Motion by mean curvature of curves on surfaces using the Allen–Cahn equation

Yongho Choi [a], Darae Jeong [a], Seunggyu Lee [a], Minhyun Yoo [b], Junseok Kim [a,*]

[a] *Department of Mathematics, Korea University, Seoul 136-713,Republic of Korea*
[b] *Department of Financial Engineering, Korea University, Seoul 136-701, Republic of Korea*

A R T I C L E   I N F O

A B S T R A C T

In this paper we develop a fast and accurate numerical method for motion by mean curvature of curves on a surface in three-dimensional space using the Allen–Cahn equation. We use a narrow band domain. We adopt a hybrid explicit numerical method which is based on an operator splitting method. First, we solve the heat equation by using an explicit standard Cartesian finite difference scheme. For the domain boundary cells, we use an interpolation using the closest point method. Then, we update the solution by using a closed-form solution. The proposing numerical algorithm is computationally efficient since we use a hybrid explicit numerical scheme and solve the governing equation only on the narrow domain. We perform a series of numerical experiments. The computational results are consistent with known analytic solutions.

1. Introduction

Many applications in the natural and applied sciences require the solutions of partial differential equations on surfaces. For example, there are texture synthesis in computer graphics (Turk, 1991), flow and solidification of a thin fluid film (Myers, Charpin, & Chapman, 2002), brain imaging (Memoli, Sapiro, & Thompson, 2004), phase separation patterns for diblock copolymers on spherical surfaces (Tang, Qiu, Zhang, & Yang, 2005), and surfactant distribution on a moving interface (Xu & Zhao, 2003). Analytic solutions to the problems on general surfaces are not always feasible, therefore it is important to be able to numerically approximate them accurately and efficiently.

In Ruuth and Merriman (2008), the authors presented the closest point method, which is an embedding method for solving partial differential equations on surfaces. In Merriman and Ruuth (2007), the diffusion generated motion algorithm has been extended to the moving curves on surfaces. Key aspects to their approach are the use of standard diffusion in one dimension higher than the underlying surface and the use of the closest point representation of the surface. The authors of Bertalmio, Cheng, Osher, and Sapiro (2001) introduced a framework for solving variational problems and partial differential equations on surfaces. The main idea is to represent the surface as the level set of a higher dimensional function and to solve the surface equations in a fixed Cartesian coordinate system.

In this article, we propose a fast and accurate numerical method on a narrow band domain for motion by mean curvature on a surface in three-dimensional space using the Allen–Cahn equation. The proposed hybrid explicit numerical method is based
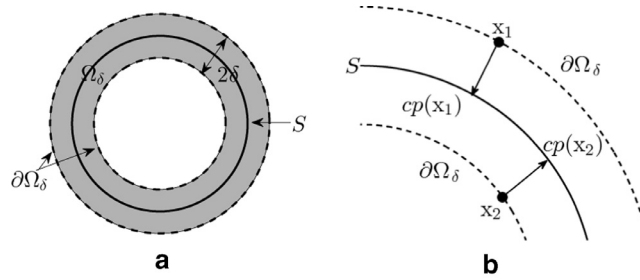
Fig. 1. (a) Schematic illustration of surface S, the narrow band domain Ω_δ with thickness 2δ, and its boundary $\partial\Omega_\delta$. (b) Closest points cp($\mathbf{x}_1$) and cp($\mathbf{x}_2$) for points $\mathbf{x}_1$ and $\mathbf{x}_2$.

on an operator splitting method. First, we solve the heat equation by using an explicit finite difference scheme. For the domain boundary cells, we use an interpolation using the closest point method. Then, we update the solution by using a closed-form solution. The proposing numerical algorithm is simple and computationally efficient since we use a hybrid explicit numerical scheme and solve the governing equation only on the narrow domain. The implementation of the algorithm is straightforward. We perform a series of numerical experiments. The computational results are consistent with known analytic solutions.

This paper is organized as follows. In Section 2, we describe the governing equation. In Section 3, we provide the numerical solution algorithm. We present the numerical results in Section 4. In Section 5, we give the paper's conclusions and list future work.

2. The Allen–Cahn equation on a narrow band domain

In this section, we consider motion by mean curvature of curves on a surface in three-dimensional space. Let S be a smooth surface in $\mathbb{R}^3$ and then we define a δ-neighborhood band of S as $\Omega_\delta = \{\mathbf{y} | \mathbf{x} \in S, \mathbf{y} = \mathbf{x} + \theta\mathbf{n}(\mathbf{x})$ for $|\theta| < \delta\}$, where $\mathbf{n}(\mathbf{x})$ is a unit normal vector at $\mathbf{x} \in S$. We use the following Allen–Cahn (AC) equation:

$$\frac{\partial\phi(\mathbf{x}, t)}{\partial t} = -\frac{F'(\phi(\mathbf{x}, t))}{\epsilon^2} + \Delta\phi(\mathbf{x}, t), \quad \mathbf{x} \in \Omega_\delta, \quad 0 < t \le T, \tag{1}$$

where $\phi(\mathbf{x}, t)$ is the difference between the concentrations of the two mixtures' components and $F(\phi) = 0.25(\phi^2 - 1)^2$. The parameter ϵ is the gradient energy coefficient related to the interfacial energy. The AC Eq. (1) was originally introduced as a mathematical model for antiphase domain coarsening in a binary alloy (Allen & Cahn, 1979). For simplicity of exposition, we shall illustrate schematics in two-dimensional space even though the actual algorithm is defined in three-dimensional space. Fig. 1(a) shows the schematic illustration of surface S, the narrow band domain Ω_δ with thickness 2δ, and its boundary $\partial\Omega_\delta$. Given a surface S, let cp($\mathbf{x}$) be a (possibly non-unique) point belonging to S which is closest to $\mathbf{x}$ (Macdonald, Brandman, & Ruuth, 2011). Fig. 1(b) shows the closest points cp($\mathbf{x}_1$) and cp($\mathbf{x}_2$) for points $\mathbf{x}_1$ and $\mathbf{x}_2$. The boundary condition is

$$\phi(\mathbf{x}, t) = \phi(\text{cp}(\mathbf{x}), t) \text{ on } \partial\Omega_\delta. \tag{2}$$

If δ is small enough, then this boundary condition allows us to use the standard Laplacian operator in the narrow band domain because the condition results in ϕ which is constant in the normal direction to the surface.

3. Numerical solution algorithm

We present a numerical scheme for the AC equation on the narrow band domain, Ω_δ. We discretize the AC equation in a three-dimensional domain $\Omega = (a, b) \times (c, d) \times (e, f)$ which includes Ω_δ. Let N_x, N_y, and N_z be positive integers, $h = (b - a)/N_x = (d - c)/N_y = (f - e)/N_z$ be the uniform mesh size, and $\Omega^h = \{(x_i = a + hi, y_j = c + hj, z_k = e + hk) | 0 \le i \le N_x, 0 \le j \le N_y, 0 \le k \le N_z\}$ be the discrete domain. Let ϕ_{ijk}^n be approximations of $\phi(x_i, y_j, z_k, n\Delta t)$, where $\Delta t = T/N_t$ is the time step, T is the final time, and N_t is the total number of time steps. We use the standard seven point discrete Laplacian:

$$\Delta_h\phi_{ijk} = \frac{\phi_{i+1,jk} + \phi_{i-1,jk} + \phi_{i,j+1,k} + \phi_{i,j-1,k} + \phi_{ij,k+1} + \phi_{ij,k-1} - 6\phi_{ijk}}{h^2} \tag{3}$$

Let $\psi : \mathbb{R}^3 \to \mathbb{R}$ be the signed distance function to S. Its zero level set is given by $S = \{\mathbf{x} \in \mathbb{R}^3 | \psi(\mathbf{x}) = 0\}$ with $\psi < 0$ inside S and $\psi > 0$ outside S. Let us define the discrete narrow band domain as $\Omega_\delta^h = \{(x_i, y_j, z_k) | |\psi(x_i, y_j, z_k)| < \delta\}$ (see Fig. 2(a)). Note that we should choose $\delta \ge \sqrt{3}h$ since the computational narrow band domain must contain the interpolation stencil for the closest points of the ghost points.

We also define an indicator function I as

$$I_{ijk} = \begin{cases} 0 & \text{if } (x_i, y_j, z_k) \in \Omega_\delta^h, \\ 1 & \text{otherwise.} \end{cases} \tag{4}$$
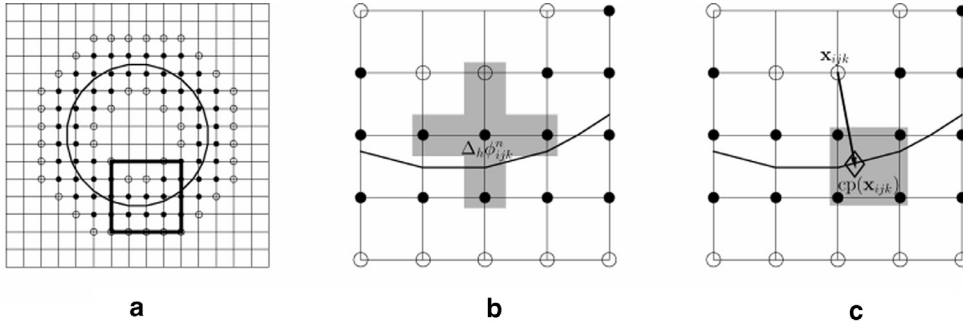
Fig. 2. (a) Schematic illustration of the discrete narrow band domain Ω_δ^h (indicated by •) and its ghost points $\partial\Omega_\delta^h$ (indicated by ○). Here, the curve S represents as a solid curve. (b) Expanded illustration of the boxed section in (a). The shaded region indicates the stencil for the discrete Laplacian operator. (c) Schematic illustration of the interpolation of cp($\mathbf{x}_{ijk}$). The shaded region indicates the interpolation stencil.

Then, we define the ghost points as $\partial\Omega_\delta^h = \{(x_i, y_j, z_k) | I_{ijk} | \nabla_h I_{ijk} | \neq 0\}$ (see Fig. 2(b)), where $\nabla_h I_{ijk} = (I_{i+1,jk} - I_{i-1,jk}, I_{i,j+1,k} - I_{i,j-1,k}, I_{ij,k+1} - I_{ij,k-1})/(2h)$.

We solve the AC Eq. (1) by using splitting method. First we solve the diffusion equation in the discrete narrow band domain Ω_δ^h using the explicit Euler's method:

$$\frac{\phi_{ijk}^* - \phi_{ijk}^n}{\Delta t} = \Delta_h \phi_{ijk}^n \tag{5}$$

with the boundary condition on $\partial\Omega_\delta^h$;

$$\phi_{ijk}^n = \phi^n(\text{cp}(\mathbf{x}_{ijk})). \tag{6}$$

The numerical closest point of $\mathbf{x}_{ijk}$ to the surface S is given as

$$\text{cp}(\mathbf{x}_{ijk}) = \mathbf{x}_{ijk} - \frac{\nabla_h |\psi_{ijk}|}{|\nabla_h |\psi_{ijk}||} |\psi_{ijk}|. \tag{7}$$

Since cp($\mathbf{x}_{ijk}$) is generally not on a grid point, $\phi^n(\text{cp}(\mathbf{x}_{ijk}))$ is obtained using the trilinearly interpolated value (see Fig. 2(c)). To compute the interpolation efficiently, we tabulated the interpolation stencil (we only save the smallest index in lexicographical order) and three fractions for each ghost point before starting time iterations. We have a stability condition, $\Delta t \leq h^2/6$, which is not that severe constraint if we are concerned with the numerical accuracy.

Next, with ϕ_{ijk}^* as the solution at $t = n\Delta t$, we solve the following equation up to $t = (n+1)\Delta t$:

$$\phi_t = \frac{\phi - \phi^3}{\epsilon^2}. \tag{8}$$

We have the following closed-form solution in the discrete narrow band domain Ω_δ^h for Eq. (8):

$$\phi_{ijk}^{n+1} = \frac{\phi_{ijk}^*}{\sqrt{e^{-2\Delta t/\epsilon^2} + (\phi_{ijk}^*)^2(1 - e^{-2\Delta t/\epsilon^2})}}. \tag{9}$$

This scheme is explicit and we only solve the AC Eq. (1) on a narrow band domain, therefore it is very fast. Full rectangular version of the hybrid explicit scheme for the AC equation will be published elsewhere.

4. Numerical experiments

We perform a series of numerical experiments to demonstrate the accuracy and efficiency of the proposed numerical algorithm. We define the width of the transition layer by using the ϵ value. From an equilibrium profile $\phi(x) = \tanh(x/(\sqrt{2}\epsilon))$, the concentration field ϕ varies from -0.9 to 0.9 over a distance of about $2\sqrt{2}\epsilon \tanh^{-1}(0.9)$. Therefore, if we want this value to be about mh (Choi, Lee, Jeong, & Kim, 2009), then we should take ϵ as $\epsilon_m = mh/[2\sqrt{2}\tanh^{-1}(0.9)] \approx 0.24mh$. Unless otherwise stated, we use $\epsilon = \epsilon_4$.

4.1. Motion by mean curvature on a sphere

We consider the motion of a circular interface on a sphere evolving according to in-surface curvature motion (Ruuth & Merriman, 2008). In Fig. 3(a), R and r are radii of the sphere and the circle, respectively. Let $V = -1/r$ be the curvature of the circle, x be
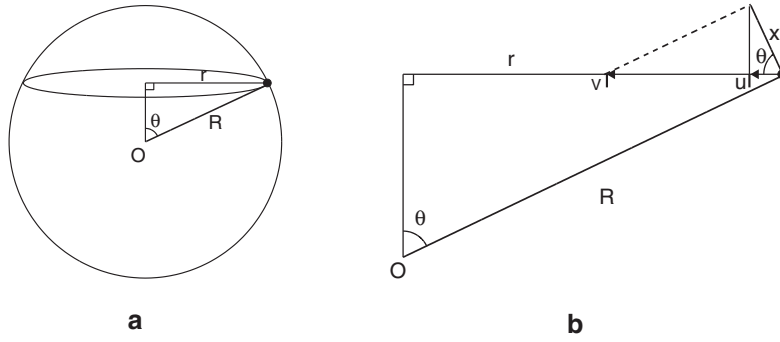
Fig. 3. Schematic of (a) parameters are on the sphere and (b) detailed description for some part of (a).
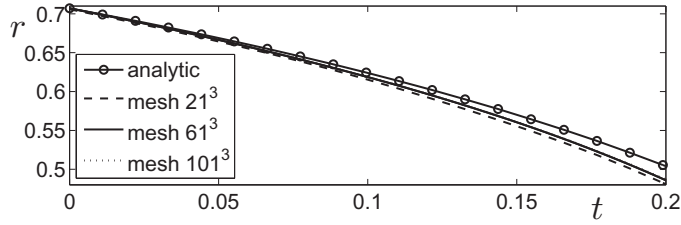

Fig. 4. Convergence test for different mesh size.

the tangential component of V, and u be the component of x toward the center of the circle. Then, we can find that $u = -\frac{1}{r}\cos^2\theta$ from Fig. 3(b). Now, the governing equation for the motion of a circular interface on a sphere is given as:

$$\frac{dr}{dt} = u = -\frac{1}{r}\cos^2\theta = \frac{r^2 - R^2}{rR^2}. \tag{10}$$

By solving Eq. (10), we have the analytic solution for $r(t)$,

$$r(t) = \sqrt{R^2 - (R^2 - r_0^2)e^{\frac{2t}{R^2}}}. \tag{11}$$

To show the convergence test, we adopt difference mesh size $n = 21, 61,$ and 101. Other conditions are same as spatial step size $h = 2/(n-1)$, sphere radius $R = 1$, circle radius $r = 1/\sqrt{2}$, total simulation time $T = 0.2$ on the computational domain $\Omega = (-1, 1) \times (-1, 1) \times (-1, 1)$.

Fig. 4 shows the convergence result with respect to grid size. Each of the mesh sizes (21^3, 61^3, and 101^3) has the radius value as 0.4809, 0.4859, and 0.4869, respectively. The errors from the analytic radius (0.5041) at $t = 0.2$ are 0.0232, 0.0182, and 0.0171 on the mesh 21^3, 61^3, and 101^3, respectively. Therefore, as we refine the mesh, the numerical result is getting closer to the analytic radius. Since we have a stability condition, $\Delta t \leq h^2/6$, we choose the proper Δt as $0.1h^2$ for the simulation.

In this simulation, the surface is defined using signed distance function:

$$d(x, y, z) = \sqrt{x^2 + y^2 + z^2} - R, \tag{12}$$

where R is sphere radius 1. And the initial condition is defined by phase-fields function:

$$\phi(x, y, z, 0) = \tanh\frac{\sqrt{x^2 + y^2} - z}{\sqrt{2}\epsilon}, \tag{13}$$

where $\epsilon = 0.1$ is transition layer. We also use other parameters are: circle radius $r = 1/\sqrt{2}$, mesh size $n = 51 \times 51 \times 51$, spatial step size $h = 2/(n-1)$, time step $\Delta t = 0.1h^2$ on the computational domain $\Omega = (-1, 1) \times (-1, 1) \times (-1, 1)$. Fig. 5(a)–(c) represent the evolution for each time (a) $t = 0$, (b) $t = 800\Delta t$, and (c) $t = 1600\Delta t$, respectively. The first column shows the surface view and second column represents grid view for calculating region only on the sphere.

4.2. Motion by mean curvature on a torus

In this section, we investigate motion by mean curvature on a torus. For all numerical simulation of torus, we use the following signed distance function as

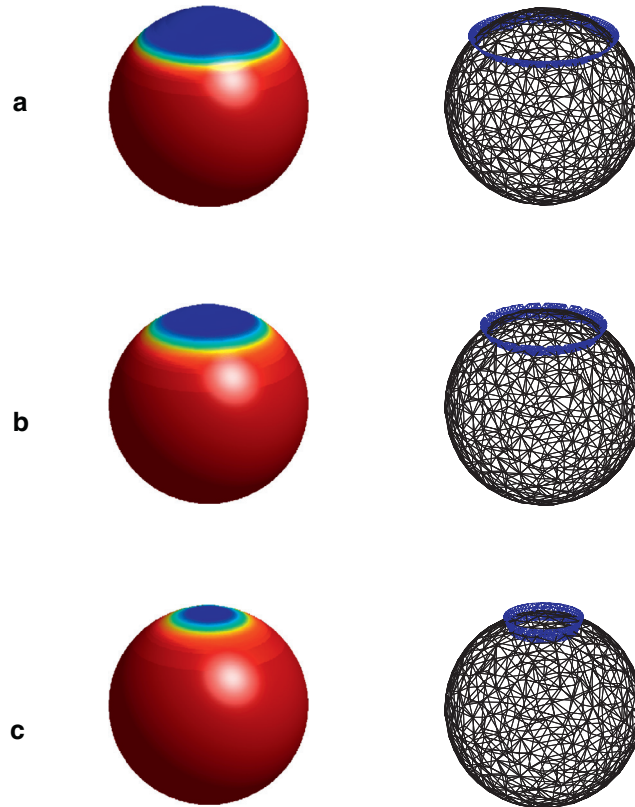$$d(x, y, z) = \sqrt{(\sqrt{x^2 + y^2} - R)^2 + z^2} - r, \tag{14}$$

Fig. 5. The first column represents the surface view and second column shows grid view at the (a) $t = 0$, (b) $t = 800\Delta t$, and (c) $t = 1600\Delta t$.
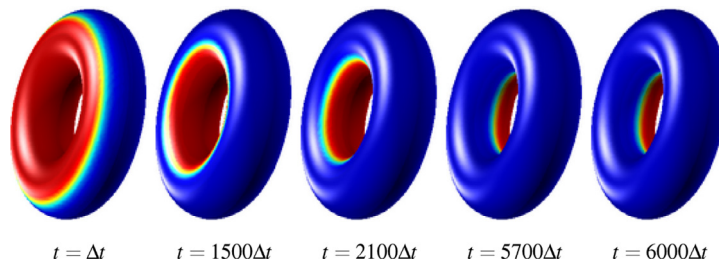

$$t = \Delta t \qquad t = 1500\Delta t \qquad t = 2100\Delta t \qquad t = 5700\Delta t \qquad t = 6000\Delta t$$

Fig. 6. Numerical solutions on the torus with the initial condition (15) at each time t. Red and blue regions denote $\phi = 1$ and $\phi = -1$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

where R represents the circumferential radius which is the size of the circle that constitutes the center of the torus tube and r is the tube radius. We take the two radii to be $R = 0.7$ and $r = 0.3$. We also use the following parameters as $\epsilon = \epsilon_4$, $h = 2/50$, $\Delta t = 0.1h^2$, $\delta = 1.01\sqrt{3}h$, and $51 \times 51 \times 16$ grids on the computational domain $\Omega = (-1, 1) \times (-1, 1) \times (-0.3, 0.3)$.

For the first example, we set the initial condition to be

$$\phi(x, y, z, 0) = \begin{cases} 1, & \text{if } \sqrt{x^2 + y^2} < 0.9 \,\&\, z > 0 \text{ and} \\ 1, & \text{if } \sqrt{x^2 + y^2} > 0.7 \,\&\, z < 0 \\ -1, & \text{otherwise.} \end{cases} \tag{15}$$

The initial curve is taken to be the intersection of the torus with the condition (15) as displayed in Fig. 6. Note that $\phi(x, y, z, t) = \phi(x, y, z, 0)$ for $z < 0.2$.

Fig. 6 illustrates the temporal behavior of the interfaces on the surface. In this figure, computational times are written below each one. The interfaces shrink by mean curvature as time evolves. In the last frame of Fig. 6, the interface reaches to the steady state.
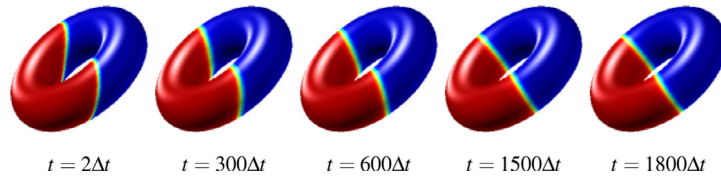
Fig. 7. Numerical solutions on the torus with the initial condition (16) at each time t. Red and blue regions denote $\phi = 1$ and $\phi = -1$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)
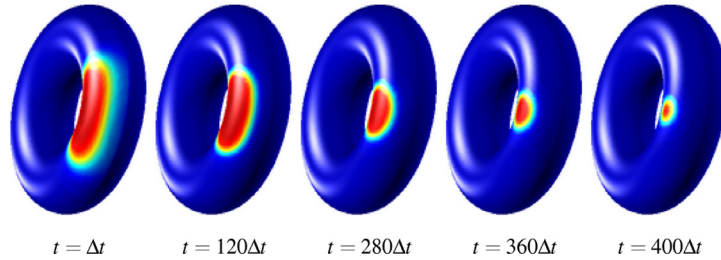

Fig. 8. Numerical solutions on the torus with the initial condition (17) at each time t. Red and blue regions denote $\phi = 1$ and $\phi = -1$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

In the second example, we use the following initial condition as

$$\phi(x, y, z, 0) = \begin{cases} 1, & \text{if} \sqrt{x^2 + z^2} < 1.2 \text{ and } y + z > 0.1 \\ -1, & \text{otherwise.} \end{cases} \qquad (16)$$

Fig. 7 displays the rendered surface of the numerical solution ϕ at $t = 2\Delta t$, $300\Delta t$, $600\Delta t$, $1500\Delta t$, and $1800\Delta t$, respectively. As time goes on, the interfaces evolve according to curvature motion. In final frame which represents a steady state, the two interfaces with ring-shape have zero curvature on the torus.

As third example, we consider the following initial condition as

$$\phi(x, y, z, 0) = \begin{cases} 1, & \text{if } \sqrt{(x - 0.7)^2 + 0.25z^2} < 0.25 \text{ and } z > 0.0 \\ -1, & \text{otherwise.} \end{cases} \qquad (17)$$

Fig. 8 shows the motion by mean curvature on torus at time t. When approaching steady state, the interface approaches a circle before disappearing.

4.3. Phase ordering on surfaces

Although our primary goal is to simulate motion by mean curvature on surfaces, we present phase ordering on curved surfaces. Phase separation on curved membranes are intriguing physical phenomena, ranging from nonequilibrium statistical physics and hydrodynamic theories to cell biology (see Marenduzzo & Orlandini, 2013 and references therein). One example is lipid bilayers (Schonborn, 1997).

First, we consider the process of phase separation on the surface of a sphere of radius R. The signed distance function is given as $d(x, y, z) = \sqrt{x^2 + y^2 + z^2} - R$, where $R = 1$. Then, the spherical surface is defined as the zero level set, i.e., $S = \{(x, y, z)|d(x, y, z) = 0\}$. The narrow band domain is defined as $\Omega_\delta = \{(x, y, z)||d(x, y, z)| < \delta\}$, where $\delta = 1.1\sqrt{3}h$. The space step $h = 2R/50$ and the time step $\Delta t = 0.1h^2$ are used. The domain is $\Omega = (-1 - 4h, 1 + 4h) \times (-1 - 4h, 1 + 4h) \times (-1 - 4h, 1 + 4h)$. The initial condition is $\phi(x, y, z, 0) = 0.5\text{rand}(x, y, z)$, where rand$(x, y, z)$ is a uniformly distributed random number between -1 and 1. Fig. 9 shows the time evolution of morphologies.

Next, we consider the process of phase separation on the surface of a torus with the signed distance function (14). And we choose the same value for numerical parameter as in the previous tests in Section 4.2. The initial condition is $\phi(x, y, z, 0) = 0.5\text{rand}(x, y, z)$, where rand$(x, y, z)$ is a uniformly distributed random number between -1 and 1. Fig. 10 shows the time evolution of morphologies of torus. As shown in this figure, spinodal decomposition proceeds to completion without arresting when approaching to steady state.

5. Conclusions

We developed a fast and accurate numerical method for motion by mean curvature of curves on a surface in three-dimensional space using the Allen–Cahn equation. By adopting operator splitting method, we solved the Allen–Cahn equation on a narrow band domain. Here, the domain boundary cells are defined by an interpolation algorithm with the closest point method. To solve
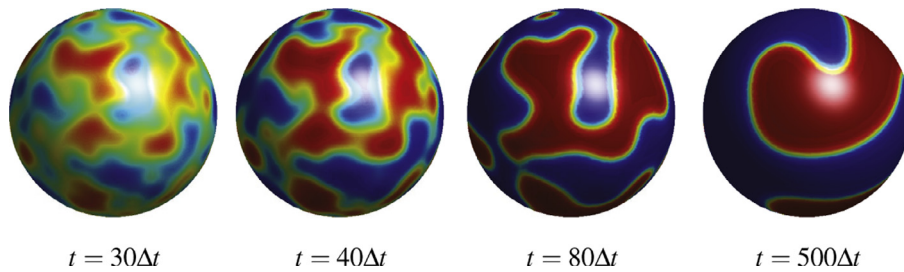
$$t = 30\Delta t \qquad t = 40\Delta t \qquad t = 80\Delta t \qquad t = 500\Delta t$$

Fig. 9. Temporal evolution of numerical solutions on a sphere. The computational times are shown below each figure.

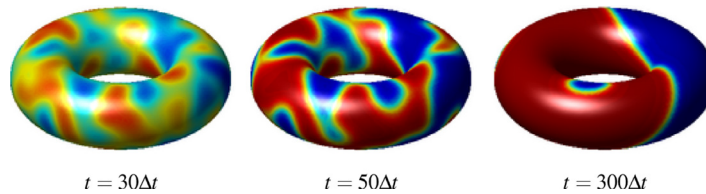$$t = 30\Delta t \qquad t = 50\Delta t \qquad t = 300\Delta t$$

Fig. 10. Temporal evolution of numerical solutions on a torus. The computational times are shown below each figure.

efficiently, we first solve the heat equation by using an explicit scheme, and then update the solution by using a closed-form solution. Through various numerical experiments such as motion by mean curvature or phase ordering on sphere and torus, we showed that the proposed numerical algorithm is computationally efficient and accurate. As a future work, we are planning to implement the implicit Euler's method for the heat equation to get rid of the time step size constraint.

Acknowledgment

References

Allen, S. M., & Cahn, J. W. (1979). A microscopic theory for antiphase boundary motion and its application to antiphase domain coarsening. *Acta Materialia, 27*, 1085–1095.

Bertalmio, M., Cheng, L. T., Osher, S., & Sapiro, G. (2001). Variational problems and partial differential equations on implicit surfaces. *Journal of Computational Physics, 174*, 759–780.

Choi, J. W., Lee, H. G., Jeong, D., & Kim, J. (2009). An unconditionally gradient stable numerical method for solving the Allen–Cahn equation. *Physica A: Statistical Mechanics and its Applications, 388*, 1791–1803.

Macdonald, C. B., Brandman, J., & Ruuth, S. J. (2011). Solving eigenvalue problems on curved surfaces using the closest point method. *Journal of Computational Physics, 230*, 7944–7956.

Marenduzzo, D., & Orlandini, E. (2013). Phase separation dynamics on curved surfaces. *Soft Matter, 9*, 1178–1187.

Memoli, F., Sapiro, G., & Thompson, P. (2004). Implicit brain imaging. *NeuroImage, 23*, S179–S188.

Merriman, B., & Ruuth, S. J. (2007). Diffusion generated motion of curves on surfaces. *Journal of Computational Physics, 225*, 2267–2282.

Myers, T. G., Charpin, J. P. F., & Chapman, S. J. (2002). The flow and solidification of a thin fluid film on an arbitrary three-dimensional surface. *Physics of Fluids (1994-present), 14*, 2788–2803.

Ruuth, S. J., & Merriman, B. (2008). A simple embedding method for solving partial differential equations on surfaces. *Journal of Computational Physics, 227*, 1943–1961.

Schonborn, O. (1997). Phase-ordering kinetics on curved surfaces. *Physica A, 239*, 412–419.

Tang, P., Qiu, F., Zhang, H., & Yang, Y. (2005). Phase separation patterns for diblock copolymers on spherical surfaces: A finite volume method. *Physical Review E, 72*, 016710.

Turk, G. (1991). Generating textures on arbitrary surfaces using reaction-diffusion. *ACM, 25*, 289–298.

Xu, J. J., & Zhao, H. K. (2003). An eulerian formulation for solving partial differential equations along a moving interface. *Journal of Scientific Computing, 19*, 573–594.