PAPER

An efficient numerical framework for the amplitude expansion of the phase-field crystal model

View the article online for updates and enhancements.

IOP ebooks™

Bringing you innovative digital publishing with leading voices to create your essential collection of books in STEM research.

Start exploring the collection - download the first chapter of every title for free.

An efficient numerical framework for the amplitude expansion of the phase-field crystal model

Simon Praetorius[1] , **Marco Salvalaglio**[1] **and Axel Voigt**[1,2]

[1] Institute of Scientific Computing, Technische Universität Dresden, D-01062 Dresden, Germany
[2] Dresden Center for Computational Materials Science (DCMS), TU Dresden, D-01062 Dresden, Germany

E-mail: marco.salvalaglio@tu-dresden.de

Abstract

The study of polycrystalline materials requires theoretical and computational techniques enabling multiscale investigations. The amplitude expansion of the phase-field crystal model allows for describing crystal lattice properties on diffusive timescales by focusing on continuous fields varying on length scales larger than the atomic spacing. Thus, it allows for the simulation of large systems still retaining details of the crystal lattice. Fostered by the applications of this approach, we present here an efficient numerical framework to solve its equations. In particular, we consider a real space approach exploiting the finite element method. An optimized preconditioner is developed in order to improve the convergence of the linear solver. Moreover, a mesh adaptivity criterion based on the local rotation of the polycrystal is used. This results in an unprecedented capability of simulating large, three-dimensional systems including the dynamical description of the microstructures in polycrystalline materials together with their dislocation networks.

Keywords: polycrystalline growth, grain boundaries, finite element method, coarse graining, dislocations, phase field modeling, phase field crystal

(Some figures may appear in colour only in the online journal)

Introduction

The study of polycrystalline materials requires modeling over various length scales. While the nucleation and structure of defects has to be considered on an atomistic resolution, the size,

shapes and arrangements of grains can be considered on mesoscopic length scales and mechanical behavior can be tackled on a macroscopic scale. However, all these phenomena strongly affect material properties, such as fracture or yield stress and thus require multiscale investigations [1].

Coarse-grained, mesoscale approaches are therefore viable tools to provide bridging-scale information of polycrystalline materials. Among the many different models reported in the literature, the so-called phase-field crystal (PFC) model has been developed in order to filter out atom vibrations and enabling the investigation of relatively long dynamics [2–4]. It describes the crystal lattice by means of a continuous field that is the atomic probability density, n, averaged over vibrational length scales. Although the original model has been further extended, as e.g. in [5–7], and applied to describe several different properties and mechanisms for crystals and quasicrystals, as e.g. in [8–13], it is restricted to relatively small sizes as the continuous density has still to be resolved at the atomic length scale.

This limit has been overcome by the so-called amplitude expansion of the PFC model (APFC), providing coarse graining in both time and space in a single framework [14–17]. This model focuses on the amplitudes of the atomic probability density which vary on a larger length scale than the atomic spacing. In its standard formulation, it is restricted to relatively small deformations [18] and approximate atomic rearrangements such as dislocation-core structures. However, it encodes a detailed description of deformations matching continuum-elasticity theories and allows for large simulations approaching the macroscopic length scales, still retaining details of the atomic length scale [19]. The original model can be further extended as it has been done, for instance, to account for binary systems [20] and to control the energy of defects and interfaces [21]. Moreover, it has been recently used to study the anisotropic shrinkage in 3D of small-angle spherical grain boundaries (GBs) regardless of the crystal lattice symmetry [22]. In addition, the APFC framework has been proven suitable to allow for the description of hydrodynamics [23], dislocation dynamics [24] and surface-energy anisotropy [25] within the more general PFC framework.

In this work, we report on the development of an efficient numerical framework to integrate the partial differential equations of the APFC model, enabling unprecedented simulations of large polycrystalline systems in two- and three-dimensions. We provide an optimized discretization of the APFC equations based on the one reported in [21]. Then, we propose a new preconditioner allowing for faster convergence of iterative linear solvers. We also illustrate a mesh-adaptivity strategy exploiting continuous fields, namely local crystal rotations, which can be derived directly by the complex amplitudes to solve for in the APFC model. Performance studies showing comparisons with a standard solver and assessment of the numerical parameter entering the method are reported. Then, we use the developed framework to simulate the large scale growth of a polycrystal in 2D and 3D accounting for both the evolution of the microstructures and the defects forming between grains having different orientations.

Model and implementation

The APFC model [14–17] is based on the representation of the atomic probability density, n, as sum of plane waves

$$n(\mathbf{r}) = n_0 + \sum_{j=1}^{J} [\eta_j(\mathbf{r}) e^{i\mathbf{k}_j \cdot \mathbf{r}} + \eta_j^*(\mathbf{r}) e^{-i\mathbf{k}_j \cdot \mathbf{r}}], \tag{1}$$

with n_0 the average density, here set to zero without loss of generality, $\eta_j(\mathbf{r})$ the amplitude of each plane wave, and $\mathbf{k}_j$ the reciprocal lattice vector representing a specific crystal symmetry. Complex amplitude functions $\eta_j : \Omega \to \mathbb{C}$ with Ω a rectangular (2D) or parallelepiped (3D) domain are considered, allowing for describing distortions and rotations of the crystal lattice with respect to a reference state described by the $\mathbf{k}_j$ vectors. The model is based on the definition of a free energy in terms of the η_j's in the approximation of slowly varying amplitudes, i.e. varying over a length scale significantly larger than the lattice spacing,

$$F(\vec{\eta}) := \int_\Omega f^{\mathrm{s}}(\vec{\eta}, \vec{\eta}^*) + \frac{1}{2} \sum_{j=1}^J |\mathcal{G}_j \eta_j|^2 \, \mathrm{d}\mathbf{r}, \tag{2}$$

with $\vec{\eta} = \{\eta_j\}$, $\mathcal{G}_j := \Delta + 2\mathrm{i}\mathbf{k}_j \cdot \nabla$, and f^{s} a polynomial bulk free energy density. For the sake of convenience, the energy is here normalized by a positive, non-zero parameter a_0 which would multiply the second term at the right-hand side of (2) [21, 26]. The evolution laws for amplitudes η_j are given by the L^2-gradient flow of the energy F

$$\frac{\partial \eta_j}{\partial t} = -\kappa_j \frac{\delta F}{\delta \eta_j^*}, \quad j = 1, \ldots, J, \tag{3}$$

with $\kappa_j = a_0 |\mathbf{k}_j|^2$ a mobility coefficient and

$$\frac{\delta F}{\delta \eta_j^*} = \mathcal{G}_j^2 \eta_j + \frac{\delta f^{\mathrm{s}}(\vec{\eta}, \vec{\eta}^*)}{\delta \eta_j^*}. \tag{4}$$

The bulk energy term $f^{\mathrm{s}}(\vec{\eta}, \vec{\eta}^*)$ entering equation (2) can be written as

$$f^{\mathrm{s}}(\vec{\eta}, \vec{\eta}^*) = \frac{a_1}{a_0} A + \frac{a_2}{a_0}\left(A^2 - 2\sum_{j=1}^J |\eta_j|^4\right) + g^{\mathrm{s}}(\vec{\eta}, \vec{\eta}^*), \tag{5}$$

with a_i positive parameters, $g^{\mathrm{s}}(\vec{\eta}, \vec{\eta}^*)$ a polynomial in $\vec{\eta}$ and $\vec{\eta}^*$ encoding the lattice symmetry, and $A = 2\sum_{j=1}^J |\eta_j|^2$. In this work we consider a triangular lattice in 2D and a face-centered cubic (fcc) lattice in 3D. The former is obtained by setting $J = 3$, g^{s} entering equation (4) as

$$g^{\mathrm{s}}(\vec{\eta}, \vec{\eta}^*) = -2\frac{a_3}{a_0}(\eta_1 \eta_2 \eta_3 + \eta_1^* \eta_2^* \eta_3^*), \tag{6}$$

and the set of $\mathbf{k}_j$ vectors as

$$\mathbf{k}_1 = \left(-\frac{\sqrt{3}}{2}, -\frac{1}{2}\right), \qquad \mathbf{k}_2 = (0, 1), \qquad \mathbf{k}_3 = \left(\frac{\sqrt{3}}{2}, -\frac{1}{2}\right). \tag{7}$$

For the fcc lattice $J = 7$, thus leading to a larger number of amplitude functions to be considered. For the sake of brevity we refer to [21] for the corresponding g^{s} polynomial and $\mathbf{k}_j$ vectors. Following [21, 22, 26], the parameters entering the energy are set to favor the growth of the solid phase as $a_0 = 0.98$, $a_1 = 0.01$, $a_2 = 0.25$, $a_3 = 0.5$.

Discretization and nonlinear solver

Throughout the following, we use the $L^2(\Omega, \mathbb{C})$ scalar product $(u, v) := \int_\Omega u^*(\mathbf{r}) \cdot v(\mathbf{r}) d\mathbf{r}$, with u, v complex scalar or vector valued functions and u^* the complex conjugate of u. By introducing a set of auxiliary variables $\zeta_j := \mathcal{G}_j \eta_j$, we derive the weak form of the evolution equations (3) using a natural splitting of the operator $\mathcal{G}_j^2$:

$$(\zeta_j, \vartheta_1) + (\nabla\eta_j, \nabla\vartheta_1) - 2\mathrm{i}(\mathbf{k}_j \cdot \nabla\eta_j, \vartheta_1) = 0$$
$$\left(\frac{\partial\eta_j}{\partial t}, \vartheta_2\right) - \kappa_j((\nabla\zeta_j, \nabla\vartheta_2) - 2\mathrm{i}(\mathbf{k}_j \cdot \nabla\zeta_j, \vartheta_2)) = \left(\frac{\delta f^{\mathrm{s}}(\vec{\eta}, \vec{\eta}^*)}{\delta\eta_j^*}, \vartheta_2\right), \qquad \forall \vartheta_1, \vartheta_2 \in H^1(\Omega, \mathbb{C}). \tag{8}$$

This set of coupled second order equations can then be discretized using conforming finite-dimensional approximations of the functions space $H^1(\Omega, \mathbb{C})$. Therefore, let us now consider a triangulation of the domain Ω, $\mathcal{T}_h$, with h the minimal diameter of the grid elements, and the finite-element space $\mathcal{V}_h \subset H^1(\Omega, \mathbb{C})$

$$\mathcal{V}_h = \{\vartheta \in C(\Omega, \mathbb{C}) : \vartheta|_S \in \mathbb{P}_1(S, \mathbb{C}), S \in \mathcal{T}_h\},$$

the continuous space of complex valued local linear polynomials on elements $S \in \mathcal{T}_h$ of the triangulation. The space discretization method of (8) corresponds to find η_j^h, $\zeta_j^h \in L^1([0, T], \mathcal{V}_h)$ such that

$$(\zeta_j^h, \vartheta_1^h) + (\nabla\eta_j^h, \nabla\vartheta_1^h) - 2\mathrm{i}(\mathbf{k}_j \cdot \nabla\eta_j^h, \vartheta_1^h) = 0$$
$$\left(\frac{\partial\eta_j^h}{\partial t}, \vartheta_2^h\right) - \kappa_j((\nabla\zeta_j^h, \nabla\vartheta_2^h) - 2\mathrm{i}(\mathbf{k}_j \cdot \nabla\zeta_j^h, \vartheta_2^h)) = \left(\frac{\delta f^{\mathrm{s}}(\vec{\eta}^h, \vec{\eta}^{h*})}{\delta\eta_j^*}, \vartheta_2^h\right), \qquad \forall \vartheta_1^h, \vartheta_2^h \in \mathcal{V}_h, \tag{9}$$

for $j = 1, \ldots, J$ in the time interval $[0, T]$ provided that $\eta_j^h(t = 0) = \eta_j^{h,0}$ is given. For better readability, we drop the superscript h in the following.

Due to the bulk polynomial f^{s}, equations (9) are a set of nonlinear equations. We combine a simplified Newton method with the idea of an operator splitting method, in order to separate the equation for each amplitude. This will allow us to efficiently solve the set of equations also if the number of amplitudes is large, as in the case of fcc symmetry. We introduce the Jacobian of $F_j^{\mathrm{s}} := \delta f^{\mathrm{s}}/\delta\eta_j^*$, but only in direction η_j, i.e.

$$\mathrm{d}F_j^{\mathrm{s}}(\vec{\eta}) := \frac{\delta F_j^{\mathrm{s}}}{\delta\eta_j}(\vec{\eta}) = \frac{\delta^2 f^s}{\delta\eta_j\,\delta\eta_j^*}(\vec{\eta}).$$

Let $\{\phi_i\}$ be a basis of $\mathcal{V}_h$. Moreover, let us introduce the mass matrix $\mathbf{M} := [(\phi_i, \phi_j)]$, the stiffness matrix $\mathbf{K} := [(\nabla\phi_i, \nabla\phi_j)]$, $\mathbf{T}^k := [(2\mathbf{k}_k \cdot \nabla\phi_i, \phi_j)]$, $\mathbf{J}^k(\vec{\eta}) := [(-\mathrm{d}F_k^{\mathrm{s}}(\vec{\eta})\phi_i, \phi_j)]$, and the vector $\mathbf{h}^k(\vec{\eta}) := [(F_k^{\mathrm{s}}(\vec{\eta}) - \mathrm{d}F_k^{\mathrm{s}}(\vec{\eta})\eta_k, \phi_j)]$. Note, that these matrices are assembled with the real valued basis functions ϕ_i.

For the time discretization we choose a backward Euler method. In the following let $0 = t_0 < t_1 < \ldots < t_M = T$ be a discretization of the time interval $[0, T]$ with $\tau_m := t_m - t_{m-1}$ and the timestep solution $\eta_j^m = \sum_i \eta_{j,i}^m \phi_i \equiv \eta_j(t_m)$ with $\boldsymbol{\eta}_j^m = [\eta_{j,i}^m]$ the vector of the coefficients for the jth amplitude at the timestep m. Analogously, we define $\zeta_j^m = \sum_i \zeta_{j,i}^m \phi_i \equiv \zeta_j(t_m)$ and $\boldsymbol{\zeta}_j^m = [\zeta_{j,i}^m]$.

Algorithm 1. Timestepping scheme and nonlinear solver for the APFC equation.

Let $\boldsymbol{\eta}_j^0, \boldsymbol{\zeta}_j^0 \in \mathbb{C}^{\dim(V_h)}$ $(j = 1, \ldots, J)$ be given.   ▷ Initial solution
for $m = 1, \ldots, M$ **do**   ▷ Loop over all timesteps
  Let $\boldsymbol{\eta}_j^{(0)} = \boldsymbol{\eta}_j^{m-1}$ and $\boldsymbol{\zeta}_j^{(0)} = \boldsymbol{\zeta}_j^{m-1}$ $(j = 1, \ldots, J)$.   ▷ Initial iteration for Newton method

  for $k=1, \ldots, K$ **do**   ▷ Newton iteration
    for $j=1, \ldots, J$ **do**   ▷ Loop over amplitudes
     Find $\boldsymbol{\eta}_j^{(k)} \in \mathbb{C}^{\dim(V_h)}$, $\boldsymbol{\zeta}_j^{(k)} \in \mathbb{C}^{\dim(V_h)}$, such that

$$
\begin{bmatrix} \mathbf{M} & \mathbf{K} - \mathrm{i}\mathbf{T}^j \\ -\kappa_j(\mathbf{K} - \mathrm{i}\mathbf{T}^j) & \frac{1}{\tau_m}\mathbf{M} + \mathbf{J}^j(\vec{\boldsymbol{\eta}}^{(k-1)}) \end{bmatrix} \begin{pmatrix} \boldsymbol{\zeta}_j^{(k)} \\ \boldsymbol{\eta}_j^{(k)} \end{pmatrix} = \begin{pmatrix} 0 \\ \frac{1}{\tau_m}\mathbf{M}\boldsymbol{\eta}_j^{m-1} + \mathbf{h}^j(\vec{\boldsymbol{\eta}}^{(k-1)}) \end{pmatrix}. \tag{10}
$$

    end for
  end for
  Update $\boldsymbol{\eta}_j^m = \boldsymbol{\eta}_j^{(K)}$ and $\boldsymbol{\zeta}_j^m = \boldsymbol{\zeta}_j^{(K)}$ $(j = 1, \ldots, J)$.
end for

The timestep iteration with an inner Picard iterative process for the linearized $F_j^{\mathfrak{s}}$ can be found in algorithm 1. The discretization is implemented in the finite-element framework AMDiS [27, 28].

Solving the linear system

The skew symmetric linear system (10) needs to be solved in each timestep and in each Newton iteration. For large-scale numerical setups in 3D, the widely used direct spare LU factorizations are out of applicability due to large memory requirements. Therefore, we concentrate on preconditioned iterative methods, belonging in particular to the Krylov subspace, to solve (10). Two strategies will be compared: domain decomposition with local sparse LU factorization also known as block Jacobi (`bjacobi`) preconditioner [29] and a dedicated preconditioner based on the schur-complement method (`apfc`), as detailed in the following.

For the construction of the schur-complement preconditioner, we follow the lines of [30, 31], and first simplify the notation by neglecting the amplitude index j. For each amplitude a structurally similar system has to be solved, so we can construct one preconditioner that can be applied to all amplitudes, with adapted coefficients. Also, we drop the timestep index m and Newton iterate index (k) and just write $\tau \equiv \tau_m$. In spite of the operator $\mathcal{G}_j$ we introduce $\mathbf{G}^j := -\mathbf{K} + \mathrm{i}\mathbf{T}^j$ and write the linear system as

$$
\underbrace{\begin{bmatrix} \mathbf{M} & -\mathbf{G} \\ \kappa\mathbf{G} & \frac{1}{\tau}\mathbf{M} + \mathbf{J} \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{pmatrix} \boldsymbol{\zeta} \\ \boldsymbol{\eta} \end{pmatrix}}_{\mathbf{x}} = \begin{pmatrix} 0 \\ \frac{1}{\tau}\mathbf{M}\boldsymbol{\eta}^m + \mathbf{h} \end{pmatrix} \tag{11}
$$

with schur-complement $\mathbf{S} = \frac{1}{\tau}\mathbf{M} + \mathbf{J} + \kappa\mathbf{G}\mathbf{M}^{-1}\mathbf{G}$.

While $\mathbf{M}$ and $\mathbf{K}$ are real matrices, $\mathbf{G}$, $\mathbf{J}$, and $\mathbf{h}$ may be complex valued. Following the ideas of [30, 31] to approximate the schur-complement matrix $\mathbf{S}$ in a way that allows for a simple factorization, we propose the preconditioner:

$$
\mathbf{P} := \begin{bmatrix} \mathbf{M} & 0 \\ \kappa\mathbf{G} & \dfrac{1}{\sqrt{\tau}}\mathbf{M} + \sqrt{\kappa}\,\mathbf{K} \end{bmatrix} \begin{bmatrix} \mathbf{I} & -\mathbf{M}^{-1}\mathbf{K} \\ 0 & \mathbf{M}^{-1}\big(\dfrac{1}{\sqrt{\tau}}\mathbf{M} + \sqrt{\kappa}\,\mathbf{K}\big) \end{bmatrix}
$$

$$
= \begin{bmatrix} \mathbf{M} & -\mathbf{K} \\ \kappa\mathbf{G} & \dfrac{1}{\tau}\mathbf{M} + 2\sqrt{\dfrac{\kappa}{\tau}}\,\mathbf{K} + \kappa(\mathbf{K}\mathbf{M}^{-1}\mathbf{K} - \mathbf{G}\mathbf{M}^{-1}\mathbf{K}) \end{bmatrix} \tag{12}
$$

with schur-complement $\mathbf{S}_P := \big(\frac{1}{\sqrt{\tau}}\mathbf{M} + \sqrt{\kappa}\,\mathbf{K}\big)\mathbf{M}^{-1}\big(\frac{1}{\sqrt{\tau}}\mathbf{M} + \sqrt{\kappa}\,\mathbf{K}\big)$. Since the highest order term in $\mathbf{G}\mathbf{M}^{-1}\mathbf{G}$ corresponds to $\mathbf{K}\mathbf{M}^{-1}\mathbf{K}$, the matrix $\mathbf{S}_P$ is expected to be a good approximation of $\mathbf{S}$.

The application of $\mathbf{P}$ to a vector $\mathbf{b} = (\beta_1, \beta_2)^\top \in \mathbb{C}^{2\cdot\dim(V_h)}$, i.e. $\mathbf{x} = \mathbf{P}_2^{-1}\mathbf{b}$, with $\mathbf{x} = (\chi_1, \chi_2)^\top$, can be performed in four steps:

1. solve $\mathbf{M}\beta'_1 = \beta_1$
2. solve $\big(\frac{1}{\sqrt{\tau}}\mathbf{M} + \sqrt{\kappa}\,\mathbf{K}\big)\beta'_2 = \beta_2 - \kappa\mathbf{G}\beta'_1$
3. solve $\big(\frac{1}{\sqrt{\tau}}\mathbf{M} + \sqrt{\kappa}\,\mathbf{K}\big)\chi_2 = \mathbf{M}\beta'_2$
4. $\chi_1 = \beta'_1 - \frac{1}{\sqrt{\kappa}}\big(\beta'_2 - \frac{1}{\sqrt{\tau}}\chi_2\big)$.

Thus, only one mass-matrix and two diffusion-matrix systems need to be solved, plus two matrix-vector products need to be performed in the application of the preconditioner. Since efficient (iterative) linear solvers exist to approximate the solution of $\mathbf{M}^{-1}\mathbf{b}$ and $\big(\frac{1}{\sqrt{\tau}}\mathbf{M} + \sqrt{\kappa}\,\mathbf{K}\big)^{-1}\mathbf{b}$, the preconditioner is cheap to apply.

In order to analyze the quality of the preconditioner $\mathbf{P}$, we look at the eigenvalue spectrum of the right-preconditioned linear system $\mathbf{A}\mathbf{P}^{-1}$. Instead of evaluating the eigenvalues directly, we consider the symbols of the differential operators and the resulting spectrum for a simplified system. Therefore, let Ω be a periodic one-dimensional domain of length 2π. We set the nonlinear term to $\mathbf{J} = 0$, and consider only lattice vectors $\mathbf{k} \in \{-1, +1\}$. Let $\mathcal{F}$ denote the non-unitary Fourier transform in angular frequency $\mathbf{q}$, i.e.

$$
\hat{\eta}(\mathbf{q}) := \mathcal{F}\eta = \int_\Omega \eta(\mathbf{r})\mathrm{e}^{-\mathrm{i}\mathbf{q}\cdot\mathbf{r}}\,\mathrm{d}\mathbf{r},
$$

with inverse transform $\mathcal{F}^{-1}$ and the linear operators $\mathbf{A}$ and $\mathbf{P}$ expressed as $\mathbf{A}\mathbf{x} = \mathcal{F}^{-1}(\mathcal{A}\hat{\mathbf{x}})$ and $\mathbf{P}\mathbf{x} = \mathcal{F}^{-1}(\mathcal{P}\hat{\mathbf{x}})$, respectively, with symbols $\mathcal{A}$ and $\mathcal{P}$. In terms of the frequency $\mathbf{q}$, these symbols can be written as

$$
\mathcal{A} = \begin{bmatrix} 1 & \mathbf{q}^2 + 2\mathbf{q}\cdot\mathbf{k} \\ -\kappa(\mathbf{q}^2 + 2\mathbf{q}\cdot\mathbf{k}) & \dfrac{1}{\tau} \end{bmatrix},
$$

$$
\mathcal{P} = \begin{bmatrix} 1 & 0 \\ -\kappa(\mathbf{q}^2 + 2\mathbf{q}\cdot\mathbf{k}) & \dfrac{1}{\sqrt{\tau}} + \sqrt{\kappa}\mathbf{q}^2 \end{bmatrix} \begin{bmatrix} 1 & -\mathbf{q}^2 \\ 0 & \dfrac{1}{\sqrt{\tau}} + \sqrt{\kappa}\mathbf{q}^2 \end{bmatrix}.
$$

For $\mathcal{Q} := \mathcal{A}\mathcal{P}^{-1}$, $\kappa = 1$, and $\tau = 1$, we obtain the spectrum

$$
\sigma(\mathcal{Q}) = \left\{ 1, \; \lambda(\mathbf{q}) := \frac{\mathbf{q}^4 \mp 4\mathbf{q}^3 + 4\mathbf{q}^2 + 1}{\mathbf{q}^4 + 2\mathbf{q}^2 + 1} \;\middle|\; \mathbf{q} \in \mathbb{R} \right\},
$$

where the sign of the second term in the nominator depends on the choice of the k-vector, see figure 1 for a visualization. The eigenfunction λ approaches the value 1 for zero and
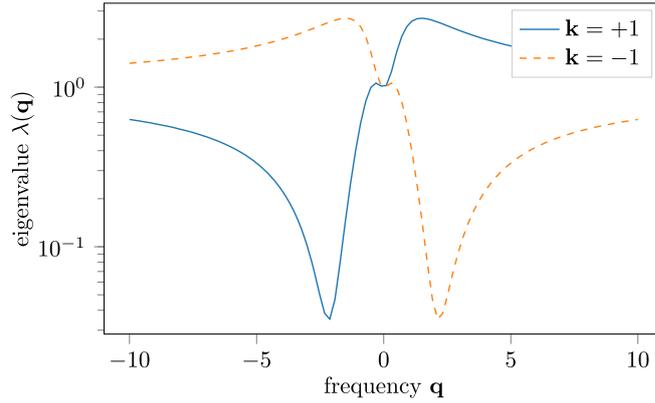
Figure 1. Visualization of the operator spectrum for $\mathbf{k} \in \{-1, +1\}$.

$\pm\infty$ frequencies and has a minimum $\min\{\lambda(\mathbf{q}) \mid \mathbf{q} \in \mathbb{R}\} \approx 0.035$ and maximum $\max\{\lambda(\mathbf{q}) \mid \mathbf{q} \in \mathbb{R}\} \approx 2.70$. The operator symbol $\mathcal{A}$ forms a normal matrix for $|\mathbf{k}_j| = 1$ and the preconditioned operator spectrum is in the positive and real interval $[0.035, 2.7]$, leading to an asymptotic convergence factor (see [29])

$$\rho := \frac{\sqrt{\lambda_{\max}/\lambda_{\min}} - 1}{\sqrt{\lambda_{\max}/\lambda_{\min}} + 1} = 0.796,$$

and thus rapid convergence of an iterative Krylov subspace method, like the flexible generalized minimal residual method [32] (known as FGMRES). See [33] for a similar analysis on a preconditioner for the discretized PFC equation.

Note, if the operator $\mathcal{G}^j$ contains higher order derivatives, a similar procedure for the construction of the preconditioner can be performed that may lead to a slightly different sequence of mass and diffusion-like systems but follows the same structure. A construction of a preconditioner for a sixth order PFC, an eighth order PFC, and a Lifschitz–Petrich type energy is discussed in [34], eventually leading to a sequence of diffusion equations to solve in the application of the preconditioner. The state potential f^s may be extended in different directions and we expect the same procedure to work as above, provided that f^s contains no derivative terms and the problem is well defined.

Mesh adaptivity

The variables to solve for within the APFC model, namely η_j's, are constant for relaxed crystals, oscillate with different periodicity according to the local distortion of the crystal with respect to the reference one, and exhibit significant variation at defects and solid-liquid interfaces. This allows for exploiting mesh adaptivity in order to optimize the numerical approach [35, 36]. We set a local grid refinement in order to resolve the oscillation of amplitudes within grains and ensuring a proper resolution at defects and interfaces.

Amplitude functions describing a rotated crystal of an angle θ can be computed as

$$\eta_j = \phi_j e^{i\delta\mathbf{k}_j(\theta)\cdot\mathbf{r}}, \tag{13}$$

with $\delta\mathbf{k}_j(\theta) = \mathbf{k}_j \cdot \mathbf{R}(\theta) - \mathbf{k}_j$, $\mathbf{R}(\theta)$ the counterclockwise rotation matrix and ϕ_j the real amplitudes describing a relaxed, unrotated crystal. Therefore, the wavelength of η_j's in the presence of a certain rotation θ is $\lambda_j(\theta) = 2\pi/|\delta\mathbf{k}_j(\theta)|$ and the spatial discretization is set as a

fraction of the smallest $\lambda_j(\theta)$ i.e. $h = \min_j(|\lambda_j(\theta)|)/n$. We determined that a proper discretization is achieved with $n \geqslant 10$. In order to practically use this idea we then need to evaluate the local rotation field during the evolution. Notice that equation (13) cannot be just inverted due to its functional form. In order to compute the local rotation we then use the approach described in [19] where the rotational tensor $\boldsymbol{\omega}$ is determined by considering the curl of the local displacement with respect to the relaxed crystal. In practice, its components ω_{ij} representing the rotation in the x_i–x_j-plane, are given by

$$\omega_{ij} = \frac{1}{2}\left[\frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i}\right], \tag{14}$$

with u_i the result of inverting the system of equations

$$\mathbf{k}_j \cdot \mathbf{u} = \arctan\left[\frac{\mathrm{Im}(\eta_j)}{\mathrm{Re}(\eta_j)}\right]. \tag{15}$$

Explicit expressions for u_i, both in 2D and 3D can be found in [19]. In 2D, equation (14) delivers just a scalar field $\omega_{ij} \equiv \omega$. In 3D, $\boldsymbol{\omega}$ has three independent components, ω_d with $d = 1, 2, 3$, and the largest rotation component is chosen in order to ensure the proper resolution for the fastest oscillation in the system. Within grains the discretization, h_{amp}, then reads

$$h_{\mathrm{amp}} = \frac{1}{n}\min_j\left(\left|\,\lambda_j[\max_d(\omega_d)]\,\right|\right). \tag{16}$$

The evaluation of rotations considered here is well posed in the solid phase and also at defects [19]. However, it is not well posed for the liquid, disordered phase as η_j's vanish. Therefore, in order to properly discretize interface regions we define an additional refinement for the interfaces controlled by h_{int} where $|\nabla A|$ is significantly larger than a relatively small threshold ϵ as done in [21]. As a result, h_{int} is ensured also at defects and we set it as the smallest resolution imposed in the system. In addition, a large discretization bound h_{max} is defined for region where $A \sim 0$ or where the local rotation vanishes. Summarizing these ideas, the local discretization, h, is set as

$$h = \begin{cases} h_{\mathrm{int}}, & \text{if } |\nabla A| \geqslant \epsilon \\ \min(\max(h_{\mathrm{amp}}, h_{\mathrm{int}}), h_{\mathrm{max}}), & \text{if } A > 0 \text{ and } |\nabla A| < \epsilon \\ h_{\mathrm{max}}, & \text{elsewhere.} \end{cases} \tag{17}$$

Although not addressed here, this criterion can be extended in order to account for amplitude oscillations due to strain fields, exploiting continuous strain-field components, as derived in [19], instead of $\boldsymbol{\omega}$.

The model and the mesh discretization are illustrated in figure 2 where a simulation reproducing the growth of 20 crystal seeds in 2D with $\theta \in (-15°, 15°)$ is illustrated. A square simulation domain with a side length of 200π is considered (more details are reported in the following section). The initial rotation of the grain is set by initializing the amplitudes by means of equation (13). A, $\mathrm{Re}(\eta_1)$ and ω at different stages during the evolution over time are shown superposed to the mesh used for the simulation. The refinement of the mesh where A changes can be recognized at the solid-liquid interface and at defects, see in particular figure 2(a). The variables to solve for are the amplitudes η_j as illustrated in figure 2(b) by $\mathrm{Re}(\eta_1)$. Notice that the correct resolution of these oscillating functions is ensured by the discretization (17), exploiting the local rotations shown in figure 2(c).
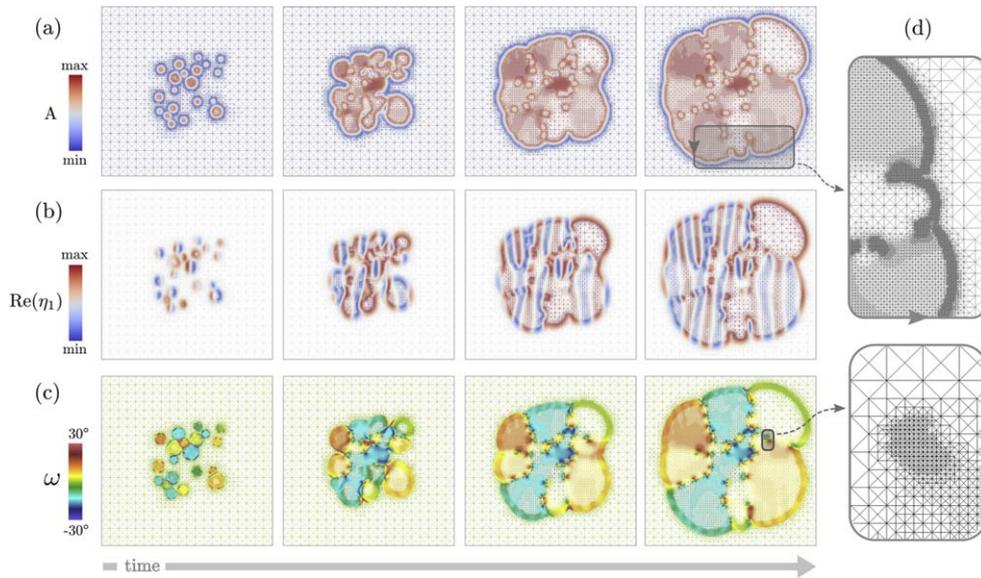
Figure 2. Growth of 20 crystal seeds (in 2D) with $\theta \in (-15°, 15°)$ randomly distributed and triangular lattice symmetry. The spatial discretization is represented by means of the mesh while colors represent: (a) A, (b) $\text{Re}(\eta_1)$, (c) ω. (d) Magnification of two regions showing the mesh on a smaller length scale at the solid-liquid interface (top) and at a defect (bottom). In this simulation the discretization bounds are $h_{\text{int}} \approx 2.0$ and $h_{\text{max}} \approx 40.0$.

The mesh adaptivity criterion illustrated in figure 2 leads to larger refined regions and, in principle, to larger computational costs than the one reported in [35] where an efficient mesh refinement strategy exploiting a polar representation of amplitudes has been proposed. However, as discussed therein, a robust regularization scheme for phase equations and special care to treat high-order derivatives were needed. This practically restricts the applicability of that method and prevents its extension to 3D calculations. An improvement of this description has been recently proposed in [36], exploiting spatially-dependent rotation of the $\mathbf{k}_j$ vectors, although it has been demonstrated only in 2D. The approach we use here allows for an optimized spatial discretization by using the standard APFC model. As it will be illustrated in the following it can be readily exploited to simulate different systems regardless of system dimensionality.

Performance studies

For the numerical evaluation of the linear solvers and the mesh adaption strategy, we concentrate on two types of setups, a triangular symmetry in 2D (TRI) and a fcc symmetry in 3D. The domain Ω is of size $[0, s \cdot 10\pi]^d$ where d is the space dimension and s a scaling factor. The unit of length and time are dimensionless as they enter in the equations of the APFC model reported above. For the setups (TRI, fcc 1, fcc 2, fcc 3), we have chosen s as (20, 7, 14, 28), respectively. To test the numerical framework we consider configurations already known and discussed in literature. The 2D triangular setup is initialized with 20 randomly positioned and oriented grains with orientation in the range $(-15°, 15°)$, corresponding to the
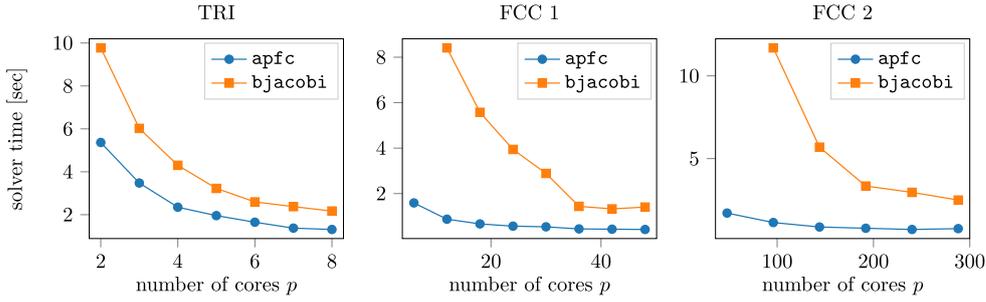
Figure 3. Comparison of `bjacobi` and `apfc` preconditioner for various setups. The large setup fcc 3 could not be solved using `bjacobi`, due to memory limitations and is thus not shown here. The TRI setup is solved with a timestep size $\tau = 2$ and the fcc setups with a timestep size $\tau = 1$.

configuration for the growth of a polycrystal as shown in figure 2 (see, e.g. [18, 19, 35]). The 3D, fcc configurations, are initialized with one grain of size 30π, 60π, or 120π, for the setup fcc 1, fcc 2, or fcc 3, respectively, rotated by $10°$ about the [111] direction with respect to a surrounding, unrotated crystal (see [22]). The discretization bounds are $h_{\mathrm{int}} \approx 2.0$ and $h_{\mathrm{max}} \approx 40.0$ in 2D and $h_{\mathrm{int}} \approx 3.0$ and $h_{\mathrm{max}} \approx 60.0$ in 3D.

In a first comparison, we evaluate the linear-solver performance for the preconditioner **P** defined in equation (12) (`apfc`) and a block Jacobi preconditioner (`bjacobi`) applied to the blocks resulting from an element-wise domain decomposition, with a local sparse direct solver UMFPACK [37]. The domain decomposition is combined with a distributed memory parallelization. In figure 3 the timings for solving the linear system per timestep are summarized for TRI, fcc 1, and fcc 2 setup. Timings for the two preconditioners are evaluated by averaging over 100 timesteps over all amplitudes. Depending on the size of the setup different improvements of the `apfc` preconditioner compared to the `bjacobi` preconditioner can be found, ranging from a factor 10 for small number of cores and thus large local partitions, to 2 for large number of cores and thus small local partitions. In 2D the improvement is nearly constant over all subdomain sizes, but in 3D the local linear system size matters for the `bjacobi` solver significantly. The three setups show a similar behavior: the time to solve one linear system goes down with increasing number of processors, while it stagnates after some threshold in the number of cores. This is due to the fact that the local domain sizes become too small and additionally the load balancing gets more and more complicated. Increasing the number of cores further would increase also the solver time. Then, communication cost imbalance of the local problems would dominate the solution procedure.

While the `bjacobi` preconditioner uses a direct solver on each local subdomain, that results in large memory costs, the `apfc` preconditioner requires the solution of mass matrix and diffusion-like equations, using an iterative solver optimized for the specific type of linear system. For the mass matrix, we use three iterations of a diagonally preconditioned conjugate gradient (CG) method. The diffusion system, on the other hand, is solved using either a diagonally preconditioned CG method with 5 iterations (parameter $= 1$), or an algebraic multigrid method (AMG, parameter $= 2$), based on the so-called Hypre BoomerAMG [38], with one V-cycle and symmetric relaxation. In figure 4(a) the difference in the solver time per timestep is plotted, depending on the timestep parameter τ that is part of the preconditioner definition (12). Increasing the timestep size results in slower convergence of all the linear solvers, but the effect is more pronounced with the `apfc` preconditioner. In the same figure also the difference in the solver time for different solver parameters is shown. For this setup, a
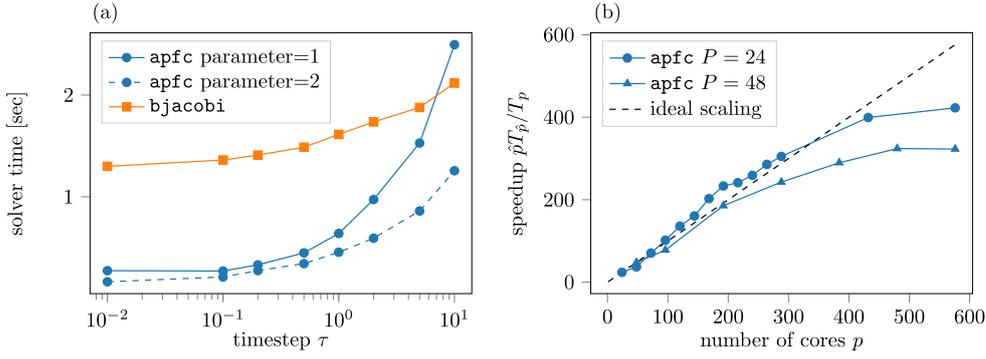
Figure 4. (a) Comparison of the solver performance per timestep of setup fcc 1 for different timestep widths τ. While the block preconditioner (`apfc`) performs better if the timestep is small, the `bjacobi` preconditioner seems more robust for large timesteps. The configuration parameter $= 1$ indicates a CG solver for the diffusion subsystem and parameter $= 2$ an AMG subsolver. (b) Strong parallel scaling for the fcc 3 setup. Computations are run on compute nodes with 2×24 cores each. Those nodes are either fully assigned ($P = 48$) or with only half the cores used ($P = 24$), resulting in different memory throughput and thus different performance. The minimal configuration p_1 for the speedup estimation is one node with 48 or 24 cores, respectively. The overstepping of the ideal solution line might be due to variations in the domain decomposition and thus in the communication pattern for different core counts. For large number of cores, the efficiency drops down to 60%. This may be because of too small local problems.

simple CG iteration outperforms the AMG subsolver. Concerning just the performance of the linear solver the optimal timestep would be as large as possible, but the accuracy of the scheme is just first order in τ so that a balance between accuracy and performance has to be found for a concrete setup.

A third benchmark considers the parallel scalability of the `apfc` preconditioner. Therefore, we run setup fcc 3 with an increasing number of subdomains, that is, an increasing number of compute cores. These benchmarks are run on the JUWELS cluster of the Jülich Supercomputing Centre on regular 2×24 core nodes. Since an allocation of a different number of running processes per node results in different node performance, e.g. due to parallel usage of memory pipelines, we have assigned a fixed number of cores per node for each scaling benchmark. In figure 4(b) the relative speedup for an allocation of 24 and 48 cores per node is plotted with respect to the smallest possible number of cores, i.e. one node with 24 or 48 cores, respectively. Using less cores per node leads to a slightly better speedup that might even exceed the ideal scaling line. Although the parallel efficiency drops down if the local problems get too small, in the intermediate range of up to 300 cores, we see very good speedup and thus parallel efficiency of the `apfc` preconditioner.

Application: growth of polycrystals with dislocations

In this section we illustrate some applications of the numerical approach illustrated above. In order to allow for estimates and comparisons in terms of number of atoms, e.g. to other methods, simulations or real systems, the length scale is here reported in terms of the lattice constant for the considered symmetries as set by the corresponding set of $\mathbf{k}_j$ vectors. They read a_{tri} and a_{fcc} for the triangular and fcc lattice symmetry, respectively. Figure 5 shows the
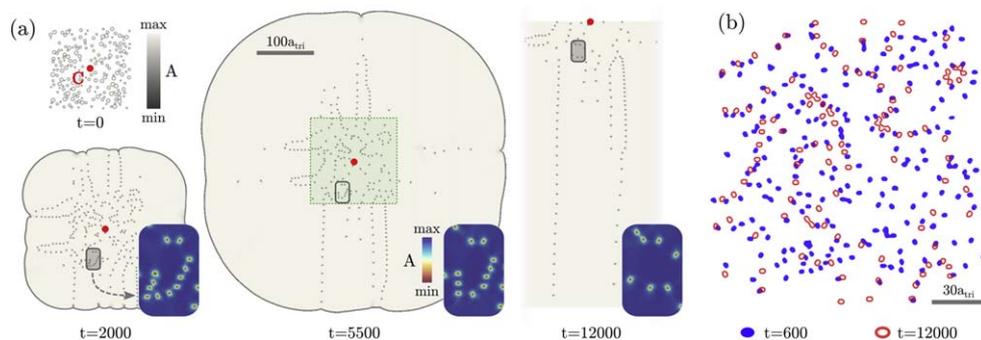
Figure 5. Large-scale simulation of the growth of a polycrystal in 2D with triangular symmetry. The initial configuration consists of 200 crystal seeds with rotation $\theta \in (-15°, 15°)$, randomly distributed around center of the simulation domain (marked by the red point C). (a) The initial configuration ($t = 0$) and three representative steps are shown in terms of the region where $A > 0$, i.e. in term of the solid phase. $t = 12\,000$, where the crystal fills the entire simulation domain, is reported by means of a portion of the crystal highlighting the formation of straight GBs. A is also shown by greyscale map, showing the presence of defects. The length scale is the same for every step. Insets show the arrangement of defects in a small portion of the crystal (gray shaded area, see t = 2000). (b) Comparison between the distribution of defects in the green shaded region of panel (a) at $t = 600$ (blue, filled) and $t = 12\,000$ (red, empty).

growth of 200 crystal seeds in 2D having triangular symmetry. A rotation of crystals $\theta \in (-15°, 15°)$ set as initial condition by means of equation (13) as in figure 2 is considered. We set here a square domain of side length $\sim 10^3 a_{\mathrm{tri}}$. The initial crystal seeds are generated in a square region at the center (see figure 5(a), $t = 0$). As shown by two representative steps during the evolution, $t = 2000$ and $t = 5500$, the growth of these seeds results in a polycrystal with several dislocations at the center while almost straight GBs form between the peripheral grains which are free to grow towards the liquid phase. The formation of such GBs is also highlighted at $t = 12\,000$ by means of a portion of the entire simulation domain. Insets of figure 5(a) illustrate the arrangement of defects on a smaller length scale in a portion of the crystal. The motion and eventual annihilation of dislocations is accounted for by the approach as illustrated in figure 5(b) where the arrangement of defects in the central region of the simulation (green shaded square superposed to the crystal at $t = 5500$) is compared at $t = 600$ (blue, filled) and $t = 12\,000$ (red, empty). The coarsening dynamics results faster in the early stages than at later times as can be noticed by comparing the main features of the dislocation networks at different times in figure 5. Indeed, small grains are present at the beginning leading to the formation of curved GBs formed by a few dislocations, which, in turn, move fast [39, 40]. Later, the grains at the center of the polycrystal are larger and the resulting GBs are more stable having smaller curvatures. However, the coarsening dynamics with motion and annihilation of defects continues as can be notice in the insets of figure 5(a), where significant changes in the arrangement of defects are observed at later times. The straight GBs forming due to the growth of the polycrystal can be considered as long-lasting defects as their curvature is negligible, while the spacing between dislocations depends on the relative tilts of grains. Notice that the growth velocity of the polycrystal is a function of the parameters entering the free energy that control the energy difference between the solid and the liquid phase [41]. Therefore, faster or slower grain growth compared to defect motion can be inspected.
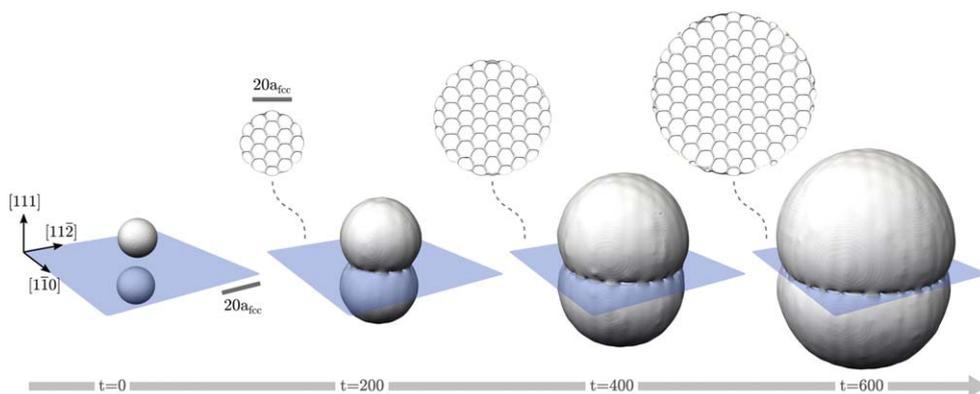
Figure 6. Growth of 2 crystal seeds in 3D with fcc lattice symmetry and rotation $\theta = \pm 5°$ about the [111] directions. Four steps during the evolution are shown by means of region where $A > 0$. Insets illustrate the the defect structure forming at the planar, twist (111) GB by means of the regions at the blue shaded plane where $0 < A < 0.8 \max_\Omega(A)$.

The main features of the dynamics obtained by the standard APFC approach considered here qualitatively correspond to predictions of classical theories and simulations [39, 40]. However, a quantitative description can be achieved by accounting for extensions of the PFC and APFC approach, including a proper description of elastic and plastic relaxation during the motion of defects [23, 42].

A 3D example is shown in figure 6. Two fcc crystals, having a rotation of $\pm 5°$ about the [111] direction are considered. The simulation size is set as in the setup fcc 3, corresponding to a square domain of side length $\sim 80 a_{\text{fcc}}$. They are also aligned along the [111] direction, therefore their growth is expected to form a (111) twisted GB with a typical hexagonal arrangement of defects [22, 43, 44]. Figure 6 reports four stages during the evolution in terms of $A > 0$ along with the dislocation networks at the resulting planar GB illustrated as insets. They correspond to regions where $0 < A < 0.8 \max_\Omega(A)$ at the (111) blue shaded plane. The study of these kind of planar GBs, in particular for what concern the morphology of the emerging dislocation network, are typically accounted for by assuming ideally infinite crystals by means of periodic boundary conditions or even just by 2D approaches. Here, the GB is obtained together with the explicit description of crystal growth which enable more general investigations tackling the simultaneous presence of different GBs with different orientations. A more general case, illustrating the general capability of the approach, is reported in figure 7. Therein 30 crystals having random rotation about the [111] direction are considered. A simulation domain that is double the size of the setup fcc 3, namely corresponding to a square domain of side length $\sim 160 a_{\text{fcc}}$, is considered here. Figure 7(a) shows the morphologies of the seeds and of the resulting, growing polycrystal in terms of $A > 0$ regions. Figure 7(b) shows half of the crystals reported in figure 7(a) by the isosurface $A = 0.8 \max_\Omega(A)$ revealing also the dislocation network forming at the internal GBs. In this case the initial seeds are distributed randomly and are not aligned along a specific direction. Therefore, together with the twist GBs as shown in figure 6, other orientations for the boundaries between grains are present thus leading to different morphologies for the dislocation networks. This can be observed in more detail in figures 7(b)–(d). Hexagonal patterns mostly lying on (111) planes, can be recognized along with elongated defects typical of pure
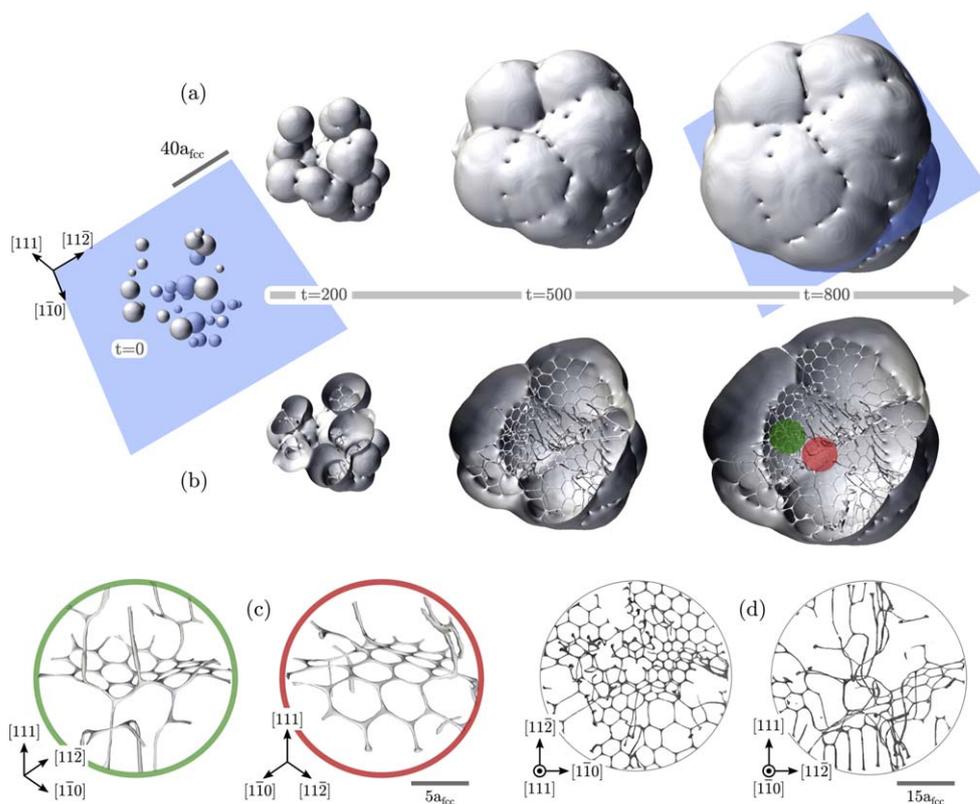
Figure 7. Growth of 30 crystal seeds in 3D with fcc lattice symmetry and random rotation $\theta \in (-15°, 15°)$ about the [111] directions. Initial seeds are randomly distributed at the center of the simulation domain. (a) Regions where $A > 0$. (b) Isosurface $A = 0.8 \max_{\Omega}(A)$ in half of the domain, cut along the (111) plane passing through its center (blue shaded plane illustrated at $t = 0$ and $t = 800$). (c) Magnification of two small spherical regions inside the polycrystal showing defect arrangements on a small length scale (see corresponding colors in panel (b), $t = 800$). (d) View of hemispheres, fully contained in the growing polycrystal, showing the arrangement of defects from two different perspectives: perpendicular to the [111] direction (left) and to the [$1\bar{1}0$] direction (right).

tilt GBs. Moreover, similar patterns having different spacing between dislocations are present due to different relative rotations between grains.

The simulations reported in this section are unprecedented in terms of sizes for what concern APFC (and PFC) approaches. Just by focusing on the largest system of figure 7, we were able to simulate a crystalline system with fcc lattice symmetry including $\sim 8 \times 10^6$ atoms. For specific materials exhibiting such a lattice symmetry as, e.g. Cu, Ag/Au, Pb this would mean a volume of $\sim(45 \text{ nm})^3$, $\sim(50 \text{ nm})^3$, $\sim(65 \text{ nm})^3$ respectively [45], which lie in the typical size range of nanoparticles and nanostructures. The simulation in figure 7 is done on 720 cores. Increasing the number of cores will also allow to consider even larger samples and thus realize the envisioned multiscale approach, ranging from atomistic details to micrometer sizes.

Conclusions

We illustrated a numerical approach to solve the equations of the APFC model efficiently. A specific discretization scheme combined with a nonlinear solver has been proved to allow for unprecedented size and performances for both 2D and 3D APFC simulations. In particular, we have constructed a schur-complement preconditioner for iterative Krylov subspace methods that outperforms a block Jacobi solver for the linear systems arising from the discretization with adaptive finite elements. The schur-complement solver requires much less memory, converges fast in terms of wall-clock time, and scales well in parallel setups. On the other hand, it is more sensitive to an increase in timestep size, compared to the `bjacobi` solver.

Moreover, an optimized criterion for mesh adaptivity has been proposed and used, exploiting features of the complex amplitude functions η_j and derived physical quantities such as the local rotation field. This can be applied in both 2D and 3D and allows for a significant reduction of the overall number of DOFs.

Some applications involving the growth of polycrystals in 2D and 3D as well as the simultaneous description of dislocations forming at GBs have been shown. They set new limits for APFC and then, more in general, PFC approaches, enabling the investigation of large systems matching the size of real nanostructures. This is a crucial step to address large, mesoscale problems still retaining details of the atomic length scale. Future work will be devoted to explicitly include further details, compatible with the APFC model, in the numerical framework presented here as, for instance, an improved description of interface-energy anisotropy [25, 46], binary systems [26], an improved description of the dynamics [23, 42], the coupling with magnetic fields [47], as well as improvements on scaling properties of the numerical approach to enable larger systems addressing grain growth in 3D.

Acknowledgments

ORCID iDs

Simon Praetorius ⬤ https://orcid.org/0000-0002-1372-4708
Marco Salvalaglio ⬤ https://orcid.org/0000-0002-4217-0951
Axel Voigt ⬤ https://orcid.org/0000-0003-2564-3697

References

[1] Rollett A, Rohrer G and Suter R 2015 *MRS Bull.* **40** 951
[2] Elder K R, Katakowski M, Haataja M and Grant M 2002 *Phys. Rev. Lett.* **88** 245701
[3] Elder K R and Grant M 2004 *Phys. Rev.* E **70** 051605
[4] Emmerich H, Löwen H, Wittkowski R, Gruhn T, Tóth G I, Tegze G and Gránásy L 2012 *Adv. Phys.* **61** 665
[5] Greenwood M, Provatas N and Rottler J 2010 *Phys. Rev. Lett.* **105** 045702
[6] Huang Z-F, Elder K R and Provatas N 2010 *Phys. Rev.* E **82** 021605
[7] Kocher G and Provatas N 2015 *Phys. Rev. Lett.* **114** 155501

 [8] Rottler J, Greenwood M and Ziebarth B 2012 *J. Phys.: Condens. Matter* **24** 135002

 [9] Berry J, Provatas N, Rottler J and Sinclair C W 2014 *Phys. Rev.* B **89** 214117

[10] Achim C V, Schmiedeberg M and Löwen H 2014 *Phys. Rev. Lett.* **112** 255501

[11] Hirvonen P, Ervasti M M, Fan Z, Jalalvand M, Seymour M, Vaez Allaei S M, Provatas N, Harju A, Elder K R and Ala-Nissila T 2016 *Phys. Rev.* B **94** 035414

[12] Wang N, Bevan K H and Provatas N 2016 *Phys. Rev. Lett.* **117** 155901

[13] Yamanaka A, McReynolds K and Voorhees P W 2017 *Acta Mater.* **133** 160

[14] Goldenfeld N, Athreya B P and Dantzig J A 2005 *Phys. Rev.* E **72** 020601

[15] Athreya B P, Goldenfeld N and Dantzig J A 2006 *Phys. Rev.* E **74** 011601

[16] Goldenfeld N, Athreya B P and Dantzig J A 2006 *J. Stat. Phys.* **125** 1015

[17] Yeon D-H, Huang Z-F, Elder K and Thornton K 2010 *Phil. Mag.* **90** 237

[18] Spatschek R and Karma A 2010 *Phys. Rev.* B **81** 214201

[19] Salvalaglio M, Voigt A and Elder K R 2019 *njp Comput. Mater.* **5** 48

[20] Elder K R, Huang Z-F and Provatas N 2010 *Phys. Rev.* E **81** 011602

[21] Salvalaglio M, Backofen R, Voigt A and Elder K R 2017 *Phys. Rev.* E **96** 023301

[22] Salvalaglio M, Backofen R, Elder K R and Voigt A 2018 *Phys. Rev. Mater.* **2** 053804

[23] Heinonen V, Achim C V, Kosterlitz J M, Ying S-C, Lowengrub J and Ala-Nissila T 2016 *Phys. Rev. Lett.* **116** 024303

[24] Skaugen A, Angheluta L and Viñals J 2018 *Phys. Rev.* B **97** 054113

[25] Ofori-Opoku N, Warren J A and Voorhees P W 2018 *Phys. Rev. Mater.* **2** 083404

[26] Elder K R, Huang Z-F and Provatas N 2010 *Phys. Rev.* E **81** 011602

[27] Vey S and Voigt A 2007 *Comput. Vis. Sci.* **10** 57

[28] Witkowski T, Ling S, Praetorius S and Voigt A 2015 *Adv. Comput. Math.* **41** 1145

[29] Saad Y 2003 *Iterative Methods for Sparse Linear Systems* 2nd edn (Philadelphia, PA: Society for Industrial and Applied Mathematics)

[30] Boyanova P, Do-Quang M and Neytcheva M 2012 *Large-Scale Scientific Computing* (*Lecture Notes in Computer Science* vol 7116) (Berlin: Springer) pp 549–57

[31] Axelsson O and Neytcheva M 2013 *Numer. Linear Algebr.* **20** 723

[32] Saad Y 1993 *SIAM J. Sci. Comput.* **14** 461

[33] Praetorius S and Voigt A 2015 *SIAM J. Sci. Comput.* **37** B425

[34] Praetorius S 2015 *Phd Thesis* Technische Universität Dresden, Germany (http://nbn-resolving.de/urn:nbn:de:bsz:14-qucosa-195532)

[35] Athreya B P, Goldenfeld N, Dantzig J A, Greenwood M and Provatas N 2007 *Phys. Rev.* E **76** 056706

[36] Berčič M and Kugler G 2018 *Phys. Rev.* E **98** 033303

[37] Davis T A 2004 *ACM Transactions on Mathematical Software* **30** 165–95

[38] Henson V E and Yang U M 2002 *Appl. Numer. Math.* **41** 155–77

[39] Mullins W W 1956 *J. Appl. Phys.* **27** 900

[40] Doherty R, Hughes D, Humphreys F, Jonas J, Jensen D, Kassner M, King W, McNelley T, McQueen H and Rollett A 1997 *Mater. Sci. Eng.* A **238** 219

[41] Elder K R, Provatas N, Berry J, Stefanovic P and Grant M 2007 *Phys. Rev.* B **75** 1

[42] Skaugen A, Angheluta L and Viñals J 2018 *Phys. Rev. Lett.* **121** 255501

[43] Scott R F and Goodhew P J 1981 *Phil. Mag.* A **44** 373

[44] De Hosson J T M and Vitek V 1990 *Phil. Mag.* A **61** 305

[45] Davey W P 1925 *Phys. Rev.* **25** 753

[46] Salvalaglio M, Backofen R, Bergamaschini R, Montalenti F and Voigt A 2015 *Cryst. Growth Des.* **15** 2787

[47] Backofen R, Elder K R and Voigt A 2019 *Phys. Rev. Lett.* **122** 126103