Received: 7 October 2022 | Accepted: 15 November 2022

DOI: 10.1002/pamm.202200191

Study on an hp-Adaptive Finite Element Solver for a Chemo-Mechanical Battery Particle Model

Giuseppe Fabian Castelli[1,*] and **Willy Dörfler**[2]

[1] Karlsruhe Institute of Technology (KIT), Institute of Thermal Process Engineering (TVT), Kaiserstr. 12, 76131 Karlsruhe, Germany

[2] Karlsruhe Institute of Technology (KIT), Institute of Applied and Numerical Mathematics (IANM), Englerstr. 2, 76131 Karlsruhe, Germany

We consider a Cahn–Hilliard-type phase-field model for phase separation and large deformations in battery electrode particles. For the numerical solution we employ an hp-adaptive finite element solution algorithm coupled to a variable-step, variable-order time stepping scheme. Numerical experiments show the adaptive meshing and distribution of the locally varying polynomial degrees of the finite element method. In particular, for a sufficient large range of polynomial degrees we achieve significant computational savings compared to an h-adaptive algorithm.

1 Introduction

Nowadays, battery research is one of the key competences of a sustainably developing society. In particular, performance optimizations of lithium-ion and next generation batteries are active research [1, 2]. For example, mechanical degradation induced by phase transformation and large deformation is one crucial aging mechanism. High occurring stresses can ultimately lead to particle fracture and thus to capacity loss [3]. To evaluate stress hotspots, numerical simulations are valuable tools, see for example [4, 5].

For our chemo-mechanically coupled Cahn–Hilliard-type phase-field model from [5] we recently developed and applied an hp-adaptive finite element solution algorithm in [6]. We performed a first study of the presented algorithm applied to the chemo-mechanical model and pointed out the general-purpose character of the algorithm. In this work, which is mainly based on our previous work [6], we continue the investigation of our hp-adaptive algorithm. In particular, we further investigate the dependence of computational complexity on the range of polynomial degrees. Due to its similarity with the previous study, this article has the same structure and may contain similar wording. In contrast, here, we consider an adapted chemo-mechanically coupled particle model from [7]. A comparison of the models, we recently discussed in [8].

The rest of this article is organized as follows: Next, in Section 2 we recall the model equations for a chemo-mechanical particle model. In Section 3 we summarize the key points of the numerical solution procedure and the hp-adaptive algorithm. Then, in Section 4 we present and discuss the numerical experiments. In the end, we conclude this article in Section 5.

2 Model Equations

In the same manner as in [6], we briefly introduce the dimensionless model equations for an adapted version of a chemo-mechanically coupled particle model for chemistry, phase separation and large deformations derived in [7]. We give two types of deformations and explain the phase-field model. For further details on the modeling we refer to [7] and the references cited therein. The normalization follows [5].

2.1 Deformation

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.

We describe the total deformation of an electrode 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 and $\boldsymbol{u}\colon [0, T_{\text{end}}] \times \overline{\Omega} \to \mathbb{R}^3$ represents the unknown *displacement*. One key of the chemo-mechanically coupled particle model is the multiplicative decomposition of the deformation gradient into a chemical and an elastic part

$$\mathbf{F} = \mathbf{F}_{\text{ch}}\mathbf{F}_{\text{el}}. \tag{2}$$

[*] Corresponding author: e-mail fabian.castelli@kit.edu, phone +49 721 608 48736, fax +49 721 608 43490

Here, the chemical deformation is caused by species intercalation and is specified by

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

where $V > 0$ is the partial molar volume and $c\colon [0, T_{\text{end}}] \times \overline{\Omega} \to \mathbb{R}$ the unknown *normalized concentration*. The elastic deformation gradient remains unspecified and is determined by $\mathbf{F}_{\text{el}} = \mathbf{F}_{\text{ch}}^{-1}\mathbf{F}$.

Throughout this article we assume that all equations are stated in the reference configuration.

2.2 Phase-Field Model

The phase-field model is based on a free energy density $\psi\colon [0,1] \times \mathbb{R}^3 \times \mathbb{R}^{3,3} \to \mathbb{R}$. It consists of three components: the first part represents the chemistry $\psi_{\text{ch}}\colon [0,1] \to \mathbb{R}$, the second part accounts for the interfacial energy $\psi_{\text{int}}\colon \mathbb{R}^3 \to \mathbb{R}$ and the third part describes the mechanical coupling $\psi_{\text{el}}\colon [0,1] \times \mathbb{R}^{3,3} \to \mathbb{R}$. These parts are given as

$$\psi(c, \nabla c, \nabla \boldsymbol{u}) = \psi_{\text{ch}}(c) + \psi_{\text{int}}(\nabla c) + \psi_{\text{el}}(c, \nabla \boldsymbol{u}), \tag{4}$$

$$\psi_{\text{ch}}(c) = \alpha_1 c + \frac{\alpha_2}{2}c^2 + c\log(c) + (1-c)\log(1-c), \tag{5}$$

$$\psi_{\text{int}}(\nabla c) = \frac{1}{2}\kappa|\nabla c|^2, \tag{6}$$

$$\psi_{\text{el}}(c, \nabla \boldsymbol{u}) = \frac{1}{2}\mathbf{E}_{\text{el}}(c, \nabla \boldsymbol{u}) : \mathbb{C}\mathbf{E}_{\text{el}}(c, \nabla \boldsymbol{u}). \tag{7}$$

The parameters α_1, $\alpha_2 \in \mathbb{R}$ of the chemical free energy density (5) determine the shape of the double-well and thus the equilibrium concentrations of the bulk phases. The parameter $\kappa > 0$ of the interfacial energy density (6) determines the thickness of the phase transition zone, see [5, Eq. (7)] for example. The third part for the elastic free energy density (7) is modeled by an elastic approach with the symmetric fourth-order elasticity tensor $\mathbb{C}$ defined by

$$\mathbb{C}\mathbf{E}_{\text{el}} = \lambda\,\mathrm{tr}(\mathbf{E}_{\text{el}})\mathbf{Id} + 2G\mathbf{E}_{\text{el}}, \tag{8}$$

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 $\nu \in (0, 1/2)$. According to [7], the elastic strain tensor is defined different from [5] and is given by

$$\mathbf{E}_{\text{el}}(c, \nabla \boldsymbol{u}) = \frac{1}{2}\left(\mathbf{F}_{\text{el}}^{\mathsf{T}}\mathbf{F}_{\text{el}} - \mathbf{Id}\right) = \frac{1}{2}\left(\lambda_{\text{ch}}^{-2}\mathbf{F}^{\mathsf{T}}\mathbf{F} - \mathbf{Id}\right). \tag{9}$$

As an additional solution variable, one introduces the *chemical potential* $\mu\colon [0, T_{\text{end}}] \times \overline{\Omega} \to \mathbb{R}$, which is defined as the variational derivative of the free energy

$$\mu = \partial_c \psi(c, \nabla c, \nabla \boldsymbol{u}) - \nabla\cdot\partial_{\nabla c}\psi(c, \nabla c, \nabla \boldsymbol{u}). \tag{10}$$

The mass flux $\boldsymbol{N}$ additionally couples chemistry and mechanics by the nonlinear mobility, which depends on the diffusion coefficient $D > 0$ and the inverse of the derivative of the chemical potential by the first argument

$$m(c, \nabla \boldsymbol{u}) = D\big(\partial_c \mu\big)^{-1} \tag{11}$$

The mass flux is then again driven by a gradient in the chemical potential multiplied with the nonlinear mobility

$$\boldsymbol{N}(c, \nabla \mu, \nabla \boldsymbol{u}) = -m(c, \nabla \boldsymbol{u})\nabla \mu. \tag{12}$$

A momentum balance law governs the first Piola–Kirchhoff stress tensor, which is defined as

$$\mathbf{P}(c, \nabla \boldsymbol{u}) = \partial_{\mathbf{F}}\psi(c, \nabla c, \nabla \boldsymbol{u}) = \lambda_{\text{ch}}^{-2}\mathbf{F}\mathbb{C}\mathbf{E}_{\text{el}}. \tag{13}$$

Finally, we obtain a similar Cahn–Hilliard-type phase-field model as considered in our previous work [6]. We look for the normalized concentration $c\colon [0, T_{\text{end}}] \times \overline{\Omega} \to [0,1]$, the chemical potential $\mu\colon [0, T_{\text{end}}] \times \overline{\Omega} \to \mathbb{R}$ and the displacement $\boldsymbol{u}\colon \overline{\Omega} \to \mathbb{R}^3$ up to rigid body motions, satisfying

$$\begin{cases} \partial_t c = -\nabla\cdot\boldsymbol{N}(c, \nabla \mu, \nabla \boldsymbol{u}) & \text{in } (0, T_{\text{end}}) \times \Omega, \\ \mu = \partial_c \psi(c, \nabla c, \nabla \boldsymbol{u}) - \nabla\cdot\partial_{\nabla c}\psi(c, \nabla c, \nabla \boldsymbol{u}) & \text{in } (0, T_{\text{end}}) \times \Omega, \\ \mathbf{0} = \nabla\cdot\mathbf{P}(c, \nabla \boldsymbol{u}) & \text{in } (0, T_{\text{end}}) \times \Omega. \end{cases} \tag{14}$$

The boundary and initial conditions remain the same as in [6]: a vanishing concentration gradient in normal direction $\nabla c \cdot \boldsymbol{n} = 0$, a predefined 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 in radial direction $\mathbf{P} \cdot \boldsymbol{n} = \mathbf{0}$. At initial time we use a given constant concentration profile $c(0, \cdot\,) = c_0 \in (0,1)$, which we assume to be consistent with the boundary conditions.

3 The Numerical Solution Procedure

The development and explanation of the numerical solution procedure for the resulting set of PDEs (14) are part of our previous works [5, 6, 9–11]. Thus, we only summarize and recall the major points of the finite element solver and the hp-marking algorithm here and refer for further details to these works and the references cited therein.

3.1 Discretization

We consider the normalized concentration c, the chemical potential μ and the displacement $\boldsymbol{u}$ as solution variables. Note, that the stresses are not part of the primarily solution variables, as we recover them in a postprocessing step. With the usage of the mixed formulation of the Cahn–Hilliard equation we avoid the C^1-regularity requirement on the finite element discretization due to the fourth-order derivative [10].

In order to derive the fully discrete problem, we start with the weak problem formulation. Let $t \in (0, T_{\text{end}}]$ be a fixed time and c, $\mu \in V$, $\boldsymbol{u} \in W$ be the solutions of the model equations (14) in appropriate function spaces [12], where the space W already includes the constraints to neglect rigid body motions. Multiplication of the PDEs (14) with smooth test functions φ, $\chi \in V$, $\boldsymbol{\xi} \in W$ and integration by parts yield the weak formulation

$$\begin{cases} \big(\varphi, \partial_t c\big) = -\big(\nabla\varphi, m(c, \nabla\boldsymbol{u})\nabla\mu\big) - \big(\varphi, N_{\text{ext}}\big)_{\partial\Omega}, \\ \qquad 0 = -\big(\chi, \mu\big) + \big(\chi, \partial_c\psi_{\text{ch}}(c) + \partial_c\psi_{\text{el}}(c, \nabla\boldsymbol{u})\big) + \kappa\big(\nabla\chi, \nabla c\big), \\ \qquad 0 = -\big(\nabla\boldsymbol{\xi}, \mathbf{P}(c, \nabla\boldsymbol{u})\big). \end{cases} \tag{15}$$

Here, we use the standard notation $(f, g) = \int_\Omega f\, g \,\mathrm{d}x$ for 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 $\partial\Omega$.

For the space discretization with finite elements we discretize the particle domain Ω by an admissible mesh $\mathcal{K}$ and distribute the degrees of freedom (DOFs) according to the isoparametric Lagrangian finite element method [12]. We choose appropriate finite dimensional subspaces $V_h \subset V$, $W_h \subset W$ and substitute the continuous functions by finite element approximations c_h, $\mu_h \in V_h$, $\boldsymbol{u}_h \in W_h$.

For the discretization in time, we gather the time-dependent coefficients of the finite element approximations in a vector-valued function

$$\boldsymbol{y}(t) = \Big[\big(c_i(t)\big)_i, \big(\mu_i(t)\big)_i, \big(u_i(t)\big)_i\Big]^{\mathsf{T}}. \tag{16}$$

Then, we employ the NDF family of linear multistep methods of order $k = 1, \ldots, 5$ [13–15] for the time integration and obtain the fully discrete problem as in [5, 6, 10]. The fully discrete problem for the computation at a subsequent time step $t_{n+1} = t_n + \tau_n$, with the time step size $\tau_n > 0$, reads as: find the discrete solution $\boldsymbol{y}^{n+1} \approx \boldsymbol{y}(t_{n+1})$ satisfying

$$\mathbf{M}(\boldsymbol{y}^{n+1} - \boldsymbol{\Phi}_k^n) = \xi_k \boldsymbol{f}(t_{n+1}, \boldsymbol{y}^{n+1}), \tag{17}$$

with the $\boldsymbol{\Phi}^n$ 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 time step size and the chosen order k, see also [10, Sect. 4.1.2]. For the definition of the singular mass matrix $\mathbf{M}$ and the right hand side function $\boldsymbol{f}$ according to the weak formulation (15) we refer to [5, 10].

3.2 Adaptive Algorithm

Based on the work by Melenk and Wohlmuth [16] we give in Algorithm 1 the pseudo-code of the hp-marking algorithm as we presented in [6]. We incorporated this marking algorithm in our general-purpose space and time adaptive solution algorithm from [5, 9, 10].

Given the control parameters θ_{r}, γ_h, γ_p and γ_n, we start computing the spatial error estimates η_K with a gradient recovery estimator, like in [10]. In addition we set the error predictions $\eta_{K,\text{pred}}$ initially to infinity to enforce p-refinement in the first adaptive cycle.

In a loop over all elements of the mesh, we decide with a maximum criterion whether an element should be considered for hp-refinement or not. The decision between h- or p-refinement is made by a comparison of the computed error estimate η_K and the predicted estimate $\eta_{K,\text{pred}}$, as in [16]. We follow [16] for the explanation: if the estimate is smaller than the prediction $\eta_K < \eta_{K,\text{pred}}$, we assume a smooth solution and thus use p-refinement. In the other case, we assume a non-smooth solution and use h-refinement. After each decision for an element, the predictions are recomputed. In case of a h-refinement step for an element with local degree p_K, the prediction is computed under the assumption of optimal convergence according to $\eta_{K_s,\text{pred}} := 1/2\gamma_h(0.5)^{p_K}\eta_K$. Beside the constant part $1/2\gamma_h$, the formula results from the expectation of algebraic convergence. In case of p-refinement, exponential convergence is expected and thus we use as new prediction $\eta_{K,\text{pred}} := \gamma_p\eta_K$, where the estimate η_K is reduced by a factor $\gamma_p \in (0, 1)$.

Algorithm 1 Marking algorithm for hp-refinement from [6].

```
Let $\theta_\mathrm{r} = 0.5$, $\gamma_h = 2$, $\gamma_p = \sqrt{0.4}$, $\gamma_n = 1.0$;
Given $\eta_K$, $\eta_{K,\mathrm{pred}}$ for all $K \in \mathcal{K}$;
Compute maximal estimate $\eta_\mathrm{max} := \max_{K \in \mathcal{K}} \eta_K$;
for all $K \in \mathcal{K}$ do
    if $\eta_K \geq \theta_\mathrm{r}\eta_\mathrm{max}$ then
        if $\eta_K > \eta_{K,\mathrm{pred}}$ then
            % $h$-refine element $K$;
            Subdivide $K$ into $K_s$ children, $s = 1, \dots, 2^d$;
            Set $\eta_{K_s,\mathrm{pred}} := 1/2\gamma_h(0.5)^{p_K}\eta_K$
                for all children $K_s$, $s = 1, \dots, 2^d$;
        else
            if $p_K < p_\mathrm{max}$ then
                % $p$-refine element $K$;
                $p_K := p_K + 1$;
                Set $\eta_{K,\mathrm{pred}} := \gamma_p \eta_K$;
            else
                % $h$-refine element $K$;
                Subdivide $K$ into $K_s$ children, $s = 1, \dots, 2^d$;
                Set $\eta_{K_s,\mathrm{pred}} := 1/2\gamma_h(0.5)^{p_K}\eta_K$
                    for all children $K_s$, $s = 1, \dots, 2^d$;
            end if
        end if
    else
        % No refinement of element $K$;
        Set $\eta_{K,\mathrm{pred}} := \gamma_n \eta_{K,\mathrm{pred}}$;
    end if
end for
```

Note, that we allow p-refinement only in a predefined range of degrees $p \in \{p_\mathrm{min}, \dots, p_\mathrm{max}\}$. If the algorithm would decide for p-refinement but the maximum degree is already reached, h-refinement is performed instead. This hp-marking algorithm replaces the h-marking procedure in our space and time adaptive algorithm from [5, 9, 10] used after every time step, if the computed solution does not fulfill the spatial tolerances.

The hp-coarsening is decoupled from the refinement step. If a certain number of time steps were computed on the same mesh, we check for coarsening with the criterion $\eta_K \leq \theta_\mathrm{c}\eta_\mathrm{max}$, with $\theta_\mathrm{c} = 0.05$, a strategy also used by Baňas and Nürnberg [17]. If an element is marked for coarsening, we performed both an h- and p-coarsening step.

4 Numerical Experiments

We investigate our hp-adaptive solution algorithm at the example of a spherical electrode particle, as in [6]. However, in contrast to our previous work, the chemo-mechanical model presented in Section 2 is different. We assume the spherical symmetry and reduce the computational domain to the one-dimensional unit interval $\Omega = (0, 1)$ representing the radial line form the particle center $\Gamma_0 = \{0\}$ to the surface $\Gamma_\mathrm{ext} = \{1\}$. The geometry is then hidden in the adapted quadrature weight.

We choose the model parameters according to lithium iron phosphate, as in [5]. The dimensionless parameters are $D = 1.6 \times 10^2$, $\alpha_1 = 4.5$, $\alpha_2 = -9$, $\kappa = 3.91 \times 10^{-4}$, $\mathrm{E_H} = 2.19 \times 10^3$, $\nu = 0.26$, $V = 6.64 \times 10^{-2}$, $N_\mathrm{ext} = -1/3$. In the particle center Γ_0 we set the artificial boundary conditions $\nabla c \cdot \boldsymbol{n} = \boldsymbol{N} \cdot \boldsymbol{n} = 0$ and $\boldsymbol{u} = \boldsymbol{0}$. At the initial time we use the constant normalized concentration $c_0 = 0.01$.

We reuse the implemented hp-adaptive algorithm from [6], which is a `C++`-code based on the functionalities of the open-source finite element library *deal.II* [18–20] and the direct LU-decomposition from [21]. For the simulations we allocated 20 Intel Xeon Gold 6230 CPUs with 2.1 GHz and 1 GB RAM on a single node of the HPC system bwUniCluster 2.0.

4.1 hp-Refinement Profile

In a first simulation, we solve the model equations with our hp-adaptive solution algorithm and visualize the concentration profile together with the refinement level n, which is related to the mesh width by $h = 2^{-n}$, and the polynomial degree of the finite element method p per element in Figure 1 at three characteristic state of charges (SOC). According to [5, Eq. (13)] we have the relation $\mathrm{SOC}(t) = c_0 + t$ between the SOC and the simulation time $t \in [0, T_\mathrm{end}]$. We employ our adaptive solution algorithm with $p \in \{2, \dots, 6\}$, which starts from a uniform discretization with $h_0 = 2^{-7}$, $p_0 = 6$, $\tau_0 = 10^{-6}$ and choose the tolerances $\mathrm{AbsTol} = 10^{-4}$ and $\mathrm{RelTol} = 10^{-6}$ for both, space and time estimates.

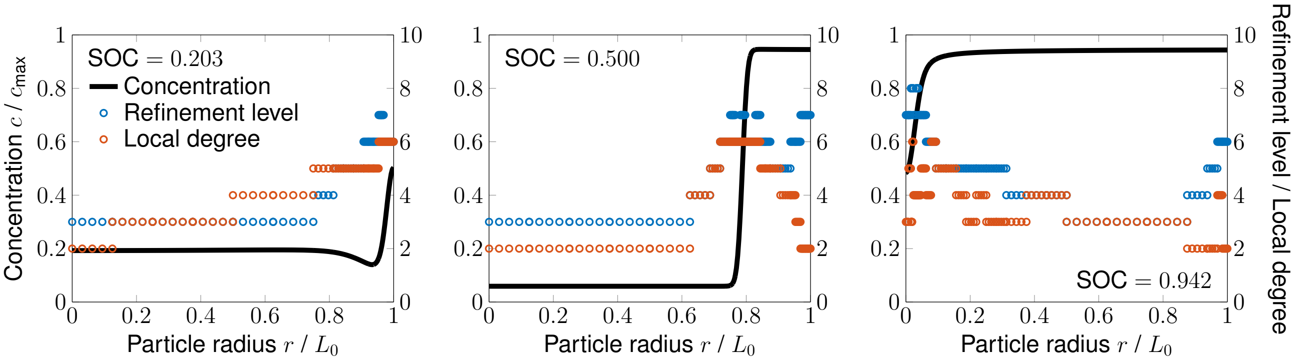

Fig. 1: Concentration profile at three characteristic state of charges (SOC) with refinement level and polynomial degree of the finite element method per element. The SOC is related to the simulation time $t \in [0, T_{\text{end}}]$ by $\text{SOC}(t) = c_0 + t$.

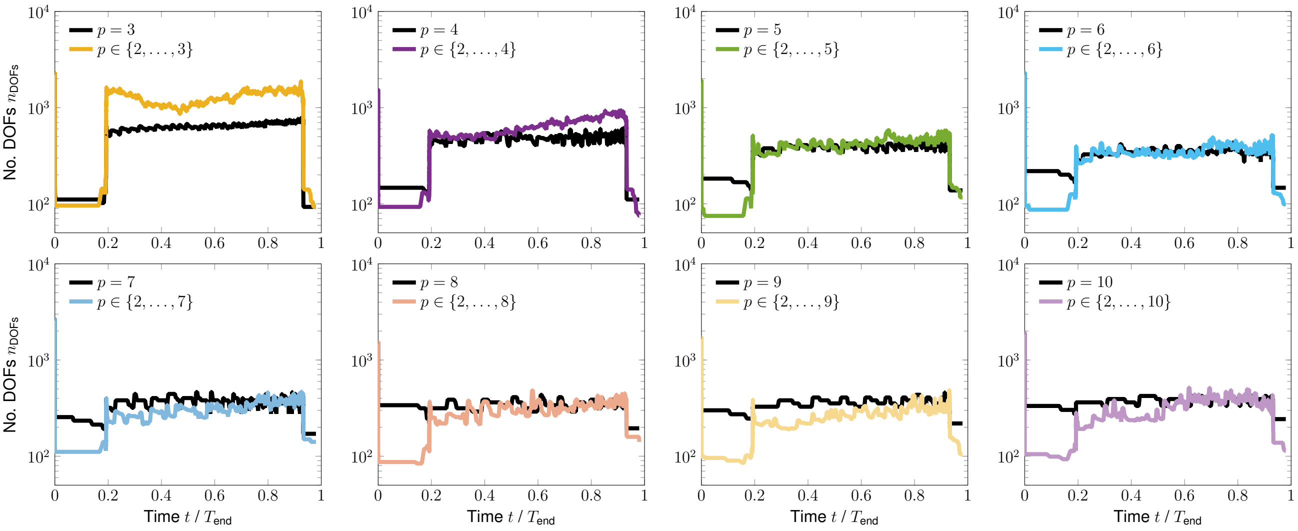

Fig. 2: Number of DOFs over time for different h- and hp-adaptive cases.

The concentration profile over time is in qualitative accordance with the results from our previous work [5]. However, the refinement profile is slightly different to our previous study [6], which may be caused by the adapted model equations. We still observe the fine resolution in h and p exactly in the phase transition zone.

4.2 Computational Complexity

In the second study, we measure the computational complexity in terms of the number of degrees of freedom (DOFs) over time. In Figure 2, we plot the number of DOFs over time for the h-adaptive algorithm with fixed polynomial degree $p = p_{\max}$ together with the number of DOFs over time for the hp-adaptive algorithm allowing $p \in \{2, \ldots, p_{\max}\}$. Again we used the same initialization and tolerances of the algorithm as above.

In the first row of Figure 2, we plotted the number of DOFs over time for the cases with a maximum polynomial degree $p_{\max} \in \{3, 4, 5, 6\}$. These cases we already studied in our previous work [6]. Again for fixed $p = p_{\max}$ we observe an increase of the number of DOFs in the single-phase diffusion stage ($t \lesssim 0.2$) before the phase separation sets in. In this stage, the hp-adaptive algorithm reduces the computational complexity. However, in the two-phase diffusion stage ($0.2 \lesssim t \lesssim 0.95$) the number of DOFs with our hp-adaptive algorithm is still greater or approximately equal compared to the case with fixed $p = p_{\max}$. In the second row of Figure 2, we plotted the numbers of DOFs over time for the cases with a maximum polynomial degree $p_{\max} \in \{7, 8, 9, 10\}$. Again we see an increase of the number of DOFs in the single-phase diffusion stage. However, the number of DOFs in the two-phase diffusion stage stagnates. For $p_{\max} > 6$, the hp-adaptive algorithm is able to reduce the number of DOFs in the single-phase as well as in the two-phase diffusion stage. So we need a sufficient large range of polynomial degrees to significantly reduce the computational complexity also in the two-phase diffusion stage.

5 Conclusion

In this article we have employed our hp-adaptive finite element solver for the numerical solution of a thermodynamically consistent electrode particle model coupling chemistry, phase separation and mechanical deformation. In the beginning we have explained the model equations and afterwards have recalled the numerical solution algorithm. We have shown the concentration profile over time for a lithium iron phosphate particle, which is in qualitative accordance to the previously studied chemo-mechanical model. Furthermore, we pointed out the reduced computationally complexity in terms of the number of DOFs for an increasing range of polynomial degrees. In sum, we have demonstrated the computational savings using our hp-adaptive solution algorithm for an application example. In particular, we have observed that for the reduction of complexity in the two-phase regime a sufficiently large range of polynomial degrees is necessary.

Our future research, we will review the coarsening procedure of the algorithm.

Acknowledgements 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. In addition, the authors acknowledge support by the state of Baden-Württemberg through bwHPC. Open access funding enabled and organized by Projekt DEAL.

References

[1] J. M. Tarascon, Phil. Trans. R. Soc. A **368**, 3227–3241 (2010).
[2] A. Manthiram, ACS Cent. Sci. **3**(10), 1063–1069 (2017).
[3] Y. Zhao, P. Stein, Y. Bai, M. Al-Siraj, Y. Yang, and B. X. Xu, J. Power Sources **413**, 259–283 (2019).
[4] Y. Zhao, P. Stein, and B. X. Xu, Comput. Methods Appl. Mech. Engrg. **297**, 325–347 (2015).
[5] G. F. Castelli, L. von Kolzenberg, B. Horstmann, A. Latz, and W. Dörfler, Energy Technol. **9**(6), 2000835 (2021).
[6] G. F. Castelli and W. Dörfler, Examples and Counterexamples **2**, 100083 (2022).
[7] L. von Kolzenberg, A. Latz, and B. Horstmann, Battery Supercaps **5**(2), e202100216 (2022).
[8] R. Schoof, G. F. Castelli, and W. Dörfler, Parallelization of a finite element solver for chemo-mechanical coupled anode and cathode particles in lithium-ion batteries, in: ECCOMAS 2022, (2022), submitted.
[9] G. F. Castelli and W. Dörfler, Study on an adaptive finite element solver for the Cahn–Hilliard equation, in: Numerical Mathematics and Advanced Applications ENUMATH 2019, edited by F. J. Vermolen and C. Vuik, Lect. Notes Comput. Sci. Eng. Vol. 139 (Springer, Cham, 2021), pp. 245–253.
[10] G. F. Castelli, Numerical Investigation of Cahn–Hilliard-Type Phase-Field Models for Battery Active Particles, PhD thesis, Karlsruhe Institute of Technology (KIT), 2021.
[11] G. F. Castelli and W. Dörfler, Proc. Appl. Math. Mech. 2021 **21**(1), e202100169 (2021).
[12] D. Braess, Finite Elements, third edition (Cambridge University Press, Cambridge, 2007).
[13] L. F. Shampine and M. W. Reichelt, SIAM J. Sci. Comput. **18**(1), 1–22 (1997).
[14] L. F. Shampine, M. W. Reichelt, and J. A. Kierzenka, SIAM Rev. **41**(3), 538–552 (1999).
[15] The MathWorks, Inc.`http://www.mathworks.com`.
[16] J. M. Melenk and B. I. Wohlmuth, Adv. Comput. Math. **15**(1–4), 311–331 (2001).
[17] L. Baňas and R. Nürnberg, J. Comput. Appl. Math. **218**(1), 2–11 (2008).
[18] W. Bangerth, R. Hartmann, and G. Kanschat, ACM Trans. Math. Software **33**(4), Art. 24, 27 (2007).
[19] W. Bangerth and O. Kayser-Herold, ACM Trans. Math. Software **36**(1), Art. 4, 31 (2009).
[20] D. Arndt, W. Bangerth, B. Blais, M. Fehling, R. Gassmöller, T. Heister, L. Heltai, U. Köcher, M. Kronbichler, M. Maier, P. Munch, J. P. Pelteret, S. Proell, K. Simon, B. Turcksin, D. Wells, and J. Zhang, J. Numer. Math. **29**(3), 171–186 (2021).
[21] T. A. Davis, ACM Trans. Math. Software **30**(2), 196–199 (2004).