A Multi-Mesh Adaptive Finite Element Approximation to Phase Field Models

Xianliang Hu[1,2], Ruo Li[3,*] and Tao Tang[4]

[1] *Department of Mathematics, Zhejiang University, Hangzhou 31027, China.*
[2] *Department of Applied and Computational Mathematics, California Institute of Technology, Pasadena, CA 91125, USA.*
[3] *CAPT, LMAM & School of Mathematical Sciences, Peking University, Beijing 100871, China.*
[4] *Department of Mathematics, Hong Kong Baptist University, Kowloon Tong, Kowloon, Hong Kong.*

Received 2 April 2008; Accepted (in revised version) 3 August 2008

Communicated by Jie Shen

Available online 20 October 2008

Abstract. In this work, we propose an efficient multi-mesh adaptive finite element method for simulating the dendritic growth in two- and three-dimensions. The governing equations used are the phase field model, where the regularity behaviors of the relevant dependent variables, namely the thermal field function and the phase field function, can be very different. To enhance the computational efficiency, we approximate these variables on different h-adaptive meshes. The coupled terms in the system are calculated based on the implementation of the multi-mesh h-adaptive algorithm proposed by Li (J. Sci. Comput., pp. 321-341, 24 (2005)). It is illustrated numerically that the multi-mesh technique is useful in solving phase field models and can save storage and the CPU time significantly.

AMS subject classifications: 65M20, 65N22, 80A22

Key words: Multi-mesh, local refinement, adaptive finite element, phase field.

1 Introduction

Dendritic growth is a main ingredient of the solidification microstructures, which has become one of the most interesting research topics in recent years. Most of the theoretical and experimental works have been devoted in understanding the mechanism of pattern selection during solidification procedure, since the microscopic properties of such

[*]Corresponding author. *Email addresses:* huxl98@yahoo.com.cn (X. Hu), rli@math.pku.edu.cn (R. Li), ttang@math.hkbu.edu.hk (T. Tang)

procedure are determined by the length scale of dendrites. With the development of the theories for mathematical models of solidification, numerical simulations have become a powerful tool in investigating dendritic growth. Most theories of solidification are based on the time dependent Stefan model, which describes the evolution of thermal field around the solidification interface by the well-known heat equation with two boundary conditions: the Stefan condition [16] and the Gibbs-Thomson condition [9]. The solution of Stefan model can be approximated by that of the phase field model [4], which avoids the task of tracking the interface. The phase field model uses a phase field variable $\phi(\mathbf{r})$, which is 1 in solid phase and -1 in liquid phase. Meanwhile, the value of ϕ decreases from 1 to -1 very rapidly within a small width W near the interface; thus the level set $\phi(\mathbf{r})=0$ represents the interface implicitly.

It was originally shown by Caginalp and Chen [3] if W is much smaller than the capillary length d_0 then the phase field model converges to the sharp interface limit. With smaller value of W, more computational cost should be paid since the smallest element parameter Δx should be much smaller than W in order to fully resolve the interface. This requirement prevents us from using very small W which is close to the physical realities. Another limitation which seriously restricts the use of the phase field simulation is that the interface kinetics β should be big enough to ensure the convergence due to the Gibbs-Thomson boundary condition. But physically, it is important to simulate solidification microstructures in the limit of zero interface kinetics because most experiments performed at low undercooling are within this limit. In 1996, Wang and Sekerka [21] showed that the phase field simulations of dendritic growth are independent of computational parameters, but the results are only feasible in a range of large undercooling. Karma and Rappel [9,10] demonstrated that such limitations can be less harmful by providing the understanding in their new asymptotic analysis. It is revealed that the phase field approach can be extended to the case of arbitrary small or even zero β, and the limitation on W is not so stringent. These results made it possible to simulate more physical cases in large interface width W and low undercooling. As an important advantage of phase field model, it should be mentioned that its extension to three dimensional case is straightforward.

For solving the phase field models for dendritic solidification, the finite difference method on uniform mesh is utilized in [9, 10, 20] due to its simplicity. In the work of Provatas et al. [15], the adaptive finite element method is applied to the phase field model of dendritic growth in pure melt. The purpose of adaptive mesh refinement is to reduce computational cost so that simulations of larger scale problems become possible with currently available computational resources. In their simulations the smallest element size δx is much smaller than W; and the ratio of the system size to the smallest element size is even greater than 2^{17}. Most numerical methods will fail to work in such case, while the adaptive mesh refinement scheme can make the computation possible. The numerical results of [15] indicated the adaptive finite element method can produce quite satisfactory phase field solutions at high undercooling.

The thermal field u is one of the variables in the phase field model. It is interesting that u and the phase function ϕ have different singularity behaviors. Since ϕ is to provide the interface information, it only needs to be fully resolved near the transient layers. On the other hand, the thermal field u is evolved based mainly on the thermal diffusive coefficients of the material under consideration. It is possible that there are large solution gradients for u, but they are often much milder than those of ϕ. Moreover, in general the locations for the large gradients for u and ϕ are very different, especially when complex dendritic structures are formatted. Solution-wise, the structure of ϕ is much simpler than u: ϕ is constant in the interior parts of both solid bulk and liquid bulk with a small distance from the interface, while u changes gradually in the whole solution domain as pointed out in [15]. To fully resolve the phase field, most of the element should be clustered in a small region around the interface with a small width W. If only a single mesh where used, then the thermal field u would be seriously over-resolved near the interface.

Motivated by the above observations, we will in this work explore the possibility of approximating ϕ and u on two different adaptive meshes. The purpose of using the multi-mesh strategy is to reduce the overall computational costs while still maintaining high accuracy. Moreover, there is an additional advantage in using the multi-mesh approach: to approximate more than one variables on one adaptive mesh, the mesh adaptive indicator is often a linear combination of the contributions from all variables. The strategy in choosing these coefficients is highly problem dependent. On the other hand, it is relatively easier to approximate only one variable on the adaptive mesh. In this case, it can be achieved by controlling the size of the minimal element. For the multi-mesh approximation method, we can take this advantage to approximate each variable on its own mesh.

In the implementation of the multi-mesh approximation, a key technical problem is how to calculate the coupled terms in the governing equations, which involves the numerical quadratures on two different meshes. This will be done by using the multi-mesh h-adaptive algorithm proposed in [12] and the finite element package AFEPack [13].

The layout of this paper is as follows. In Section 2 we introduce the phase field model and in Section 3 we give the finite element discretization for the governing equations on static meshes and on multi-meshes. In Section 4, we demonstrate the performances of our adaptive scheme in two- and three-dimensions, respectively. The concluding remarks will be given in the final section.

2 The phase field model for dendritic growth

We consider the phase field model in two-dimensions as proposed by [15]

$$\frac{\partial u}{\partial t} = D\triangle u + \frac{1}{2}\frac{\partial \phi}{\partial t}, \tag{2.1}$$

$$\begin{aligned}\tau(\mathbf{n})\frac{\partial\phi}{\partial t}=&\nabla\cdot(W^2(\mathbf{n})\nabla\phi)+(\phi-\lambda u(1-\phi^2))(1-\phi^2)\\&+\frac{\partial}{\partial x}\left(|\nabla\phi|^2W(\mathbf{n})\frac{\partial W(\mathbf{n})}{\partial\phi_x}\right)+\frac{\partial}{\partial y}\left(|\nabla\phi|^2W(\mathbf{n})\frac{\partial W(\mathbf{n})}{\partial\phi_y}\right),\end{aligned}\tag{2.2}$$

where $u=c_P(T-T_M)/L$ is the rescaled thermal field, c_P the specific heat at constant pressure, L is the latent heat of fusion and T_M is the melting temperature of the solid phase; D is the diffusion parameter of the thermal field, supposing it take the same value both in solid phase and liquid phase. In the case of different values for D in solid and liquid phases, the corresponding convergence analysis can be found in Almgren [1]. The interface width $W(\mathbf{n})$ and $\tau(\mathbf{n})$ are defined as following

$$W(\mathbf{n})=W_0A(\mathbf{n}),\tag{2.3}$$

$$\tau(\mathbf{n})=\tau_0A(\mathbf{n})^2,\tag{2.4}$$

where W_0 and τ_0 are constants for the scale of spatial and the time respectively, and

$$A(\mathbf{n})=(1-3\epsilon_4)\left(1+\frac{4\epsilon_4}{1-3\epsilon_4}\frac{\phi_x^4+\phi_y^4}{|\nabla\phi|^4}\right)$$

introduces the anisotropic factor. Here the parameter ϵ_4 represents the measure of the anisotropy which causes the deviation of W from W_0. The unit vector

$$\mathbf{n}=(\phi_x^2+\phi_y^2)^{-\frac{1}{2}}\left[\phi_x,\phi_y\right]^T\tag{2.5}$$

is the normal direction to the contour of ϕ, implicitly defined in (2.3), where ϕ_x and ϕ_y represent the partial derivatives of ϕ. There are several other ways to define anisotropy, and here we adopt the way proposed in [10] as described above.

We focus on the simulation in the limit of zero kinetics, which can approximate the Stefan problem more accurately. Due to the asymptotic analysis by Karma and Rappel [9], it is reasonable to simulate phase field model for dendritic growth with zero kinetic parameter β, which is defined by

$$\beta=\frac{a_1\tau_0}{\lambda W_0}\left[1-\lambda a_2\frac{W_0^2}{D\tau_0}\right],\tag{2.6}$$

where $a_1=0.8839$ is for the particular form of the free energy of surface tension and $a_2=0.6267$ is for the choice of the free energy functional as proposed in [10]. Obviously, $\beta=0$ means that

$$\lambda=\frac{1}{a_2}\frac{D\tau_0}{W_0^2},\tag{2.7}$$

which help us to simulate the limit cases of vanishing interface kinetics.

In (2.2), the nonlinear term

$$f(\phi,u;\lambda)=(\phi-\lambda u(1-\phi^2))(1-\phi^2)$$

is referred to as the derivative of double well potential with respect to ϕ. In this case, u and ϕ are coupled via the constant λ of order $\mathcal{O}(1)$ and come from so called the Isothermal Variational Form (IVF) as described in [15]. Another simpler choice of $f(\phi,u;\lambda)$ is called the Variational Form (VF), however, it is pointed out in [10] that the IVF model can provide more rapid convergence rate while the VF can produce an unphysical spurious branch of steady state growth solutions in high velocity. Therefore, we choose the IVF for f in current research. Moreover, there is an important quantity named capillary length, which can be approximated by

$$d_0(\mathbf{n})=\frac{a_1}{\lambda}W(\mathbf{n}).$$

in the isotropic case. It reflects the interface width in some sense, and the minimum element size for ϕ near the interface should not larger than this value. For further description on the model problem, we refer to [10,15].

In three dimensional case, the system is only slightly different:

$$\frac{\partial u}{\partial t}=D\triangle u+\frac{1}{2}\frac{\partial\phi}{\partial t}, \tag{2.8}$$

$$\begin{aligned}\tau(\mathbf{n})\frac{\partial\phi}{\partial t}=&\nabla\cdot(W^2(\mathbf{n})\nabla\phi)+(\phi-\lambda u(1-\phi^2))(1-\phi^2)\\&+\frac{\partial}{\partial x}\left(|\nabla\phi|^2W(\mathbf{n})\frac{\partial W(\mathbf{n})}{\partial\phi_x}\right)+\frac{\partial}{\partial y}\left(|\nabla\phi|^2W(\mathbf{n})\frac{\partial W(\mathbf{n})}{\partial\phi_y}\right)\\&+\frac{\partial}{\partial z}\left(|\nabla\phi|^2W(\mathbf{n})\frac{\partial W(\mathbf{n})}{\partial\phi_z}\right),\end{aligned} \tag{2.9}$$

where the definition of anisotropic term is given by

$$W(\mathbf{n})=W_0(1-3\epsilon_4)\left(1+\frac{4\epsilon_4}{1-3\epsilon_4}\frac{\phi_x^4+\phi_y^4+\phi_z^4}{|\nabla\phi|^4}\right). \tag{2.10}$$

The two- and three-dimensional models were solved in Wang et al. [19] using a r-adaptive finite element method. The r-adaptive methods for phase-field problems have been studied recently, see, e.g., [2] for solving one- and two-dimensional phase-field problems. In [19], the governing equations are solved in the physical domain and the meshes are redistributed in the physical domain using a strategy proposed in [5]. This approach makes efficient simulations in three space dimensions possible. It will be demonstrated in this work that the efficiency can be further enhanced by using multi-mesh adaptive methods.

3 The adaptive finite element solution on two meshes

For ease of demonstrations, we only describe the discretization for two dimensional case. The three-dimensional implementation can be obtained similarly particularly with the use of AFEPack software. We first introduce the finite element discretization of the phase field models (2.1)-(2.2) using multi-mesh strategy. The main difference between our scheme and the composite field scheme [15] is that we discretize the two equations on two different meshes by considering solution regularities of multi-variables.

3.1 The finite element discretization

The computational domain is as simple as a rectangle $[-L_x,L_x]\times[-L_y,L_y]$. As has been done in the references, it is sufficient to consider the problem on the positive quarter of the domain: $\Omega=[0,L_x]\times[0,L_y]$. We can further choose the computational domain even half of the quarter domain due to the symmetry to the line $y=x$. For the two variables ϕ and u, we denote their own meshes by $\triangle_\phi$ and $\triangle_u$, respectively. It is pointed out that they are both generated from the same background mesh on Ω. We also denote the piecewise linear finite element spaces by $V_h(\triangle_\phi)$ and $V_h(\triangle_u)$ corresponding to $\triangle_\phi$ and $\triangle_u$. Then we consider the finite element approximations of unknown functions ϕ and u as

$$\phi_h=\sum_{i=1}^{N_\phi}\phi_i N_i(\triangle_\phi),\quad u_h=\sum_{i=1}^{N_u}u_i N_i(\triangle_u),$$

where $\{\phi_i\}_{i=1}^{N_\phi}$ and $\{u_i\}_{i=1}^{N_u}$ are the coefficients for phase field variable and thermal field variable, with N_ϕ and N_u being the dimensions of $V_h(\triangle_\phi)$ and $V_h(\triangle_u)$, respectively. $\{N_i(\triangle_\phi)\}_{i=1}^{N_\phi}$ and $\{N_i(\triangle_u)\}_{i=1}^{N_u}$ are the finite element basis for $V_h(\triangle_\phi)$ and $V_h(\triangle_u)$. For convenience, we also denote their coefficients as $\mathbf{\Phi}$ and $\mathbf{u}$ if there is no confusion would arise. Then the standard Galerkin finite element method yields the semi-discretization formulation for (2.1)-(2.2):

$$M_u\frac{\partial\mathbf{u}}{\partial t}=DK_u\mathbf{u}+\frac{1}{2}\frac{\partial\mathbf{\Phi}}{\partial t}, \tag{3.1}$$

$$M_\phi\frac{\partial\mathbf{\Phi}}{\partial t}=K_\phi\mathbf{\Phi}+F_\phi, \tag{3.2}$$

where the M_u,K_u,M_ϕ,K_ϕ are matrix whose entries are

$$\begin{aligned}
M_u(i,j)&=\int_\Omega N_i(\triangle_u)N_j(\triangle_u)dxdy,\\
K_u(i,j)&=\int_\Omega \nabla N_i(\triangle_u)\cdot\nabla N_j(\triangle_u)dxdy,\\
M_\phi(i,j)&=\int_\Omega \tau(\mathbf{n})N_i(\triangle_\phi)N_j(\triangle_\phi)dxdy,\\
K_\phi(i,j)&=\int_\Omega W(\phi_h)^2\nabla N_i(\triangle_\phi)\cdot\nabla N_j(\triangle_\phi)dxdy,
\end{aligned}$$

and F_ϕ is a vector whose i-th entry is:

$$(F_\phi)_i = \int_\Omega \Big[f(\phi_h,u_h;\lambda) N_i(\triangle_\phi) - |\nabla\phi_h|^2 W(\phi_h) \\ \times \Big(\frac{\partial W(\phi_h)}{\partial\phi_x}\frac{\partial N_i(\triangle_\phi)}{\partial x} + \frac{\partial W(\phi_h)}{\partial\phi_y}\frac{\partial N_i(\triangle_\phi)}{\partial y} \Big) \Big] dxdy,$$

for $1 \le i \le N_\phi$. The system (3.1)-(3.2) turns out to be a nonlinear ODE system since M_ϕ, K_ϕ and F_ϕ all depend on ϕ_h. Following [18], we adopt the alternating Crank-Nicholson scheme for the time discretization, which is actually composed of two steps of implicit Euler scheme. Denote the time step length to be δt. Then the scheme can be formulated by

$$(M_\phi(\phi_h^{(i)}) + \delta t K_\phi(\phi_h^{(i)}))\mathbf{\Phi}^{(i+1)} = M_\phi(\phi_h^{(i)})\mathbf{\Phi}^{(i)} + \delta t F_\phi(\phi_h^{(i)}), \tag{3.3}$$

$$(M_u + \delta t D K_u)\mathbf{u}^{(i+1)} = M_u \left(\mathbf{u}^{(i)} + \frac{1}{2}\Big(\mathbf{\Phi}^{(i+1)} + \mathbf{\Phi}^{(i)} \Big) \right), \tag{3.4}$$

In [15], Eq. (3.2) is solved by the forward-difference (explicit) scheme and Eq. (3.1) is discretized by the Crank-Nicholson scheme. It is known that the explicit scheme is more efficient than the implicit one but the former is stable only with sufficiently small time steps. In our implementation, the implicit method is used for both (3.3) and (3.4), since the multigrid solver for the resulting linear system is used which can speed up the solution procedure. Furthermore, it is observed that the implicit scheme used here for time stepping is equivalent to second-order Crank-Nicolson scheme in terms of accuracy, but its cost is similar to that of the implicit Euler scheme, see also [6, 18]. To have a better understanding of our scheme, below we list the flow-chart of the algorithm.

```
Algorithm 1:  Multi-mesh adaptive FEM for the phase field model.
Prepare the background mesh $\triangle_0$ for $\Omega$;
Set the initial value for $\phi$ and $u$ and obtain the initial mesh for $\triangle_\phi$ and $\triangle_u$;
while $t < T$ do
    Solve (3.3) and (3.4);
    $t = t + \delta t$;
    if the meshes are not updated after $m$ (fixed constant) steps then
        Update mesh $\triangle_\phi$ and $\phi_h$;
        Update mesh $\triangle_u$ and $u_h$;
    end
end
```

The right hand-sides of (3.3) and (3.4) are the coupled terms of the two variables defined on different meshes, so the calculation of those terms is a key point in solving (3.3)-(3.4). We use the same algorithm proposed in [12], which takes the full advantages of hierarchical mesh refinement algorithm and the tree type data structure for adaptive meshes, see also [11]. The main idea there is to use the fact that any intersection of two elements from such two meshes is either empty or a simplex. Under such observation, the interpolation for ϕ and u between the two meshes costs almost nothing. For more details about the implementation, we refer to [12]. The resulting linear systems are solved by the algebraic multigrid method [7,8], which is implemented in [13]. In our simulations, 3 or 4 multigrid iterations are always enough for solving each linear system.

3.2 The mesh adaptation

A reasonable error indicator is important for the mesh adaptation. There are several choices to construct error indicators and many works have been devoted to this subject, among which [17] is a useful reference. It is known that the patch recovery technique [22] is one of the most commonly adopted techniques in practice, see also [15], while the residual type error estimator is simple and convenient to use. The latter one is based on the gradient jump of finite element solutions on the interface of adjacent elements. In [16], an explicit L_2 error estimate for the associated elliptic problem of the thermal field equation is utilized. In our work, the jumps of finite element solution on the interfaces are used for both ϕ and u. For example, if u_h is the finite element solution for the thermal field equation then its heuristic error indicator can be chosen as

$$\eta_T(u_h)=\left(\sum_{e\in\partial T}\int_e h^3|[\![\nabla u_h\cdot n_e]\!]|^2 d\Gamma\right)^{1/2},$$

where $[\![\cdot]\!]$ denotes the jump on the interface and h is the length of edge e. Actually, the above formulation is a residual-based explicit L_2 error estimator for the Laplacian operator as given in [17], and the inner residual is vanished since we use linear finite element spaces here. It is pointed out that the error estimators are generally not accurate enough although it can indicate the error magnitudes. In our simulations, the simple error estimator mentioned above is found effective for both ϕ and u.

The strategy for mesh refinement and coarsening is another important aspect in the adaptive mesh algorithms. For single variable problems, there are some effective strategy such as the fixed threshold rule and the equi-distribution principle [17]. The fixed threshold rule is easy and stable in many cases, which is to equi-distribute the errors on each element by ensuring the element error indicators η_T satisfying

$$\underline{\theta}\cdot tol\le\eta_T\le\bar{\theta}\cdot tol,$$

where tol is the prescribed tolerance and $0<\bar{\theta},\ \underline{\theta}<1$ are two constants. As a result, it is sufficient to refine the elements where $\eta_T>\bar{\theta}$ and to coarse the elements where $\eta_T<\underline{\theta}$. In practice, the tolerance is often chosen as

$$tol=\frac{\eta}{\sqrt{N}}=\left(\sum_{t\in\triangle}\eta_T^2/N\right)^{1/2}.$$

In fact, we observe that the choice of tolerance is not so sensitive to the convergence of the adaptive algorithms. However, it can affect the density of elements at the interface. Thus one can control the minimal element size near the interface by adjusting the tolerance to a certain value. The numerical tests show that a tolerance between 0.001$\sim$0.05 is reasonable for ϕ and u.

The problems involving two or more dependent variables are more complex when the variables are of different regularity behaviors. One feasible way is to define an composite field as proposed in [15]:

$$\psi=\phi+\gamma u,$$

where $\gamma>0$ is a constant representing the weight of the contribution from u on the mesh adaptation. The strategy to find the parameter γ is highly artificial. In [15], $\gamma=0$ to $\gamma=4$ are tested; but both variables can be over-refined or over-coarsened in certain parts of the solution domain due to the different regularity behaviors in the solutions. The adaptive mesh will generate additional transient region from the interface of ϕ to outside when $\gamma>0$. As pointed out in [15], this is due to the fact that the main physics of solidification only occurs around the interface whose area is much smaller than the whole domain. Near the interface, ϕ varies significantly but becomes constant away from it; while the thermal field u extends well beyond the interface varying much more gradually, which permits a much coarser mesh to resolve it.

With the use of our multi-mesh algorithm, it is found that the number of elements for $\triangle_u$ is much less than that for $\triangle_\phi$. In particular, when the topology of the interfaces is more complicated, the ratio of number of elements for $\triangle_u$ to that of $\triangle_\phi$ is even smaller. As for $\triangle_\phi$, the number of elements can also be reduced slightly due to the lack of influence from u.

4 Numerical simulations

In this section, we present the numerical results for both two- and three-dimensional simulations. All computations are carried out on a Dell Precision 490 with core speed 3.0G. The initial uniform meshes in 2D are generated by software Easymesh [14]. We will make full use of the symmetries of the problems in our implementation.

4.1 Two-dimensional validations: simple structure

Several two-dimensional benchmark results for dendritic growth are available in [10,15]. We take the same parameters as theirs for comparison. The following two cases are used to validate our multi-mesh algorithm by verifying relevant known properties such as the velocity of the steady state tips under different undercoolings. For high undercooling cases, the triangle domain Ω_c with vertices (0,0), (400,0) and (200,200) is employed for computation instead of using the whole domain with $L_x=L_y=800$ (due to the symmetry of the solution). The initial mesh for both ϕ and u is a uniform mesh obtained by globally refining Ω_c five times. The initial phase field function is given by

$$\phi_0(\mathbf{x}) = -\tanh\left(\frac{|\mathbf{x}|-R_0}{\sqrt{2}}\right),$$

where R_0 is set as a small number (relative to the domain size 800), say $R_0=3$, to approximate a very small core of the initial solidification seed. The initial temperature decays exponentially from $u=0$ at interface to $-\Delta$ as $\mathbf{x}\to\infty$. Fig. 1 presents some typical dendritic tip shapes and corresponding meshes for ϕ under different high undercoolings. The shapes of the dendritic tips in these figures agree well with those given by [10]. It is also observed that the adapted meshes for ϕ can capture the interface accurately.

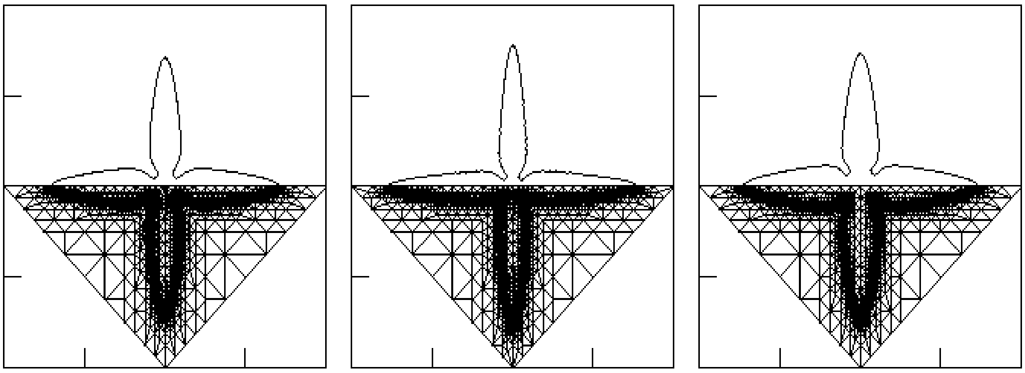

Figure 1: Typical shapes of the dendritic tips with the domain size $[-800,800]\times[-800,800]$ and with $\Delta=0.45$, 0.55 and 0.65 (from left to right). The contours of $\phi=0$ are shown on the top and the corresponding meshes for ϕ lie in the bottom. In all cases, $\epsilon=0.05$.

The tip velocity is an important quantity which can be used to validate the numerical results, where the value of tip velocity is rescaled by

$$\widetilde{V}_{tip}=\frac{V_{tip}d_0}{D}.$$

As in [10, 15], we plot in Fig. 2 the numerical and theoretical tip velocities under high undercoolings, where the theoretical solvability constants are given in [10]. It is observed that there is a good agreement between our numerical solutions and those predicated by the solvability theory for high undercooling cases. For low undercoolings, it is observed in [15] that the domain size can affect the dendritic growth significantly. In our computations, a quite large domain of size 6400×400 is used for cases of $\Delta=0.25$ and $\Delta=0.3$. It is noted that the size in the y-direction is much smaller than that in the x-direction. As in low undercooling cases, the size of domain can affect the crystal shape significantly, and the use of a domain with small y-x ratio can avoid considering the influence between the two directions. In other words, the small ratio simplifies the study of the 2D dendritic

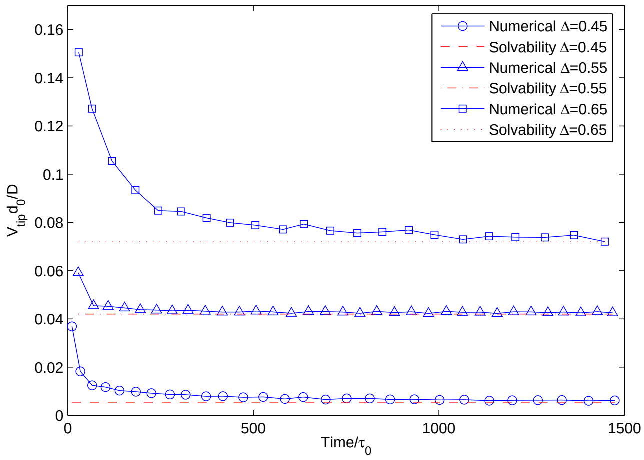

Figure 2: The time evolution of the tip-velocity for high undercoolings $\Delta = 0.45, 0.55, 0.65$. The data for $\Delta = 0.55$ and 0.65 have been shifted up by 0.025 for clarity. $\epsilon = 0.05$ in all cases.

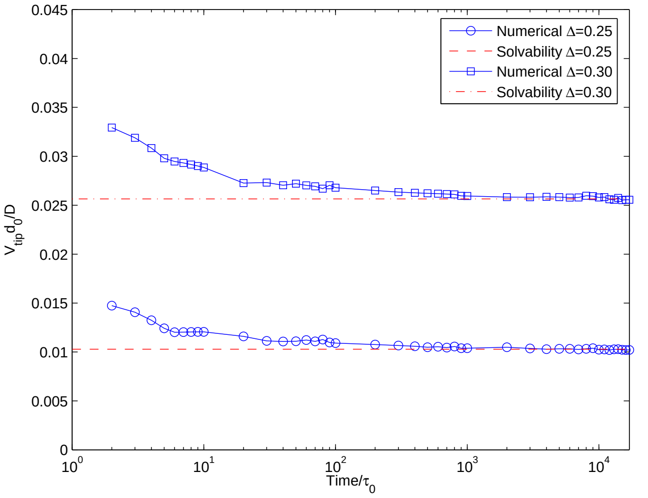

Figure 3: The time evolution of the tip-velocity for low undercoolings $\Delta = 0.25$ and 0.3. The data are shifted up by 0.01 for case $\Delta = 0.25$ and 0.025 for case $\Delta = 0.3$. $\epsilon = 0.05$ in both cases.

shape to some variations in one direction. The value of λ in (2.7) is suitably chosen to correspond the cases of $\beta = 0$ and $D = 13$. A proper tolerance for mesh adaptation is used to ensure that the minimal grid size is about 0.8, by which we can resolve the interface as demonstrated in [10]. Smaller minimal grid sizes are also tested, and the resulting interfaces are graphically indistinguishable with those obtained by using the minimal grid size 0.8. In Fig. 3, we show the time evolution of the tip-velocity for the low undercooling cases. It is observed that initially the tip velocity is larger than that predicated by the solvability theory but approaches to the theoretical predications with the time evolution. This phenomena is also reported in [15].

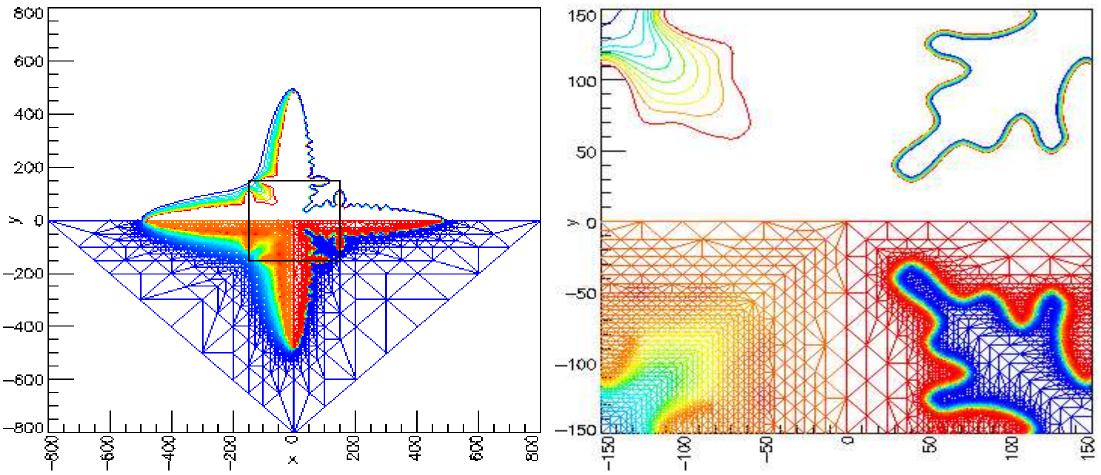

Figure 4: The simulation result with $\Delta = 0.70$, $D = 2$, $\Delta t = 0.1$ and $\epsilon = 0.05$. Left: clock-wisely from upper-left is the contour of u, contour of $\phi = 0$, $\triangle_\phi$ and $\triangle_u$ at $T = 1000$. Right: zoomed part inside the marked box in the left figure.

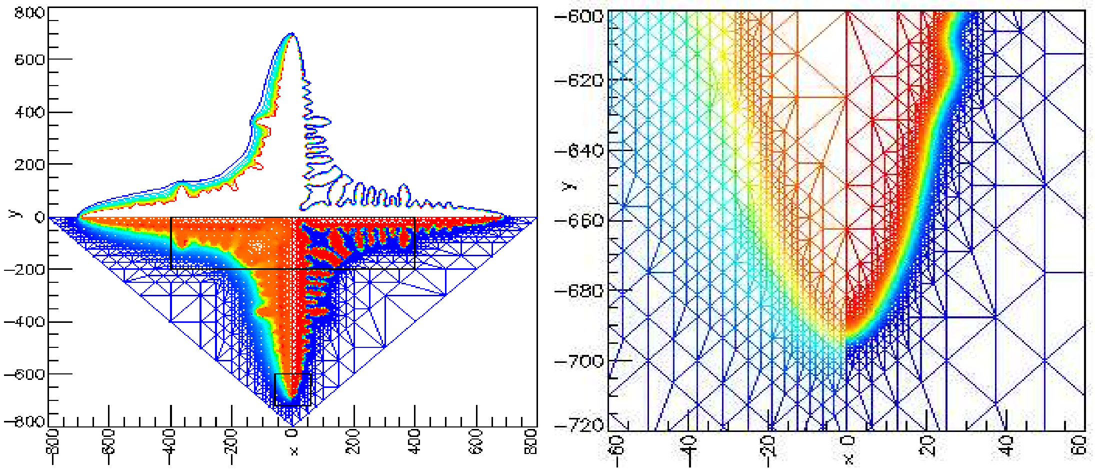

Figure 5: Left: the adapted meshes (bottom) and the solution profiles (top) for the dendritic structures with $\Delta = 0.65$, $D = 3$ and $\epsilon = 0.05$ at $T = 1000$. Right: zoomed part at the bottom of the left figure.

4.2 Two-dimensional validations: complex structures

Now we present two more examples to demonstrate the advantages of the multi-mesh approach. In the first example, the parameters used are $\Delta = 0.70$, $D = 2$, $\Delta t = 0.1$, $\epsilon = 0.05$ and λ is still chosen to make $\beta = 0$. The tolerance for mesh adaptation is also chosen to ensure that the smallest element size for $\triangle_\phi$ is approximately 0.8. Fig. 4 shows the solution contours and the corresponding meshes for both ϕ and u at 10^4 time steps (i.e., $T = 1000$). It is obvious that $\triangle_\phi$ is much dense than $\triangle_u$, which is easily seen from the right figure of Fig. 4. In fact, the ratio of the numbers of element $\triangle_\phi$ to that of $\triangle_u$ is about

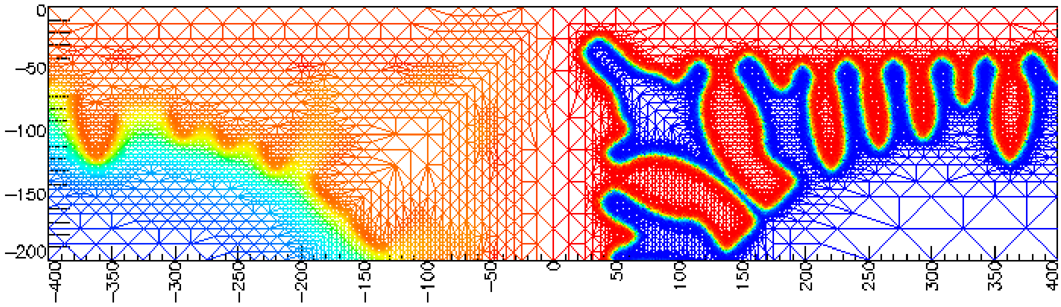

Figure 6: Meshes for u (left) and ϕ (right) which are given in the center boxing area in Fig. 5.

10 : 1. Since a great deal of elements are saved, the present simulation costs only about 3 hours on the Desktop computer.

The final 2D example is to consider a more complex dendritic structure, with $\Delta = 0.65$ and $D = 3$. In Fig. 5, we plot the adapted meshes for ϕ and u together with their solution profiles; while in Fig. 6, a zoomed part in Fig. 5 is plotted. It is observed from both figures that the structure of the dendritic growth becomes very complex. As a result, the configurations of $\triangle_\phi$ and $\triangle_u$ are very different. It is obvious that with a single mesh much more elements are needed to resolve such configurations. In this case, to reach the same accuracy the multi-mesh approach requires at most half of the CPU times needed for a single mesh.

4.3 Three-dimensional simulations

The three-dimensional simulations are more challenging, partly due to the need of a large storage. Thus there have been very few three-dimensional results available in the literature. Our simulations are performed on a tetrahedron domain with vertices (0,0,0), (300,0,0), (150,150,0) and (100,100,100), which utilizes the solution symmetry. The initial meshes are obtained by using 5 uniform refinements from the background mesh with only one element. Since the anisotropic cases are of the major interests in the present work, the parameter ϵ in (2.10) is always chosen as 0.05. We perform the simulations with several groups of other physical parameters.

The first case corresponds to $\Delta = 0.55$, $D = 1$, and the resulting dendritic shapes at three different time steps are presented in Fig. 7. In this case, the shape of the crystal is relatively simple, so that the degree of freedoms (DOFs) used for u becomes almost a constant after $t \approx 200$, see Fig. 10(a). When D becomes larger, more complex structure are presented, so that $D = 1.4$ is used the next two simulations. The results for three different time levels are shown in Fig. 8 for $\Delta = 0.45$ and in Fig. 9 for $\Delta = 0.55$. In both simulations, the side-branchings are observed which agrees well with the observation in [15]. It is also noted that the shapes are also very similar in both cases, but the case $\Delta = 0.55$ yields much faster growing rate. It seems that D dominates the complexity of the shapes and Δ affects the tip shape and velocity significantly.

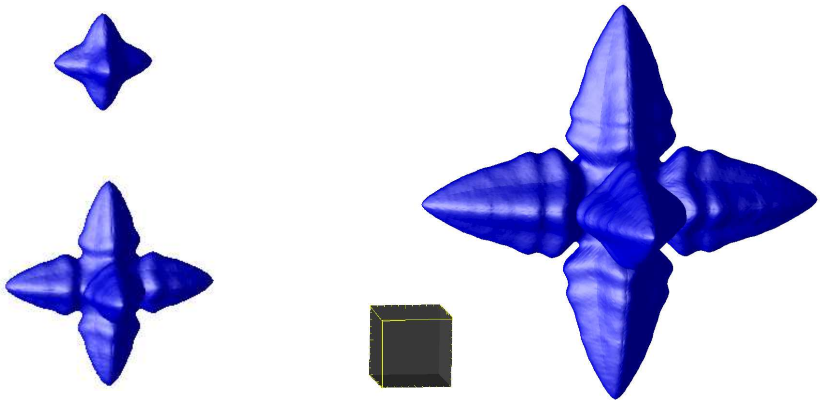

Figure 7: The patterns with $\Delta = 0.55, D = 1, \Delta = 0.1$ and $\epsilon = 0.05$ at $T = 50$ (top left), 130 (bottom left) and 200 (right). The coordinate box is sized $100 \times 100 \times 100$.

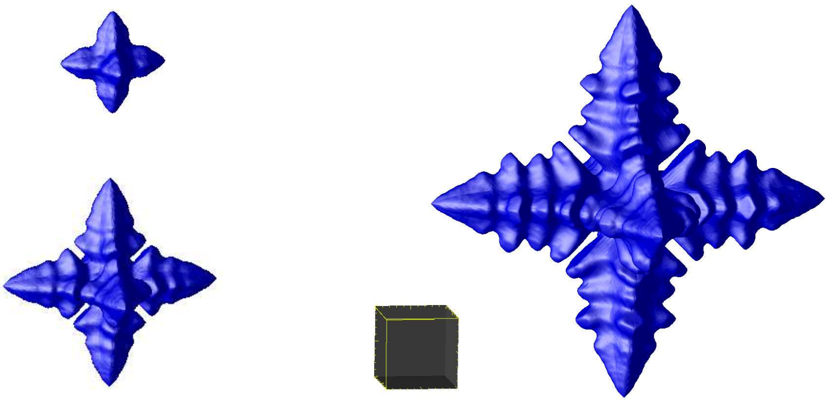

Figure 8: The dentritic patterns for $\Delta = 0.45$, $D = 1.4$ and $\epsilon = 0.05$ at $T = 50$ (top left), 110 (bottom left) and 160 (right). The time-step is $\Delta = 0.1$ and the coordinate box is sized $100 \times 100 \times 100$.

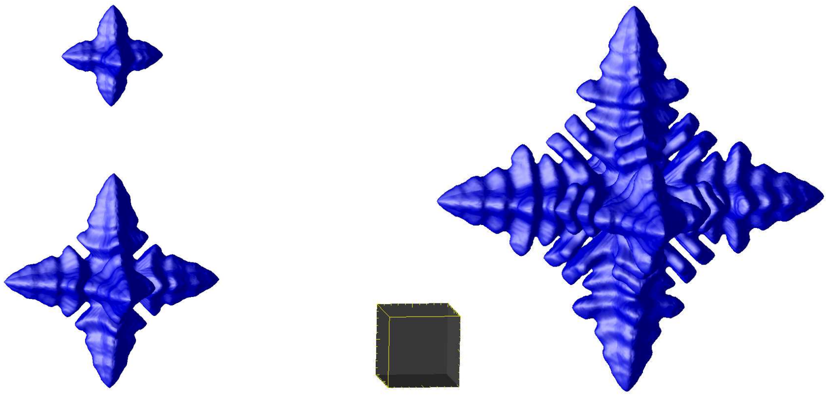

Figure 9: The dentritic patterns for $\Delta = 0.55$, $D = 1.4$ and $\epsilon = 0.05$ at $T = 26$ (top left), 52 (bottom left) and 78 (right). The time-step is $\Delta = 0.1$ and the coordinate box is sized $100 \times 100 \times 100$.

All the above simulations cost less than 5 hours each using our desktop computers. To show the desired features of the multi-mesh algorithm, we further compare the DOFs needed for ϕ and u in Fig. 10. As expected, initially both ϕ and u use similar number of DOFs since they start from the same initial mesh. This is necessary since the resolution of u should be high enough to catch the temperature distribution around the tiny initial core. As the dendrites grow, the costs for ϕ increases almost exponentially, while the cost for u keeps almost a constant after some integration time. This is due to the mild regularity of u. On the other hand, the number of unknowns for ϕ becomes large due to the fast development of the phase interfaces. In fact, when the dentritic structures become more complicated, our multi-mesh algorithm is more useful due to the storage and CPU savings in computing u.

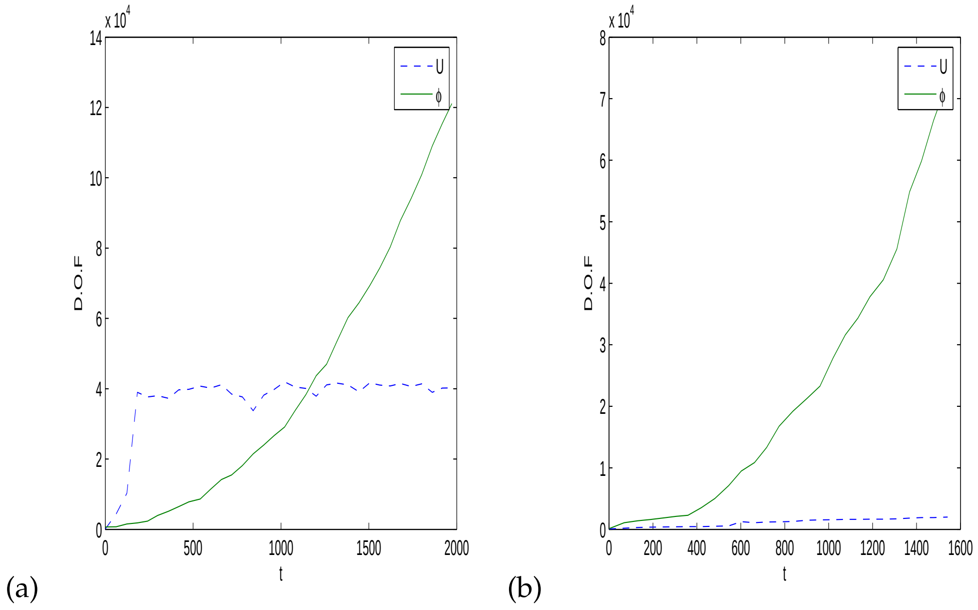

Figure 10: Degree of freedoms for ϕ and u with (a): $\Delta=0.55$, $D=1$, and (b): $\Delta=0.45$, $D=1.4$.

It is observed from Fig. 10 that the DOF of temperature becomes almost a constant after some time. The dendrite is still growing, so the interface becomes larger and larger. It may be of concern that the smaller portion of DOF for the temperature may cause the lose of accuracy. This turns out not be a problem since the temperature changes much more slowly than the phase variable. As the variation for the temperature field is slow, the small number of DOFs used for u is found sufficient in obtaining reasonable resolution even after a long integration time.

Several other types of dendritic pattern can be obtained by using different parameters. As an example, Fig. 11 presents a typical complex pattern obtained with $\Delta=0.45$ and $D=3.5$. It is also observed that the ratio of the DOFs used for ϕ and u is even beyond 30:1 for this test case.

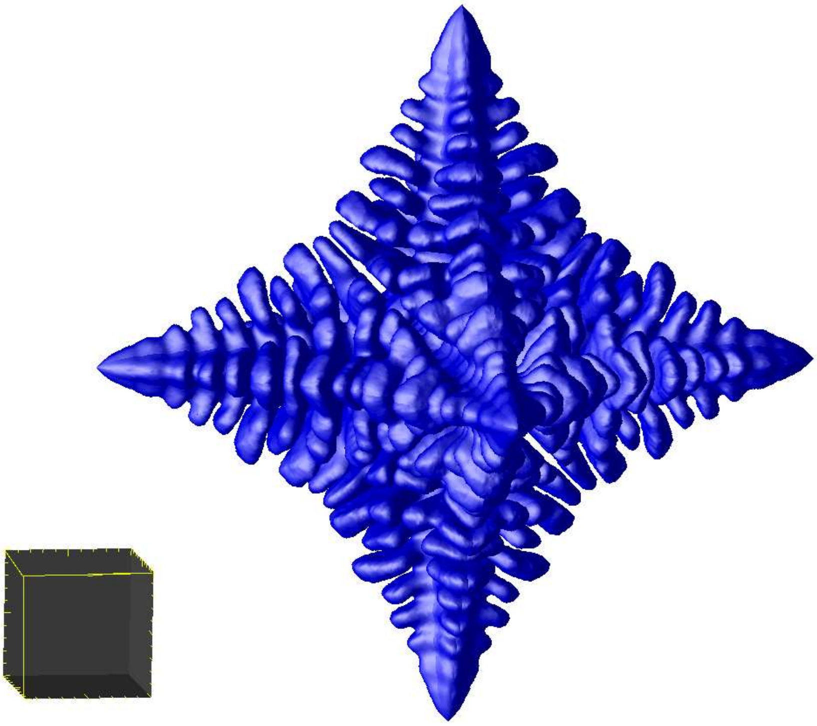

Figure 11: A complex 3D pattern with $\Delta = 0.45$, $D = 3.5$ and $\epsilon = 0.05$, which is compared to the coordinate box sized $100 \times 100 \times 100$.

5 Concluding remarks

In this work, we have developed a multi-mesh adaptive finite element method to simulate the crystallization of a pure melt governed by the anisotropic phase field model. The thermal field variable and the phase field variable are approximated on different meshes. As a result, the computational cost can be saved significantly as different solution behaviors of the dependent variables are utilized. The numerical results of the tip velocity in two space dimensions are in good agreement with the predictions of the solvability theory, both for the high and low undercooling cases. It is demonstrated that our algorithm can be used to simulate very complex dendritic structures in both two- and three-dimensions. Our next step is to simulate multi-component alloy crystallization and large steel ingot solidification by using the method proposed in this work.

Acknowledgments

Part of Hu's research was carried out while visiting Hong Kong Baptist University. His research was also supported by an National Basic Research Program of China under the grant 2005CB32170. Li's research was partially supported by the National Basic Research

Program of China under the grant 2005CB321701, Foundation for National Excellent Doctoral Dissertation Award of China and the Joint Applied Mathematics Research Institute between Peking University and Hong Kong Baptist University. Tang's research was supported by CERG Grants of Hong Kong Research Grant Council and FRG grants of Hong Kong Baptist University.

References

[1] R. Almgrem. Variational algorithms and pattern formation in dendritic solidification. *J. Comput. Phys.*, 106:337, 1993.

[2] G. Beckett, J. A. Mackenzie, and M. L. Robertson. An r-adaptive finite element method for the solution of the two-dimensional phase-field equations. *Commun. Comput. Phys.*, 1:805–826, 2006.

[3] G. Caginalp and X. Chen. On the evolution of phase boundaries. In M. E. Gurtin and G. B. McFadden, editors, *The IMA volumes in mathematics and its applications*, page 1. Springer-Verlag, New York, 1992.

[4] J. B. Collins and H. Levine. Diffuse interface model of diffusion-limited crystal growth. *Phys. Rev. B*, 31:6119–6122, 1985.

[5] Y. Di, R. Li, and T. Tang. A general moving mesh framework in 3D and its application for simulating the mixture of multi-phase flows. *Commun. Comput. Phys.*, 3:582–602, 2008.

[6] Y. Di, T. Tang R. Li, and P.-W. Zhang. Moving mesh finite element methods for the incompressible navier-stokes equations. *SIAM J Sci. Comput.*, 26:1036–1056, 2005.

[7] A. J. Cleary etc. Robustness and scalability of algebraic multigrid. *SIAM J. Sci. Comput.*, 21(5):1886–1908, 1999.

[8] M. Brezina etc. Algebraic multigrid based on element interpolation (AMGe). *SIAM J. Sci. Comput.*, 22(5):1570–1592, 2000.

[9] A. Karma and W.-H. Rappel. Phase-field method for computationally efficient modeling of solidification with arbitrary interface kinetics. *Phys. Rev. E*, 53(4):3017–3020, 1996.

[10] A. Karma and W.-J. Rappel. Quantitative phase-field modeling of dendritic growth in two and three dimensions. *Phys. Rev. E*, 57:4323–4349, 1998.

[11] A. Khamayseh and G. Hansen. Use of the spatial kD-tree in computational physics applications. *Commun. Comput. Phys.*, 2:545–576, 2007.

[12] R. Li. On multi-mesh h-adaptive algorithm. *J. Sci. Comput.*, 24(3):321–341, 2005.

[13] R. Li and W. B. Liu. `http://circus.math.pku.edu.cn/AFEPack`.

[14] B. Niceno. `http://www-dinma.univ.trieste.it/nirftc/research/easymesh/`.

[15] N. Provatas, N. Goldenfeld, and J. Dantzig. Adaptive mesh refinement of solidification microstructures using dynamic data structures. *J. Comput. Phys.*, 148:265–290, 1999.

[16] A. Schmidt. Computation of three dimensional dendrites with finite elements. *J. Comput. Phys.*, 125:293–312, 1996.

[17] R. Verfürth. *A Review of A Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques*. Wiley-Teubner, 1996.

[18] H.-Y. Wang and R. Li. Mesh sensitivity for numerical solutions of phase-field equations using r-adaptive finite element methods. *Commun. Comput. Phys.*, 3:357–375, 2008.

[19] H.-Y. Wang, R. Li, and T. Tang. Efficient computation of dentritic growth with r-adaptive finite element methods. *J. Comput. Phys.*, page Accepted, 2008.

[20] S.-L. Wang and R. F. Sekerka. Algorithms for phase field computation of the dendritic operating state at large supercoolings. *J. Comput. Phys.*, 127:110–117, 1996.

[21] S.-L. Wang and R. F. Sekerka. Computation of the dendritic operating state at large supercoolings by the phase field model. *Phys. Rev. E*, 53:3760–3776, 1996.

[22] O. C. Zienkiewicz and J. Z. Zhu. A simple error estimator and adaptive procedure for practical engineering analysis. *Int. J. Numer. Meth. Eng.*, 24:337–357, 1987.