Examples and Counterexamples 2 (2022) 100083

Contents lists available at ScienceDirect

Examples and Counterexamples

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

Comparison of an *h*- and *hp*-adaptive finite element solver for chemo-mechanically coupled battery electrode particles

G.F. Castelli [a,*], W. Dörfler [b]

[a] Karlsruhe Institute of Technology (KIT), Institute of Thermal Process Engineering, Kaiserstr. 12, 76131 Karlsruhe, Germany
[b] Karlsruhe Institute of Technology (KIT), Institute of Applied and Numerical Mathematics, Englerstr. 2, 76131 Karlsruhe, Germany

ARTICLE INFO

Keywords:
hp-adaptivity
Finite elements
Battery
Cahn–Hilliard
Phase separation
Mechanics

ABSTRACT

Numerical investigations of mechanical stresses for phase transforming battery electrode materials on the particle scale are computationally highly demanding. The limitations are mainly induced by the strongly varying spatial and temporal scales of the underlying phase field model, which require an ultra fine mesh and time resolution, however, solely at specific stages in space and time. To overcome these numerical difficulties we present a general-purpose space and time adaptive solution algorithm based on an *hp*-adaptive finite element method and a variable-step, variable-order time integrator. At the example of a chemo-mechanical electrode particle model we demonstrate the computational savings gained by the *hp*-adaptivity. In particular, we compare the results to an *h*-adaptive finite element method and show the reduction of computational complexity.

1. Introduction

In the last decades battery research has become a key competence for a sustainable energy management in the civil and industrial society. Currently, lithium-ion batteries are the state-of-the-art technology and are used in many kinds of mobile power devices [1]. Thus, optimizing the performance and lifetime is an important issue for the next generation batteries [2].

On our end, we are therefore interested in the mechanical degradation of electrode materials. In particular, the coupling of chemistry, phase separation and large deformations in single particles of battery electrodes implies high stresses, which can lead to particle fracture and thus capacity loss [3]. In this context the mathematical modeling and numerical simulation are powerful tools to identify hotspots of stress generation.

In literature, Cahn–Hilliard-type phase-field models are commonly used to describe phase transformation and the mechanical coupling, see [4–7] and others. Based on these models, a wide variety of numerical investigations can be carried out. As an example, the dependence of the maximum arising stresses on the particle shape [4,5] or the material parameters [6,7] can be investigated.

However, numerical simulations of the underlying phase-field models in crucial parameter regimes and on realistic geometries are computationally expensive, compare [7,8]. Motivated by the lack of efficient simulations in the field of battery research, we started in [7–10] the systematic development of a general-purpose adaptive solution algorithm. This is based on an *h*-adaptive finite element method and a variable-step, variable-order time integration scheme, see [8] and the references cited therein. In particular, we found superior efficiency for the phase field models compared to solvers with constant time step size and fixed mesh, see [7,8].

For the classical Cahn–Hilliard equation, *h*-adaptive finite element solvers are widely used. However, the influence of the uniform variation [11,12] as well as the adaptive variation [13] of the polynomial degree of the finite element method on the accuracy of the solution has been studied only in a few cases [13]. Moreover, we have not found any established work on *hp*-adaptive finite element methods for the Cahn–Hilliard equation. This is despite the fact that *hp*-adaptive finite element methods and *hp*-adaptive algorithms are already established for stationary problems [14–17]. Furthermore, in these cases they are known for the possibility to achieve exponential convergence [17].

We therefore extended our existing solver for the time-dependent Cahn–Hilliard-type phase-field model to an *hp*-adaptive finite element method, which we present in this article. In order to obtain a general-purpose solution algorithm for time-dependent problems we adapted the *hp*-refinement algorithm from [15]. Using a concrete application example, a chemo-mechanical electrode particle model, we show the functionality of our *hp*-adaptive algorithm and compare the computational savings to our previous *h*-adaptive finite element solver.

The rest of this article is organized as follows: In Section 2 we recall the model equations for a chemo-mechanical particle model. Then in Section 3 we describe the discretization and explain the *hp*-adaptive

* Corresponding author.
E-mail addresses: fabian.castelli@kit.edu (G.F. Castelli), willy.doerfler@kit.edu (W. Dörfler).

https://doi.org/10.1016/j.exco.2022.100083
Received 11 November 2021; Accepted 16 August 2022

solution algorithm. Afterwards, in Section 4 we present the numerical experiments and discuss the results. Finally, we conclude this article in Section 5 by pointing out our findings.

2. Model equations

In the following section, we give the dimensionless model equations for the chemo-mechanical particle model for chemistry, phase separation and large deformations that we developed in our previous work [7]. Therefore, we first explain the deformation and afterwards introduce the phase-field model. Since we are primarily interested in the numerical treatment of the model equations, we repeat the meaning of individual parts only briefly. Further details of the modeling and normalization can be found in [7,8] and the references cited therein.

2.1. Deformation

In the following let $\Omega \subset \mathbb{R}^3$ be a bounded domain representing an electrode particle in the stress-free reference configuration and let $T_{\text{end}} > 0$ be a final simulation time.

With the displacement $\boldsymbol{u} : [0, T_{\text{end}}] \times \overline{\Omega} \to \mathbb{R}^3$ we describe the deformation of a particle by the deformation gradient

$$\mathbf{F}(\nabla \boldsymbol{u}) = \mathbf{Id} + \nabla \boldsymbol{u}, \tag{1}$$

where $\mathbf{Id} \in \mathbb{R}^{3,3}$ is the three-dimensional unit tensor. In order to separate the chemical and the elastic contribution to the deformation, we multiplicatively decompose the deformation gradient into an elastic and a chemical part $\mathbf{F} = \mathbf{F}_{\text{el}}\mathbf{F}_{\text{ch}}$. Especially the chemical deformation is due to species insertion and given by

$$\mathbf{F}_{\text{ch}}(c) = \lambda_{\text{ch}}(c)\mathbf{Id} = \sqrt[3]{1 + Vc}\,\mathbf{Id}, \tag{2}$$

where $V > 0$ is the partial molar volume and $c : [0, T_{\text{end}}] \times \overline{\Omega} \to \mathbb{R}$ the normalized concentration we are looking for.

For the rest of this article we assume all equations to be stated in the reference configuration.

2.2. Phase-field model

A free energy density $\psi : [0,1] \times \mathbb{R}^3 \times \mathbb{R}^{3,3} \to \mathbb{R}$ forms the basis of the chemo-mechanical phase-field model. It consists of three components: A part for the chemistry $\psi_{\text{ch}} : [0,1] \to \mathbb{R}$, an interfacial energy term $\psi_{\text{int}} : \mathbb{R}^3 \to \mathbb{R}$ and a mechanical coupling $\psi_{\text{el}} : [0,1] \times \mathbb{R}^{3,3} \to \mathbb{R}$. These parts are given as

$$\psi(c, \nabla c, \nabla \boldsymbol{u}) = \psi_{\text{ch}}(c) + \psi_{\text{int}}(\nabla c) + \psi_{\text{el}}(c, \nabla \boldsymbol{u}), \tag{3}$$

$$\psi_{\text{ch}}(c) = \alpha_1 c + \frac{\alpha_2}{2} c^2 + c\log(c) + (1-c)\log(1-c), \tag{4}$$

$$\psi_{\text{int}}(\nabla c) = \frac{1}{2}\kappa |\nabla c|^2, \tag{5}$$

$$\psi_{\text{el}}(c, \nabla \boldsymbol{u}) = \frac{1}{2}\mathbf{E}_{\text{el}}(c, \nabla \boldsymbol{u}) : \mathbb{C}\mathbf{E}_{\text{el}}(c, \nabla \boldsymbol{u}). \tag{6}$$

For the chemical part (4), the parameters $\alpha_1, \alpha_2 \in \mathbb{R}$ control the shape of the double-well and thus the equilibrium concentrations of the bulk phases. In the interfacial energy contribution (5) the parameter $\kappa > 0$ determines the thickness of the phase transition zone. Finally, we describe the mechanical coupling (6) with an elastic approach through the symmetric fourth-order elasticity tensor $\mathbb{C}$ defined by

$$\mathbb{C}\mathbf{E}_{\text{el}} = \lambda \operatorname{tr}(\mathbf{E}_{\text{el}})\mathbf{Id} + 2G\mathbf{E}_{\text{el}}, \tag{7}$$

with $\lambda = 2G\nu/(1-2\nu)$ and $G = \mathrm{E}_{\mathrm{H}}/\big(2(1+\nu)\big)$ the first and second Lamé constants depending on the elastic modulus $\mathrm{E}_{\mathrm{H}} > 0$ and the Poisson's ratio $\nu \in (0, 1/2)$. The elastic strain tensor is defined as

$$\mathbf{E}_{\text{el}}(c, \nabla \boldsymbol{u}) = \frac{1}{2}\left(\mathbf{F}^{\mathsf{T}}\mathbf{F} - \mathbf{F}_{\text{ch}}^{\mathsf{T}}\mathbf{F}_{\text{ch}}\right). \tag{8}$$

To arrive at the final set of equations, we introduce the chemical potential $\mu : [0, T_{\text{end}}] \times \overline{\Omega} \to \mathbb{R}$. This additional variable is defined as the variational derivative of the free energy

$$\mu = \partial_c \psi(c, \nabla c, \nabla \boldsymbol{u}) - \nabla \cdot \partial_{\nabla c}\psi(c, \nabla c, \nabla \boldsymbol{u}). \tag{9}$$

In particular, the mass flux is driven by a gradient in the chemical potential

$$\boldsymbol{N}(c, \nabla \mu) = -m(c)\nabla \mu, \tag{10}$$

with the nonlinear mobility $m(c) = Dc(1-c)$ and the diffusion coefficient $D > 0$.

Finally, we formulate the momentum balance of the first Piola–Kirchhoff stress tensor as

$$\mathbf{P}(c, \nabla \boldsymbol{u}) = \partial_{\mathbf{F}}\psi(c, \nabla c, \nabla \boldsymbol{u}) = \mathbf{F}\mathbb{C}\mathbf{E}_{\text{el}}. \tag{11}$$

The resulting dimensionless Cahn–Hilliard-type phase-field model describes the normalized concentration $c : [0, T_{\text{end}}] \times \overline{\Omega} \to [0,1]$, the chemical potential $\mu : [0, T_{\text{end}}] \times \overline{\Omega} \to \mathbb{R}$ and the displacement $\boldsymbol{u} : [0, T_{\text{end}}] \times \overline{\Omega} \to \mathbb{R}^3$, up to rigid body motions, satisfying

$$\begin{cases} \partial_t c = \nabla \cdot \boldsymbol{N}(c, \nabla \mu) & \text{in } (0, T_{\text{end}}) \times \Omega, \\ \mu = \partial_c \psi(c, \nabla c, \nabla \boldsymbol{u}) & \text{in } (0, T_{\text{end}}) \times \Omega, \\ \qquad - \nabla \cdot \partial_{\nabla c}\psi(c, \nabla c, \nabla \boldsymbol{u}) & \\ \mathbf{0} = \nabla \cdot \mathbf{P}(c, \nabla \boldsymbol{u}) & \text{in } (0, T_{\text{end}}) \times \Omega. \end{cases} \tag{12}$$

These equations come along with the following boundary conditions on $(0, T_{\text{end}}) \times \partial\Omega$: a vanishing normal derivative of the concentration $\nabla c \cdot \boldsymbol{n} = 0$, a given particle surface flux $N_{\text{ext}} \in \mathbb{R}$ modeling insertion or extraction $\boldsymbol{N} \cdot \boldsymbol{n} = N_{\text{ext}}$ and a stress-free condition $\mathbf{P} \cdot \boldsymbol{n} = \mathbf{0}$. At initial time we prescribe a constant concentration profile $c(0, \cdot) = c_0 \in (0,1)$, which we assume to be consistent with the boundary conditions.

Note that from a mathematical point of view, the model equations are also valid in arbitrary spatial dimensions. However, in the modeling of the mechanical theory, the assumption of a three-dimensional particle entered. Furthermore, we did not specify the material at all, because the general model framework can be used for arbitrary phase transforming materials, as for example lithium iron phosphate or sodium iron phosphate, see [18] for a similar model used for both materials.

3. The *hp*-adaptive finite element solver

After stating the model equations, we next describe the discretization in space and time with finite elements and multistep methods as in [7, Sect. 3.2.1]. Then, for the fully discrete problem, we explain our *hp*-adaptive solution algorithm. Again, further details on the discretization and the core solution algorithm can be found in [7,8] and the references cited therein.

3.1. Discretization

In the modeling, we have already introduced the chemical potential μ as an additional variable. Therefore, we consider the mixed formulation of the Cahn–Hilliard-type model and thus avoid the C^1-regularity requirement on the finite element discretization due to the fourth-order derivative [7,8]. Furthermore, we solve the PDEs (12) for the concentration c, the chemical potential μ and the displacement $\boldsymbol{u}$ and recover the stress in a postprocessing step if necessary.

For fixed $t \in (0, T_{\text{end}}]$ let c, $\mu \in V$, $\boldsymbol{u} \in W$ be the solutions of the model Eqs. (12) in appropriate function spaces [19]. Here we assume that the space W includes already the correct constraints to neglect rigid body motions. Then, we multiply the PDEs (12) with smooth functions φ, $\chi \in V$, $\xi \in W$ and integrate over the domain Ω. We arrive at the weak formulation

$$\begin{cases} \big(\varphi, \partial_t c\big) = -\big(\nabla \varphi, m(c)\nabla \mu\big) - \big(\varphi, N_{\text{ext}}\big)_{\partial\Omega}, \\ 0 = -\big(\chi, \mu\big) + \big(\chi, \partial_c \psi_{\text{ch}}(c) + \partial_c \psi_{\text{el}}(c, \nabla \boldsymbol{u})\big) \\ \qquad\qquad + \kappa\big(\nabla \chi, \nabla c\big), \\ 0 = -\big(\nabla \xi, \mathbf{P}(c, \nabla \boldsymbol{u})\big). \end{cases} \tag{13}$$

Here, $(f, g) = \int_\Omega f\, g \,\mathrm{d}x$ denotes the L^2-inner product of two functions f, g and analogously for two vectors or tensors. The boundary integral we indicate with the subscript $\partial\Omega$.

We discretize the domain Ω with an admissible mesh $\mathcal{K}$ and employ the isoparametric Lagrangian finite element method [19]. Then, we replace the continuous functions by spatial approximations c_h, $\mu_h \in V_h$, $\boldsymbol{u}_h \in W_h$ in the finite dimensional subspaces $V_h \subset V$, $W_h \subset W$.

In view of the discretization in time, we gather the time-dependent coefficients of a basis representation of the finite element functions in a vector-valued function

$$\boldsymbol{y}(t) = \left[\left(c_i(t)\right)_i, \left(\mu_i(t)\right)_i, \left(u_i(t)\right)_i\right]^{\mathsf{T}}. \tag{14}$$

Finally, we employ the NDF family of linear multistep methods [20–22] for the time integration and obtain the fully discrete problem as in [7,8]. For advancing one time step from time t_n to $t_{n+1} = t_n + \tau_n$ with the time step size $\tau_n > 0$ find the discrete solution $\boldsymbol{y}^{n+1} \approx \boldsymbol{y}(t_{n+1})$ satisfying

$$\mathbf{M}(\boldsymbol{y}^{n+1} - \boldsymbol{\Phi}_k^n) = \xi_k \boldsymbol{f}(t_{n+1}, \boldsymbol{y}^{n+1}), \tag{15}$$

with the $\boldsymbol{\Phi}_k^n$ depending on the solutions at previous time steps $\boldsymbol{y}^n$, $\ldots$, $\boldsymbol{y}^{n-k}$ and a generalized time step size $\xi_k > 0$ depending on the chosen order k, see also [8, Sect. 4.1.2]. The singular mass matrix $\mathbf{M}$ and the right hand side function $\boldsymbol{f}$ are defined according to the weak formulation (13) and can be looked up also in [7,8].

3.2. Adaptive algorithm

Functions in V_h are polynomials of a degree p_K on $K \in \mathcal{K}$. The approximation properties of V_h depend on the diameter of K and the degree p_K and can be improved by either refining K (*h-method*) or increasing the local polynomial degree (*p-method*). A combination of both is called *hp-method*. Instead of an a priori selection of V_h we can use methods that improve the approximation properties in either ways by evaluation of local error indicators η_K, that depend on data and the actual discrete solution, and modification of V_h. Such a method is called *adaptive* and it produces a sequence of spaces and approximate solutions until certain criteria are met.

Here, we present an *hp*-adaptive method for (15). An *h*-adaptive version was previously proposed in our works [7–9]. For η_K we used the gradient recovery estimator. The *hp*-adaptivity is an extension of our adaptive solution algorithm presented in [7,9] and optimized in [8]. More details on the single steps are described in [8]. Our extension for *hp*-adaptivity is based on the work by Melenk and Wohlmuth [15]. In the following we discuss only the adapted steps and refer to [7,8] for further details.

Our *hp*-marking strategy is explained in Algorithm 1. Besides η_K we also need a field of predicted errors $\eta_{K,\text{pred}}$ for all $K \in \mathcal{K}$. A set of four parameters θ_r, γ_h, γ_p, γ_n is needed for fine-tuning the algorithm.

As outcome we obtain a new finite element space based on a modified mesh and with modified local polynomial degrees. We finalize this by performing a smoothing step, where we try to avoid oscillations in the level difference in *h*- as well as *p*-refinement by marking additional cells, when all neighbors are already marked.

We took [15, Alg. 4.4] as a base for our method, however, we deviate from it as follows:

(a) [15] selected a cell $K \in \mathcal{K}$ for *h*- or *p*-refinement if $\eta_K^2 > \theta_\text{r}\overline{\eta}_K^2 := \frac{1}{|\mathcal{K}|}\sum_{K\in\mathcal{K}} \eta_K^2$. We instead used the comparison with the maximal indicator, i.e., we select all cells $K \in \mathcal{K}$ for *h*- or *p*-refinement if $\eta_K > \theta_\text{r}\eta_\text{max}$ with $\theta_\text{r} = 0.5$, which seems to work best in our numerical experiments.

(b) [15] initialized the predicted error estimates by $\eta_{K,\text{pred}} := 0$, so they start with a *h*-refinement step. We instead initialize the predicted error estimates as $\eta_{K,\text{pred}} := \texttt{inf}$, so that we start with a *p*-refinement step. In our time-dependent problem, we reset the predicted error estimates to $\eta_{K,\text{pred}} := \texttt{inf}$ in each time step, so that always *p*-refinement is enforced as first adaptive refinement step.

Algorithm 1 Marking algorithm for *hp*-refinement.

```
Let θ_r = 0.5, γ_h = 2, γ_p = √0.4, γ_n = 1.0;
Given η_K, η_K,pred for all K ∈ 𝒦;
Compute maximal estimate η_max := max_{K∈𝒦} η_K;
for all K ∈ 𝒦 do
    if η_K ≥ θ_r η_max then
        if η_K > η_K,pred then
            % h-refine element K;
            Subdivide K into K_s children, s = 1, ..., 2^d;
            Set η_{K_s,pred} := 1/2 γ_h (0.5)^{p_K} η_K
                for all children K_s, s = 1, ..., 2^d;
        else
            if p_K < p_max then
                % p-refine element K;
                p_K := p_K + 1;
                Set η_K,pred := γ_p η_K;
            else
                % h-refine element K;
                Subdivide K into K_s children, s = 1, ..., 2^d;
                Set η_{K_s,pred} := 1/2 γ_h (0.5)^{p_K} η_K
                    for all children K_s, s = 1, ..., 2^d;
            end if
        end if
    else
        % No refinement of element K;
        Set η_K,pred := γ_n η_K,pred;
    end if
end for
```

(c) We choose $\gamma_h = 2$, $\gamma_p = \sqrt{0.4}$ and $\gamma_n = 1$ for the parameters controlling the marking algorithm for *hp*-refinement, see [15]. The influence of these parameters are as follows:

γ_h: After a *h*-step, a larger value of this parameter makes *p*-refinement more likely.

γ_p: After a *p*-step, a larger value of this parameter makes *h*-refinement more likely.

γ_n: If a cell was not marked for refinement, scale the predicted error estimate with this factor. A larger value makes *p*-refinement more likely, a smaller one makes *h*-refinement more likely.

(d) As previously, we split coarsening and refinement. We checked for coarsening after each step using

$$\eta_K \le \theta_\text{c}\eta_\text{max}, \tag{16}$$

with $\theta_\text{c} = 0.05$, a strategy also used by Baňas and Nürnberg [23]. If we marked elements for local coarsening, we performed both a *h*- and *p*-coarsening step.

(e) We allow *p*-adaption in a given range $p \in \{p_\text{min}, \ldots, p_\text{max}\}$. If the maximum polynomial degree is reached, we do *h*-refinements instead.

(f) We solve the nonlinear problem in each time step with the Newton method.

(g) We assemble one big system matrix which takes all equations into account and solve the system as a whole.

(h) The necessary interpolation between the different *hp*-adapted finite element spaces is done within the *deal.II* library[1].

The algorithm is simple, quite general and can be applied to a wide range of application problems, beyond this exemplary problem. No additional decision markers for *p*-refinement need to be computed. An overview on some other *hp*-adaptive strategies are given in [17]. The algorithm from [15] is also among those compared algorithms and is said to be a cheap alternative.

[1] See the `SolutionTransfer` class

4. Numerical experiments

In this section we present and discuss the results of our numerical experiments. At the beginning we give details on the application problem and the implementation.

4.1. Application example

We analyze our *hp*-adaptive solution algorithm in case of a spherical electrode particle. Hereby, the chemo-mechanical particle model from [7], which we recalled in Section 2, describes the chemistry, phase separation and mechanical coupling during species insertion or extraction. In the scope of this article, we assume a spherical symmetric solution, like in [7]. Hence, we obtain the one-dimensional unit interval $\Omega = (0,1)$ as computational domain representing the radial line form the particle center $\Gamma_0 = \{0\}$ to the surface $\Gamma_{\text{ext}} = \{1\}$. With a modified integration weight $\mathrm{d}x = 4\pi r^2\,\mathrm{d}r$ we encode the original three-dimensional geometry in the one-dimensional simulation.

As in [7], we choose the values for the material parameters corresponding to lithium iron phosphate. The dimensionless parameters are $D = 1.6\times10^{2}$ for the diffusion coefficient, $\alpha_1 = 4.5$, $\alpha_2 = -9$, $\kappa = 3.91\times10^{-4}$ for the chemical and the interfacial free energy density and the Young's modulus $\mathrm{E_H} = 2.19\times10^{3}$, the Poisson ratio $\nu = 0.26$, the partial molar volume $V = 6.64\times10^{-2}$ for the mechanical coupling. We model the lithium insertion over the particle surface Γ_{ext} with the boundary condition $N_{\text{ext}} = -1/3$. In the particle center Γ_0 we preserve the symmetry with no-flux conditions $\nabla c\cdot\boldsymbol{n} = \boldsymbol{N}\cdot\boldsymbol{n} = 0$ and fix the displacement $\boldsymbol{u} = \boldsymbol{0}$. Finally, we assume a constant concentration $c_0 = 0.01$ at the initial time.

4.2. Implementation details

We implemented the *hp*-adaptive algorithm from Section 3 as a new feature of our C++-code that relies on the functionalities of the open-source finite element library *deal.II* [24] and the direct LU-decomposition from [25]. Thus, for further details on the solver we refer to our previous works [7–9]. The simulations for this article were performed on a laptop pc with 16 GB RAM and an Intel i7-7500U CPU with 2.70 GHz.

4.3. Refinement profile

In a first experiment we demonstrate the functionality of our proposed *hp*-adaptive solution algorithm. Therefore we solve the model equations of our application example, where we allow an adaptively chosen local degree $p \in \{2,\dots,6\}$. The simulation starts from a uniform spatial discretization with $h_0 = 2^{-7}$, $p_0 = 6$ and the initial time step size is set to $\tau_0 = 10^{-6}$. As tolerances for the adaptive algorithm were set $\mathrm{AbsTol} = 10^{-4}$ and $\mathrm{RelTol} = 10^{-6}$ for both, space and time estimates.

Finally, in Fig. 1 we visualize the solution for the concentration at time $t = 0.5$. In addition, we depict the *h*-refinement level n, which is related to the mesh width by $h = 2^{-n}$, as well as the local degree p for each mesh element. The whole solution over time is in excellent qualitative accordance with the results from our previous work [7] without *p*-adaptivity. In particular, as one would expect, we observe a primarily *hp*-refinement in the phase transition zone, whereas away from the transition zone the mesh and the local degrees are coarsened. Note, that the refinement indicators are computed with respect to all solution variables (c, μ, $\boldsymbol{u}$), which influences the overall local refinement and coarsening profile.

4.4. Computational complexity

Next, we study the dependency of the computational complexity. The computational complecity is measured in the number of degrees of freedom (DOFs) in the *hp*-adapted mesh. In all following simulations we use the same tolerances and initial time step size as above. For the initial spatial discretization we use the maximal allowed global degree and a mesh width such that the phase transition is resolved with at least ten DOFs.

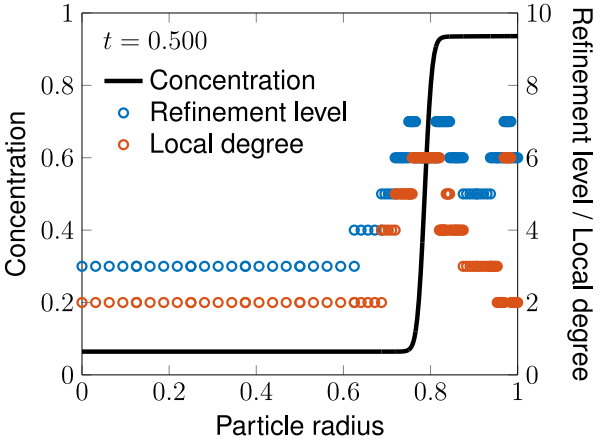

Fig. 1. Concentration profile at time $t = 0.5$ with refinement level and local polynomial degree of the finite element method per element.

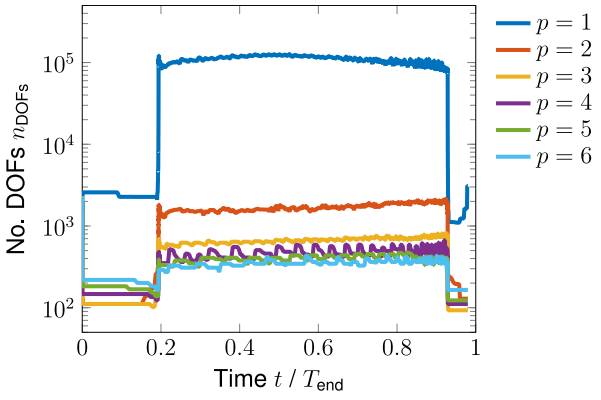

Fig. 2. Number of DOFs over time for different fixed polynomial degrees.

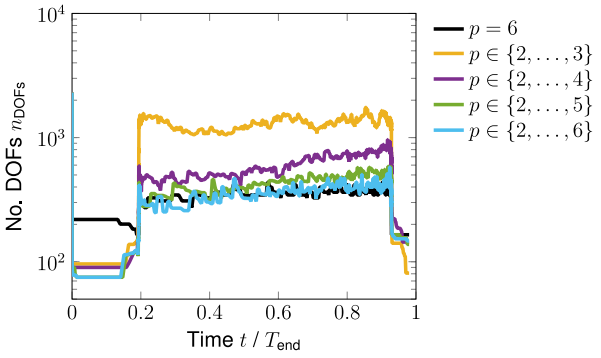

Fig. 3. Number of DOFs over time for different fully *hp*-adapted cases.

h-FEM with fixed p. We will first have a look on the computational complexity for methods with fixed global degree. In Fig. 2 we plotted the number of DOFs over the time for fixed $p \in \{1,\dots,6\}$. We find the expected improved properties for higher order methods, especially when we switch from linear ($p = 1$) to quadratic ($p = 2$) elements.

hp-FEM. We analyze the computational complexity of the full *hp*-adaptivity. As a reference we use *h*-FEM with fixed polynomial degree $p = 6$ and compare the number of DOFs for increasing local degree ranges. From Fig. 3 we conclude, that a larger range of degrees reduces the number of DOFs significantly. Moreover, for the case $p \in \{2,\dots,6\}$ we reach approximately the same complexity in the two-phase diffusion regime and keep the complexity also in the single-phase diffusion state

at a low number. This computational savings makes the full *hp*-adaptive algorithm attractive.

5. Conclusion

In this work, we have recalled the model equations for chemistry, phase separation and mechanical deformation in a battery electrode particle. For the numerical solution of the resulting initial boundary value problem we have designed a robust *hp*-adaptive finite element solution algorithm. The advantage of this adaptive algorithm is its simplicity, which makes it straightforward to implement and applicable to a wide range of problems. At the example of a spherical electrode particle we have demonstrated the functionality and the reduction of computational complexity in terms of DOFs depending on the range of the polynomial degree of the finite element method.

The presented results in the spherical symmetric example indicate the potential of *hp*-adaptivity for even larger computational savings in multidimensional applications. In future research, we will focus on the efficient matrix-free implementation to investigate the chemo-mechanical interplay in arbitrary shaped electrode particles. Therefore, we will review the *hp*-coarsening procedure and develop a robust linear solver for the underlying model equations.

Declaration of competing interest

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

Acknowledgments

The authors thank Fleurianne Bertrand for the kind invitation to submit this article. G.F.C. acknowledges financial support by the German Research Foundation (DFG) through the Research Training Group 2218 SiMET – Simulation of mechano-electro-thermal processes in lithium-ion batteries, project number 281041241.

References

[1] J.-M. Tarascon, Key challenges in future Li-battery research, Phil. Trans. R. Soc. A 368 (2010) 3227–3241.
[2] A. Manthiram, An outlook on lithium ion battery technology, ACS Cent. Sci. 3 (10) (2017) 1063–1069.
[3] Y. Zhao, P. Stein, Y. Bai, M. Al-Siraj, Y. Yang, B.-X. Xu, A review on modeling of electro-chemo-mechanics in lithium-ion batteries, J. Power Sources 413 (2019) 259–283.
[4] Y. Zhao, P. Stein, B.-X. Xu, Isogeometric analysis of mechanically coupled Cahn–Hilliard phase segregation in hyperelastic electrodes of Li-ion batteries, Comput. Methods Appl. Mech. Engrg. 297 (2015) 325–347.
[5] L. Wu, V.D. Andrade, X. Xiao, J. Zhang, Phase field modeling of coupled phase separation and diffusion-induced stress in lithium iron phosphate particles reconstructed from synchrotron nano X-ray tomography, ASME J. Electrochem. En. Conv. Stor. 16 (4) (2019) 041006.
[6] T. Zhang, M. Kamlah, Mechanically coupled phase-field modeling of microstructure evolution in sodium ion batteries particles of Na_xFePO_4, J. Electrochem. Soc. 167 (2) (2020) 020508.
[7] G.F. Castelli, L. von Kolzenberg, B. Horstmann, A. Latz, W. Dörfler, Efficient simulation of chemical-mechanical coupling in battery active particles, Energy Technol. 9 (6) (2021) 2000835.
[8] G.F. Castelli, Numerical Investigation of Cahn–Hilliard-Type Phase-Field Models for Battery Active Particles (Ph.D. thesis), Karlsruhe Institute of Technology (KIT), 2021.
[9] G.F. Castelli, W. Dörfler, Study on an adaptive finite element solver for the cahn–hilliard equation, in: F.J. Vermolen, C. Vuik (Eds.), Numerical Mathematics and Advanced Applications, ENUMATH 2019, in: Lect. Notes Comput. Sci. Eng., vol. 139, Springer, Cham, 2021, pp. 245–253.
[10] G.F. Castelli, W. Dörfler, A parallel matrix-free finite element solver for phase separation in electrode particles of lithium ion batteries, in: Proceedings in Applied Mathematics & Mechanics, 21, (1) 2021, p. e202100169.
[11] D. Kay, V. Styles, E. Süli, Discontinuous Galerkin finite element approximation of the Cahn–Hilliard equation with convection, SIAM J. Numer. Anal. 47 (4) (2009) 2660–2685.
[12] L. Goudenège, D. Martin, G. Vial, High order finite element calculations for the Cahn–Hilliard equation, J. Sci. Comput. 52 (2) (2012) 294–321.
[13] G. Ntoukas, J. Manzanero, G. Rubio, E. Valero, E. Ferrer, A free-energy stable p-adaptive nodal discontinuous Galerkin for the Cahn–Hilliard equation, J. Comput. Phys. 442 (2021) Paper No. 110409, 31.
[14] C. Schwab, *p*- and *hp*-Finite Element Methods, in: Numerical Mathematics and Scientific Computation, The Clarendon Press, Oxford University Press, New York, 1998.
[15] J.M. Melenk, B.I. Wohlmuth, On residual-based a posteriori error estimation in *hp*-FEM, Adv. Comput. Math. 15 (1–4) (2001) 311–331.
[16] P. Houston, E. Süli, A note on the design of *hp*-adaptive finite element methods for elliptic partial differential equations, Comput. Methods Appl. Mech. Engrg. 194 (2–5) (2005) 229–243.
[17] W.F. Mitchell, M.A. McClain, A comparison of *hp*-adaptive strategies for elliptic partial differential equations, ACM Trans. Math. Software 41 (1) (2014) Art. 2, 39.
[18] T. Zhang, M. Kamlah, Sodium ion batteries particles: Phase-field modeling with coupling of Cahn–Hilliard equation and finite deformation elasticity, J. Electrochem. Soc. 165 (10) (2018) A1997–A2007.
[19] D. Braess, Finite elements, third ed., Cambridge University Press, Cambridge, 2007.
[20] L.F. Shampine, M.W. Reichelt, The MATLAB ODE suite, SIAM J. Sci. Comput. 18 (1) (1997) 1–22.
[21] L.F. Shampine, M.W. Reichelt, J.A. Kierzenka, Solving index-1 DAEs in MATLAB and simulink, SIAM Rev. 41 (3) (1999) 538–552.
[22] The MathWorks, Inc., MATLAB, http://www.mathworks.com.
[23] L. Baňas, R. Nürnberg, Adaptive finite element methods for Cahn–Hilliard equations, J. Comput. Appl. Math. 218 (1) (2008) 2–11.
[24] W. Bangerth, R. Hartmann, G. Kanschat, deal.II—A general-purpose object-oriented finite element library, ACM Trans. Math. Software 33 (4) (2007) Art. 24, 27.
[25] T.A. Davis, Algorithm 832: UMFPACK V4.3—an unsymmetric-pattern multifrontal method, ACM Trans. Math. Software 30 (2) (2004) 196–199.