Computers and Mathematics with Applications 73 (2017) 1855–1864

Contents lists available at ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa

Computationally efficient adaptive time step method for the Cahn–Hilliard equation

Yibao Li [a], Yongho Choi [b], Junseok Kim [b,*]

[a] School of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an 710049, China
[b] Department of Mathematics, Korea University, Seoul, 02841, Republic of Korea

ARTICLE INFO

Article history:
Received 18 October 2016
Received in revised form 17 January 2017
Accepted 20 February 2017
Available online 8 March 2017

Keywords:
Cahn–Hilliard equation
Adaptive time-stepping method
Unconditionally stable scheme

ABSTRACT

In this work, we propose a fast and efficient adaptive time step procedure for the Cahn–Hilliard equation. The temporal evolution of the Cahn–Hilliard equation has multiple time scales. For spinodal decomposition simulation, an initial random perturbation evolves on a fast time scale, and later coarsening evolves on a very slow time scale. Therefore, if a small time step is used to capture the fast dynamics, the computation is quite costly. On the other hand, if a large time step is used, fast time evolutions may be missed. Hence, it is essential to use an adaptive time step method to simulate phenomena with multiple time scales. The proposed time adaptivity algorithm is based on the discrete maximum norm of the difference between two consecutive time step numerical solutions. Numerical experiments in one, two, and three dimensions are presented to demonstrate the performance and effectiveness of the adaptive time-stepping algorithm.

1. Introduction

In this paper, we propose a simple and efficient adaptive time-stepping method for the Cahn–Hilliard (CH) equation with the homogeneous Neumann boundary condition:

$$\frac{\partial \phi(\mathbf{x},t)}{\partial t} = \Delta \mu(\mathbf{x},t), \quad \mathbf{x} \in \Omega,\ t > 0, \tag{1}$$

$$\mu(\mathbf{x},t) = F'(\phi(\mathbf{x},t)) - \epsilon^2 \Delta \phi(\mathbf{x},t), \tag{2}$$

$$\mathbf{n} \cdot \nabla \phi(\mathbf{x},t) = \mathbf{n} \cdot \nabla \mu(\mathbf{x},t) = 0, \quad \mathbf{x} \in \partial\Omega, \tag{3}$$

where $\Omega \subset \mathbb{R}^d$ $(d = 1, 2, 3)$ and $\mathbf{n}$ is the unit vector normal to the domain boundary $\partial\Omega$. The quantity ϕ is the difference between the mole fractions of a binary mixture (e.g., $\phi = m_\alpha - m_\beta$, where m_α and m_β are the mole fractions phases α and β) [1]. The CH equation can be derived as the H^{-1}-gradient flow of the Ginzburg–Landau energy functional [2],

$$\mathcal{E}(\phi) = \int_\Omega \left[F(\phi) + \frac{\epsilon^2}{2} |\nabla \phi|^2 \right] d\mathbf{x}, \tag{4}$$

* Corresponding author. Fax: +82 2 929 8562.
E-mail addresses: yibaoli@xjtu.edu.cn (Y. Li), cfdkim@korea.ac.kr (J. Kim).
URLs: http://gr.xjtu.edu.cn/web/yibaoli (Y. Li), http://math.korea.ac.kr/~cfdkim (J. Kim).

http://dx.doi.org/10.1016/j.camwa.2017.02.021
0898-1221/

where the first term, $F(\phi) = 0.25(\phi^2-1)^2$, is the homogeneous free energy and the second term, $0.5\epsilon^2|\nabla\phi|^2$, penalizes large gradients and models capillary effects [3]. The small constant ϵ is the gradient energy coefficient related to the interfacial energy. The chemical potential μ is defined as the variational derivative of $\mathcal{E}(\phi)$ with respect to ϕ, i.e., $\mu = \delta\mathcal{E}(\phi)/\delta\phi = F'(\phi) - \epsilon^2\Delta\phi$. Then, using the CH equation and total energy functional, we have the following mass conservation and decrease of total energy, respectively [1]:

$$\frac{d}{dt}\int_\Omega \phi(\mathbf{x}, t)d\mathbf{x} = 0, \tag{5}$$

$$\frac{d}{dt}\mathcal{E}(\phi) = -\int_\Omega |\nabla\mu(\mathbf{x}, t)|^2 d\mathbf{x}. \tag{6}$$

The CH equation was proposed as a phenomenological model of phase separation in a binary alloy [4]. A review of the physical, mathematical, and numerical derivations of the CH equation can be found in [5]. The CH equation has been applied as a model for various problems: spinodal decomposition [4,6], which is a mechanism for the phase separation of a mixture of liquids or solids from one thermodynamic phase; microphase separation of diblock copolymers, which is two or more different polymer chains linked together [7]; image inpainting, which is the process of reconstructing lost parts of images [8]; volume reconstruction [9]; co-continuous binary polymer microstructures [10]; topology optimization [11]; microstructures with elastic inhomogeneity [12], phase-field modeling for tumor growth simulation [13], and multiphase fluid flows [14–16].

The temporal evolution of the CH equation involves multiple time scales. For example, in the spinodal decomposition simulation, an initial random perturbation evolves on a fast time scale and later coarsening evolves on a very slow time scale. Therefore, if a uniform small time step is used to capture the fast dynamics, the computation is very costly. On the other hand, if a uniform large time step is used, fast time evolutions may be overlooked. Hence, it is essential to use an adaptive time step method to simulate phenomena with multiple time scales.

This paper is organized as follows. We briefly review the related previous adaptive time step methods in Section 2. In Section 3, we describe the numerical solution algorithm. The one-, two-, and three-dimensional numerical results demonstrating the performance and effectiveness of the adaptive time-stepping methods are discussed in Section 4. Finally, a discussion is presented in Section 5.

2. Related works

The authors in [17] proposed an adaptive time-stepping method using a criterion related to a residual of the discrete energy law of Eq. (6), specifically, given ϕ^n, ϕ^{n-1}, Δt^n, Δt^{n-1}, and a parameter $\theta > 1$, chosen to be $\theta = 1.1$, the method is as follows:

Step 1. Compute ϕ^{n+1} and obtain

$$RE^{n+1} := \frac{\mathcal{E}(\phi^{n+1}) - \mathcal{E}(\phi^n)}{\Delta t^n} + \int_\Omega \left|\nabla\mu^{n+\frac{1}{2}}\right|^2 d\mathbf{x}. \tag{7}$$

Step 2. If $|RE^{n+1}| >$ **resmax**, take $\Delta t^n = \Delta t^n/\theta$ and go to *Step* 1.
Step 3. If $|RE^{n+1}| <$ **resmin**, take $\Delta t^{n+1} = \theta\Delta t^n$.
Here, a trial and error choice of **resmax** and **resmin** was used.
An adaptive time-stepping method using the time derivative of the total energy was considered in [18]; that is,

$$\Delta t = \max\left(\Delta t_{\min}, \frac{\Delta t_{\max}}{\sqrt{1+\alpha|\mathcal{E}'(t)|^2}}\right), \tag{8}$$

where the constant α is chosen in experience to adjust the level of adaptivity. A large value of $|\mathcal{E}'(t)|$ leads to a small time step, whereas a small $|\mathcal{E}'(t)|$ value yields a large time step.

The following adaptive time-stepping technique was developed in [19]:

Step 1. Calculate $e = \|\phi_{BE}^{n+1} - \phi_\alpha^{n+1}\|/\|\phi_\alpha^{n+1}\|$, where ϕ_{BE}^{n+1} and ϕ_α^{n+1} are the numerical solutions using the backward Euler method and a second-order generalized-α method with Δt^n, respectively.

Step 2. If $e > tol$, then reset the time step size $\Delta t^n = \rho\sqrt{tol/e}\Delta t^n$ and return to *Step* 1, where $\rho = 0.9$ and $tol = 10^{-3}$ are used. Otherwise, set $\Delta t^{n+1} = \rho\sqrt{tol/e}\Delta t^n$.

To equally distribute the locally computable discretization error, the authors in [20] proposed the following algorithm:

Step 1. The global relative time discretization error estimate is defined by

$$e = \frac{\|\phi^{n+1} - \hat{\phi}^{n+1}\|}{\Delta t^n \max(\|\phi^{n+1}\|, \|\hat{\phi}^{n+1}\|)}, \tag{9}$$

where ϕ^{n+1} and $\hat{\phi}^{n+1}$ are the numerical solutions obtained by taking two time steps of size $\Delta t^n/2$ and a single time step of size Δt^n from ϕ^n, respectively.

Step 2. If $e < e_{MAX}$, then set

$$\Delta t^{n+1} = \Delta t^n \left(\frac{e_{TOL}}{e}\right)^{\frac{1}{p}}, \tag{10}$$

where e_{MAX} is a target error tolerance and p is the global convergence rate of the time-stepping algorithm being used. If $e \geq e_{MAX}$, then that time step is rejected and go to *Step* 1, and recalculated with halving, i.e., $\Delta t^n = \Delta t^n/2$.

Kuhl and Schmidt [21] suggested an adaptive time-stepping scheme based on the convergence behavior of Newton–Raphson iterations. If more than six Newton iterations are required to reach the incremental equilibrium state, then the time step size is divided by two. Otherwise, the time step is increased by 10%.

The majority of previous adaptive time-stepping algorithms are based on the difference between a high- and low-order numerical solution or a temporal derivative of the total energy functional. Moreover, the algorithms are iterative, which is costlier. In this paper, we propose a very simple and efficient adaptive time-stepping procedure based on the maximum norm of the difference between two consecutive numerical solutions.

3. Numerical solution algorithm

In this section, we present the proposed adaptive time-stepping method for the CH equation. We discretize the CH equations (1) and (2) in one-dimensional space, i.e., $\Omega = (a, b)$ for the clear exposition. Let N_x be a positive even integer, $h = (b-a)/N_x$ be the uniform mesh size, and $\Omega_h = \{x_i : x_i = a + (i-0.5)h,\ 1 \leq i \leq N_x\}$ be the set of cell-centers. Let ϕ_i^n and μ_i^n be approximations of $\phi(x_i, t^n)$ and $\mu(x_i, t^n)$, respectively. Here, $1 \leq n \leq N_t$ and N_t is the total number of time steps. $t^n = t^{n-1} + \Delta t^{n-1}$ and Δt^{n-1} is the nonuniform time step. We then discretize Eqs. (1) and (2) in time by a nonlinear splitting algorithm [22]:

$$\frac{\phi_i^{n+1} - \phi_i^n}{\Delta t^n} = \Delta_d \mu_i^{n+1}, \tag{11}$$

$$\mu_i^{n+1} = (\phi_i^{n+1})^3 - \phi_i^n - \epsilon^2 \Delta_d \phi_i^{n+1}, \tag{12}$$

where the discrete Laplacian is defined by $\Delta_d \psi_i = (\psi_{i-1} - 2\psi_i + \psi_{i+1})/h^2$. The resulting system of discrete equations is solved by using a multigrid method [23,24]. The nonlinear scheme (11) and (12) is unconditionally energy gradient stable, which implies arbitrary large time steps are allowed. However, for the sake of accuracy, a very large time step is unacceptable. To improve the efficiency of the scheme without losing accuracy, we propose an adaptive time-stepping strategy in the following procedure.

Set the minimum and maximum time step sizes, $\Delta t_{\min}$ and $\Delta t_{\max}$, respectively. Let Δt^0 be an initial time step (in this paper, we set $\Delta t^0 = \Delta t_{\min}$). Then, for $n = 1, 2, \ldots$, we define the next time step to be

$$\Delta t^n = \min\left[\max\left(\frac{tol}{\|\phi^n - \phi^{n-1}\|_\infty}, \Delta t_{\min}\right), \Delta t_{\max}\right], \tag{13}$$

where $\|\phi\|_\infty = \max_{1\leq i\leq N_x} |\phi_i|$ and *tol* is a tolerance, which is important for choosing time steps. Too small a *tol* will lead to time steps close to $\Delta t_{\min}$, and too large will produce time steps close to $\Delta t_{\max}$.

As remarked in [17], the choice of **resmax** and **resmin** in the scheme using the residual (7) is not a trivial task because its values depend on the physical and discrete parameters. If **resmin** is large, then the accuracy of the solutions could be deteriorated. If **resmax** is small, then no improvement in the computational cost is obtained. Therefore, depending on the numerical results, a trial and error choice of **resmax** and **resmin** must be made. Most other adaptive time-stepping algorithms [18–21] choose the maximum and minimum time steps or a tolerance by trial and error. We also choose the parameters using trial and error. In general, the choice of the minimum and maximum time steps is case by case depending on the specific experiment of interest. Typically, the following time steps were used in the numerical simulations of the CH equation: $\Delta t = 10h^2$ [25], $\Delta t = h^2 \sim 4h^2$ [26], $\Delta t = 0.5h^2$ [27], $\Delta t = 25h^2/3$ [28], and $\Delta t = h^2$ [29]. Therefore, unless otherwise specified, we use $\Delta t_{\min} = h^2$ and $\Delta t_{\max} = 10\Delta t_{\min} \sim 100\Delta t_{\min}$.

4. Numerical results

In this section, we present numerical experiments for the CH equation obtained using the proposed adaptive time-stepping algorithm to demonstrate the effectiveness of the method. The phase-field ϕ varies from -0.9 to 0.9 over a length of approximately $2\sqrt{2}\epsilon \tanh^{-1}(0.9)$ across the interfacial transition layer. Therefore, if we want the length to be approximately hm, then $\epsilon = \epsilon_m = hm/[2\sqrt{2}\tanh^{-1}(0.9)]$ should be used [30]. Here, m is the number of grid cells. All the algorithms are implemented using generic C on a Windows 8 platform with Dual 3.20 GHz CPU and 8 GB of RAM.

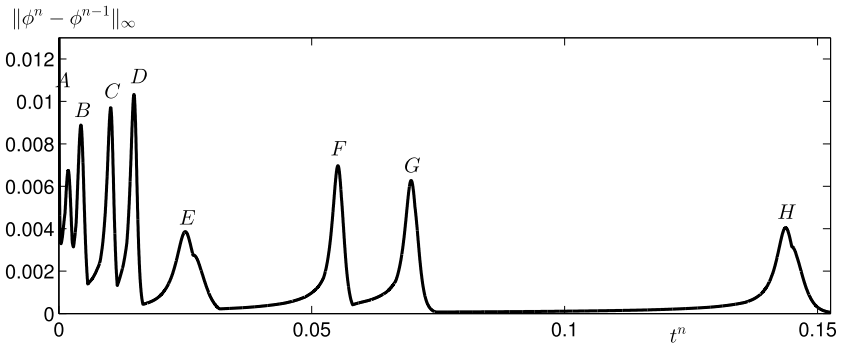

Fig. 1. $\|\phi^n - \phi^{n-1}\|_\infty$ versus time t^n.

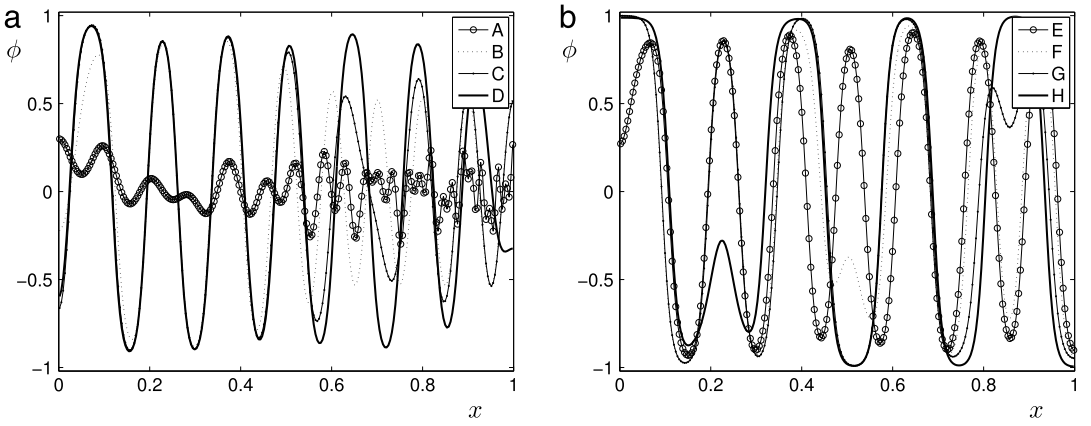

Fig. 2. Numerical solutions at the times as indicated in Fig. 1.

4.1. Temporal evolution with a uniform time step

As a first test, we perform a temporal evolution of the CH equation with a uniform time step. The computational domain is $\Omega = (0, 1)$ with $h = 1/256$, $\Delta t = h^2$, $N_t = 10\,000$, and $\epsilon = \epsilon_{15}$. The initial condition is

$$\phi(x, 0) = 0.1\cos(20\pi x) + 0.1\cos(30\pi x^2) + 0.1\cos(40\pi x^3). \tag{14}$$

Note that although most initial conditions have been generated using a random perturbation for phase separation simulations, we use sinusoidal perturbation for the purpose of reproductivity of the tests. In this paper, we use the same Ω, h, $\epsilon = \epsilon_{15}$, and initial condition, unless otherwise specified.

Fig. 1 shows $\|\phi^n - \phi^{n-1}\|_\infty$ versus the time t^n. Notice several picks in the figure, which imply that there are fast evolutions such as growth and coalescence. Fig. 2(a) and (b) show the numerical solutions at the times indicated in Fig. 1.

Fig. 3(a) plots $\|\phi^n - \phi^{n-1}\|_\infty$ versus the time t^n with three different time steps, $\Delta t = 100h^2$, $10h^2$, and h^2. The results for $\Delta t = 10h^2$ and h^2 are observed be almost similar. However, the result with $\Delta t = 100h^2$ is significantly different. In Fig. 3(b), numerical results obtained by using three different time steps at $t = 10\,000h^2$ are shown. If a large time step $\Delta t = 100h^2$ is used, then we obtain a qualitatively different result, which implies that a sufficiently small time step should be used to accurately capture the evolution dynamics.

4.2. Temporal evolution with the proposed adaptive time step

In this section, we present the performance of the proposed adaptive time step. Here, we set $\Delta t_{\min} = h^2$, $\Delta t_{\max} = 20h^2$, and $tol = 2 \times 10^{-6}$. Fig. 4(a) shows $\|\phi^n - \phi^{n-1}\|_\infty$ (solid line) and $50\Delta t^n$ (dotted line) versus the time t^n and Δt^n is scaled for better visualization. Notice that the time step adaptively adjusts its size according to the magnitude of $\|\phi^n - \phi^{n-1}\|_\infty$. Fig. 4(b) shows the numerical results obtained by using two uniform time steps ($\Delta t = 10h^2$, h^2) and the adaptive time step at $t = 10\,000h^2$. The numbers of the time steps are $N_t = 1000$, $10\,000$, and 851 for $\Delta t = 10h^2$, h^2, and the adaptive time step, respectively. As can be seen from Fig. 4, the numerical solution using the adaptive time step is closer to the solution with $\Delta t = h^2$ than that with $\Delta t = 10h^2$. Table 1 lists CPU times (in seconds) for $\Delta t = 10h^2$, $\Delta t = h^2$, and the adaptive time step. The extra computational cost for the adaptive time step is marginal.

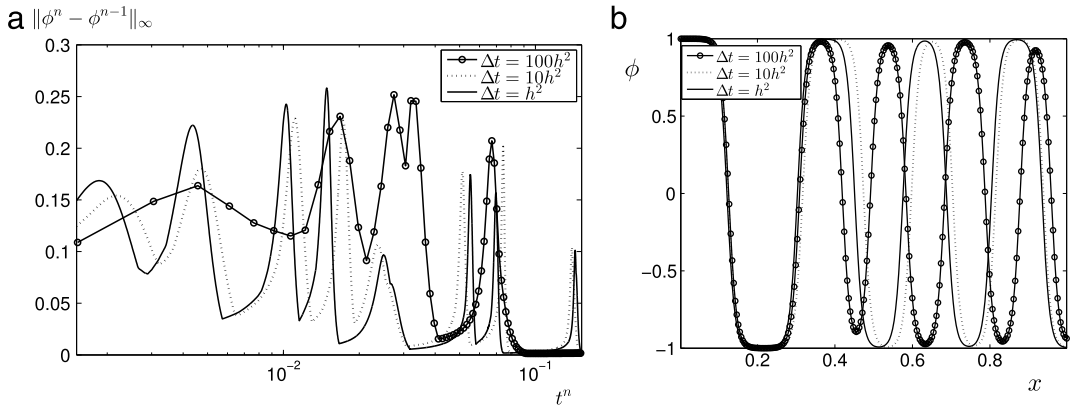

Fig. 3. (a) $\|\phi^n - \phi^{n-1}\|_\infty$ versus time t^n for three different time steps, $\Delta t = 100h^2$, $10h^2$, and h^2. A logarithmic scale for the time axis is used for better visualization. (b) Numerical results obtained by using three different time steps at $t = 10\,000h^2$.

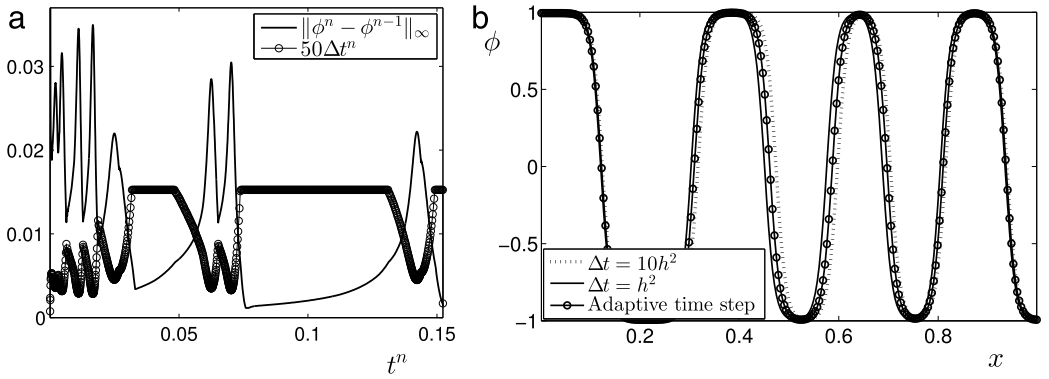

Fig. 4. (a) $\|\phi^n - \phi^{n-1}\|_\infty$ and $50\Delta t^n$ versus time t^n. Here, Δt^n is scaled for better visualization. (b) Numerical results obtained by using two uniform time steps ($\Delta t = 10h^2, h^2$) and the adaptive time step at $t = 10\,000h^2$.

Table 1
CPU times (in seconds) for $\Delta t = 10h^2$, $\Delta t = h^2$, and the adaptive time step $\Delta t_{\min} = h^2$, $\Delta t_{\max} = 20h^2$.

Time step	$\Delta t = 10h^2$	$\Delta t = h^2$	Adaptive time step
CPU time	1.421	10.390	1.265

Table 2
Effect of the parameters: tolerance, $\Delta t_{\min}$, and $\Delta t_{\max}$ on the CPU time (in seconds) and accuracy. The CPU time for the uniform time step ($\Delta t = 10h^2$) is 10.390.

$\Delta t_{\min}$	$\Delta t_{\max}$	*tol*	CPU time	$\|e\|_2$
h^2	$1.1h^2$	2×10^{-6}	9.608	0.0033
h^2	$2h^2$	2×10^{-6}	5.499	0.0330
h^2	$5h^2$	2×10^{-6}	2.577	0.1288
h^2	$20h^2$	2×10^{-6}	1.265	0.1634
h^2	$80h^2$	2×10^{-6}	1.171	0.1732
h^2	$20h^2$	1×10^{-6}	1.546	0.1051
h^2	$20h^2$	3×10^{-6}	1.125	0.2116

Next, we consider the effect of the parameters: tolerance, $\Delta t_{\min}$, and $\Delta t_{\max}$ on the CPU time and accuracy. Table 2 lists the CPU times and errors with a reference solution, which is the numerical result using the uniform time step, $\Delta t = h^2$, at $t = 10\,000h^2$. The CPU time for the uniform time step ($\Delta t = 10h^2$) is 10.390. Here, the error is defined as the difference between the adaptive time step result and the reference solution, which is obtained with fine uniform time step $\Delta t = h^2$. The discrete l_2-norm is defined as $\|e\|_2 = \sqrt{\sum_{i=1}^{N_x} e_i^2 / N_x}$.

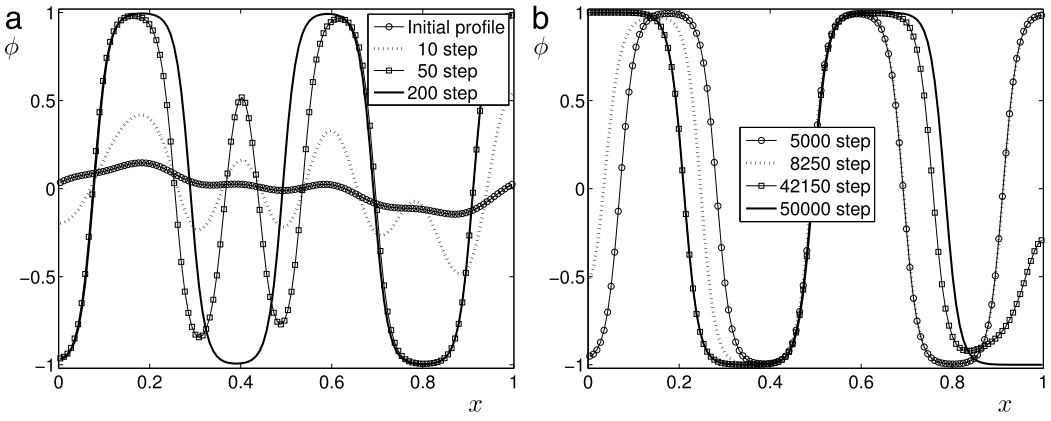

Fig. 5. (a) Early evolution and (b) later evolution.

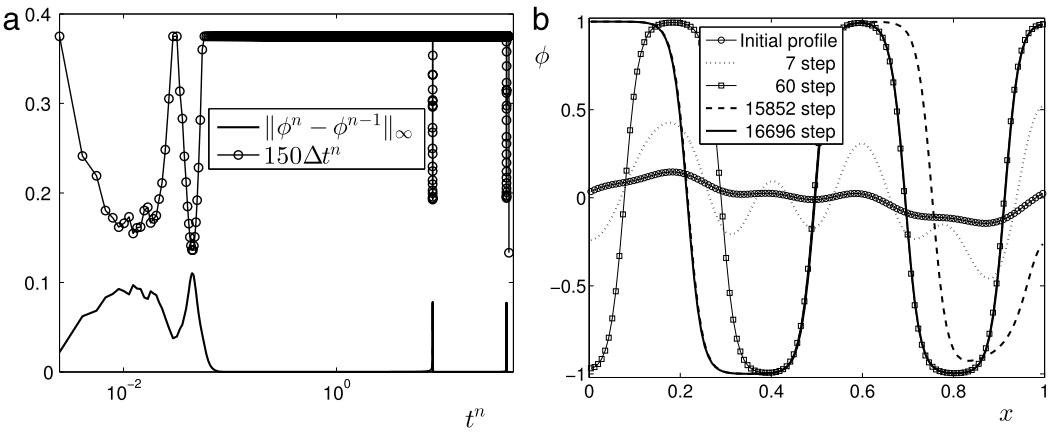

Fig. 6. (a) $\|\phi^n - \phi^{n-1}\|_\infty$ and $150\Delta t^n$ versus time t^n. (b) Temporal evolution of ϕ.

As can be observed from Table 2, CPU time decreases as $\Delta t_{\max}$ or tolerance is increased. For a fixed tolerance, the accuracy increases as we decrease the maximum time step size $\Delta t_{\max}$. On the other hand, for a fixed $\Delta t_{\max}$, the error decreases as tolerance decreases.

4.3. Comparison with a previous study

We consider a test example from [28]. The initial condition is

$$\phi(x, 0) = 0.1\sin(2\pi x) + 0.01\cos(4\pi x) + 0.05\sin(4\pi x) + 0.02\cos(10\pi x)$$

and the parameters are $h = 1/128$, $\Delta t = 1/1200$, and $\epsilon = \sqrt{0.0005}$. Fig. 5(a) and (b) show the early and later evolutions, respectively.

Next, we run the same test with the adaptive time-stepping algorithm. We set $\Delta t_{\min} = 1/1200$, $\Delta t_{\max} = 3\Delta t_{\min}$, and $tol = 10^{-3}$. Fig. 6(a) shows $\|\phi^n - \phi^{n-1}\|_\infty$ and $150\Delta t^n$ versus the time t^n. Observe that the time step adaptively adjusts itself according to the magnitude of the consecutive time step solutions. Fig. 6(b) shows some selected profiles of ϕ at intermediate times. If we compare this result to the previous test with a uniform time step, it is clear that the main features of the dynamics can be captured with less time step iterations by using the adaptive time step procedure.

4.4. Comparison test with a previous method

In this section, we perform a comparison test with the previous adaptive time-stepping method [20]. The initial condition on a computational domain $\Omega = (0, 1)$ is

$$\phi(x, 0) = 0.1\cos(20\pi x) + 0.1\cos(30\pi x^2) + 0.1\cos(40\pi x^3). \tag{15}$$

We take a reference solution with a uniform time step. The parameters used are $h = 1/256$, $\Delta t = h^2$, $N_t = 10\,000$, and $\epsilon = \epsilon_{15}$. The CPU time for the uniform time step is 10.390. We set $\Delta t_{\min} = h^2$ and $\Delta t_{\max} = 50h^2$ for both the methods and change e_{TOL} and tol for the previous method and the proposed method, respectively (see Fig. 7).

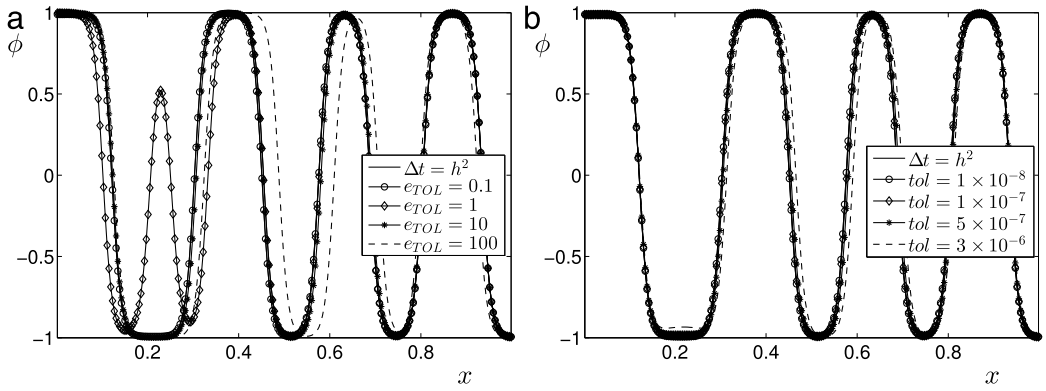

Fig. 7. Solutions obtained from the uniform and adaptive time-stepping methods with various tolerances at $t = 10\,000h^2$: (a) iterative method [20] and (b) proposed method.

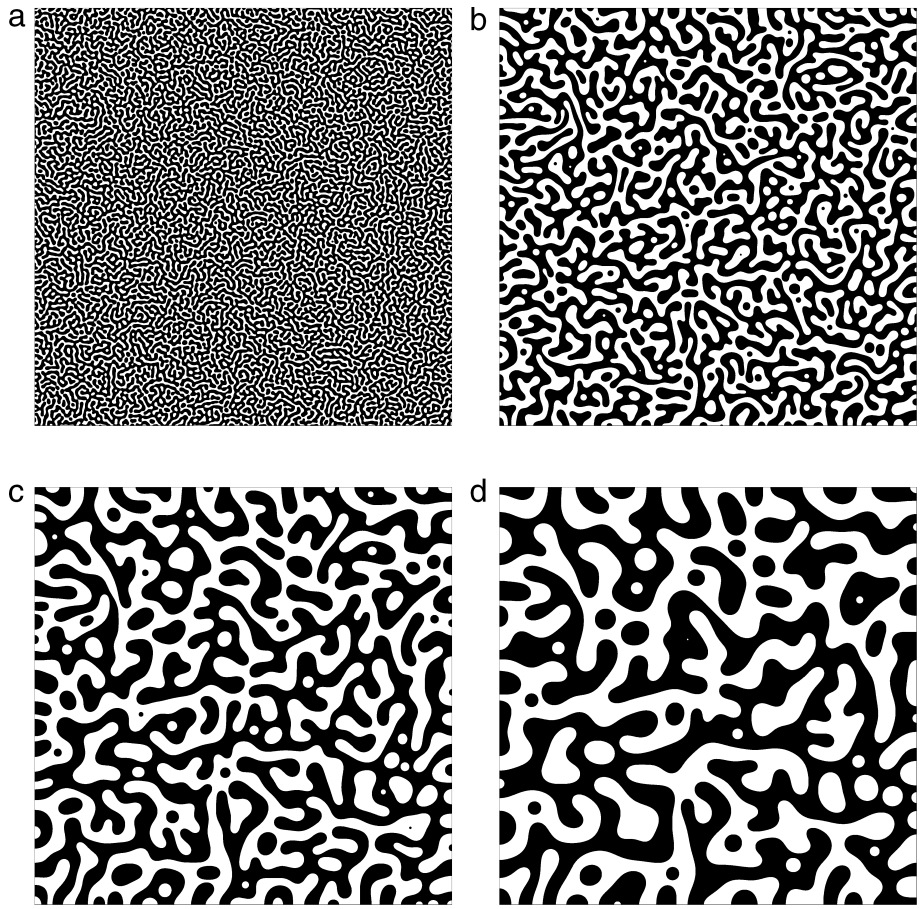

Fig. 8. Morphological patterns during spinodal decomposition: (a) $t = 100$, (b) $t = 2000$, (c) $t = 10\,000$, (d) $t = 30\,000$.

Table 3 lists the CPU times in seconds: the iterative algorithm [20] (left column) and proposed algorithm (right column). We can observe that the proposed algorithm is better than the iterative algorithm in terms of CPU time under equivalent accuracy. Moreover, we can see the error decreases as *tol* is reduced.

4.5. Spinodal decomposition

The computational domain $\Omega = (0, 1024) \times (0, 1024)$, $h = 1$, ϵ_4, $tol = 1$, $\Delta t_{\min} = h^2$, and $\Delta t_{\max} = 100h^2$ are used. The initial condition is $\phi(x, y, 0) = 0.1\text{rand}(x, y)$, where $\text{rand}(x, y)$ is a random number between 1 and -1. Fig. 8 shows

Table 3
CPU times in seconds: the iterative algorithm [20] (left column) and proposed algorithm (right column).

e_{TOL}	$N_{\max}$	CPU time	$\|e\|_2$	*tol*	$N_{\max}$	CPU time	$\|e\|_2$
0.1	7996	33.218	0.0016	1×10^{-8}	7427	8.625	0.0011
1	2692	13.968	0.4095	1×10^{-7}	3493	4.703	0.0123
10	631	3.890	0.0667	5×10^{-7}	1516	2.265	0.0668
100	230	1.453	0.4519	3×10^{-6}	630	1.015	0.2180

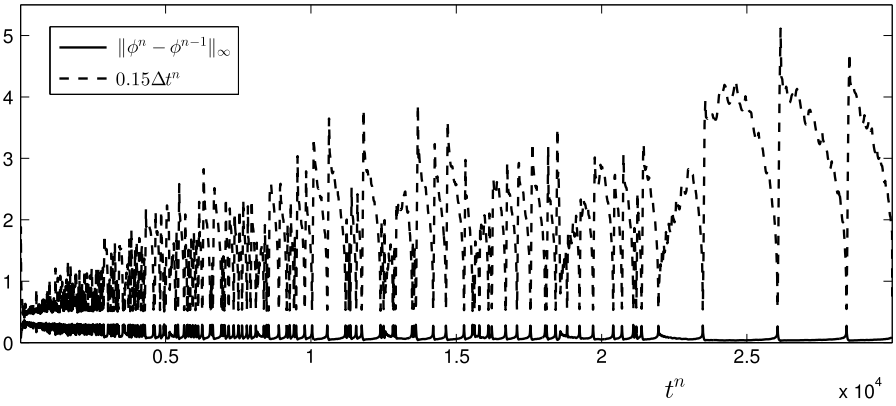

Fig. 9. $\|\phi^n - \phi^{n-1}\|_\infty$ and $0.15\Delta t^n$ versus time t^n.

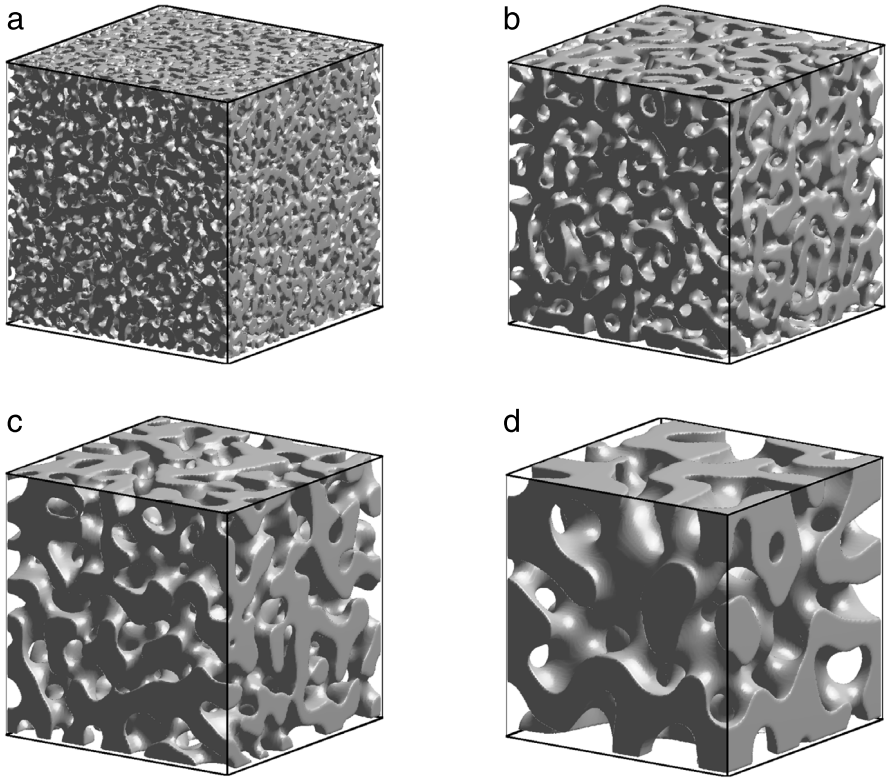

Fig. 10. Morphological patterns during spinodal decomposition: (a) $t = 116.96$, (b) $t = 1508.6$, (c) $t = 5001$, (d) $t = 25\,010$.

temporal evolution of morphological patterns during spinodal decomposition and subsequent coarsening. We obtain the result in Fig. 8(d) with only 3077 time step iterations. Previously, it was obtained with 30 000 iterations [29].

Fig. 9 shows $\|\phi^n - \phi^{n-1}\|_\infty$ and $0.15\Delta t^n$ versus the time t^n. Notice that the time step changes adaptively according to the maximum norm of the difference between two consecutive numerical solutions.

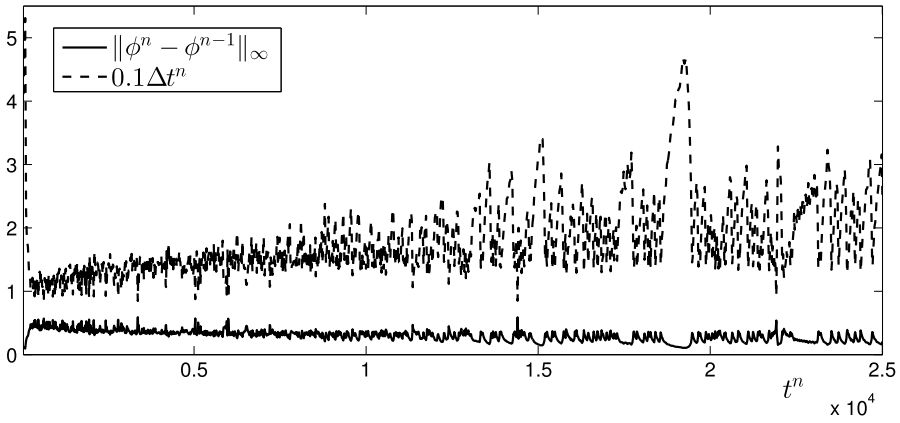

Fig. 11. $\|\phi^n - \phi^{n-1}\|_\infty$ and $0.1\Delta t^n$ versus time t^n.

Finally, we perform a three-dimensional test on $\Omega = (0, 256) \times (0, 256) \times (0, 256)$ with $h = 1, \epsilon_4, tol = 5, \Delta t_{\min} = h^2$, and $\Delta t_{\max} = 100h^2$. The initial condition is $\phi(x, y, 0) = 0.1\text{rand}(x, y, z)$, where $\text{rand}(x, y, z)$ is a random number between 1 and -1. Fig. 10 shows temporal evolution of morphological patterns during spinodal decomposition and subsequent coarsening. We obtain the result in Fig. 10(d) with only 1483 time step iterations. Fig. 11 shows $\|\phi^n - \phi^{n-1}\|_\infty$ and $0.1\Delta t^n$ versus the time t^n. We can observe that the time step changes adaptively according to the maximum norm of the difference between two consecutive numerical solutions.

5. Conclusions

In this paper, we proposed a fast and simple adaptive time step algorithm for the CH equation, which has multiple time scales for temporal evolution. To capture multiple time scale dynamics, we proposed an adaptive time step technique. Since the proposed time adaptivity is based on the discrete maximum norm of the difference between two consecutive time step numerical solutions, the algorithm is very simple. If the norm is large or small, then small or large time steps are used, respectively. Furthermore, after a small modification of the pre-existing codes for the CH equation, the new adaptive algorithm can easily be implemented. The one-, two-, and three-dimensional numerical experiments demonstrated the effectiveness of the adaptive time-stepping algorithm.

Acknowledgments

Y.B. Li is supported by National Natural Science Foundation of China (Nos. 11601416, and 11631012). The corresponding author (J.S. Kim) was supported by Korea University Future Research Grant. The authors greatly appreciate the reviewers for their constructive comments and suggestions, which have improved the quality of this paper.

References

[1] Y. Li, D. Jeong, J. Shin, J. Kim, A conservative numerical method for the Cahn–Hilliard equation with Dirichlet boundary conditions in complex domains, Comput. Math. Appl. 65 (2013) 102–115.
[2] J. Bosch, D. Kay, M. Stoll, A.J. Wathen, Fast solvers for Cahn–Hilliard inpainting, SIAM J. Imaging Sci. 7 (2014) 67–97.
[3] L. Cueto-Felgueroso, J. Peraire, A time-adaptive finite volume method for the Cahn–Hilliard and Kuramoto–Sivashinsky equations, J. Comput. Phys. 227 (2008) 9985–10017.
[4] J.W. Cahn, On spinodal decomposition, Acta Metall. 9 (1961) 795–801.
[5] D. Lee, J.Y. Huh, D. Jeong, J. Shin, A. Yun, J.S. Kim, Physical, mathematical, and numerical derivations of the Cahn–Hilliard equation, Comput. Mater. Sci. 81 (2014) 216–225.
[6] L. Ju, J. Zhang, Q. Du, Fast and accurate algorithms for simulating coarsening dynamics of Cahn–Hilliard equations, Comput. Mater. Sci. 108 (2015) 272–282.
[7] R. Choksi, M.A. Peletier, J.F. Williams, On the phase diagram for microphase separation of diblock copolymers: an approach via a nonlocal Cahn–Hilliard functional, SIAM J. Appl. Math. 69 (2009) 1712–1738.
[8] A. Bertozzi, S. Esedoğlu, A. Gillette, Inpainting of binary images using the Cahn–Hilliard equation, IEEE Trans. Image Process. 16 (2007) 285–291.
[9] Y. Li, J. Shin, Y. Choi, J. Kim, Three-dimensional volume reconstruction from slice data using phase-field models, Comput. Vis. Image Underst. 137 (2015) 115–124.
[10] D. Carolan, H.M. Chong, A. Ivankovic, A.J. Kinloch, A.C. Taylor, Co-continuous polymer systems: A numerical investigation, Comput. Mater. Sci. 98 (2015) 24–33.
[11] S. Zhou, M. Wang, Multimaterial structural topology optimization with a generalized Cahn–Hilliard model of multiphase transition, Struct. Multidiscip. Optim. 33 (2007) 89–111.
[12] M.A. Zaeem, H.E. Kadiri, M.F. Horstemeyer, M. Khafizov, Z. Utegulov, Effects of internal stresses and intermediate phases on the coarsening of coherent precipitates: A phase-field study, Curr. Appl. Phys. 12 (2012) 570–580.
[13] S.M. Wise, J.S. Lowengrub, H.B. Frieboes, V. Cristini, Three-dimensional multispecies nonlinear tumor growth-I: model and numerical method, J. Theoret. Biol. 253 (2008) 524–543.

[14] H. Heida, On the derivation of thermodynamically consistent boundary conditions for the Cahn–Hilliard–Navier–Stokes system, Internat. J. Engrg. Sci. 62 (2013) 126–156.
[15] Y. Li, J,-I. Choi, J. Kim, Multi-component Cahn–Hilliard system with different boundary conditions in complex domains, J. Comput. Phys. 323 (2016) 1–16.
[16] Y. Li, J,-I. Choi, J. Kim, A phase-field fluid modeling and computation with interfacial profile correction term, Commun. Nonlinear Sci. Numer. Simul. 30 (2016) 84–100.
[17] F. Guillén-González, G. Tierra, Second order schemes and time-step adaptivity for Allen–Cahn and Cahn–Hilliard models, Comput. Math. Appl. 68 (2014) 821–846.
[18] Z. Zhang, Z. Qiao, An adaptive time-stepping strategy for the Cahn–Hilliard equation, Commun. Comput. Phys. 11 (2012) 1261–1278.
[19] H. Gomez, V.M. Calo, Y. Bazilevs, T.J.R. Hughes, Isogeometric analysis of the Cahn–Hilliard phase-field model, Comput. Methods Appl. Mech. Engrg. 197 (2008) 4333–4352.
[20] R.H. Stogner, G.F. Carey, B.T. Murray, Approximation of Cahn–Hilliard diffuse interface models using parallel adaptive mesh refinement and coarsening with C^1 elements, Internat. J. Numer. Methods Engrg. 76 (2008) 636–661.
[21] E. Kuhl, D.W. Schmidt, Computational modeling of mineral unmixing and growth, Comput. Mech. 39 (2007) 439–451.
[22] D.J. Eyre, Unconditionally gradient stable time marching the Cahn–Hilliard equation, in: Computational and Mathematical Models of Microstructural Evolution (San Francisco, CA, 1998), Mater. Res. Soc. Sympos. Proc. Warrendale, PA, vol. 529, 1998, pp. 39–46.
[23] U. Trottenberg, A. Schüller, C. Oosterlee, Multigrid, Academic Press, New York, 2000.
[24] J.S. Kim, A numericalmethod for the Cahn–Hilliard equation with a variable mobility, Commun. Nonlinear Sci. Numer. Simul. 12 (2007) 1560–1571.
[25] H. Song, Energy stable and large time-stepping methods for the Cahn–Hilliard equation, Int. J. Comput. Math. 92 (2015) 2091–2108.
[26] G. Sheng, T. Wang, Q. Du, K.G. Wang, Z.K. Liu, L.Q. Chen, Coarsening kinetics of a two phase mixture with highly disparate diffusion mobility, Commun. Comput. Phys. 8 (2010) 249–264.
[27] L. He, Y. Liu, A class of stable spectral methods for the Cahn–Hilliard equation, J. Comput. Phys. 228 (2009) 5101–5110.
[28] D. Furihata, A stable and conservative finite difference scheme for the Cahn–Hilliard equation, Numer. Math. 87 (2001) 675–699.
[29] J.Z. Zhu, L.Q. Chen, J. Shen, V. Tikare, Coarsening kinetics from a variable-mobility Cahn–Hilliard equation: Application of a semi-implicit Fourier spectral method, Phys. Rev. E 60 (1999) 3564–3572.
[30] J.S. Kim, Phase-field models for multi-component fluid flows, Commun. Comput. Phys. 12 (2012) 613–661.