

An r -Adaptive Finite Element Method for the Solution of the Two-Dimensional Phase-Field Equations

G. Beckett¹, J. A. Mackenzie^{1,*} and M. L. Robertson¹

¹ *Department of Mathematics, University of Strathclyde, Livingstone Tower, 26 Richmond Street, Glasgow G1 1XH, Scotland.*

Received 17 October 2005; Accepted (in revised version) 6 February, 2006

Abstract. An adaptive moving mesh method is developed for the numerical solution of two-dimensional phase change problems modelled by the phase-field equations. The numerical algorithm is relatively simple and is shown to be more efficient than fixed grid methods. The phase-field equations are discretised by a Galerkin finite element method. An adaptivity criterion is used that ensures that the mesh spacing at the phase front scales with the diffuse interface thickness.

Key words: Phase change; phase-field; equidistribution; moving meshes; adaptive method.

1 Introduction

There has been much recent interest in the modelling of solidification processes. The main challenge is to incorporate events on the smallest microstructural scales to the larger macroscopic scales. Classical Stefan models do not take account of important physical properties such as undercooling and surface tension. These effects are normally incorporated within modified Stefan models (see Section 2 below). The numerical simulation of the modified model requires the estimation of the curvature of the interface between the solid and liquid phases. This is often a difficult task, especially in three dimensions or when phase fronts merge.

Modified Stefan problems have become more amenable to numerical solution by the introduction of phase-field models. Front-tracking is avoided by introducing an auxiliary continuous order parameter p that interpolates between the solid and liquid phases, attaining two different constant values in each phase (e.g, $p = \pm 1$), with a rapid transition

*Correspondence to: J. A. Mackenzie, Department of Mathematics, University of Strathclyde, Livingstone Tower, 26 Richmond Street, Glasgow G1 1XH, Scotland. Email: jam@maths.strath.ac.uk

region at the solidification front. The level set $p = 0$ is identified with the front and p is assumed to evolve in such a way that it minimises a free energy functional consistent with thermodynamics.

Most numerical methods used to solve the phase-field equations have utilised stationary uniform meshes [9,20,29,31]. However, it is well known that it is important that the diffuse interface is well resolved if the correct dynamics are to be reproduced. As the phase interface moves in time it is clear that an efficient numerical method must use some form of mesh adaptivity. Within a finite element context this is usually achieved using the h -method of adaptation, where the mesh is locally refined or coarsened by adding or deleting points. This strategy has been used successfully in [6, 25, 26]. A less popular approach is to use the so-called r -adaptive method where mesh points are moved throughout the domain while the connectivity of the mesh is kept fixed. The main reason for the lack of popularity of this approach is the difficulty involved in controlling the geometric quality of the mesh elements. However, the development of a robust r -adaptive method is attractive in that it intuitively should be able to accurately resolve and follow important solution features. The coding involved in an r -adaptive method is also simpler than an h -method, which requires a considerably more complicated data structure.

In this paper we will concentrate on an r -adaptive method where movement of the mesh is based on a variational formulation used by Huang and Russell [18], Cao *et al.* [10], and Huang [17]. The basic idea is to move the mesh so that it attempts to minimise a weighted quadratic functional where the weights are based on some local adaptivity criterion. Such an approach has been successfully used to solve a regularised formulation of two-dimensional classical Stefan problems [4] and convective heat transfer problems involving a change of phase [28]. Moving mesh methods have also been used to solve the phase-field equations in one dimension [22, 23] and in two dimensions [27]. In particular, in [22] we have been able to suggest an appropriate adaptivity criterion based on an asymptotic expansion of the interface region of a planar travelling wave solution. The resulting mesh has been shown to automatically scale with the diffuse interface thickness.

The main aim of this paper is to investigate the use of a moving mesh approach to solve the two-dimensional phase-field equations. The layout of this paper is as follows: in the next section we present the sharp and diffuse interface models for heat conduction with a change of phase. In Section 3 we discuss how the moving mesh is generated along with specific adaptivity criteria for the phase-field equations. In Section 4 we describe a semi-implicit moving finite element discretisation of the phase-field model. Finally, we apply the moving mesh method to a number of test cases in Section 5.

2 Sharp and diffuse interface models

Let $\Omega \in \mathbb{R}^2$ be a bounded domain with a Lipschitz continuous boundary $\partial\Omega$. For each $t \in \mathbb{R}^+$ we will assume we have a decomposition of Ω into subdomains $\Omega^+(t)$ and $\Omega^-(t)$ so that $\Omega = \Omega^+(t) \cup \Omega^-(t) \cup \Gamma(t)$, where the interface $\Gamma(t) = \partial\Omega^+(t) \cup \partial\Omega^-(t)$ is smooth.

Let $T_f > 0$ and set $Q := \Omega \times (0, T_f)$. We are interested in the class of sharp interface problems that takes the form [13]

$$\rho c T_t = k \nabla^2 T, \quad \mathbf{x} \in \Omega^+(t) \cup \Omega^-(t), \quad (2.1)$$

$$\rho l v = k [\nabla T \cdot \mathbf{n}]_+^-, \quad \mathbf{x} \in \Gamma(t), \quad (2.2)$$

$$T - T_m = -\frac{\sigma}{[s]_m} \kappa - \frac{\alpha \sigma}{[s]_m} v, \quad \mathbf{x} \in \Gamma(t). \quad (2.3)$$

Here T_m is the equilibrium melting temperature, l is the latent heat per unit mass, k is the thermal conductivity, σ is the surface tension, ρ is the density, c is the specific heat, $[s]_m$ is the entropy difference per unit volume ($[s]_m = 4$ in the normalisation used here), v is the normal velocity of the interface, κ is the sum of the principal curvatures, $[\nabla T \cdot \mathbf{n}]_+^-$ is the jump in the normal component of the temperature (from solid to liquid), and α is a kinetic undercooling coefficient.

If we define a dimensionless temperature $\theta = c(T - T_m)/l$, a diffusion parameter $D = k/\rho c$, and a capillary length $d_0 = \sigma c/(l[s]_m)$ then we can write (2.1)-(2.3) in the form

$$\theta_t = D \nabla^2 \theta, \quad \mathbf{x} \in \Omega^+(t) \cup \Omega^-(t), \quad (2.4)$$

$$v = D [\nabla \theta \cdot \mathbf{n}]_+^-, \quad \mathbf{x} \in \Gamma(t), \quad (2.5)$$

$$\theta = -d_0 \kappa - \alpha d_0 v, \quad \mathbf{x} \in \Gamma(t). \quad (2.6)$$

The classical Stefan problem incorporates heat transfer by diffusion (2.4) and the latent heat due to fusion (2.5). In addition, the interfacial temperature is equal to the equilibrium melting temperature, i.e. $\theta = 0$ replaces (2.6) which is equivalent to setting $d_0 = 0$. The classical Stefan model neglects the effects of surface tension and surface kinetics but these are modelled by finite values of σ and α . However, (2.6) then becomes a difficult condition to impose numerically due to the presence of the curvature term.

Using a scaling introduced in [7] we consider the phase-field model

$$\theta_t + \frac{1}{2} p_t = D \nabla^2 \theta, \quad (2.7)$$

$$\alpha \varepsilon^2 p_t = \varepsilon^2 \nabla^2 p + \frac{1}{2} (p - p^3) + \frac{\varepsilon}{3d_0} \theta, \quad (2.8)$$

where ε is a measure of the diffuse interface thickness. In [8] Caginalp showed that if $\varepsilon \rightarrow 0$ while all other parameters are held fixed, then the asymptotic solutions of (2.7), (2.8) are, to leading order, solutions of the modified Stefan problem (2.1)-(2.3).

The boundary conditions for the phase-field equations are the same as the sharp interface model for θ , with compatible conditions for p . For example, if Dirichlet conditions are imposed on $\theta = \theta_{\partial\pm}$, where \pm denotes the liquid and solid boundaries respectively, then the corresponding values of p are the roots of

$$f(p, \theta) = \frac{1}{2} (p_{\pm} - p_{\pm}^3) + \frac{\varepsilon}{3d_0} \theta_{\partial\pm} = 0. \quad (2.9)$$

Other phase-field models have been proposed that allow a simpler implementation of boundary conditions for p that do not depend on the temperature on the boundary, (see for example Wang *et al.* [30]).

In this paper we will concentrate on the phase-field model (2.7)-(2.8). The adaptive moving mesh algorithm can easily be modified and applied to the solution of other phase-field models such as the model proposed by Karma and Rappel [20], which approximates the sharp interface model to $\mathcal{O}(\varepsilon^2)$, rather than the $\mathcal{O}(\varepsilon)$ rate of approximation of the Caginalp model.

3 Adaptive mesh generation

3.1 Variational formulation

To generate an adaptive mesh it is useful to regard the physical domain Ω_p as the image of a computational (logical) domain Ω_c under the invertible maps

$$x = x(\xi, \eta), \quad y = y(\xi, \eta) \quad \text{and} \quad \xi = \xi(x, y), \quad \eta = \eta(x, y), \quad (3.1)$$

where $\mathbf{x} = (x, y)$ and $\boldsymbol{\xi} = (\xi, \eta)$ are the physical and computational coordinates, respectively. A mesh covering Ω_p is obtained by applying the mapping given in (3.1) to a partitioning of Ω_c .

A popular way to choose the coordinate transformation for steady problems is to require that it minimises a functional of the form

$$F(\xi, \eta) = \frac{1}{2} \int_{\Omega_p} (\nabla \xi^T G^{-1} \nabla \xi + \nabla \eta^T G^{-1} \nabla \eta) dx dy, \quad (3.2)$$

where $\nabla = (\partial/\partial x, \partial/\partial y)$ and $G(x, y)$ is a 2×2 symmetric positive definite (SPD) matrix, often referred to as a monitor matrix. The idea in adaptive mesh generation is to choose G to concentrate mesh points in Ω_p where the solution of the PDE is difficult to solve. One example of a monitor matrix is $G = M/\sqrt{\det(M)}$, where M is a SPD matrix. This gives rise to a method based on Harmonic mappings [14]. Alternatively, if $G = \lambda I$, where $\lambda > 0$ is a scalar adaptivity function, then we arrive at a method based on the minimisation of Winslow's functional [32]. An analysis of the effect of G on the clustering and orientation of the mesh is presented in Cao *et al* [11].

For time dependent problems Huang and Russell [18, 19] propose that the coordinate transformation from the physical domain to the computational domain should satisfy the modified gradient flow equation

$$\frac{\partial \boldsymbol{\xi}}{\partial t} = \frac{Q}{\tau} \nabla \cdot (G^{-1} \nabla \boldsymbol{\xi}), \quad (3.3)$$

where $\tau > 0$ is a temporal relaxation parameter which determines the rate at which the computational mesh attempts to minimise the functional, and Q is a differential operator with a positive spectrum which is chosen to make the mesh generation more robust.

In practice, we are interested in the inverse map $\mathbf{x}(\boldsymbol{\xi})$ which defines the mesh in physical space. By changing the independent and dependent variables the gradient flow equations can be written as the following moving mesh PDEs (MMPDEs)

$$\frac{\partial \mathbf{x}}{\partial t} = \frac{Q}{\tau} \left[\sum_{ij} a_{ij} \frac{\partial^2 \mathbf{x}}{\partial \xi^i \partial \xi^j} + \sum_i b_i \frac{\partial \mathbf{x}}{\partial \xi^i} \right], \quad (3.4)$$

where

$$a_{ij} = \mathbf{a}^i \cdot G^{-1} \mathbf{a}^j, \quad b_i = - \sum_j \mathbf{a}^i \frac{\partial G^{-1}}{\partial \xi^j} \mathbf{a}^j, \quad (3.5)$$

and the contravariant base vectors $\mathbf{a}^i = \nabla \xi^i$, $i = 1, 2$.

The aim of introducing the function Q is to make the resulting MMPDEs easier to integrate forward in time. Huang [17] suggests setting

$$Q = \frac{1}{\sqrt{\sum_i (a_{ii}^2 + b_i^2)}}.$$

This choice of Q is designed so that the coefficients of the spatial derivatives in (3.4), especially the second-order derivatives, change evenly over the domain. For this reason Huang refers to this scaling as ‘‘spatial balancing’’. Although it is difficult to theoretically understand the precise effect of Q , our experience is that this choice of Q results in a more robust algorithm than simply setting $Q = 1$. Furthermore, the resulting MMPDE is invariant under the scaling transformations $\mathbf{x} \rightarrow \delta \mathbf{x}$ and $G \rightarrow \delta G$ for all $\delta > 0$. These properties indicate the robustness of the MMPDEs to changes in the size of the physical domain and to scalings of the monitor matrix.

3.2 Discretisation and boundary conditions

The temporal discretisation of (3.4) is achieved using a semi-implicit method where

$$\tau(\mathbf{x}^{n+1} - \mathbf{x}^n) = \Delta t Q^n (a_{11}^n \mathbf{x}_{\xi\xi}^{n+1} + 2a_{12}^n \mathbf{x}_{\xi\eta}^{n+1} + a_{22}^n \mathbf{x}_{\eta\eta}^{n+1} + b_1^n \mathbf{x}_{\xi}^{n+1} + b_2^n \mathbf{x}_{\eta}^{n+1}). \quad (3.6)$$

Freezing the coefficients a_{ij} , b_i , and Q has two very important effects. The first is that it linearises the equations that define the mesh \mathbf{x}^{n+1} . Moreover, the system of PDEs represented by (3.6) for $\mathbf{x}^{n+1}(\xi, \eta)$ and $\mathbf{y}^{n+1}(\xi, \eta)$ decouples into two scalar equations which have the same spatial derivatives. For simplicity, the spatial discretisation of (3.6) is performed using second-order central finite differences on a $N \times N$ uniform partition of $\Omega_c = (0, 1) \times (0, 1)$. Since the spatial derivatives are identical the same coefficient matrix needs to be inverted to find the x and y coordinates. We therefore first solve for the x coordinates using an ILU-preconditioned BICGstab method until the maximum norm of the difference in iterates is less than ε/N . The same preconditioner is then used to solve for the y coordinates. In general, we have found that convergence of the iterative solver

is achieved in around 3 – 4 iterations per time step. Additional details about the spatial discretisation of the MMPDEs and the performance of the ILU-preconditioned BICGstab routine can be found in [3].

Dirichlet boundary conditions for the above system are obtained by solving a one-dimensional MMPDE. If $\partial_p \in \partial\Omega_p$ and $\partial_c \in \partial\Omega_c$ denote the physical and computational boundary segments with arc-lengths l and l_c respectively, then the mesh on ∂_p is the solution of

$$\tau \frac{\partial s}{\partial t} = \frac{1}{\sqrt{(M^2 + (M_\zeta)^2}} \frac{\partial}{\partial \zeta} \left(M \frac{\partial s}{\partial \zeta} \right), \quad \zeta \in (0, l_c), \quad (3.7)$$

with $s(0) = 0$ and $s(l_c) = l$. Here M is the one-dimensional projection of the two-dimensional monitor function along the boundary. That is, if \mathbf{t} is a unit tangent vector along the boundary then $M(s, t) = \mathbf{t}^T G \mathbf{t}$.

3.3 Choice of G

The choice of an appropriate monitor matrix is crucial to the success of the moving mesh method. Ideally, for the phase-field equations we would like the mesh to be clustered within the interface region to correctly capture the effects of surface tension, while at the same time providing sufficient resolution away from the interface to accurately model the diffusion of heat. For robustness we would also like to choose a monitor matrix which leads to a mesh that uniformly resolves the interface independently of its thickness.

Let $r(\mathbf{x})$ denote a signed normal distance from the point \mathbf{x} to the interface Γ (i.e $r(\mathbf{x})$ is the distance to the interface if it is in the liquid region and minus the distance if the point is in the solid region). Based on a local asymptotic expansion it can be shown that close to the interface [8]

$$p(\mathbf{x}) \approx \tanh \left(\frac{r(\mathbf{x})}{2\varepsilon} \right). \quad (3.8)$$

Therefore, one idea would be to choose a monitor function that is tailored towards this profile. For the one-dimensional phase-field equations Mackenzie and Robertson [22] proposed an algorithm based on equidistribution of the monitor function

$$M(x) = 1 + \frac{1}{\gamma\beta} \operatorname{sech} \left(\frac{r(x)}{2\varepsilon} \right) \approx 1 + \frac{1}{\gamma\beta} |p_x|^{\frac{1}{2}}, \quad x \in (x_L, x_R), \quad (3.9)$$

where

$$\beta = \frac{1}{x_R - x_L} \int_{x_L}^{x_R} \operatorname{sech} \left(\frac{r}{2\varepsilon} \right) dx, \quad \gamma > 0. \quad (3.10)$$

If the mesh has N grid points then with this choice of monitor function a constant proportion of the mesh points are placed in the interfacial region, namely $N/(1 + \gamma)$, and this proportion is independent of ε . For example, if $\gamma = 1$ then half of the mesh points are placed within the interface region while the other half are uniformly distributed throughout the rest of the domain. Furthermore, it has been shown by Beckett *et al* [2] that the

error in the piecewise linear interpolant of (3.8) converges at the optimal rate of $O(N^{-2})$, and that this rate is independent of ε . It should be noted that equidistribution of the popular solution arc-length monitor function $M = \sqrt{1 + (p_x)^2}$ leads to a mesh on which the linear interpolant of (3.8) converges suboptimally at $O(N^{-1})$ when $\varepsilon \ll N^{-1}$.

The simplest way of extending this idea for two-dimensional problems is to follow the Winslow approach and choose a monitor matrix of the form

$$G(\mathbf{x}) = \left(1 + \frac{1}{\gamma\beta} \operatorname{sech} \left(\frac{r(\mathbf{x})}{2\varepsilon} \right) \right) I. \tag{3.11}$$

The calculation of r is achieved using the same approach used by Beckett *et al* [4]. At time $t = t^n$ a piecewise linear approximation of the phase field is obtained from a finite element discretisation which is described in the next section. A piecewise linear representation of a phase interface is then obtained from the ordered list of points $\{\mathbf{x}_i^c\}_{i=1}^{N_c}$ such that $P(\mathbf{x}_i^c) = 0$. These points are obtained from a plotting routine used to display the numerical results. This routine can also detect if more than one phase front is present and hence changes in the topology of the phase interface can easily be dealt with. A smooth representation of an interface I^n is then obtained by an arc-length parameterised spline passing through the points $\{\mathbf{x}_i^c\}_{i=1}^{N_c}$. This curve is then partitioned by a set of points $\{\mathbf{x}_i^s\}_{i=1}^{N_s}$ which are uniformly distributed along I^n with $\mathbf{x}_1^s = \mathbf{x}_1^c$ and $\mathbf{x}_{N_s}^s = \mathbf{x}_{N_c}^c$. Finally, we set $r(\mathbf{x}) = \pm \min_{1 \leq i \leq N_s} |\mathbf{x}_i^s - \mathbf{x}|$, where \pm denotes points within the liquid and solid regions, respectively. In all the calculations performed in Section 5 we have set $N_s = 100$.

4 A moving finite element discretisation

We will assume that $[0, T_f]$ is partitioned by uniform time intervals $\Delta t = T_f/Nt$ such that

$$0 = t^0 < t^1 < \dots < t^{Nt-1} < t^{Nt} = T_f.$$

Using the procedure described in the previous section we will assume that at time $t = t^{n+1}$ we have a regular triangular mesh \mathcal{S}^{n+1} that has the same connectivity as the mesh \mathcal{S}^n at the previous time step. Therefore, each element of \mathcal{S}^{n+1} corresponds to a unique element of \mathcal{S}^n .

We consider approximations of the form

$$P(x, y, t) = \sum_j P_j(t) \phi_j(x(t), y(t)), \quad \Theta(x, y, t) = \sum_j \Theta_j(t) \phi_j(x(t), y(t)),$$

where $\phi_j(x(t), y(t))$ is the usual piecewise linear basis function associated with the node $\mathbf{x}_j = (x_j(t), y_j(t))^T$. As the mesh will be moving the temporal derivatives take the form

$$P_t = \sum_j \{\dot{P}_j \phi_j - P_j(\dot{\mathbf{x}}_j \cdot \nabla \phi_j)\}, \quad \text{and} \quad \Theta_t = \sum_j \{\dot{\Theta}_j \phi_j - \Theta_j(\dot{\mathbf{x}}_j \cdot \nabla \phi_j)\}, \tag{4.1}$$

where \cdot denotes differentiation in time along the trajectory of a mesh point. Let \mathcal{J} denote the set of indices of the mesh points and \mathcal{J}_0 the subset of \mathcal{J} excluding those corresponding to Dirichlet boundary conditions. Our finite element approximation satisfies the weak formulation of the phase-field equations such that $\forall i \in \mathcal{J}_0$ we have

$$(\Theta_t + \frac{1}{2}P_t, \phi_i) - (D\nabla^2\Theta, \phi_i) = 0, \quad (4.2)$$

$$(\alpha\varepsilon^2 P_t, \phi_i) - (\varepsilon^2 \nabla^2 P, \phi_i) - (\frac{1}{2}(P - P^3), \phi_i) - (\frac{\varepsilon}{3d_0}\Theta, \phi_i) = 0, \quad (4.3)$$

where (\cdot, \cdot) denotes the L_2 inner product over Ω . If we define the vectors of unknowns

$$\Theta = (\Theta_1, \Theta_2, \dots, \Theta_{(N+1)^2})^T, \quad \mathbf{P} = (P_1, P_2, \dots, P_{(N+1)^2})^T,$$

then using (4.1) and integration by parts we can write equations (4.2) and (4.3) as the block system

$$\begin{bmatrix} M & \frac{1}{2}M \\ 0 & \alpha\varepsilon^2 M \end{bmatrix} \begin{bmatrix} \dot{\Theta} \\ \dot{\mathbf{P}} \end{bmatrix} + \begin{bmatrix} B + DK & \frac{1}{2}B \\ 0 & \alpha\varepsilon^2 B + \varepsilon^2 K \end{bmatrix} \begin{bmatrix} \Theta \\ \mathbf{P} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{f}(\Theta, \mathbf{P}) \end{bmatrix} = \mathbf{0}, \quad (4.4)$$

where

$$M_{ij} = (\phi_j, \phi_i), \quad B_{ij} = -(\dot{\mathbf{x}}_j \cdot \nabla \phi_j, \phi_i), \quad \text{and} \quad K_{ij} = (\nabla \phi_j, \nabla \phi_i)$$

are the usual mass matrix, a convection-like matrix due to the mesh movement, and the stiffness matrix. Using a product approximation of the nonlinear terms in the phase equation we have

$$\mathbf{f}(\Theta, \mathbf{P}) = M[-\frac{\varepsilon}{3d_0}\Theta - \frac{1}{2}(\mathbf{P} - \mathbf{P}^3)]. \quad (4.5)$$

To perform the temporal integration of (4.4) we treat all diffusion and source terms implicitly and explicitly treat the convective-like terms arising from the mesh movement. Thus we get the system

$$\begin{bmatrix} M^{n+1} + \Delta t DK^{n+1} & \frac{1}{2}M^{n+1} \\ 0 & \varepsilon^2(\alpha M^{n+1} + \Delta t K^{n+1}) \end{bmatrix} \begin{bmatrix} \Theta^{n+1} \\ \mathbf{P}^{n+1} \end{bmatrix} + \begin{bmatrix} S & \frac{1}{2}S \\ 0 & \alpha\varepsilon^2 S \end{bmatrix} \begin{bmatrix} \Theta^n \\ \mathbf{P}^n \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \Delta t \mathbf{f}^{n+1} \end{bmatrix} = \mathbf{0},$$

where $S = \Delta t B^n - M^{n+1}$. To solve this nonlinear system we consider the Newton iteration

$$\begin{bmatrix} J_{11} & J_{12} \\ J_{21} & J_{22} \end{bmatrix} \begin{bmatrix} \Theta^{(n+1,s+1)} \\ \mathbf{P}^{(n+1,s+1)} \end{bmatrix} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix}, \quad (4.6)$$

where $J_{11} = M^{n+1} + \Delta t DK^{n+1}$, $J_{12} = \frac{1}{2}M^{n+1}$, $J_{21} = \Delta t \mathbf{f}_{\Theta}$, $J_{22} = \varepsilon^2(\alpha M^{n+1} + \Delta t K^{n+1}) + \Delta t \mathbf{f}_{\mathbf{P}}$, and \mathbf{b}_1 and \mathbf{b}_2 are vectors which are independent of $\Theta^{(n+1,s+1)}$ and $\mathbf{P}^{(n+1,s+1)}$. Each Newton iteration therefore involves the solution of a (2×2) block linear system. From the first block of equations (4.6) we find that

$$\mathbf{P}^{(n+1,s+1)} = J_{12}^{-1}(\mathbf{b}_1 - J_{11}\Theta^{(n+1,s+1)}). \quad (4.7)$$

Substituting (4.7) into the second block of equations we have

$$(J_{21} - J_{22}J_{12}^{-1}J_{11})\Theta^{(n+1,s+1)} = (\mathbf{b}_2 - J_{22}J_{12}^{-1}\mathbf{b}_1). \quad (4.8)$$

To efficiently solve this linear system we replace the mass matrix by a diagonal matrix by mass lumping so that J_{12} is diagonal and hence is easily inverted. The linear system (4.8) is solved iteratively using an ILU-preconditioned BICGstab routine to a tolerance of 10^{-15} . Thereafter, we find $\mathbf{P}^{(n+1,s+1)}$ directly from (4.7) which is simply a matrix vector multiplication. For efficiency only one Newton iteration is performed as the computed results were found to be almost identical to those obtained by solving the nonlinear system to convergence. Note that if we had used a fully implicit treatment of the mesh convection term then the matrix J_{12} would have been modified by the introduction of the non-symmetric matrix B . The inverse J_{12}^{-1} would then be full and we would not have an efficient sequential solution procedure. A similar semi-implicit method was used successfully in the moving mesh methods appearing in [21] and [4].

If we assume that at time $t = t^n$ we have an approximation of the physical solution P^n and Θ^n , and a mesh \mathbf{x}^n , then we integrate forward in time using the algorithm below.

1. Compute the monitor matrix $G^n(\mathbf{x}) = G(\mathbf{x}^n, t^n)$ using P^n and \mathbf{x}^n .
2. Integrate the discretised MMPDE to get the mesh \mathbf{x}^{n+1} .
3. Integrate the physical PDEs using the meshes \mathbf{x}^n and \mathbf{x}^{n+1} to generate M , K , and B .
4. Goto the next time step.

In theory this algorithm could be modified to return to step 1 after P^{n+1} has been obtained in step 3 and the coefficients a_{ij} , b_i in (3.5) have been evaluated at t^{n+1} and \mathbf{x}^{n+1} . A new estimate of the mesh \mathbf{x}^{n+1} could then be found by resolving the MMPDEs. This process could be repeated a fixed number of times or until some measure of grid convergence is reached. A similar approach was investigated in [2] and [3] where it was found that larger time steps could be used without affecting accuracy and that the gain in efficiency outweighed the additional cost of performing the additional steps. As an initial step to solving the phase-field equations in two dimensions, in this work we have used a fixed time step and the one pass algorithm above to obtain the results of the next section. Higher order temporal integration and adaptive time-stepping will be considered within a multi-pass formulation in future work.

5 Numerical results

5.1 Planar solidification

A popular test case is to reproduce the travelling wave solution of the growth of a solid planar interface within an undercooled melt [5, 9, 16, 20]. The sharp interface equations

(2.4)-(2.6) are solved with boundary conditions

$$\theta(\infty, y) = \theta_{cool}, \quad \theta(-\infty, y) = \theta_{cool} + 1, \quad y \in (-\infty, \infty).$$

It is easy to show that there exists a travelling wave solution which takes the form

$$\theta(x, y, t) = \begin{cases} \theta_{cool} + e^{-v^*(x-v^*t)/D}, & x > v^*t, \\ \theta_{cool} + 1, & x \leq v^*t, \end{cases} \quad (5.1)$$

where the velocity

$$v^* = -\frac{1}{\alpha d_0} (\theta_{cool} + 1). \quad (5.2)$$

We consider the case where $\alpha = 1$, $d_0 = 0.05$ and $\theta_{cool} = 1.05$ which leads to a travelling wave with unit velocity.

We solve the problem in the two-dimensional geometry $\Omega = [-0.25, 0.25]^2$ using a moving mesh with $N = 50$, $\Delta t = 10^{-3}$, and $\tau = 0.1$. For the phase-field model we let $\varepsilon = 0.00625$. Fig. 1 shows the computed profiles of θ and p along the bottom boundary $y = -0.25$ at the times $t = 0.02, 0.04, 0.06, 0.08$ and $t = 0.1$. The solid lines represent the numerical results and the dashed lines the travelling wave profile of the phase-field equations translated to the right at the speed $v(\varepsilon)$ which was obtained using the one-dimensional adaptive moving mesh method described in [22]. We can see that we have excellent agreement. Note also that the choice of the parameter $\gamma = 1$ leads as predicted to a mesh that approximately places half of the mesh points within the interface region. Fig. 2 shows the computed interface positions which again agree well with the reference solution. Note also that no grid anisotropy has been introduced. The L_2 error in the temperature at $t = 0.1$ along the bottom boundary was found to be 6.45×10^{-4} which compares favourably with the one-dimensional calculations presented in [22]. Finally, in Fig. 3 we can see the meshes used and the corresponding front predictions.

5.2 Critical radius of equilibrium

The second problem we consider involves the stability of a solid sphere in equilibrium with its undercooled liquid melt. Let us consider a domain Ω which has no heat flux into it and within this domain the initial temperature is equal to a constant, θ_{cool} . Let us introduce an initial ball of solid of radius, R_o , centred at the origin, lying inside the undercooled liquid. It is well known that there exists a steady state solution of (2.4)–(2.6) where the solid ball is in equilibrium with its melt [12]. This occurs when the radius of the solid ball, R_c , is given by

$$R_c = -\frac{d_o}{\theta_{cool}}.$$

This equilibrium is unstable in that if $R_o < R_c$ then the solid sphere will melt and the radius will decrease to zero. On the other hand if $R_o > R_c$ then the solid will expand into the undercooled melt and the radius will increase.

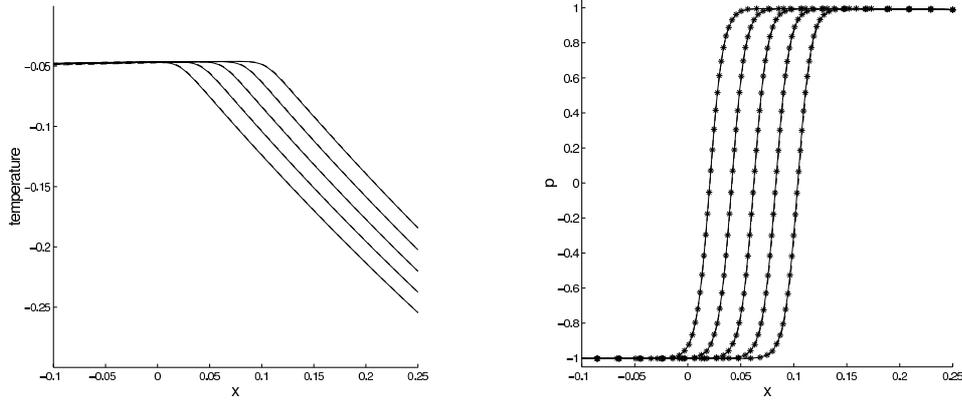


Figure 1: Temperature (left) and phase variable (right) profiles for two-dimensional planar solidification with $\varepsilon=0.00625$ and $d_o = 0.05$.

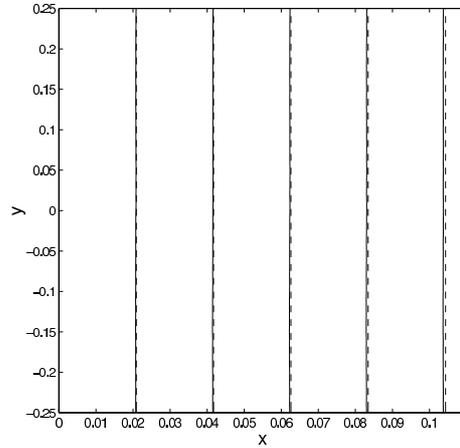


Figure 2: Exact and approximate front positions for two-dimensional planar solidification with $\varepsilon=0.00625$ and $d_o = 0.05$.

If we take our initial temperature to be $\theta_{cool} = -2$ and $d_o = 0.5$ then $R_c = 1/4$. The phase-field calculations are performed with $\varepsilon = 1/(160\sqrt{2})$. The initial phase profile is given by

$$p(\mathbf{x}, 0) = p_{bc} \tanh\left(\frac{r(\mathbf{x})}{2\varepsilon}\right),$$

where

$$p_{bc} = \begin{cases} \min_p\{p : f(p, \theta_{cool}) = 0\}, & \text{closest to } -1, \quad r(\mathbf{x}) < 0, \\ \max_p\{p : f(p, \theta_{cool}) = 0\}, & \text{closest to } 1, \quad r(\mathbf{x}) \geq 0, \end{cases}$$

and $f(p, \theta)$ is given by (2.9). We consider two cases where the initial radius is $R_o = 0.24$ and $R_o = 0.26$ with $N = 50$, $\Delta t = 10^{-3}$, $\tau = 0.1$ in the domain $[0, 0.5]^2$. The mesh is adapted with $\gamma = 1$. Fig. 4 shows the grids and front positions at times $t = 0.02, 0.08, 0.14$, and $t = 0.16$ when $R_o = 0.24$. As expected the interface moves inwards and the radius decreases

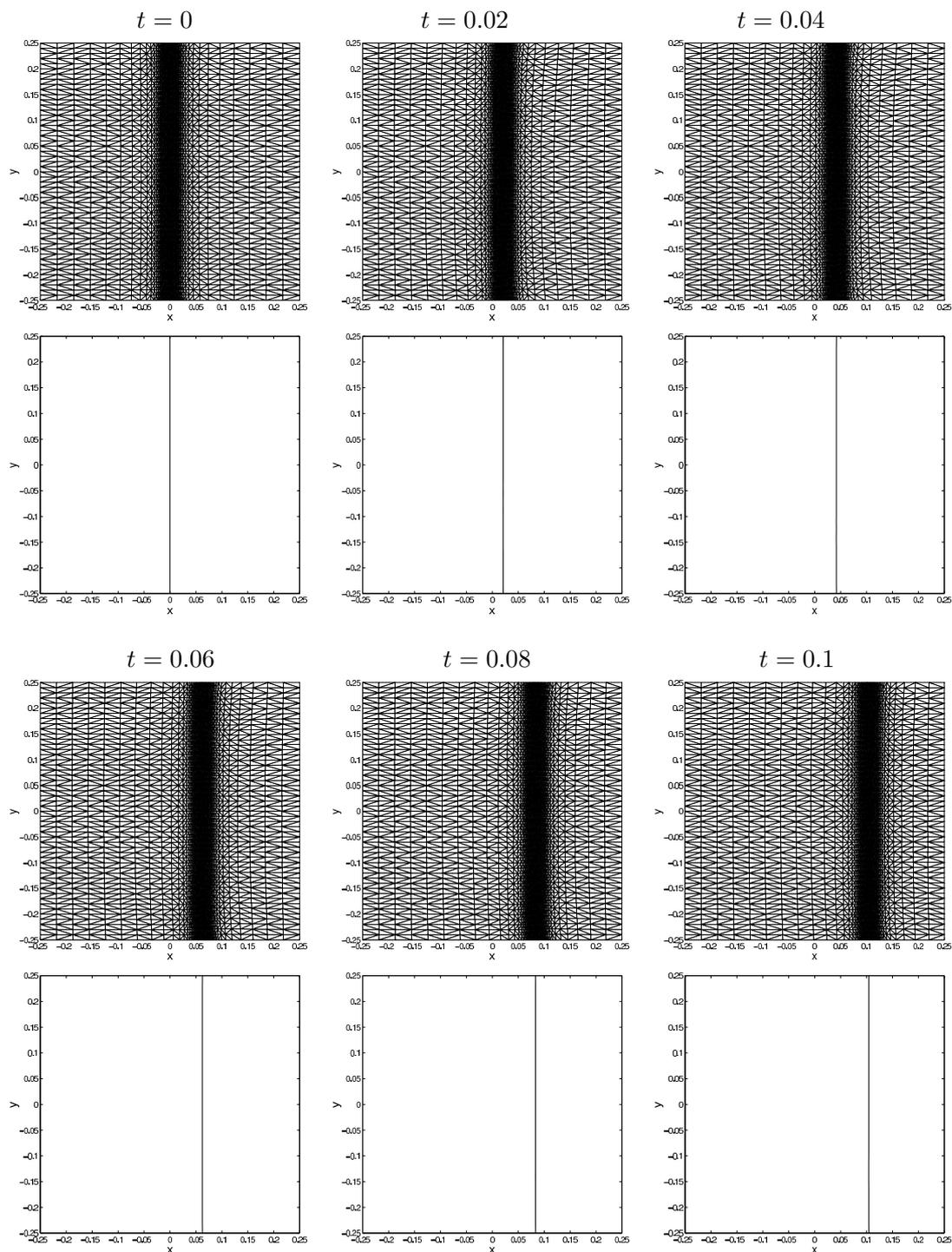


Figure 3: Grid and interface predictions for Problem 1 with $\varepsilon=0.00625$ and $d_o = 0.05$.

until at $t = 0.1950$ the interface vanishes. The results obtained when $R_o = 0.26$ were also as expected, with the ball solidifying outwards. Figs. 5(b) and (c) shows the interface position at time intervals of 2×10^{-2} for $R_o = 0.24$ and $R_o = 0.26$, respectively. The radial positions for both interfaces are given in Fig. 5(a). The radii have been calculated by inverse linear interpolation of the zero level set of the phase order parameter along the boundary $x = 0$. Although we have only used a 50×50 mesh, these results compare favourably to those obtained in [15] where a uniform 128×128 mesh was used with a time step $\Delta t = 10^{-5}$.

Fig. 6 shows the solidification of two solid spheres which are surrounded by undercooled liquid. The initial radii of both spheres is 0.26 and the centres are located at (0.75, 1.25) and (1.25, 0.75). We can see that the spheres increase in size until they touch, at which point they merge into one. At the point of contact there is a region of very high negative curvature which diminishes rapidly due the interface conditions that are implicit in the phase-field equations. Note that the adaptive mesh does an excellent job of following the evolution of both the interfaces even when there is a change in topology as they merge.

5.3 Scaled viscous Cahn-Hilliard problem

The viscous Cahn-Hilliard equation [1, 24] arises from the phase-field model by eliminating θ_t from (2.7). This equation was derived to include certain viscous effects neglected in the original Cahn-Hilliard model for spinodal decomposition - a process by which isothermal phase separation occurs in a binary alloy after the temperature has been reduced beneath its critical value. In this setting θ represents a generalised chemical potential and p is a concentration variable.

Let us consider a circular domain Ω of radius R and that homogeneous Neumann data for θ is imposed on the fixed boundary $r = R$. Initially we have two circular interfaces with $R_I = 0.15$ and $R_O = 0.30$. If we assume radial symmetry then the sharp interface problem is equivalent to solving Laplace's equation

$$\frac{1}{r} (r\theta_r(r, t))_r = 0,$$

which has solution

$$\theta(r, t) = \begin{cases} B(t) & 0 < r < R_I(t), \\ A^+(t) \ln r + B^+(t) & R_I(t) < r < R_O(t), \\ B^-(t) & r > R_O(t). \end{cases}$$

Using condition (2.6) we have that

$$B(t) = \theta(R_I(t), t) = -\frac{d_o}{R_I(t)} - d_o\alpha \frac{dR_I(t)}{dt},$$

and

$$B^-(t) = \theta(R_O(t), t) = \frac{d_o}{R_O(t)} + d_o\alpha \frac{dR_O(t)}{dt}.$$

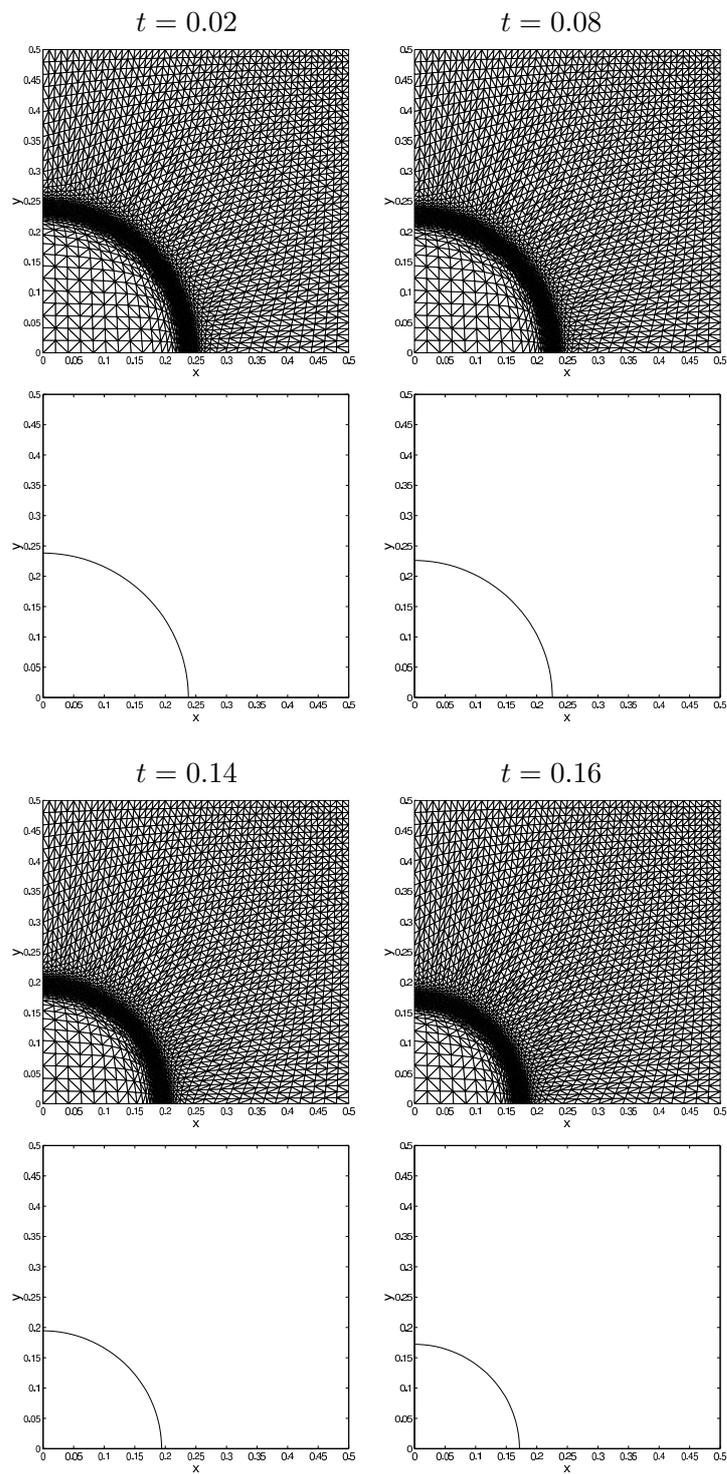
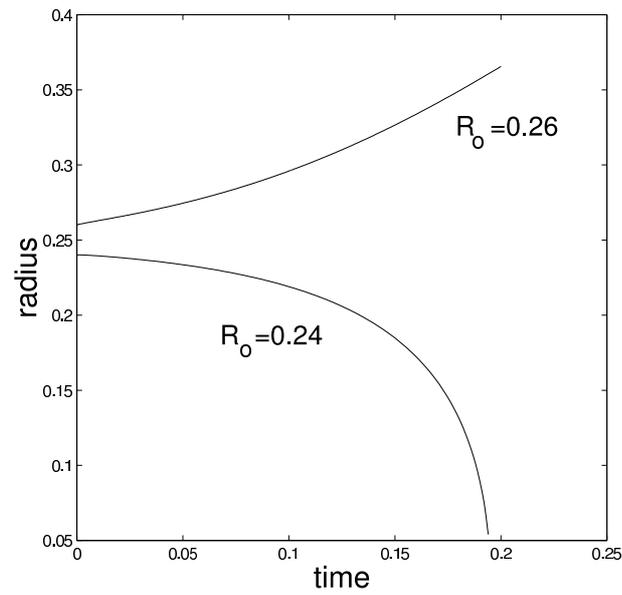
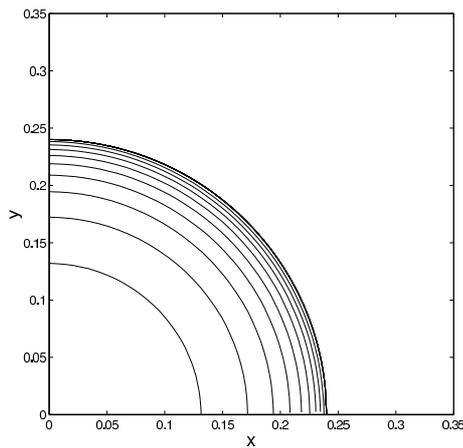


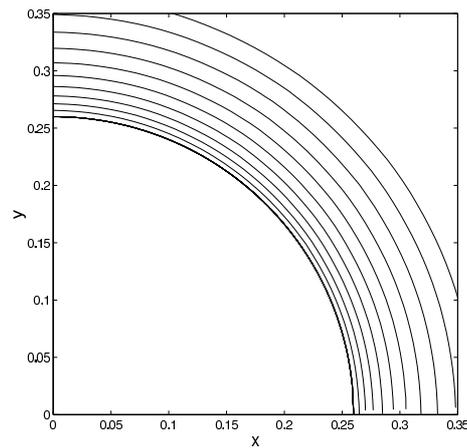
Figure 4: Grid and interface predictions for Problem 2 with $\varepsilon = 1/(160\sqrt{2})$, $d_o = 0.5$ and $R_o = 0.24$.



(a)



(b)



(c)

Figure 5: (a) Radius, (b) front positions for $R_o = 0.24$, (c) front positions for $R_o = 0.26$ for $\varepsilon = 1/(160\sqrt{2})$ and $d_o = 0.5$.

In addition, from the jump condition (2.5) we have

$$R_I \frac{dR_I(t)}{dt} = R_O \frac{dR_O(t)}{dt} = -A^+(t), \quad (5.3)$$

and to ensure conservation of mass we require

$$R_O^2(t) = R_I^2(t) + \delta, \quad (5.4)$$

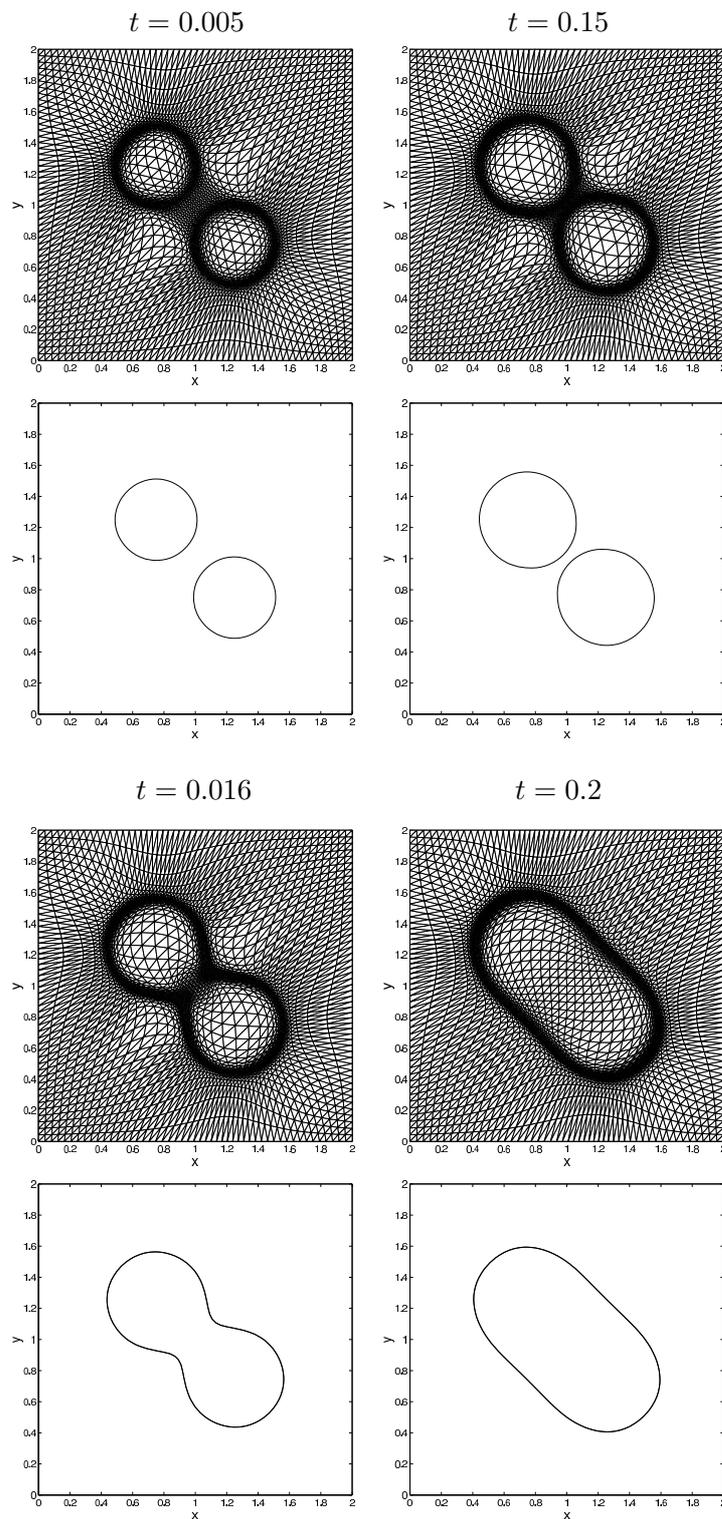


Figure 6: Two seeds of solid are surrounded by an undercooled melt. The intersection of the interfaces poses no difficulty for the adaptive moving mesh computation of the phase-field equations.

where δ is a constant determined by the initial values of the radii. Using the above information we find that

$$\frac{dR_I(t)}{dt} = \frac{-d_o \left(\frac{1}{R_O(t)} + \frac{1}{R_I(t)} \right)}{-d_o \left(1 + \frac{R_I(t)}{R_O(t)} \right) + R_I(t) \ln \left(\frac{R_O(t)}{R_I(t)} \right)}. \quad (5.5)$$

We solve (5.5) using an adaptive Runge-Kutta method. Fig. 9 (b) shows $R_I(t)$ and $R_O(t)$ as dashed lines when $R_I(0) = 0.15$, $R_O(0) = 0.3$, $\alpha = 1$, and $d_o = 0.5$. We can see that both balls decrease in size with the inner ball shrinking faster than the outer ball. The time at which the inner interface $R_I(t)$ vanishes is approximately $t \approx 0.01287$.

We now consider solving the phase-field equations with $\varepsilon = 1/(320\sqrt{2})$ using an adaptive mesh with $N = 50$, $\Delta t = 6.5 \times 10^{-5}$, $\tau = 0.1$, $\gamma = 1$ on the domain $[0, 0.5]^2$. This test case has also been considered in [15]. Fig. 7 shows the grids and front positions at times $t = 0.0013$, 0.0039 , 0.0078 , and $t = 0.0117$. We can see that the mesh has no difficulty adapting to the two interfaces. As with the sharp interface problem, both balls move inwards and the radius of the inner ball decreases until at $t = 0.01287$ the interface vanishes. The temperature and phase variable profiles along the bottom boundary at these times are shown in Fig. 8(a) and (b), respectively. It can be seen in Fig. 8(b) that a significant number of points lie in both interfacial regions. Fig. 9(a) shows the interface positions at time intervals 1.3×10^{-3} where we can see that the inner ball shrinks faster than the outer ball. Finally, the radial positions for both interfaces are given in Fig. 9(b). These results are similar to those obtained in [15] and also approximate well the solution of the sharp interface problem which is shown as dashed lines in Fig. 9(b).

5.4 Faceted crystal growth

Faceted crystal growth, where corners in the interface can develop, provides a difficult test for the phase-field approach. From smooth initial data corners and facets will propagate along planar fronts dictated by the material anisotropy and the domain geometry. Anisotropy in the phase-field equations can involve both $\sigma(\psi)$ and $\alpha(\psi)$, where ψ is the angle between the interface and the x -axis given by

$$\cos(\psi) = \frac{\nabla p}{|\nabla p|} \cdot n_x.$$

For this example, we consider the anisotropy

$$\alpha^{-1}(\psi) = \alpha_0^{-1} [1 + \delta_\alpha \cos(M(\psi - \psi_\alpha))], \quad (5.6)$$

where M is an integer, $\delta_\alpha > 0$ is the magnitude of the anisotropy, and ψ_α is the preferred angle for growth. Fig. 10 shows the growth of two circular solid seeds of unit radii centred at $(0.75, 1.25)$ and $(1.25, 0.75)$. The parameters chosen for (5.6) are $M = 4$, $\alpha_0 = 0.065$, $\delta_\alpha = 0.818$, and $\psi_\alpha = \pi/4$. We can see clearly the effect of the anisotropy as the initial

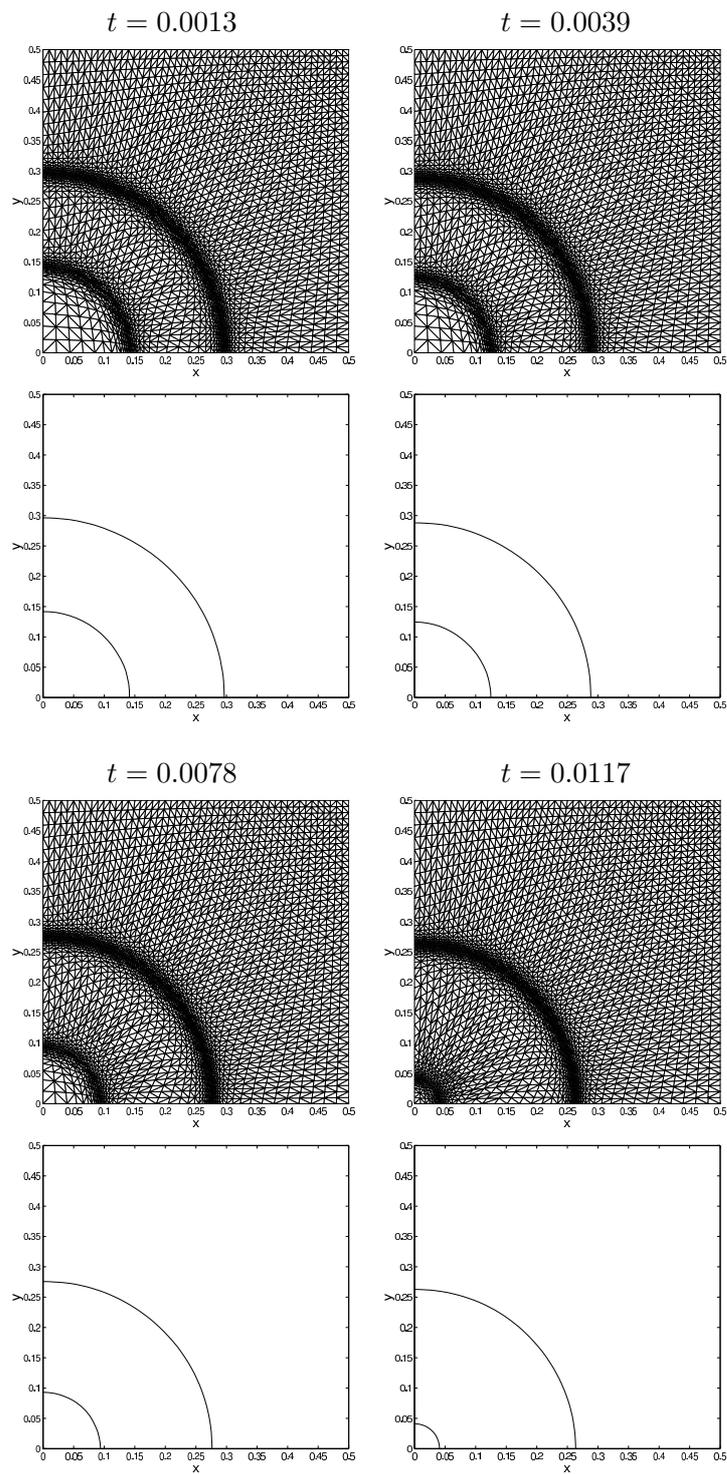


Figure 7: Grid and interface predictions for viscous Cahn-Hilliard problem with $\varepsilon = 1/(160\sqrt{2})$, $d_o = 0.5$, $R_I = 0.15$ and $R_O = 0.3$.

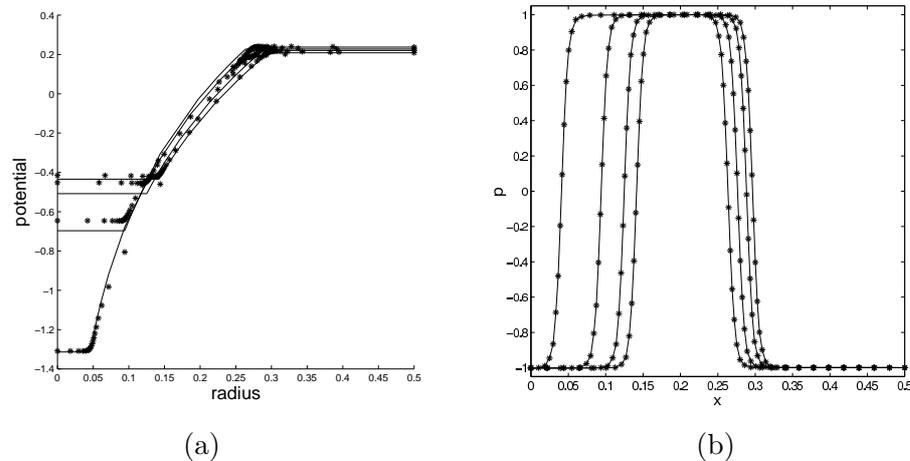


Figure 8: (a) Potential (—sharp interface, * phase-field) and (b) concentration profiles at $t = 0.0013, 0.0039, 0.0078, 0.0117$ for viscous Cahn-Hilliard simulation.

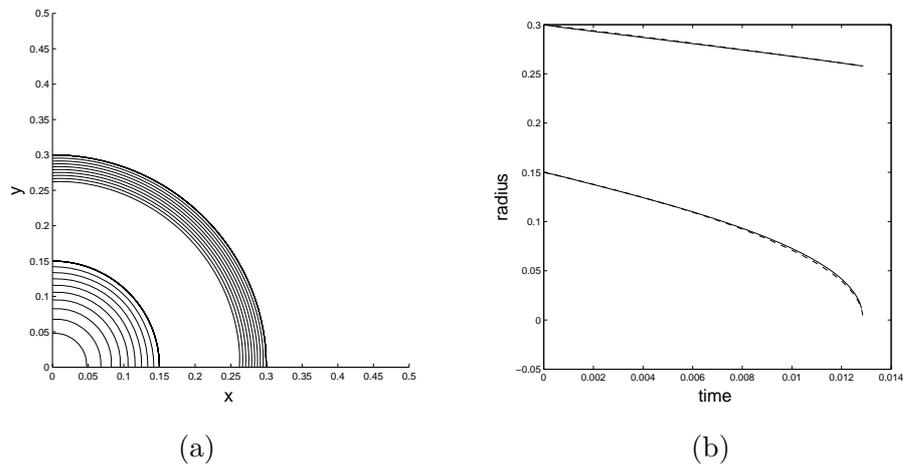


Figure 9: (a) Front positions, and (b) radii for viscous Cahn-Hilliard problem with $\varepsilon = 1/(320\sqrt{2})$, $d_o = 0.5$, $R_I = 0.15$ and $R_O = 0.3$; -- sharp interface, — phase-field.

circles quickly develop slightly rounded corners. As the simulation proceeds the “rounded squares” merge and the interface develops regions of very high negative curvature. Once again we can see that the adaptive moving mesh does an excellent job of tracking the interfaces and has no difficulty in dealing with the change in topology.

6 Conclusions

We have developed an adaptive moving mesh method that has been successfully applied to solve the two-dimensional phase-field equations. The algorithm is relatively simple and

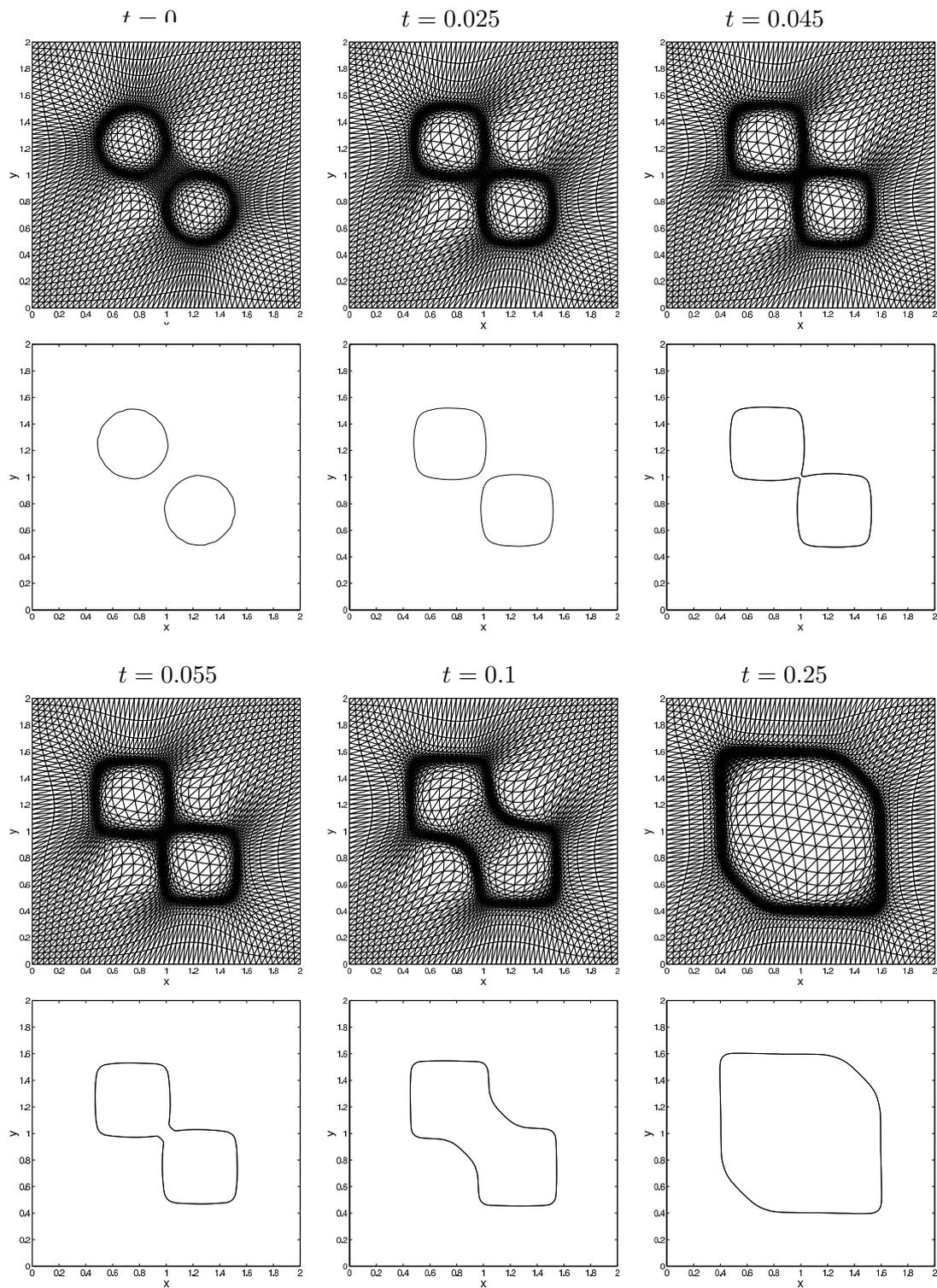


Figure 10: Grid and interface predictions for faceted crystal growth $\varepsilon = 5 \times 10^{-3}$, $d_o = 0.01$, $\alpha_0 = 0.1$

has been shown to handle easily changes in topology of evolving interfaces.

Future work will concentrate on combining the present method with an h -refinement strategy for problems that develop topologically complex solidification fronts that appear, for example, in the growth of dendrites.

References

- [1] F. Bai, C. M. Elliott, A. Gardiner, A. Spence and A. M. Stuart, The viscous Cahn-Hilliard equation. Part I: Computations, *Nonlinearity*, 8 (1995), 131–160.
- [2] G. Beckett, J. A. Mackenzie, A. Ramage and D. M. Sloan, On the numerical solution of one-dimensional PDEs using adaptive methods based on equidistribution, *J. Comput. Phys.*, 167 (2001), 372–392.
- [3] G. Beckett, J. A. Mackenzie, A. Ramage and D. M. Sloan, Computational solution of two-dimensional unsteady PDEs using moving mesh methods, *J. Comput. Phys.*, 182 (2002), 478–495.
- [4] G. Beckett, J. A. Mackenzie and M. L. Robertson, A moving mesh finite element method for the solution of two-dimensional Stefan problems, *J. Comput. Phys.*, 168 (2001), 500–518.
- [5] R. J. Braun, G. B. McFadden and S. R. Coriell, Morphological instability in phase-field models of solidification, *Phys. Rev. E*, 49 (1994), 4336–4352.
- [6] R. J. Braun and B. T. Murray, Adaptive phase-field computations of dendritic growth, *J. Cryst. Growth*, 174 (1997), 41–53.
- [7] G. Caginalp, Mathematical models of phase boundaries, in: J. Ball (Ed.), *Material Instabilities in Continuum Mechanics and Related Mathematical Problems*, Oxford University Press, USA, 1988, pp. 35–52.
- [8] G. Caginalp, Stefan and Hele-Shaw type models as asymptotic limits of the phase-field equations, *Phys. Rev. A.*, 39 (1989), 5887–5896.
- [9] G. Caginalp and E. A. Socolovsky, Phase field computations of single-needle crystals, crystal growth, and motion by mean curvature, *SIAM J. Sci. Comput.*, 15 (1994), 106–126.
- [10] W. Cao, W. Huang and R. D. Russell, An r-adaptive finite element method based upon moving mesh PDEs, *J. Comput. Phys.*, 149 (1999), 221–244.
- [11] W. Cao, W. Huang and R. D. Russell, A study of monitor functions for two-dimensional adaptive mesh generation, *SIAM J. Sci. Comput.*, 20 (1999), 1978–1994.
- [12] B. Chalmers, *Principles of Solidification*, Krieger, New York, 1977.
- [13] P. I. Crumpton and G. Shaw, A vertex-centred finite volume method with shock detection, *Int. J. Numer. Meth. Fl.*, 18 (1994), 605–625.
- [14] A. S. Dvinsky, Adaptive grid generation from harmonic maps on Riemannian manifolds, *J. Comput. Phys.*, 95 (1991), 450–476.
- [15] C. M. Elliott and A. R. Gardiner, Phase field equations, in: D. Stewart, H. Gardner and D. Singleton (Eds.), *Computational Techniques and Applications, CTAC93*, World Scientific, Singapore, 1993.
- [16] K. Glasner, Nonlinear preconditioning for diffuse interfaces, *J. Comput. Phys.*, 174 (2001), 695–711.
- [17] W. Huang, Practical aspects of formulation and solution of moving mesh partial differential equations, *J. Comput. Phys.*, 171 (2001), 753–775.
- [18] W. Huang and R. D. Russell, A high dimensional moving mesh strategy, *Appl. Numer. Math.*, 26 (1997), 63–76.

- [19] W. Huang and R. D. Russell, Moving mesh strategy based on a gradient flow equation for two-dimensional problems, *SIAM J. Sci. Comput.*, 20(3) (1999), 998–1015.
- [20] A. Karma and W-J. Rappel, Quantitative phase-field modeling of dendritic growth in two and three dimensions, *Phys. Rev. E*, 57 (1998), 4323–4349.
- [21] J. A. Mackenzie and M. L. Robertson, The numerical solution of one-dimensional phase change problems using an adaptive moving mesh method, *J. Comput. Phys.*, 161 (2000), 537–557.
- [22] J. A. Mackenzie and M. L. Robertson, A moving mesh method for the solution of one-dimensional phase-field equations, *J. Comput. Phys.*, 181(2) (2002), 526–544.
- [23] J. F. McCarthy, One-dimensional phase field models with adaptive grids, *Trans. ASME*, 120 (1998), 956–964.
- [24] A. Novick-Cohen, On the viscous Cahn-Hilliard equation, in: J. Ball (Ed.), *Material Instabilities in Continuum Mechanics and Related Mathematical Problems*, Oxford University Press, USA, 1988, pp. 329–342.
- [25] N. Provatas, N. Goldenfeld and J. Dantzig, Efficient computation of dendritic microstructures using adaptive mesh refinement, *Phys. Rev. Lett.*, 80(15) (1998), 3308–3311.
- [26] N. Provatas, N. Goldenfeld and J. Dantzig, Adaptive mesh refinement computation of solidification microstructures using dynamic data structures, *J. Comput. Phys.*, 148 (1999), 265–290.
- [27] Z. Tan, T. Tang and Z. Zhang, A simple moving mesh method for one- and two-dimensional phase-field equations, *J. Comput. Appl. Math.*, 190 (2006), 252–269.
- [28] R. T. Tenchev, J. A. Mackenzie, T. J. Scanlon and M. T. Stickland, Finite element moving mesh analysis of phase change problems with natural convection, *Int. J. Heat Fluid Fl.*, 26 (2005), 597–612.
- [29] S-L. Wang and R. F. Sekerka, Algorithms for phase field computation of the dendritic operation state at large supercoolings, *J. Comput. Phys.*, 127 (1996), 110–117.
- [30] S-L. Wang, R. F. Sekerka, A. A. Wheeler, B. T. Murray, S. R. Coriell, R. J. Braun and G.B. McFadden, Thermodynamically-consistent phase field-models for solidification, *Physica D*, 69 (1993), 189–200.
- [31] A. A. Wheeler, B. T. Murray and R. J. Schaefer, Computation of dendrites using a phase field model, *Physica D*, 66 (1993), 243–262.
- [32] A. Winslow, Numerical solution of the quasilinear Poisson equation in a nonuniform mesh, *J. Comput. Phys.*, 2 (1967), 149–172.