a limiting factor that significantly affects battery performance.[89] The study of this class of problems is beyond the scope of the present work. Given the conflicting roles of separator pore size in electrochemical performance and mechanical robustness, developing a mechanical–electrochemical model to determine the optimal pore size is of practical significance and will be considered in a future work.

2. Idealized microstructure composed of tapered channels

A series of idealized microstructures, composed of tapered channels, was created to investigate the impact of the constrictivity

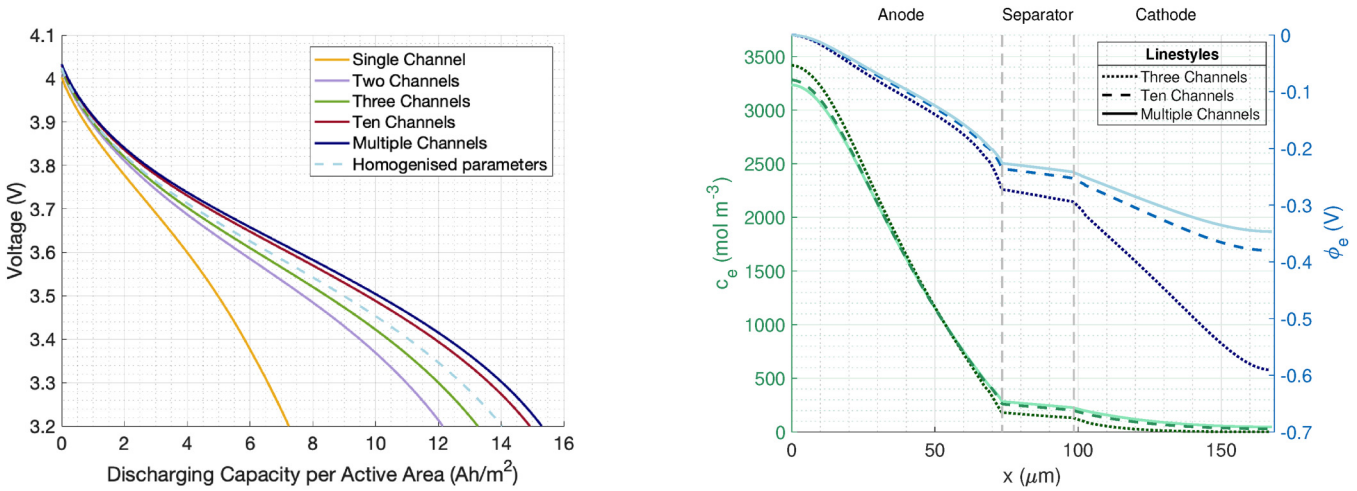


FIG. 13. Computed results for the 5C discharge rate test simulated using various idealized separator geometries composed of straight parallel channels. (a) Computed cell voltage profiles. The blue dashed curve represents the result of the reference test simulated with the use of homogenized parameters for the separator region. (b) Computed spatial distributions of electrolyte concentration and electrolyte potential around the mid-height of the separator at T = 512 s.

of the separator geometry on battery performance. Constrictivity χ is defined as the ratio of the cross-sectional area between A_{out} and A_{in},

$$\chi = \frac{A_{out}}{A_{in}}, \tag{14}$$

where A_{out} and A_{in} are the cross-sectional areas of the pore adjacent to the anode and cathode, respectively.[46] An illustration of the definition of A_{out} and A_{in} can be found in Fig. 15(a).

As shown by the delivered capacity vs constrictivity plot presented in Fig. 15(b), battery performance appears to be nearly independent of the microstructure constrictivity, consistent with the findings reported by Cannon and Ryan.[46]

3. Realistic microstructure

We conclude the investigation into the effects of separator microstructure by performing two simulations using separator geometries based on the realistic microstructure of the Celgard 2500 separator, a commonly used commercial separator made of polypropylene (PP). The computed results also serve to further demonstrate the capability of our solver and the techniques presented in this paper to resolve complex, realistic geometric configurations in the context of battery simulation.

In the first test, we use the separator microstructure reconstructed by Xu *et al.* using 2D SEM and 3D XCT data.[56] Figure 16 illustrates the geometries used in our simulation, which were created with minor adjustments to the microstructure reconstructed by Xu *et al.* to ensure open pores for 2D simulation.

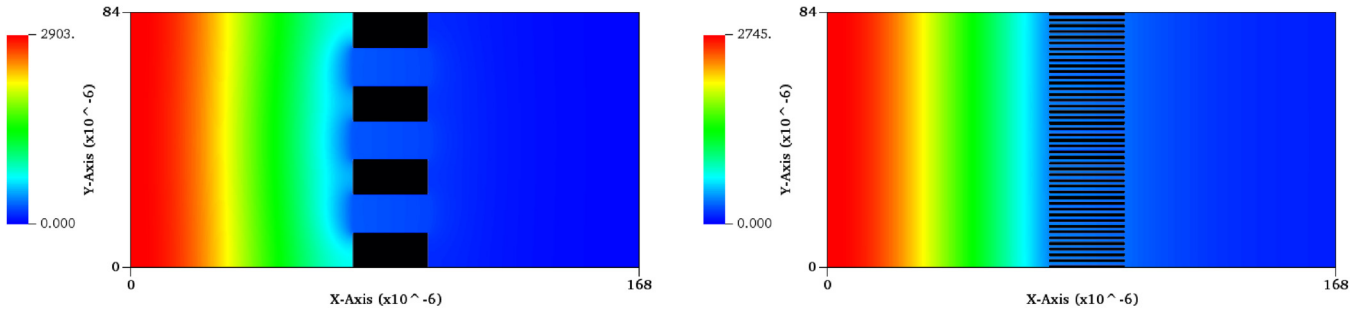


FIG. 14. Electrolyte concentration plots (unit: mol m^{-3}) at T = 384 s for the 5C discharge rate test, simulated using various idealized separator geometries. The geometries colored in black represent the idealized separator microstructure. (a) Three straight parallel channels. (b) Multiple straight parallel channels with a pore size of 0.86 μm.

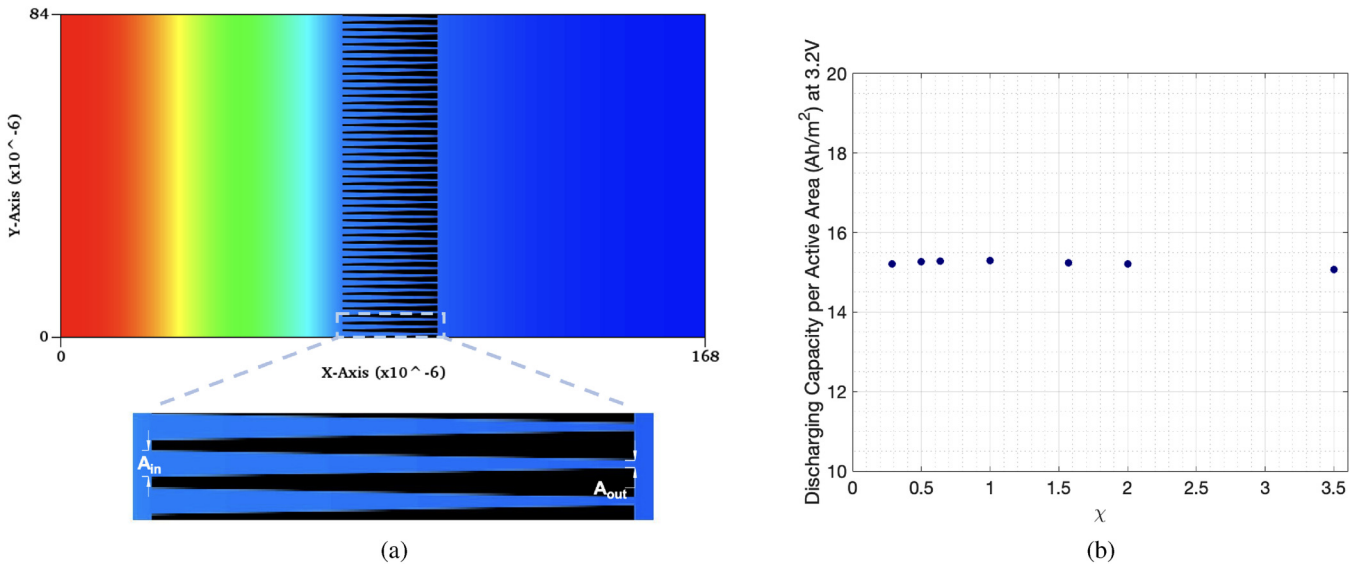


FIG. 15. Computed results for the 5C discharge rate test simulated using idealized separator geometries with varying constrictivity. (a) Illustration of the definition of constrictivity. (b) Computed discharging delivered capacity at a cell voltage of 3.2 V.

Such modifications, commonly applied to adapt real microstructures for 2D modeling, do not alter the bulk physical characteristics of the separator, as demonstrated by Cannon and Ryan[46] (who base their work on the microstructure reconstructed by Lagadec[90]). While inevitable discrepancies may arise from these edits, this simulation serves as a first step toward accurately resolving the realistic separator microstructure within a complete model of battery physics. Extending our solver to perform full 3D simulations is part of our future work. It is also worth noting that the microstructure provided by Xu *et al.*[56] has a separator thickness of 15 μm, whereas our benchmark test uses a separator with a thickness of 25 μm. To match the thickness of the benchmark test, we extend the microstructure by Xu *et al.* for $x \in [15, 25]\,\mu$m using a mirror image, symmetric around $x = 15\,\mu$m. The porosity abstracted from this realistic microstructure is approximately 0.63, which is higher than the value in the benchmark test. Nevertheless, it still falls within the typical range for a commercial separator, which generally has a porosity between 0.2 and 0.8 and a thickness of the order of 25 μm.[45]

The real microstructure of the separator is extremely complex, with intricate features at small length scales, introducing numerous challenges for numerical studies. Conventional numerical techniques often struggle with this due to the cumbersome and time-consuming process of generating high-quality conformal meshes. This challenge may explain why such problems have not been previously investigated numerically. The solver we have developed uses a Cartesian grid for the entire electrode-pair domain. With arbitrarily complex geometries embedded in the grid, the mesh generation process is carried out automatically and rapidly. The fact that the governing equations are discretized and solved on the AMR hierarchy significantly improves computational efficiency. The computational cost is further reduced through parallel computing. These features enable our solver to effectively resolve the realistic microstructure of the separator within a complete model of battery physics. We simulate the test with the use of a fixed time step size of 4 s, on a grid that uses a base resolution of 1380 × 60 cells and one level of AMR with refinement ratio 4 on the electrode-pair domain, and 16 unigrid cells on the particle domain. As a result, the separator geometry is resolved with an effective grid size of approximately 30 nm. The simulation completed in approximately 9.7 min of wall-time on 64 CPU cores, showcasing the potential of our solver in being efficient enough to be used as a design tool for optimizing sophisticated separator microstructures.

It is evident that the electrolyte concentration distributes non-uniformly across the y direction, as shown in Figs. 16(a) and 16(b). This non-uniformity is dictated by the complex separator microstructure, which affects regional ionic transport. Such a phenomenon could not be captured previously using homogenized parameters simulated via the P2D model, although these regional behaviors can be crucial to performance when the separator operates under extreme conditions. On the other hand, our solver successfully predict battery performance and its internal states *in operando*, offering a more comprehensive understanding of the regional behaviors inside the separator.

Figure 16(c) presents a comparison of the computed cell voltage from this test with the two predictions made using the P2D model with homogenized parameters. These two P2D tests are conducted using the same bulk porosity as that abstracted from the realistic separator microstructure, but with Bruggeman factors of 1.5 and 2.3, respectively. These two Bruggeman factors are selected because 2.3 is the value used in the benchmark test, while 1.5 is a commonly used choice in the literature.[46,91] As shown in Fig. 16(c),

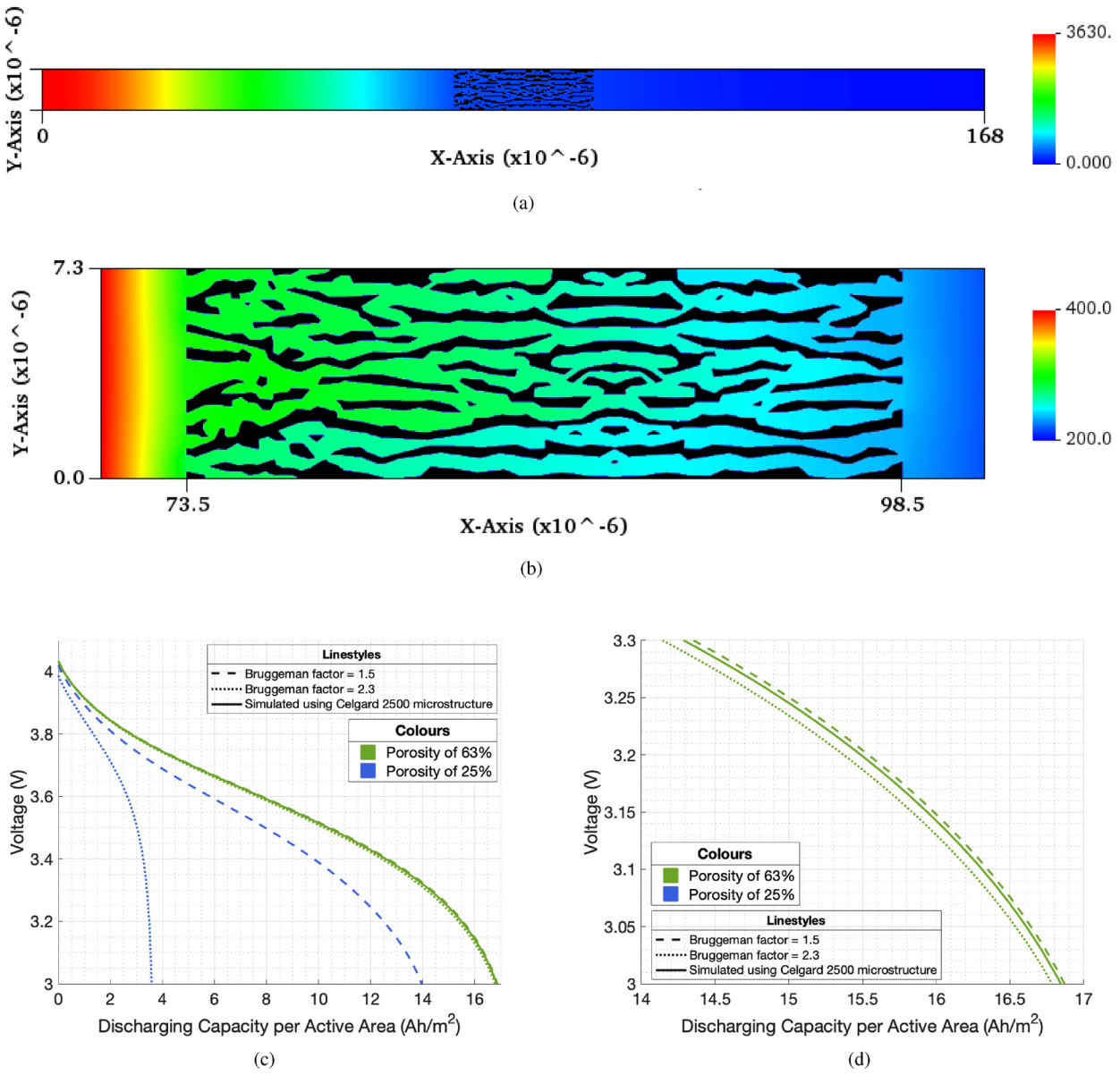


FIG. 16. Computed results for the 5C discharge rate test, simulated using a reconstructed microstructure of the Celgard 2500 separator with a porosity of 0.63. The geometries colored in black represent the separator membrane. (a) Electrolyte concentration plots (unit: mol m^{-3}) at T = 560 s. (b) Close-up of electrolyte concentration (unit: mol m^{-3}) in the separator region at T = 560 s. (c) Comparison against the results obtained using homogenized material parameters for the separator region, of the computed cell voltage from the simulation resolving a reconstructed microstructure of the Celgard 2500 separator. (d) Close-up of the cell voltage profiles near the end of the discharge.

the computed cell voltage for the case that uses the explicit representation of the realistic separator microstructure falls between the two curves predicted by the P2D model. This indicates that the results of our P3D simulation with the separator resolved are physically representative. Note that while the differences between the solutions simulated using the two different Bruggeman factors are small for this case, this will not always be the case in practical scenarios if, for example, a higher discharge rate is used, or if the porosity is lower. For instance, the blue lines in Fig. 16(c) show the results simulated with the same Bruggeman factors, but at a lower porosity of 0.25, where a much larger difference between the results simulated with Bruggeman factors of 1.5 and 2.3 is observed. This

result indicates that the Bruggeman factor has an increased impact on cell performance as porosity decreases. Since the Bruggeman coefficient reflects the tortuosity, which in turn corresponds to the morphology of the separator microstructure, the result further highlights the importance of understanding the correlation between the separator microstructure and battery performance, especially for separators with low porosity.

In the second test conducted, we use the Celgard 2500 separator microstructure rendered by Finegan *et al.*,[92] who employed a Zernike phase contrast setting in lab-based nano-scale X-ray microscopy to image microstructural characteristics. As illustrated in Fig. 17(a), the separator geometry used in this simulation has a porosity of 0.53, matching the porosity reported by Finegan *et al.*[92] We chose the domain size in the y direction to be the same as that used in the first test. According to the representative volume element analysis conducted by Finegan *et al.*,[92] this size is sufficiently large to be representative of the bulk material.

Like in the first test, the computed cell voltage profile simulated using the Celgard 2500 separator microstructure, shown in Fig. 17(b), sits between two bounds predicted using homogenized parameters with Bruggeman factors of 1.5 and 2.3. This provides further evidence that the computed results from the P3D simulation with the separator microstructure resolved are physically representative. The voltage curve of this test is relatively close to the one predicted with a Bruggeman factor of 1.5, revealing that the tortuosity of the microstructure involved in this test is low. This inference is in line with the tortuosity value of 1.43 reported by Finegan *et al.*[92] for the Celgard 2500 microstructure.

As shown in Fig. 17(a), the electrolyte concentration distribution in this test exhibits a lower maximum concentration value compared to that of the first test, depicted in Fig. 16(a). The discrepancies between the two tests arise primarily from differences in the porosity of the separator microstructures, even though both are based on the Celgard 2500 separator. This is due to the fact that the two microstructures were imaged and reconstructed by different groups using different techniques. The microstructure used in the second test is reconstructed using an imaging technique that struggles to identify nano-fibrils in the separator membrane due to

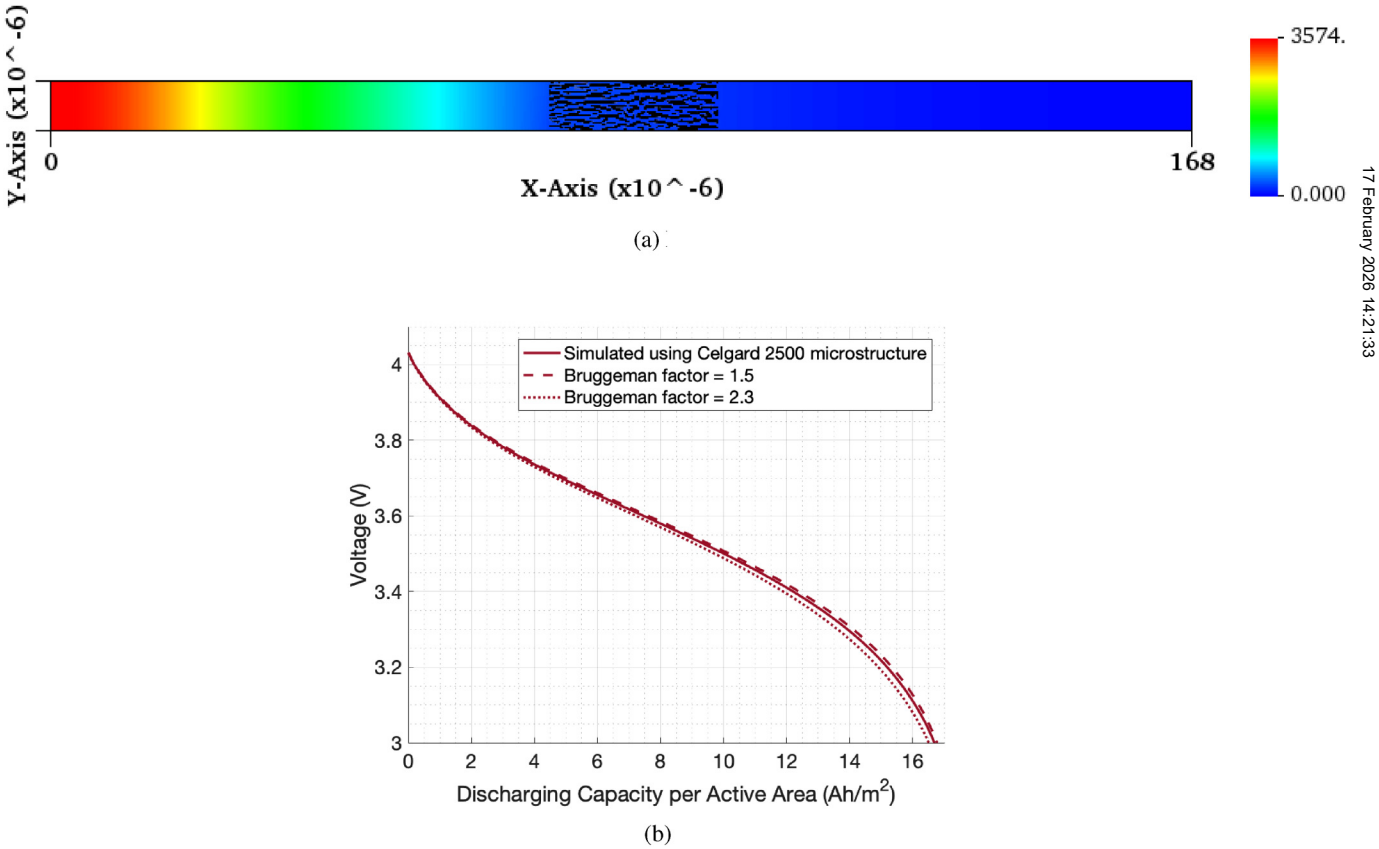


FIG. 17. Computed results for the 5C discharge rate test, simulated using the microstructure of the Celgard 2500 separator with a porosity of 0.53. The geometries colored in black represent the separator membrane. (a) Electrolyte concentration plots (unit: mol m^{-3}) at T = 560 s. (b) Comparison against the results obtained using homogenized material parameters for the separator region, of the computed cell voltage from the simulation resolving a reconstructed microstructure of the Celgard 2500 separator.

systematic resolution limitations.[92] As a result, in the reconstructed microstructure, nano-fibrils merge with the lamellae geometries, resulting in a porosity that closely matches the value stated in the manufacturer's specifications. In contrast, the microstructure used in our first test, reconstructed by Xu *et al.*[56] using 2D SEM and 3D XCT data, managed to separate fibrils and lamellae geometries. However, as the current work models the electrode-pair domain in two dimensions, fibrils are not included, resulting in a lower porosity than that of the microstructure used in the second test. In our ongoing research, we plan to extend the current model to three dimensions, which will enable the prediction of battery performance with the separator geometry modeled using 3D image-based microstructures.

V. CONCLUSIONS

In this paper, we presented a parallelizable, hierarchical AMR-compatible, Cartesian immersed interface solver that is suitable for solving multi-dimensional battery models. Diffuse and sharp (cut-cell) Cartesian immersed interface methods were introduced to model interactions at material interfaces, allowing for the mesh generation process to be carried out rapidly, with arbitrarily complex geometries embedded into the Cartesian grids. To reduce the computational expense involved in multi-dimensional simulations, the solver integrates seamlessly with AMR, which allows the use of a coarse mesh covering the entire computational domain, with local grid refinement applied only where needed, such as near the material interfaces. The solver was parallelized with the use of MPI, making it compatible be run on a single-core processor as well as on massively parallel supercomputers. While our solver was designed to be applicable for solving general multi-dimensional battery models, this work uses the P3D model, with a focus on resolving the microstructure in the separator region, as a working example to demonstrate the advantages of using the aforementioned numerical techniques in handling multi-dimensional battery simulations. The model we chose allows for the study of two-dimensional variations on the electrode-pair domain, with the microstructures of separator membranes being carefully resolved to investigate their effects on the battery electrochemical performance, an aspect that has received little attention in the literature. The governing equations of the P3D model were discretized using the finite volume method and solved implicitly using an AMR-compatible multigrid solver, with the sequential SFOM approach.

To validate the solver, we computed solutions to the well-known constant discharge rate, isothermal tests of Rahimian *et al.*[77] in two configurations: one with the electrode-pair direction oriented horizontally and the other oriented diagonally. The computed results in all three cases were consistent with the P2D solutions from the literature. The simulation of the diagonal case demonstrated that the solver is capable of solving problems that involve complex, non-grid-aligned battery boundary shapes, such as those in 3D batteries. The use of AMR was systematically explored through the test with the electrode-pair direction orientated horizontally, as well as through another test that resolved separator geometries. In both cases, we demonstrated that the use of AMR provided speed-ups in compute time while preserving solution accuracy, making simulations that would otherwise be computationally prohibitive with a unigrid mesh practical. The test that resolved separator geometries was also used to study the strong scaling parallel performance of the solver. For an AMR simulation, the solver demonstrated almost ideal strong scaling performance when run on two and four cores. For the subsequent doubling of cores from 4 to 8, 8 to 16, 16 to 32, and 32 to 64 cores, the compute time decreased by approximately 1.34, 1.64, 1.62, and 1.24 times respectively, resulting in an overall speed-up of 17.4 times when run on 64 cores, compared to the wall-time spent on a single-core simulation. It was concluded that the value of the parallelization was of greater significance for larger problem sizes, such as those involving the use of fine resolutions to resolve intrinsic geometric features, or indeed for three-dimensional scenarios. While the latter were beyond the scope of this work, it may be noted that the techniques we have introduced are readily extendable to three-dimensions.

We used our solver as a tool to directly investigate the effects of separator membrane microstructure on the electrochemical performance of lithium-ion batteries. To the best of our knowledge, this is the first time such an investigation has been achieved in the computational community. Through simulations with idealized separator geometries composed of straight parallel channels, it was found that for a fixed porosity and tortuosity in the separator region, battery performance gets improved with a reduction in pore size. The constrictivity of the separator geometry was found to have a nearly negligible influence on battery performance, consistent with findings reported in Ref. 46. We concluded the study by performing two tests with the separator membrane modeled based on the microstructure of a commercial separator, demonstrating that the solver is capable of performing simulations involving realistic microstructures.

While this paper has demonstrated one possible application of our solver, the ability to resolving complex microstructures provides the potential for the solver to be used in investigating battery degradation mechanisms,[93] as well as in the material selection, design, and optimization of various battery components. These studies, and the application of the solver in three dimensions, are expected to be the focus of future publications.

NOMENCLATURE

a_s	Specific interfacial area of electrode particles (m^{-1})
c_e	Electrolyte phase Li^+ concentration ($mol\,m^{-3}$)
c_s	Solid phase Li^+ concentration in electrode ($mol\,m^{-3}$)
$c_{s,max}$	Maximum Li^+ concentration ($mol\,m^{-3}$)
$c_{s,surf}$	Surface solid phase Li^+ concentration ($mol\,m^{-3}$)
D_e	Effective diffusivity of Li^+ in the electrolyte ($m^2\,s^{-1}$)
D_s	Solid phase diffusivity in the solid particles ($m^2\,s^{-1}$)
F	Faraday's constant ($96485.33\,C\,mol^{-1}$)
$f_\pm$	Electrolyte mean molar activity coefficient
H	Height in the y direction (m)
$I_{applied}$	Applied current density ($A\,cm^{-2}$)
k_0	Reaction rate constant ($m^{2.5}\,mol^{-0.5}\,s^{-1}$)
L	Thickness of the battery cell (m)
R	Ideal gas constant ($= 8.314\,J\,mol^{-1}\,K^{-1}$)
R_s	Radius of the electroactive particles (m)
r_{Li}	Reaction rate (pore wall flux) of Li^+ ($mol\,m^{-2}\,s^{-1}$)
T	Temperature (K)
t_+	Transference number of Li^+

U Open circuit potential (V)

Greek

α_a Anodic charge transfer coefficient
α_c Cathodic charge transfer coefficient
ε_e Volume fraction of an electrolyte (also known as porosity)
η Overpotential (V)
κ_e Effective ionic conductivity of an electrolyte phase ($S\,m^{-1}$)
ϕ_e Ionic potential in electrolyte phase (V)
ϕ_s Electric potential in a solid phase (V)
σ_s Effective solid phase electronic conductivity ($S\,m^{-1}$)

Subscripts

a, s, c Properties for anode, separator, and cathode, respectively.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Jiawei Lu: Conceptualization (equal); Formal analysis (lead); Investigation (lead); Methodology (equal); Software (lead); Validation (lead); Visualization (lead); Writing – original draft (lead); Writing – review & editing (equal). **Nandan Gokhale:** Conceptualization (equal); Formal analysis (supporting); Methodology (equal); Supervision (equal); Writing – original draft (supporting); Writing – review & editing (lead). **Nikolaos Nikiforakis:** Conceptualization (equal); Funding acquisition (lead); Methodology (supporting); Project administration (lead); Supervision (equal); Writing – review & editing (supporting).

DATA AVAILABILITY

The data that support the findings of this study are available within the article.

REFERENCES

[1]S. Wappelhorst, "The end of the road? An overview of combustion engine car phase-out announcements across Europe" (International Council on Clean Transportation, 2020).
[2]S. Wappelhorst and H. Cui, "Growing momentum: Global overview of government targets for phasing out sales of new internal combustion engine vehicles" (International Council on Clean Transportation, 2020), available at https://theicct.org/blog/staff/global-ice-phaseout-nov2020.
[3]G. E. Blomgren, "The development and future of lithium ion batteries," J. Electrochem. Soc. **164**(1), A5019 (2017).
[4]Z. P. Cano, D. Banham, S. Ye, A. Hintennach, J. Lu, M. Fowler, and Z. Chen, "Batteries and fuel cells for emerging electric vehicle markets," Nat. Energy **3**(4), 279–289 (2018).
[5]X. Lu, A. Bertei, D. P. Finegan, C. Tan, S. R. Daemi, J. S. Weaving, K. B. O'Regan, T. M. Heenan, G. Hinds, E. Kendrick *et al.*, "3D microstructure design of lithium-ion battery electrodes assisted by X-ray nano-computed tomography and modelling," Nat. Commun. **11**(1), 2079 (2020).
[6]W. Mai, M. Yang, and S. Soghrati, "A particle-resolved 3D finite element model to study the effect of cathode microstructure on the behavior of lithium ion batteries," Electrochim. Acta **294**, 192–209 (2019).
[7]M. Kespe and H. Nirschl, "Numerical simulation of lithium-ion battery performance considering electrode microstructure," Int. J. Energy Res. **39**(15), 2062–2074 (2015).
[8]J. Le Houx and D. Kramer, "Finite volume modelling of Li-ion composite cathodes based on micro-scale X-ray computed tomography," in *Electrochemical Society Meeting Abstracts 237* (The Electrochemical Society, Inc., 2020), Vol. 4, pp. 539–539.
[9]D. Miranda, R. Gonçalves, S. Wuttke, C. M. Costa, and S. Lanceros-Méndez, "Overview on theoretical simulations of lithium-ion batteries and their application to battery separators," Adv. Energy Mater. **13**(13), 2203874 (2023).
[10]V. Goel and K. Thornton, "Enabling the electrochemical simulation of Li-ion battery electrodes with anisotropic tortuosity in COMSOL Multiphysics®," MethodsX **8**, 101425 (2021).
[11]M. M. Forouzan, B. A. Mazzeo, and D. R. Wheeler, "Modeling the effects of electrode microstructural heterogeneities on Li-ion battery performance and lifetime," J. Electrochem. Soc. **165**(10), A2127–A2144 (2018).
[12]L. Cai and R. E. White, "Mathematical modeling of a lithium ion battery with thermal effects in COMSOL Inc. Multiphysics (MP) software," J. Power Sources **196**(14), 5985–5989 (2011).
[13]P. W. Northrop, B. Suthar, V. Ramadesigan, S. Santhanagopalan, R. D. Braatz, and V. R. Subramanian, "Efficient simulation and reformulation of lithium-ion battery models for enabling electric transportation," J. Electrochem. Soc. **161**(8), E3149 (2014).
[14]M. Chouchane and A. A. Franco, "An invitation to engage with computational modeling: User-friendly tool for in silico battery component generation and meshing," Batter. Supercaps **4**(9), 1451–1456 (2021).
[15]A. Malik and H.-C. Yu, "Complex electrode microstructure simulations using a smoothed boundary method with adaptive mesh refinement," J. Electrochem. Soc. **169**(7), 070527 (2022).
[16]G. F. Castelli and W. Dörfler, "The numerical study of a microscale model for lithium-ion batteries," Comput. Math. Appl. **77**(6), 1527–1540 (2019).
[17]M. Klinsmann, D. Rosato, M. Kamlah, and R. M. McMeeking, "Modeling crack growth during Li insertion in storage particles using a fracture phase field approach," J. Mech. Phys. Solids **92**, 313–344 (2016).
[18]J. Le Houx and D. Kramer, "OpenImpala: Open source image based parallisable linear algebra solver," SoftwareX **15**, 100729 (2021).
[19]S. Dussi and C. H. Rycroft, "Less can be more: Insights on the role of electrode microstructure in redox flow batteries from two-dimensional direct numerical simulations," Phys. Fluids **34**(4), 043111 (2022).
[20]M. D. Murbach and D. T. Schwartz, "Extending Newman's pseudo-two-dimensional lithium-ion battery impedance simulation approach to include the nonlinear harmonic response," J. Electrochem. Soc. **164**(11), E3311 (2017).
[21]S. E. Wetterauer, "Design of numerical methods for the simulation of multiscale lithium ion battery models," Ph.D. dissertation (Heidelberg University, 2020).
[22]J. Mora-Paz, "High-order transient multidimensional simulation of a thermo-electro-chemo-mechanical model for lithium-ion batteries," J. Comput. Phys. **519**, 113392 (2024).
[23]N. Gokhale, N. Nikiforakis, and R. Klein, "A dimensionally split Cartesian cut cell method for hyperbolic conservation laws," J. Comput. Phys. **364**, 186–208 (2018).
[24]H. A. Muir and N. Nikiforakis, "Numerical modeling of imposed magnetohydrodynamic effects in hypersonic flows," Phys. Fluids **34**(10), 107114 (2022).
[25]A. Zhang, S. Millmore, and N. Nikiforakis, "Thermal simulation of millimetre wave ablation of geological materials," Comput. Geotech. **161**, 105571 (2023).
[26]D. Fan, A. Nonaka, A. S. Almgren, A. Harpole, and M. Zingale, "MAESTROeX: A massively parallel low mach number astrophysical solver," Astrophys. J. **887**(2), 212 (2019).
[27]M. Zingale, A. Almgren, M. B. Sazo, V. Beckner, J. Bell, B. Friesen, A. Jacobs, M. Katz, C. Malone, A. Nonaka *et al.*, "Meeting the challenges of modeling astrophysical thermonuclear explosions: Castro, MAESTRO, and the AMReX astrophysics suite," in *Journal of Physics: Conference Series* (IOP Publishing, 2018), Vol. 1031(1), p. 012024.

[28]A. S. Almgren, J. B. Bell, P. Colella, L. H. Howell, and M. L. Welcome, "A conservative adaptive projection method for the variable density incompressible Navier–Stokes equations," J. Comput. Phys. **142**(1), 1–46 (1998).
[29]J. Lu, N. Gokhale, and N. Nikiforakis, "An immersed interface adaptive mesh refinement algorithm for Li-ion battery simulations. I. Development of a fast P2D solver," J. Appl. Phys. **138**(4), (2025).
[30]R. Mittal and G. Iaccarino, "Immersed boundary methods," Annu. Rev. Fluid Mech. **37**(1), 239–261 (2005).
[31]P. W. Northrop, M. Pathak, D. Rife, S. De, S. Santhanagopalan, and V. R. Subramanian, "Efficient simulation and model reformulation of two-dimensional electrochemical thermal behavior of lithium-ion batteries," J. Electrochem. Soc. **162**(6), A940 (2015).
[32]A. P. Monasterial, P. J. Weddle, K. Atkinson, D. S. Wragg, A. M. Colclasure, F. L. Usseglio-Viretta, N. Seitzman, J.-S. Park, J. Almer, K. Smith *et al.*, "Dynamic in-plane heterogeneous and inverted response of graphite to fast charging and discharging conditions in lithium-ion pouch cells," Small Sci. **3**(7), 2200067 (2023).
[33]T. Gao, A. Kim, and W. Lu, "Modeling electrode-level crack and quantifying its effect on battery performance and impedance," Electrochim. Acta **363**, 137197 (2020).
[34]P. Luo, P. Li, D. Ma, K. Wang, and H. Zhang, "Coupled electrochemical-thermal-mechanical modeling and simulation of lithium-ion batteries," J. Electrochem. Soc. **169**(10), 100535 (2022).
[35]C. Kupper and W. G. Bessler, "Multi-scale thermo-electrochemical modeling of performance and aging of a $LiFePO_4$/graphite lithium-ion cell," J. Electrochem. Soc. **164**(2), A304 (2017).
[36]M. Xu, Z. Zhang, X. Wang, L. Jia, and L. Yang, "A pseudo three-dimensional electrochemical–thermal model of a prismatic $LiFePO_4$ battery during discharge process," Energy **80**, 303–317 (2015).
[37]S. Carelli, M. Quarti, M. C. Yagci, and W. G. Bessler, "Modeling and experimental validation of a high-power lithium-ion pouch cell with LCO/NCA blend cathode," J. Electrochem. Soc. **166**(13), A2990 (2019).
[38]D. Schmider and W. G. Bessler, "Thermo-electro-mechanical modeling and experimental validation of thickness change of a lithium-ion pouch cell with blend positive electrode," Batteries **9**(7), 354 (2023).
[39]R. C. Aylagas, C. Ganuza, R. Parra, M. Yañez, and E. Ayerbe, "cideMOD: An open source tool for battery cell inhomogeneous performance understanding," J. Electrochem. Soc. **169**(9), 090528 (2022).
[40]M. R. Palacín and A. de Guibert, "Why do batteries fail? Science **351**(6273), 1253292 (2016).
[41]H. Lee, M. Yanilmaz, O. Toprakci, K. Fu, and X. Zhang, "A review of recent developments in membrane separators for rechargeable lithium-ion batteries," Energy Environ. Sci. **7**(12), 3857–3886 (2014).
[42]M. Plaimer, C. Breitfuß, W. Sinz, S. F. Heindl, C. Ellersdorfer, H. Steffan, M. Wilkening, V. Hennige, R. Tatschl, A. Geier *et al.*, "Evaluating the trade-off between mechanical and electrochemical performance of separators for lithium-ion batteries: Methodology and application," J. Power Sources **306**, 702–710 (2016).
[43]A. Li, A. C. Y. Yuen, W. Wang, I. M. De Cachinho Cordeiro, C. Wang, T. B. Y. Chen, J. Zhang, Q. N. Chan, and G. H. Yeoh, "A review on lithium-ion battery separators towards enhanced safety performances and modelling approaches," Molecules **26**(2), 478 (2021).
[44]S. S. Zhang, "A review on the separators of liquid electrolyte Li-ion batteries," J. Power Sources **164**(1), 351–364 (2007).
[45]A. Jana, D. R. Ely, and R. E. García, "Dendrite-separator interactions in lithium-based batteries," J. Power Sources **275**, 912–921 (2015).
[46]A. Cannon and E. M. Ryan, "Characterizing the microstructure of separators in lithium batteries and their effects on dendritic growth," ACS Appl. Energy Mater. **4**(8), 7848–7861 (2021).
[47]M. F. Lagadec, R. Zahn, and V. Wood, "Characterization and performance evaluation of lithium-ion battery separators," Nat. Energy **4**(1), 16–25 (2019).
[48]Y. Yu, B. Xiong, F. Zeng, R. Xu, F. Yang, J. Kang, M. Xiang, L. Li, X. Sheng, and Z. Hao, "Influences of compression on the mechanical behavior and electrochemical performances of separators for lithium ion batteries," Ind. Eng. Chem. Res. **57**(50), 17142–17151 (2018).
[49]K. Smith and C.-Y. Wang, "Power and thermal characterization of a lithium-ion battery pack for hybrid-electric vehicles," J. Power Sources **160**(1), 662–673 (2006).
[50]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. **163**(7), A1192 (2016).
[51]I. Korotkin, S. Sahu, S. E. O'Kane, G. Richardson, and J. M. Foster, "DandeLion v1: An extremely fast solver for the Newman model of lithium-ion battery (dis) charge," J. Electrochem. Soc. **168**(6), 060544 (2021).
[52]V. Sulzer, S. G. Marquis, R. Timms, M. Robinson, and S. J. Chapman, "Python battery mathematical modelling (PyBaMM)," J. Open Res. Softw.**9**(1), 14 (2021).
[53]Y. Wang, Q. Li, and Y. Xing, "Porosity variation of lithium-ion battery separators under uniaxial tension," Int. J. Mech. Sci. **174**, 105496 (2020).
[54]M. F. Lagadec, R. Zahn, S. Müller, and V. Wood, "Topological and network analysis of lithium ion battery components: The importance of pore space connectivity for cell operation," Energy Environ. Sci. **11**(11), 3194–3200 (2018).
[55]R. Chandrasekaran, "Quantification of contributions to the cell overpotential during galvanostatic discharge of a lithium-ion cell," J. Power Sources **262**, 501–513 (2014).
[56]H. Xu, F. Usseglio-Viretta, S. Kench, S. J. Cooper, and D. P. Finegan, "Microstructure reconstruction of battery polymer separators by fusing 2D and 3D image data for transport property analysis," J. Power Sources **480**, 229101 (2020).
[57]T. Wallis, P. T. Barton, and N. Nikiforakis, "A unified diffuse interface method for the interaction of rigid bodies with elastoplastic solids and multi-phase mixtures," J. Appl. Phys. **131**(10), 104901 (2022).
[58]S. Osher and R. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces*, 1st ed., Applied Mathematical Sciences (Springer, New York, 2002), Vol. 153.
[59]W. Zhang, A. Almgren, V. Beckner, J. Bell, J. Blaschke, C. Chan, M. Day, B. Friesen, K. Gott, D. Graves, M. Katz, A. Myers, T. Nguyen, A. Nonaka, M. Rosso, S. Williams, and M. Zingale, "AMReX: A framework for block-structured adaptive mesh refinement," J. Open Source Softw. **4**(37), 1370 (2019).
[60]M. Frey, A. Adelmann, and U. Locans, "On architecture and performance of adaptive mesh refinement in an electrostatics Particle-In-Cell code," Comput. Phys. Commun. **247**, 106912 (2020).
[61]J. Teunissen and U. Ebert, "Afivo: A framework for quadtree/octree AMR with shared-memory parallelization and geometric multigrid methods," Comput. Phys. Commun. **233**, 156–166 (2018).
[62]W. Zhang, A. Myers, K. Gott, A. Almgren, and J. Bell, "AMReX: Block-structured adaptive mesh refinement for multiphysics applications," Int. J. High Perform. Comput. Appl. **35**(6), 508–526 (2021).
[63]C. A. Rendleman, V. E. Beckner, M. Lijewski, W. Crutchfield, and J. B. Bell, "Parallelization of structured, hierarchical adaptive mesh refinement algorithms," Comput. Vis. Sci. **3**, 147–157 (2000).
[64]K. Sverdrup, A. Almgren, and N. Nikiforakis, "An embedded boundary approach for efficient simulations of viscoplastic fluids in three dimensions," Phys. Fluids **31**(9), 093102 (2019).
[65]A. Giuliani, A. S. Almgren, J. B. Bell, M. J. Berger, M. H. de Frahan, and D. Rangarajan, "A weighted state redistribution algorithm for embedded boundary grids," J. Comput. Phys. **464**, 111305 (2022).
[66]H. Johansen and P. Colella, "A Cartesian grid embedded boundary method for Poisson's equation on irregular domains," J. Comput. Phys. **147**(1), 60–85 (1998).
[67]M. S. Day, P. Colella, M. J. Lijewski, C. A. Rendleman, and D. L. Marcus, "Embedded boundary algorithms for solving the Poisson equation on complex domains" (Lawrence Berkeley National Laboratory, 1998), available at https://escholarship.org/uc/item/7xq402q1.
[68]E. Bourne, P. Leleux, K. Kormann, C. Kruse, V. Grandgirard, Y. Güçlü, M. J. Kühn, U. Rüde, E. Sonnendrücker, and E. Zoni, "Solver comparison for

Poisson-like equations on tokamak geometries," J. Comput. Phys. **488**, 112249 (2023).
[69]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 **490**, 229571 (2021).
[70]D. F. Martin and K. L. Cartwright, *Solving Poisson's Equation Using Adaptive Mesh Refinement* (Lawrence Berkeley National Laboratory, 1996).
[71]D. F. Martin and P. Colella, "A cell-centered adaptive projection method for the incompressible Euler equations," J. Comput. Phys. **163**(2), 271–312 (2000).
[72]A. Afzal, Z. Ansari, A. R. Faizabadi, and M. Ramis, "Parallelization strategies for computational fluid dynamics software: State of the art review," Arch. Comput. Methods Eng. **24**(2), 337–363 (2017).
[73]K. Sverdrup, N. Nikiforakis, and A. Almgren, "Highly parallelisable simulations of time-dependent viscoplastic fluid flow with structured adaptive mesh refinement," Phys. Fluids **30**(9), 093102 (2018).
[74]A. Myers, W. Zhang, A. Almgren, T. Antoun, J. Bell, A. Huebl, and A. Sinn, "AMReX and pyAMReX: Looking beyond the Exascale Computing Project," Int. J. High Perform. Comput. Appl. **38**(6), 599–611 (2024).
[75]G. Guennebaud, B. Jacob *et al.*, "Eigen," version 3 (2010); see http://eigen.tuxfamily.org.
[76]CSD3, "Research Computing Services HPC Documentation"; see https://docs.hpc.cam.ac.uk/hpc/ (accessed Febuary 12, 2025).
[77]S. K. Rahimian, S. Rayman, and R. E. White, "Extension of physics-based single particle model for higher charge–discharge rates," J. Power Sources **224**, 180–194 (2013).
[78]V. Esfahanian, F. Chaychizadeh, H. Dehghandorost, and H. Shokouhmand, "An efficient thermal-electrochemical simulation of lithium-ion battery using proper mathematical-physical CFD schemes," J. Electrochem. Soc. **166**(8), A1520 (2019).
[79]J. Kim, A. Mallarapu, S. Santhanagopalan, and J. Newman, "A robust numerical treatment of solid phase diffusion in pseudo two-dimensional lithium-ion battery models," J. Power Sources **556**, 232413 (2023).
[80]M. Apsley, "Multiphysics modelling of lithium-ion batteries," Ph.D. dissertation (University of Cambridge, 2022).
[81]P. Ramadass, B. Haran, P. M. Gomadam, R. White, and B. N. Popov, "Development of first principles capacity fade model for Li-ion cells," J. Electrochem. Soc. **151**(2), A196 (2004).
[82]L. O. Valøen and J. N. Reimers, "Transport properties of LiPF6-based Li-ion battery electrolytes," J. Electrochem. Soc. **152**(5), A882 (2005).
[83]Z. Wang, J. Ni, L. Li, and J. Lu, "Theoretical simulation and modeling of three-dimensional batteries," Cell Rep. Phys. Sci. **1**(6), 100078 (2020).
[84]D. Miranda, C. Costa, A. Almeida, and S. Lanceros-Méndez, "Modeling separator membranes physical characteristics for optimized lithium ion battery performance," Solid State Ionics **278**, 78–84 (2015).
[85]M. F. Lagadec, R. Zahn, and V. Wood, "Designing polyolefin separators to minimize the impact of local compressive stresses on lithium ion battery performance," J. Electrochem. Soc. **165**(9), A1829 (2018).
[86]C. M. Costa, M. M. Silva, and S. Lanceros-Méndez, "Battery separators based on vinylidene fluoride (VDF) polymers and copolymers for lithium ion battery applications," RSC Adv. **3**(29), 11404–11417 (2013).
[87]W. Pantoja, J. A. Perez-Taborda, and A. Avila, "Tug-of-war in the selection of materials for battery technologies," Batteries **8**(9), 105 (2022).
[88]A. Davoodabadi, C. Jin, D. L. Wood III, T. J. Singler, and J. Li, "On electrolyte wetting through lithium-ion battery separators," Extreme Mech. Lett. **40**, 100960 (2020).
[89]C. Sauter, R. Zahn, and V. Wood, "Understanding electrolyte infilling of lithium ion batteries," J. Electrochem. Soc. **167**(10), 100546 (2020).
[90]M. F. Lagadec, "Microstructure of Celgard® PP1615 lithium-ion battery separator," Dataset (ETH Zurich, 2018), available at https://doi.org/10.3929/ethz-b-000265085.
[91]S. Yu, Y. Chung, M. S. Song, J. H. Nam, and W. I. Cho, "Investigation of design parameter effects on high current performance of lithium-ion cells with $LiFePO_4$/graphite electrodes," J. Appl. Electrochem. **42**, 443–453 (2012).
[92]D. P. Finegan, S. J. Cooper, B. Tjaden, O. O. Taiwo, J. Gelb, G. Hinds, D. J. Brett, and P. R. Shearing, "Characterising the structural properties of polymer separators for lithium-ion batteries in 3D using phase contrast X-ray microscopy," J. Power Sources **333**, 184–192 (2016).
[93]M. F. Lagadec, M. Ebner, R. Zahn, and V. Wood, "Communication—Technique for visualization and quantification of lithium-ion battery separator microstructure," J. Electrochem. Soc. **163**(6), A992 (2016).