COMMUNICATIONS IN COMPUTATIONAL PHYSICS
Vol. **3**, No. 2, pp. 357-375

Commun. Comput. Phys.
February 2008

Mesh Sensitivity for Numerical Solutions of Phase-Field Equations Using r-Adaptive Finite Element Methods

Heyu Wang[1,2,*] and Ruo Li[3]

[1] *College of Computer Science, Zhejiang University, Hangzhou 310027, China.*
[2] *Department of Mathematics, Hong Kong Baptist University, Kowloon Tong, Hong Kong.*
[3] *LMAM and School of Mathematical Sciences, Peking University, Beijing 100871, China.*

Received 18 January 2007; Accepted (in revised version) 25 May 2007

Available online 27 September 2007

Abstract. There have been several recent papers on developing moving mesh methods for solving phase-field equations. However, it is observed that some of these moving mesh solutions are essentially different from the solutions on very fine fixed meshes. One of the purposes of this paper is to understand the reason for the differences. We carried out numerical sensitivity studies systematically in this paper and it can be concluded that for the phase-field equations, the numerical solutions are very sensitive to the starting mesh and the monitor function. As a separate issue, an efficient alternating Crank-Nicolson time discretization scheme is developed for solving the nonlinear system resulting from a finite element approximation to the phase-field equations.

AMS subject classifications: 65M20, 65M50, 65M60, 80A22

Key words: Phase-field equations, moving mesh method, Crank-Nicolson scheme, numerical sensitivity.

1 Introduction

Numerical methods have been proposed to resolve phase change interface between the solid and liquid regions. To avoid the calculation of the position and curvature of the interface, an alternative is to use a so-called diffuse interface model that implicitly defines the position of the interface, see, e.g., [3]. In this model, a phase indicator parameter p is assumed to be smooth on the whole solution domain, which has distinctive values in solid and liquid. With this idea and using the Ginzburg-Landau theory, the phase-field

*Corresponding author. *Email addresses:* hywang@math.hkbu.edu.hk (H. Wang), rli@math.pku.edu.cn (R. Li)

 ©2008 Global-Science Press

equations can be deduced by requiring that the temperature and phase-field evolve such that the free energy decreases [11].

Most numerical methods to solve the phase-field equations have used stationary uniform meshes, see, e.g., [4, 5, 10, 26]. However, it is important that the diffused 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 approach must involve some form of mesh adaptivity. There have been two approaches in doing this. One is to use the local mesh refinement method, i.e. h-method, see, e.g., [2, 21–23]. The other is to use moving mesh method which is simpler in implementation and able to resolve the structures as the phase interface with highly anisotropic mesh grids [1, 16, 17, 24].

Recently, Beckett et al. [1] developed a moving mesh strategy for two-dimensional phase-field equations. Their computational mesh was obtained by equidistributing a monitor function tailored for the functional variation of the phase-field in the interfacial region. The same problems were also computed by Tan et al. [24] using the moving mesh finite volume methods. For the solidification of a single solid sphere which is surrounded by uncooled liquid, although the moving mesh results of [1,24] on the radial positions are in quite good agreement with each other, it is found that they are qualitatively different from the (very fine) uniform mesh results given by Elliott and Gardiner [9]. In the Elliott and Gardiner's model, the parameter of the diffuse interface thickness ϵ is taken as $1/80=0.0125$, while in [1, 24], ε is taken as $1/(160\sqrt{2})\approx 0.0044$. The smaller value of ε has the impact that very fine meshes have to be used in order to resolve the very small transition interfaces. Nevertheless, it is found that the radial position is in fact quite insensitive to the choice of the parameter ε. Therefore the differences between the solutions from different authors should be due to some other reasons, such as the numerical methods adopted.

In this work, we try to understand the reason why the results of [1,24] have unreasonable differences from the results obtained on uniform meshes. From the references, one can find that the possible factors leading to the differences among the numerical results therein are relevant to the starting mesh, the monitor function and the time integrating scheme. We first make a numerical convergence study to reveal the fact that the radial position is fairly insensitive to the choice of the parameter ε by computing for both parameters on a sequence of refined uniform meshes. Then we keep the same parameters ε and $p_\pm$ as [1, 24] and carry out a sequence of computations using different starting meshes, monitor functions and time integrating schemes. The numerical evidences demonstrated that among these three possible factors, the variation in time integrating scheme contributes only slight differences to the numerical solutions, while the other two are on the very contrary. It can be imagined that an inappropriate monitor function will introduce additional error to the numerical solutions, but it is such a surprising fact that the starting mesh can have a similar effect for this problem. Ideally, the moving mesh solutions should not be dependent on the choice of the starting mesh, since the starting mesh will be adapted immediately based on the initial values after the computations begin. However, the numerical results showed that the starting mesh may affect the phase-field

solutions essentially, in particular when the initial mesh is not fine enough to resolve the sharp interface.

As an appendant contribution, an efficient time discretization scheme is proposed for solving nonlinear phase-field equations. The two phase-field equations are nonlinear and coupled. Typically, a fully implicit scheme [1] or a semi-implicit scheme [24] is used to solve the ODE systems resulting from the spatial discretization (method of line). Then Newton-iteration will be applied to find the solution of the resulting nonlinear system. In this case, the Jacobian matrix is non-symmetric which adds additional difficulties in finding solutions efficiently. To speed up the solution process, we propose in this work an alternating Crank-Nicolson scheme to handle the phase-field equations. Applying the alternating Crank-Nicolson scheme to the phase-field equations leads to two *decoupled* algebraic subsystems, one is linear and another is nonlinear. Both have nice structures and can be solved by efficient matrix iteration methods. Our approach is motivated by earlier work of Mu and Huang [18, 19] who used a similar idea to solve the Ginzburg-Landau models.

The layout of this paper is as follows. In Section 2 we describe the phase-field model in 2D. The numerical methods consisting of the alternating Crank-Nicolson scheme and moving mesh techniques is presented for phase-field equation. In Section 3, numerical sensitivity studies for the solidification of a single solid sphere which is surrounded by uncooled liquid are carried out. The dependence of the numerical solutions on the starting mesh and the monitor function will be reported. Numerical results for solidification of two spheres will be illustrated in Section 4, to show the solution quality and efficiency of using the alternating Crank-Nicolson scheme as the time integrator. Some concluding remarks are drawn in the final section.

2 The phase-field equations and numerical methods

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

$$\begin{aligned}
&\rho cT_t=k\triangle T, && x\in\Omega^+(t)\cup\Omega^-(t),\\
&\rho lv=k[\nabla T\cdot\mathbf{n}]_+^-, && x\in\Gamma(t),\\
&T-T_m=-\frac{\sigma}{[s]_m}\kappa-\frac{\alpha\sigma}{[s]_m}v, && x\in\Gamma(t).
\end{aligned}\tag{2.1}$$

Here T_m is the equilibrium melting temperature, l is the latent 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 normalization used here), v is the normal velocity of the interface, κ is the sum of the principle 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 under-cooling 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 the system (2.1) in the following dimensionless form

$$\begin{aligned}&\theta_t=D\triangle\theta, && x\in\Omega^+(t)\cup\Omega^-(t),\\ &v=D[\nabla\theta\cdot\mathbf{n}]_+^-, && x\in\Gamma(t)\\ &\theta=-d_0\kappa-\alpha d_0 v, && x\in\Gamma(t).\end{aligned} \tag{2.2}$$

Using a scaling introduced in [3] we consider the phase-field model

$$\begin{aligned}&\theta_t+\frac{1}{2}p_t=D\triangle\theta,\\ &\alpha\varepsilon^2 p_t=\varepsilon^2\triangle p+\frac{1}{2}(p-p^3)+\frac{\varepsilon}{3d_0}\theta,\end{aligned} \tag{2.3}$$

where ε is a measure of the diffuse interface thickness. 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 largest (p_+) and smallest (p_-) roots of

$$f(p,\theta)=\frac{1}{2}(p_\pm-p_\pm^3)+\frac{\varepsilon}{3d_0}\theta_{\partial_\pm}=0. \tag{2.4}$$

The above requirement ensures that there is no mass flow out of the system (see, e.g. [4, 5]). Then the two phases are characterized by p taking values close to p_+ and p_- in each phase. In this work, the Neumann boundary conditions

$$\frac{\partial p}{\partial\mathbf{n}}=0,\quad \frac{\partial\theta}{\partial\mathbf{n}}=0$$

will be imposed on $\partial\Omega$.

We solve the phase-field equations using finite element discretization. At first, we triangulate the domain Ω into a triangle mesh $\mathcal{T}_h$ with N nodes. Let $V_h\subset H_0^1(\Omega)$ be the piecewise linear finite element space with basis functions ϕ_i, $i=1,\cdots,N$. Moreover, assume that $[0,T_f]$ is partitioned by $0=t^0<t^1<\cdots<t^{N_t}=T_f$ with a constant time stepping, i.e., $\Delta t=t^{n+1}-t^n$. We also denote $t^{n+\frac{1}{2}}=(t^{n+1}-t^n)/2$.

For solidification and melting problems the liquid and solid regions change continuously with time. To model this situation we use finite elements that continuously move and deform. We will assume that, local to an element, the finite element approximations at time t can be expressed in the form

$$p_h(x,y,t)=\sum_{i=1}^N p_i(t)\phi_i(x,y), \tag{2.5a}$$

$$\theta_h(x,y,t)=\sum_{i=1}^N \theta_i(t)\phi_i(x,y), \tag{2.5b}$$

where $p_i(t)$ and $\theta_i(t)$ denote values for p and θ at the node i. With a standard finite element approach, the weak formulation for the phase-field equation (2.3) is given by

$$\begin{aligned}&\left(\dot{\theta}_h+\frac{1}{2}\dot{p}_h,\phi_j\right)-\left(D\triangle\theta_h,\phi_j\right)=0\\&\left(\alpha\varepsilon^2\dot{p}_h,\phi_j\right)-\left(\varepsilon^2\triangle p_h,\phi_j\right)-\left(\frac{1}{2}(p_h-p_h^3),\phi_j\right)-\left(\frac{\varepsilon}{3d_0}\theta_h,\phi_j\right)=0,\end{aligned}\tag{2.6}$$

for $j=1,\cdots,N$, where $(\cdot,\cdot)$ is the L_2 inner product over Ω. Denote

$$P=(p_1,p_2,\cdots,p_N)^T,\quad \Theta=(\theta_1,\theta_2,\cdots,\theta_N)^T.$$

Using (2.5) and integration by parts, we can obtain from (2.6) the block system

$$\begin{bmatrix} M & \frac{1}{2}M \\ 0 & \alpha\varepsilon^2 M \end{bmatrix}\begin{bmatrix} \dot{\Theta} \\ \dot{P} \end{bmatrix}+\begin{bmatrix} DK & 0 \\ -\frac{\varepsilon}{3d_0}M & \varepsilon^2 K \end{bmatrix}\begin{bmatrix} \Theta \\ P \end{bmatrix}+\begin{bmatrix} 0 \\ \mathbf{f}(P) \end{bmatrix}=0,\tag{2.7}$$

where the $N\times N$ matrix M and K are the usual mass matrix and the stiffness matrix with their (i,j) entries have the form like

$$M_{ij}=(\phi_i,\phi_j),\qquad K_{ij}=(\nabla\phi_i,\nabla\phi_j).$$

With a product approximation, the nonlinear term $\mathbf{f}$ in (2.7) has the form

$$\mathbf{f}(P)=-\frac{1}{2}(P-P^3)M.\tag{2.8}$$

The nonlinear ODE system (2.7) thus can be solved by a standard ODE solver. Here we present a new ODE solver deliberately designed for this system with second-order accuracy. As a temporal discretization with second-order accuracy, the alternating Crank-Nicolson scheme presented below is a quite efficient one. To enhance the numerical efficiency, it is useful to decouple $\dot{\Theta}$ and $\dot{P}$. This can be done by using a backward substitution in (2.7), which gives

$$\alpha\varepsilon^2\begin{bmatrix} M & 0 \\ 0 & M \end{bmatrix}\begin{bmatrix} \dot{\Theta} \\ \dot{P} \end{bmatrix}+\begin{bmatrix} \alpha\varepsilon^2 D\cdot K+\frac{\varepsilon}{6d_0}M & -\frac{\varepsilon^2}{2}K \\ -\frac{\varepsilon}{3d_0}M & \varepsilon^2 K \end{bmatrix}\begin{bmatrix} \Theta \\ P \end{bmatrix}+\begin{bmatrix} -\frac{1}{2}\mathbf{f}(P) \\ \mathbf{f}(P) \end{bmatrix}=0.\tag{2.9}$$

With the form (2.9), we can use an alternating Crank-Nicolson scheme to obtain a second-order discretization with improved efficiency. To demonstrate the idea clearly, suppose we are given a system of the form

$$P_t=\rho_{11}(P)+\rho_{12}(\Theta),\tag{2.10a}$$

$$\Theta_t=\rho_{21}(P)+\rho_{22}(\Theta).\tag{2.10b}$$

The above ODE system is discretized as

$$\frac{P^{(n+1)}-P^{(n)}}{\Delta t}=\frac{\rho_{11}(P^{(n+1)})+\rho_{11}(P^{(n)})}{2}+\rho_{12}(\Theta^{(n+\frac{1}{2})}), \tag{2.11a}$$

$$\frac{\Theta^{(n+\frac{3}{2})}-\Theta^{(n+\frac{1}{2})}}{\Delta t}=\rho_{21}(P^{(n+1)})+\frac{\rho_{22}(\Theta^{(n+\frac{3}{2})})+\rho_{22}(\Theta^{(n+\frac{1}{2})})}{2}, \tag{2.11b}$$

for $n=0,\cdots,N_t$. Applying the alternating Crank-Nicolson scheme (2.11) to the system (2.9) gives

$$\begin{aligned}&\alpha\varepsilon^2 M\frac{\Theta^{(n+\frac{3}{2})}-\Theta^{(n+\frac{1}{2})}}{\Delta t}+\left(\alpha\varepsilon^2 D\cdot K+\frac{\varepsilon}{6d_0}M\right)\frac{\Theta^{(n+\frac{3}{2})}+\Theta^{(n+\frac{1}{2})}}{2}\\ &=\frac{\varepsilon^2}{2}KP^{(n+1)}-\frac{M}{4}[P^{(n+1)}-(P^{(n+1)})^3],\end{aligned} \tag{2.12}$$

$$\begin{aligned}&\alpha\varepsilon^2 M\frac{P^{(n+1)}-P^{(n)}}{\Delta t}+\frac{\varepsilon^2}{2}K(P^{(n+1)}+P^{(n)})\\ &-\frac{M}{4}[P^{(n+1)}-(P^{(n+1)})^3+P^{(n)}-(P^{(n)})^3]-\frac{\varepsilon}{3d_0}M\Theta^{(n+\frac{1}{2})}=0.\end{aligned} \tag{2.13}$$

Denote the left-hand side of (2.13) by $H(P^{(n+1)})$. For nonlinear system (2.12)-(2.13), we solve the nonlinear equations (2.13) first. After obtaining the updating value $P^{(n+1)}$, then we can solve the *linear* system (2.12). The cycle leads to a solution sequence $\{P^{(n)}\}_{n=0}^{N_t}$ and $\{\Theta^{(n+\frac{1}{2})}\}_{n=0}^{N_t}$.

The alternative approach described above solves a much smaller nonlinear system compared with the larger system solved by Beckett et al. [1] and Tan et al. [24]. For the small nonlinear system (2.13), we use Newton-iteration

$$JP^{(n+1,s+1)}=-H(P^{(n+1,s)}), \tag{2.14}$$

where

$$J=\frac{\partial H}{\partial P^{(n+1)}}=\frac{\alpha\varepsilon^2}{\Delta t}M+\frac{\varepsilon^2}{2}K-\frac{M}{4}+\frac{3}{4}M(P^{(n+1)})^2.$$

As the bootstrap of the computation, we need the value of $\Theta^{(\frac{1}{2})}$. This can be done by using a second-order approximation based on Taylor expansion:

$$\Theta^{(\frac{1}{2})}\approx\Theta^{(0)}+\frac{\Delta t}{2}\dot{\Theta}^{(0)}, \tag{2.15}$$

where the value of $\dot{\Theta}^{(0)}$ is given explicitly by using the initial conditions $\Theta^{(0)}$ and $P^{(0)}$ as well as the first equation in (2.9), i.e.,

$$\alpha\varepsilon^2 M\dot{\Theta}^{(0)}=-[\alpha\varepsilon^2 D\cdot K+\frac{\varepsilon}{6d_0}M]\Theta^{(0)}+\frac{\varepsilon^2}{2}KP^{(0)}-\frac{M}{4}[P^{(0)}-(P^{(0)})^3]. \tag{2.16}$$

The moving mesh schemes used in this paper is following the method proposed in [8, 13, 14], including the mesh moving algorithm and the solution update algorithm. In our moving mesh computations, we choose the commonly used [7, 25] heuristical gradient-based monitor as

$$m=\sqrt{1+\beta|\nabla p|^2}, \tag{2.17}$$

where the parameter β is a positive constant.

3 Mesh sensitivity

To compare with [1, 9, 24], we will consider exactly the same configuration as therein, which is a phase-field problem involving the stability of a solid sphere in equilibrium with its undercooled liquid melt. More precisely, 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 R_0 lying inside the undercooled liquid. It is known that there exists a steady state solution of (2.2) where the solid ball is in equilibrium with its melt [6]. This occurs when the radius of the solid ball, R_c, is given by

$$R_c=-\frac{d_0}{\theta_{cool}}. \tag{3.1}$$

This equilibrium is unstable when $R_0<R_c$: in this case the solid sphere will melt and the radius will decrease to zero. On the other hand if $R_0>R_c$ then the solid will expand into the undercooled liquid and the radius will increase.

Following [1], we take the initial temperature to be $\theta_{cool}=-2$ and $d_0=1/2$ with parameters $D=1$, $\alpha=1$. It follows from (3.1) that $R_c=0.25$. The initial phase function is given by

$$p(\mathbf{x},0)=p_{bc}\tanh(\frac{r(\mathbf{x})}{2\varepsilon}), \tag{3.2}$$

where ε is the parameter to control the thickness of the transient layer, and $f(p,\theta)$ is given by (2.4),

$$p_{bc}=\begin{cases}\min\limits_p\{p:f(p,\theta_{cool})=0\}, \text{ closest to } -1, \ r(x)<0,\\ \max\limits_p\{p:f(p,\theta_{cool})=0\}, \text{ closest to } 1, \ r(x)\geq 0,\end{cases}$$

and $r(\mathbf{x})$ denote the signed normal distance from the point $\mathbf{x}\in\Omega$ to the interface, 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. We consider one case where the initial radius is $R_0=0.24$, which corresponds to the unstable case of $R_0<R_c$, in the domain $\Omega=[0,1/2]\times[0,1/2]$. The codes for all the computations below are based on the adaptive finite element package AFEPack [12].

In our numerical sensitivity studies, the time integrator for the ODE system (2.7) is found to be an insensitive factor. We tried the backward Euler scheme, the three-step

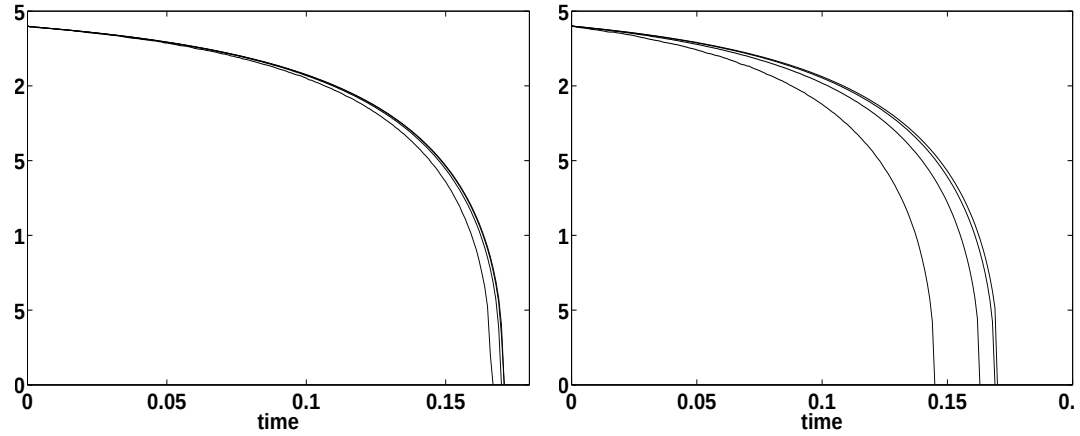

Figure 1: The radial positions in time computed using different parameter ε on a sequence of stationary uniform grids, with increase of resolution from 64×64, 128×128, 256×256 to 512×512, for $\varepsilon=1/80$ (left) and $\varepsilon=1/(160\sqrt{2})$ (right).

Runge-Kutta scheme and the alternating Crank-Nicolson scheme described in the last section. The results show that the time integrators, with different time integrators and time step length, provide the numerical solutions essentially the same, even when the numerical solutions are seriously under-resolved. The accuracy order of the temporal discretization has only small impact to the numerical solutions. The time step length of the time integration is not an important factor, either.

As a first step to reveal the main concern of this work, we solve the above problem on a sequence of uniform meshes and check the dependence of the limits of the numerical solutions on the parameter ε. In Fig. 1, we plot the radial positions as a function of time obtained using two different parameters ε on the uniform mesh sequence. It can be seen that for both parameters, the numerical solutions converge to their own limits, while the two limits are quite close to each other. This agrees with the theoretical results for ε going to zero [4]. As expected, for the smaller value $\varepsilon=1/(160\sqrt{2})$, the convergence rate of the numerical solutions is slower than that for the larger value of $\varepsilon=1/80$. Fig. 2 shows that our numerical solutions with $\varepsilon=1/80$ are in good agreement with the data in [9] on 128×128 mesh grids.

From now on, we set the parameter ε as $1/(160\sqrt{2})$ and carry out a group of computations systematically. The meshes under consideration are of four types, as demonstrated in Fig. 3, which are referred as /-type, \-type, V-type and D-type. The D-type mesh is the Delaunay triangulation generated by the mesh generating software Easymesh [20]. The other three types of meshes are the uniform meshes generated using varied cell patterns. These patterns are chosen since the behavior of the solutions are found different on these meshes. Moreover, the V-type mesh is the one used in [1].

At first, the numerical results on the fixed mesh at resolution of 512×512 are used as the reference solutions. For all four types of mesh, the reference solutions are matched

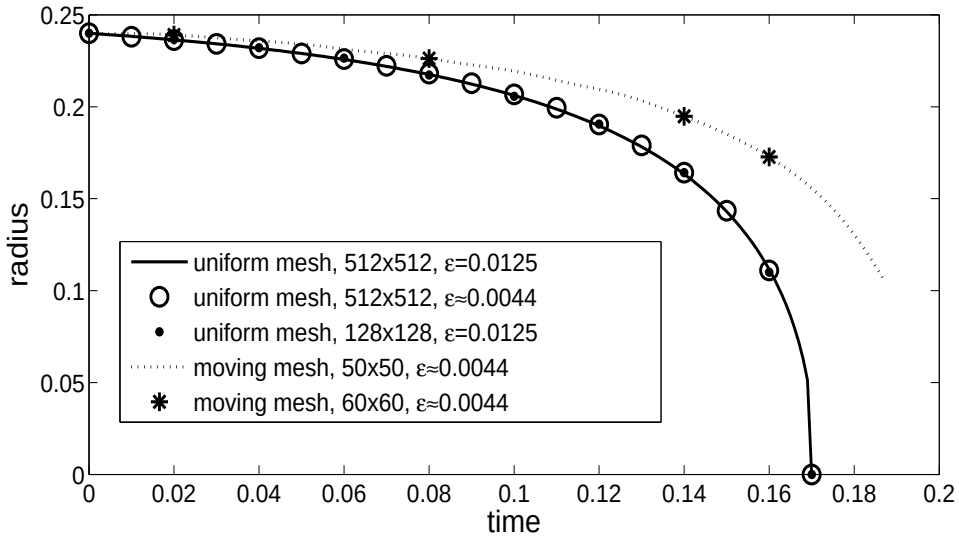

Figure 2: The solid line and the circles are our results on 512×512 grids for $\varepsilon=1/80,1/(160\sqrt{2})$, respectively, which are very close to each other. The dots are coming from [9] on 128×128 mesh grids which are agreed with our lines on 512×512 mesh grids. The dotted line is the results presented in [1], which is reproduced in [24] and denoted by asterisks.

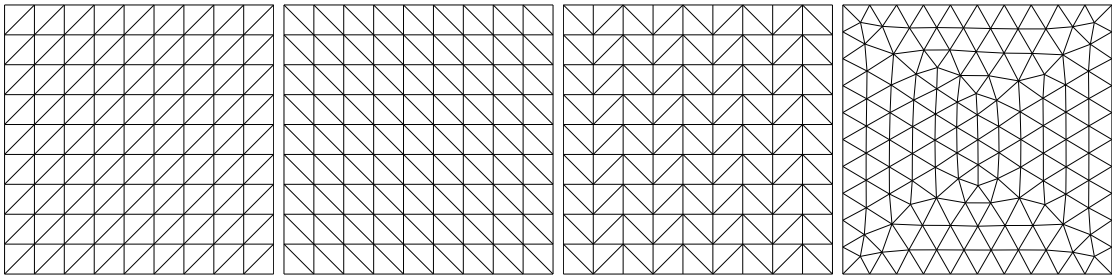

Figure 3: Profile of the different mesh types used. The figures from left to right are referred as /-type, \-type, V-type and D-type.

perfectly. The computations on the uniform meshes are calculated with time step length $\Delta t=10^{-3}$. We have checked that for the uniform mesh with resolutions 512×512, the numerical solutions matched perfectly using $\Delta t=10^{-3}$ and $\Delta t=10^{-4}$ with \-type and D-type meshes.

For each mesh type, we use four mesh resolutions for the starting mesh of the moving mesh method for our numerical convergence studies. More precisely, the resolutions of the meshes are 48×48, 64×64, 80×80 and 96×96. It can be seen clearly that for all cases, the moving mesh results converge to the reference solutions. For the moving mesh computation, we used $\Delta t=10^{-4}$. For $\Delta t=10^{-3}$, some cases in the computations are unstable based on our numerical experiments, while $\Delta t=10^{-4}$ can guarantee the numerical stability in all cases.

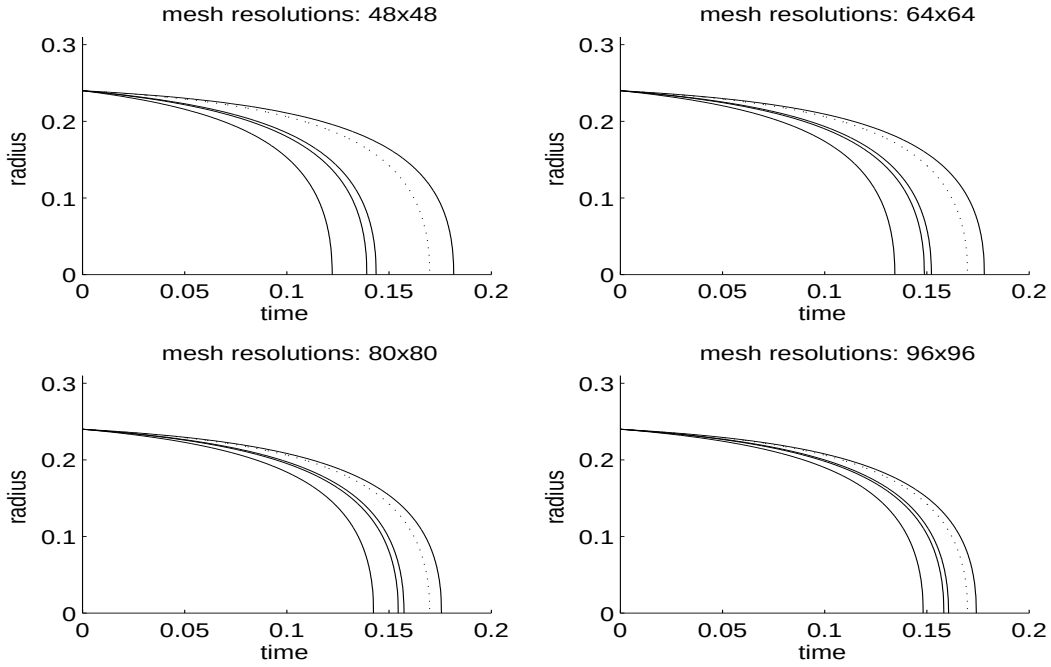

Figure 4: Starting mesh dependence for $\beta=2$. The solid line in each figure, from left to right: /-type, D-type, V-type and \-type. The dotted line is the reference result on 512×512 uniform mesh.

For the starting mesh for each mesh type and on every resolution, we present the results for the parameter β in the monitor function (2.17) at four typical values, namely, $\beta=2$, 4, 8 and 16. These values are chosen to present richer numerical phenomena with fewer data.

Sensitivity to the starting mesh. Using the same parameter β and resolutions, we plot the radial position on different types of starting mesh against time. Figs. 4-7 are the results for $\beta=2$, 4, 8 and 16, respectively. It can be seen that the \-type starting mesh is the case with the worst solution quality. The D-type is the best one. The /-type starting mesh provides numerical solutions with comparatively lower quality, but the improvement is better than the \-type starting mesh. The results on the V-type starting mesh are the intermediate ones. By a detailed observation of the local mesh structures in the transient layer in different numerical solutions (see Fig. 8), it is found that for the starting mesh with better numerical solutions, the spatial mesh step size perpendicular to the radial direction is smaller, thus the curvature of the interface can be better resolved. This is one of the possible reasons why the computations depended so seriously on the starting mesh. In spite of the differences on the mesh types, all numerical results on the moving mesh converge to the reference solutions obtained on the fixed 512×512 mesh, with the increasing resolution.

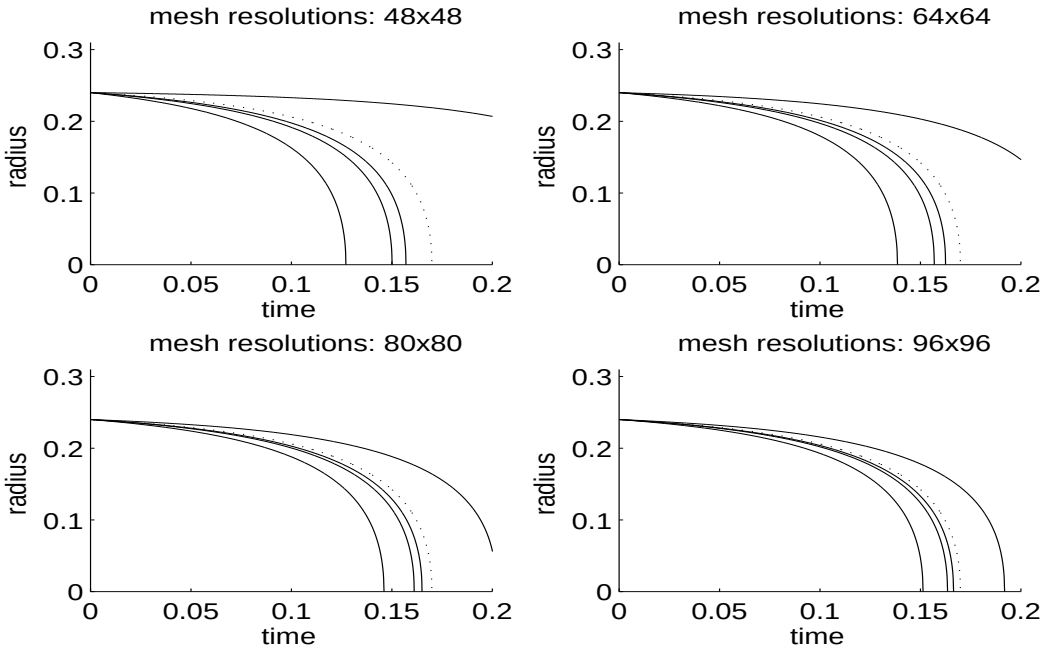

Figure 5: Same as Fig. 4, except $\beta=4$.

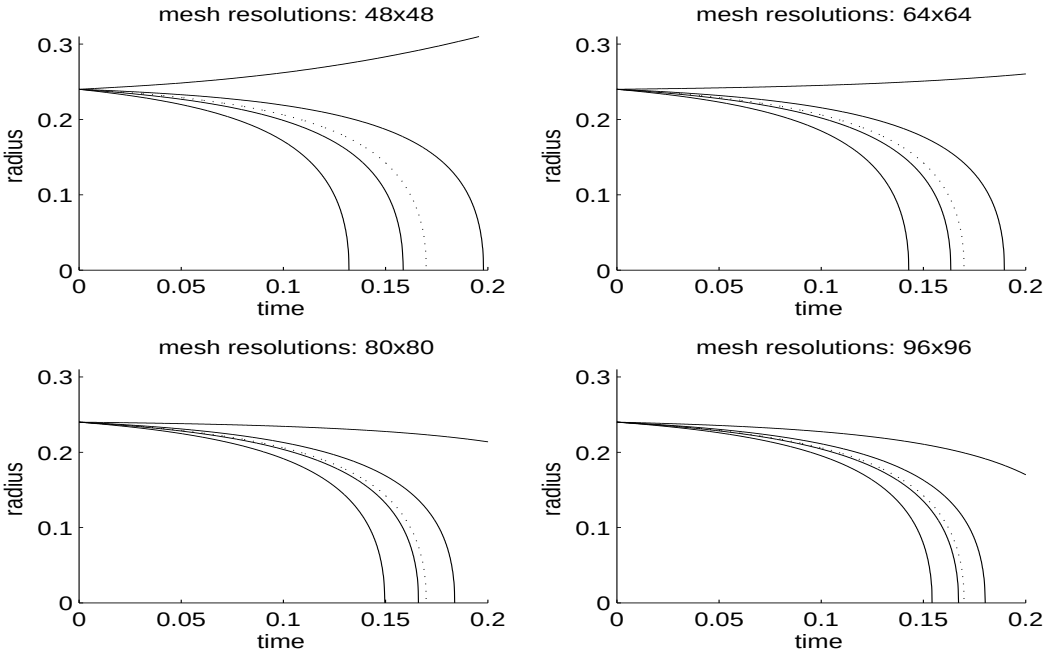

Figure 6: Same as Fig. 4, except $\beta=8$.

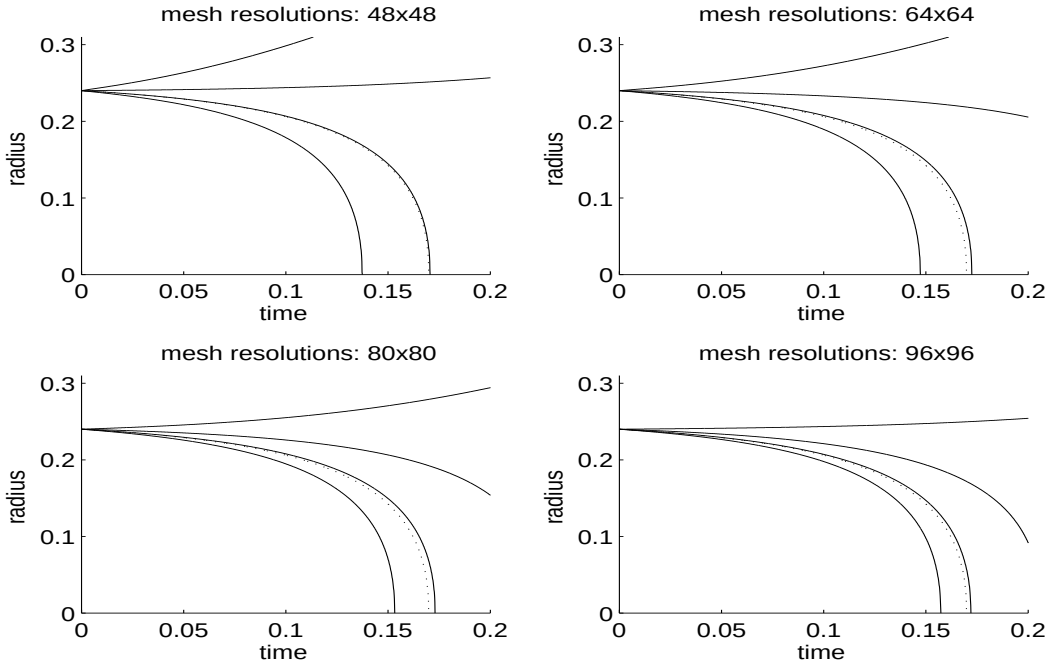

Figure 7: Same as Fig. 4, except $\beta=16$.

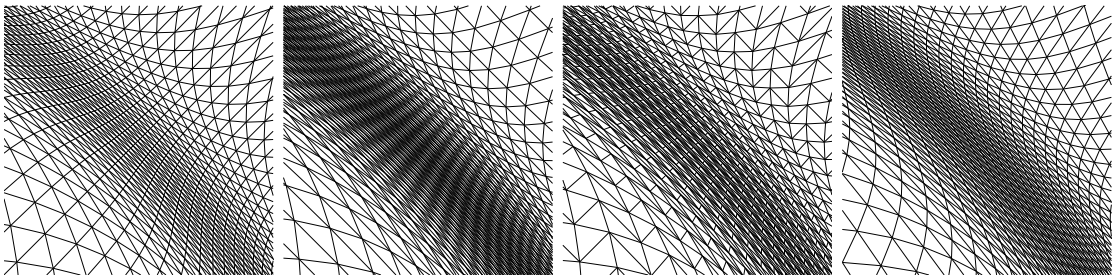

Figure 8: The local mesh structure in the transient layer with different types of starting mesh. The figures from left to right are given by /-type, \-type, V-type and D-type starting mesh.

Sensitivity to the monitor function. It is also observed that the moving mesh solutions are very sensitive to the constant β in the monitor function (2.17). Figs. 9-12 are the results for /-type, \-type, V-type and D-type, respectively. On the \-type starting mesh, the solutions are substantially incorrect and the sphere can even grow instead of melting. On the D-type starting mesh, the convergence of the numerical solutions to the reference solution seems the best. The results for the /-type starting mesh is quite inertial to the variation of the parameter β while there is still a big gap between its solutions and the reference solution. It is interesting to see that the numerical solutions, though they all converge to the reference solution with increasing resolution, behavior distinctive at dif-

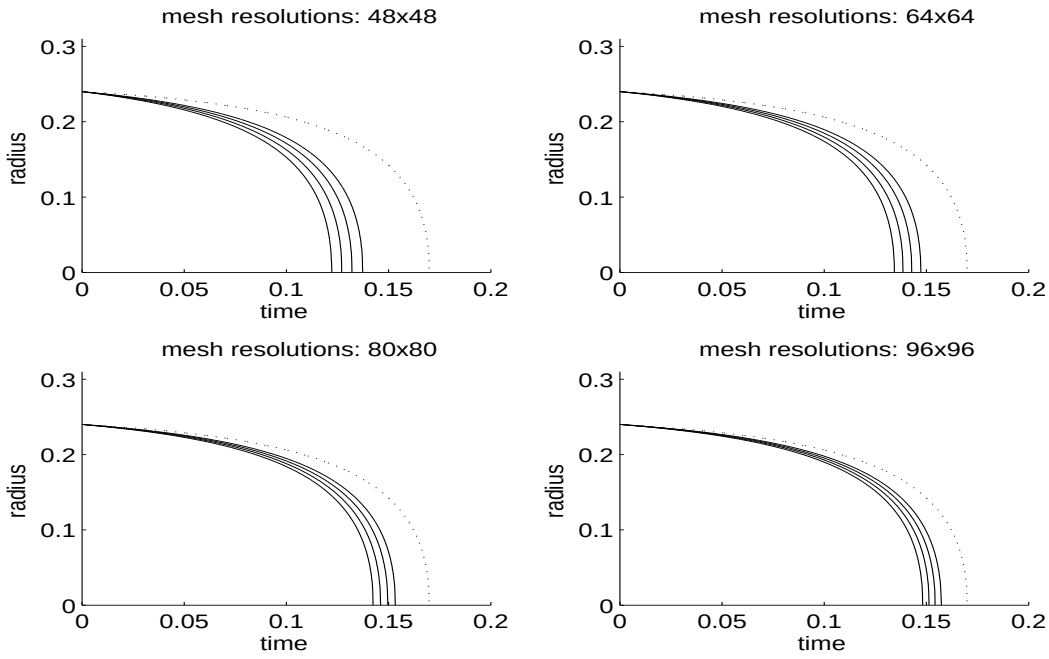

Figure 9: Monitor dependence for /-type mesh. The real line in each figure, from left to right, β=2, 4, 8 and 16. The dotted line is the reference result of 512×512 uniform mesh.

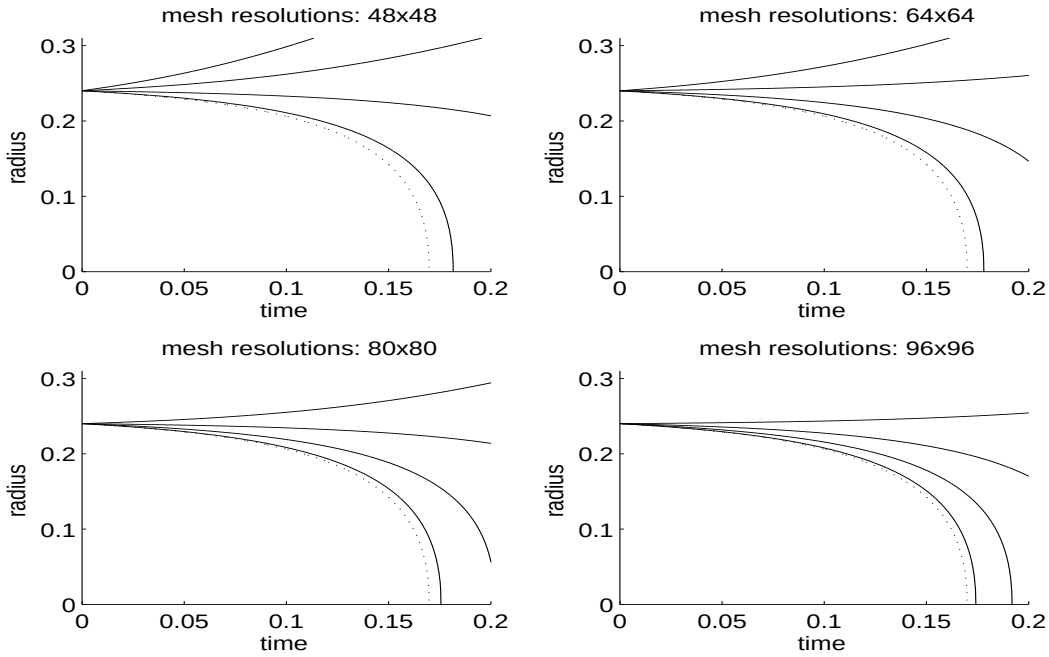

Figure 10: Same as Fig. 9, except the starting mesh is \-type mesh.

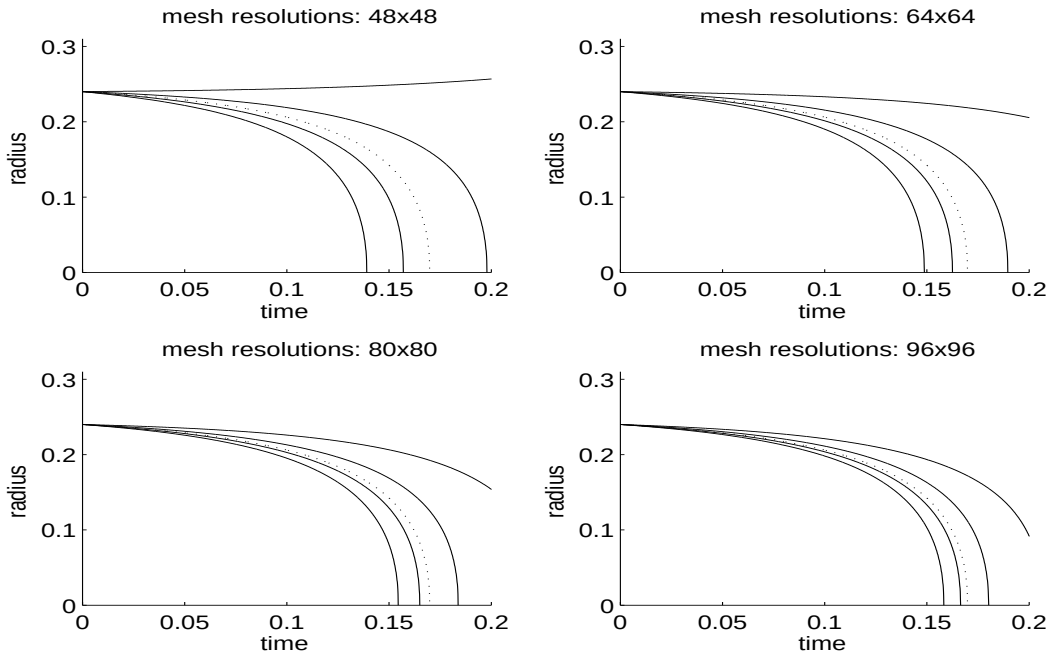

Figure 11: Same as Fig. 9, except the starting mesh is V-type mesh.

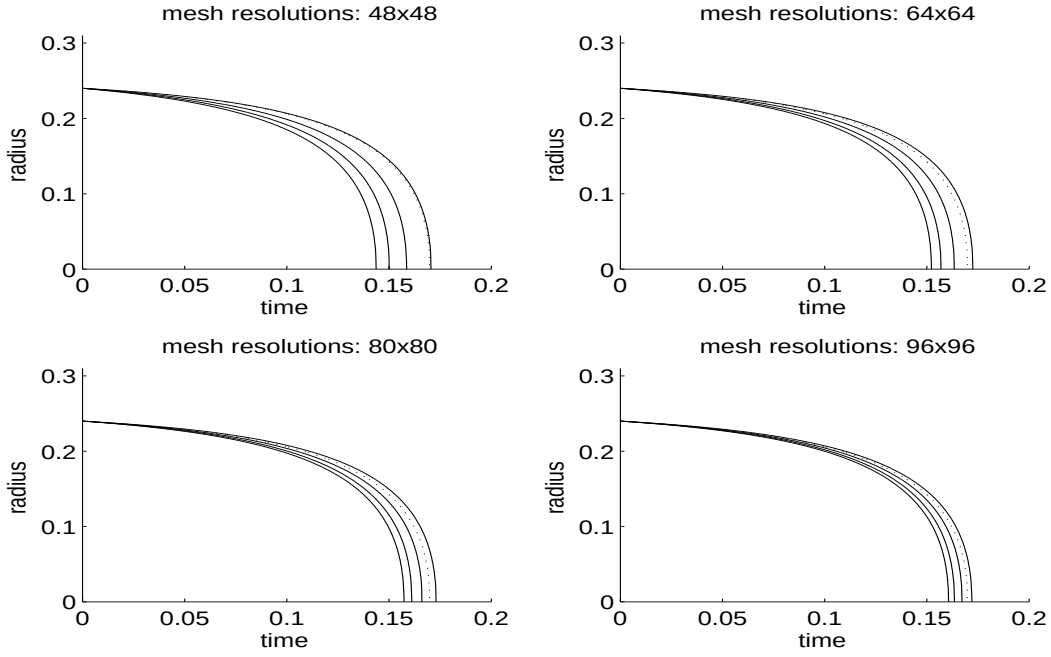

Figure 12: Same as Fig. 9, except the starting mesh is D-type mesh.

ferent β values and mesh types. Either too large or too small of β can be harmful to the quality of the numerical solution, and the highest convergence rate can be achieved at a certain optimal value of β. The behavior of the numerical solutions on the D-type mesh in Fig. 12 give us a delusion that for this mesh type, a bigger β can be better. However, for $\beta=32$, the numerical solution will jump to the other side of the reference solutions. Hence all the mesh types have a limited range of β values. It is observed from Figs. 9-12 that the D-type has the widest range of β among the four types of meshes.

4 Solutions for the solidification of two solid spheres

In this section, we solve the same equations with a more complex configuration, i.e., the solidification of two solid spheres which are surrounded by undercooled liquid, using the alternating Crank-Nicolson scheme as the time integrator. The alternating Crank-Nicolson scheme adopted can not reduce the iteration steps in the nonlinear iteration, but the iteration is carried out on a positive defined system with halved size. Thus in total the alternating Crank-Nicolson scheme can save the computation time cost up to at least 3 times. The improvement in numerical efficiency makes it possible to obtain reference solutions on very fine uniform meshes, up to a 800×800 resolution.

Based on the initial configuration, the two spheres will grow, then meet, and then merge into one sphere. From the numerical results, we can see that the solution is resolved quite well, even at the moment when the two spheres merge and the topological changes appear. The numerical results obtained using the moving mesh method match the results obtained on very fine fixed uniform meshes perfectly. The easy aspect of this example is that the radii of both spheres are increasing, thus the curvature of the interfaces are kept well-resolved during the whole computation. This fact partially explains the good agreement between the data on the moving mesh and the reference solutions.

The initial radius of both spheres is 0.26 and the centers are located at $(0.75,1.25)$ and $(1.25,0.75)$ in the domain $\Omega=[0,2]\times[0,2]$. The numerical results presented in this section uses the D-type mesh for computations on fixed mesh. All the other parameters are same as the one sphere case except that the constant in the monitor function (2.17) is $\beta=15$ for the D-type mesh and $\beta=4$ for the V-type mesh. We use the 100×100 resolution in the moving mesh computation and 400×400, 800×800 resolutions in the fixed mesh computation.

From Fig. 13, we can see that the adaptive grids follow the evolution of the interfaces with good quality when the two spheres merge together. In Fig. 14, the computational mesh at $t=0.206$ and 0.207 is replotted and magnified to show the local mesh structure. Fig. 15 shows the radial position of one of the two spheres against the time t. In the uniform mesh case, the parameter $\varepsilon=1/(160\sqrt{2})\approx0.0044$ implies that the mesh resolution should be about 455×455 in order to get the resolved numerical solutions on uniform meshes. By comparing the numerical results obtained using the moving mesh method with resolution 100×100 and that on the fixed meshes with resolutions 400×400 and

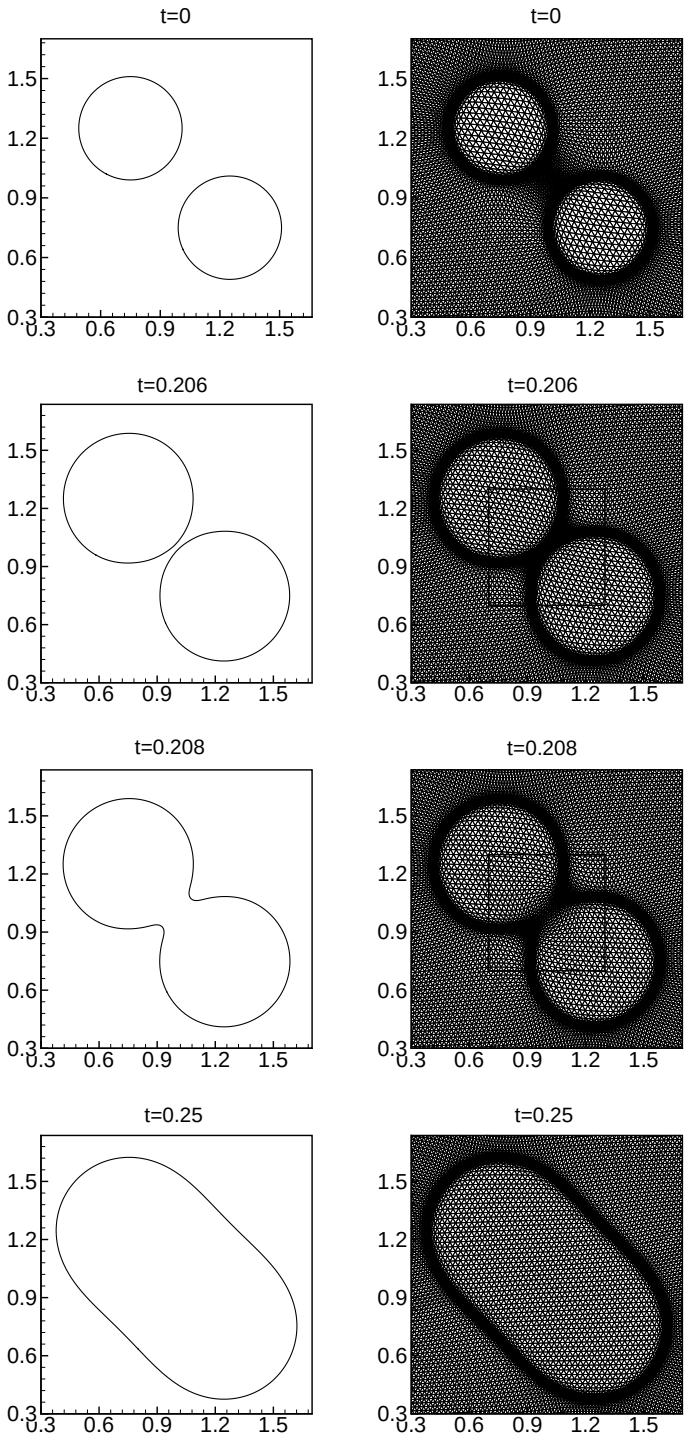

Figure 13: The interface and the mesh structure in two spheres case at $t=0$, 0.206, 0.208 and 0.25.

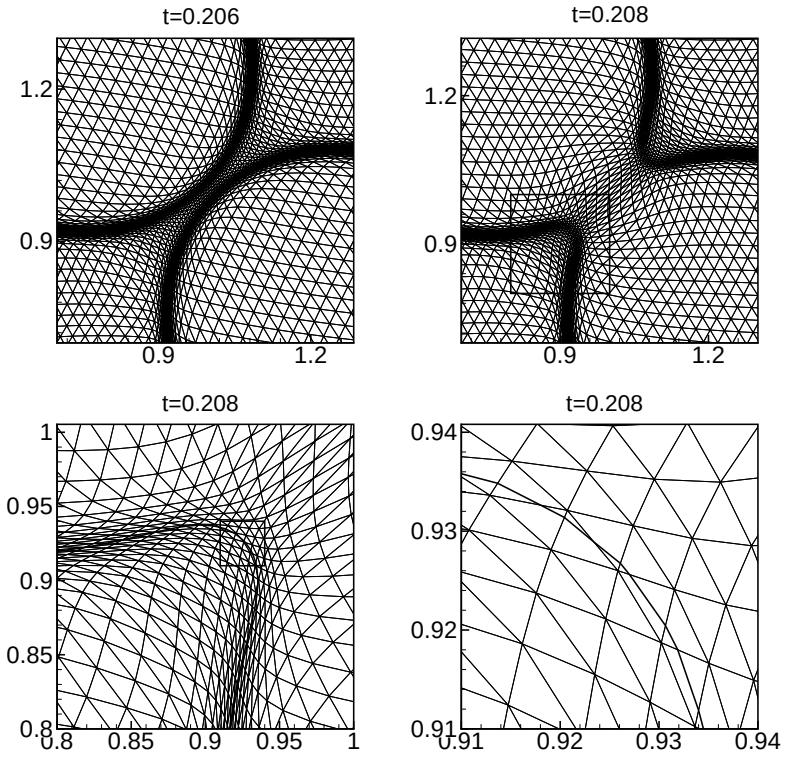

Figure 14: The local mesh structure in the domain marked by a box in Fig. 13 at $t=0.206$, and the more zoomed mesh structures at $t=0.208$.

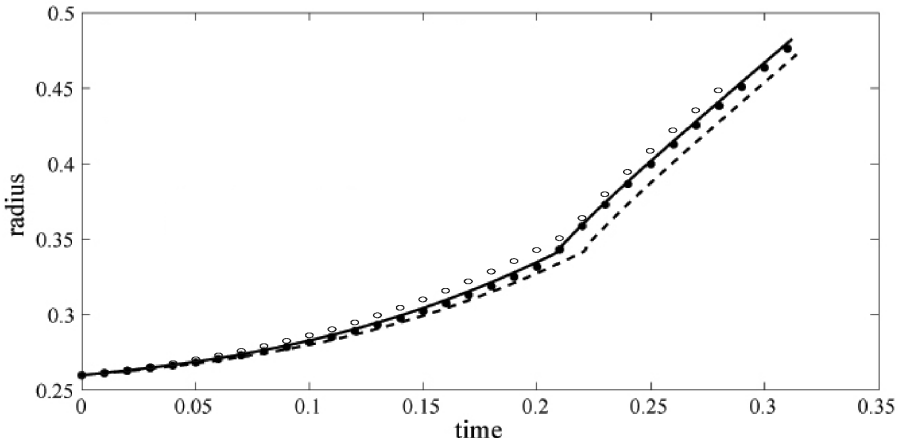

Figure 15: Radial position against time on the moving mesh and stationary meshes. The dots are the result from the moving mesh method with D-type mesh, $\beta=15$, which clearly agree with the results on the 800×800 uniform mesh grids (solid line) better than those from moving mesh with V-type mesh, $\beta=4$ (circles) and the 400×400 uniform mesh grids (dashed line).

800×800, it can be found that the moving mesh results are in good agreement with the finest fixed mesh results. As expected, the moving mesh result with the V-type starting mesh and $\beta=4$ has bigger error than that with the D-type mesh and $\beta=15$.

5 Conclusions

In this work, we observed that for a classical phase-field model, the efficiency of the moving mesh methods depends on the starting mesh and the monitor function. This phenomena brings us the reminiscences of the observation of Li [15], who considered the finite element approximations of microstructures. It is known that the approximations of the microstructures are strongly mesh dependent, e.g., the mesh distribution plays an important role in the computation of the crystalline microstructure. Although the reason for the starting mesh dependence is not clarified at this stage for the phase-field problems, the observed phenomena provide a serious numerical challenge. The present numerical evidence indicates that for the phase-field problems, the under-resolving of the interface curvature maybe the source of the qualitatively error of the numerical solutions. Moreover, we developed an alternating Crank-Nicolson time-stepping method which solves a smaller nonlinear system at a second-order accuracy in time with a remarkable saving in computational time.

Acknowledgments

The authors are grateful to Professor Tao Tang for many helpful discussions. The research of the first author was supported by Hong Kong Baptist University through an RGC Grant. The second author was partially supported by the National Basic Research Program of China under the grant 2005CB321701 and the Joint Applied Mathematics Research Institute between Peking University and Hong Kong Baptist University.

References

[1] 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 (2006), 805-826.
[2] R. J. Braun and B. T. Murray, Adaptive phase-field computations of dendritic growth, J. Cryst. Growth, 174 (1997), 41-53.
[3] G. Caginalp, Mathematical models of phase boundaries, in: J. Ball (Ed.), Material instabilities in continuum mechanics and related mathematical problems, Oxford University Press, Oxford, 1988, pp. 35-52.
[4] G. Caginalp and E. A. Socolovsky, Computation of sharp phase boundaries by spreading: The planar and spherically symmetric cases, J. Comput. Phys., 95 (1991), 85-100.
[5] 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.

[6] B. Chalmers, Principles of Solidification, Krieger, New York, 1977.
[7] Y. Di and P. Zhang, Moving mesh kinetic simulation for sheared rodlike polymers with high potential intensities, Commun. Comput. Phys., 1 (2006), 859-873.
[8] Y. N. Di, R. Li, T. Tang and P. W. Zhang, Moving mesh finite element methods for the incompressible Navier-Stokes equations, SIAM J. Sci. Comput., 26 (2005), 1036-1056.
[9] C. M. Elliott and A. R. Gardiner, Phase field equations, in: D. E. Stewart, H. J. Gardner and D. B. Singleton (Eds.), Computational Techniques and Applications Conference (CTAC93), World Scientific, Singapore, 1994, pp. 12-25.
[10] 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.
[11] J. S. Langer, Models of pattern formation in first-order phase transitions, in: G. Grinsteil and G. Mazenko (Eds.), Directions in Condensed Matter Physics: Memorial Volume in Honor of Shang-Keng Ma , Series on Directions in Condensed Matter Physics, World Scientific Press, Singapore, vol. 1, 1986.
[12] R. Li and W. B. Liu, http://circus.math.pku.edu.cn/AFEPack.
[13] R. Li, T. Tang and P. W. Zhang, A moving mesh methods in multiple dimensions based on harmonic maps, J. Comput. Phys., 170 (2001), 562-588.
[14] R. Li, T. Tang and P. W. Zhang, A moving mesh finite element algorithm for singular problems in two and three space dimensions, J. Comput. Phys., 177 (2002), 365-393.
[15] Z. P. Li, Computation of crystalline microstructures with the mesh transformation method, in: T. Tang and J. Xu (Eds.), Adaptive Computations: Theory and Algorithms, Science Press, Beijing, 2006.
[16] 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.
[17] J. A. Mackenzie and M. L. Robertson, A moving mesh method for the solution of the one-dimensional phase-field equations, J. Comput. Phys., 181 (2002), 526-544.
[18] M. Mu, A linearized Crank-Nicolson-Galerkin method for the Ginzburg-Landau model, SIAM J. Sci. Comput., 18 (1997), 1028-1039.
[19] M. Mu and Y. Q. Huang, An alternating Crank-Nicolson method for decoupling the Ginzburg-Landau equations, SIAM J. Numer. Anal., 35 (1998), 1740-1761.
[20] B. Niceno, http://www-dinma.univ.trieste.it/nirftc/research/easymesh/.
[21] R. H. Nochetto, M. Paolini and C. Verdi, An adaptive finite element method for two-phase Stefan problems in two space dimensions. Part ii: Implementation and numerical experiments, SIAM J. Sci. Comput., 12 (1991), 1207-1244.
[22] N. Provatas, N. Goldenfeld and J. Dantzig, Efficient computation of dendritic microstructures using adaptive mesh refinement, Phys. Rev. Lett., 80 (1998), 3308-3311.
[23] 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.
[24] Z. J. Tan, T. Tang and Z. R. Zhang, A simple moving mesh method for one- and two-dimensional phase-field equations, J. Comput. Appl. Math., 190 (2006), 252-269.
[25] H.-Z. Tang, A moving mesh method for the Euler flow calculations using a directional monitor function, Commun. Comput. Phys., 1 (2006), 656-676.
[26] S. L. Wang, S. L. Wang, R. F. Sekerka, A. A. Wheeler, 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.