



Efficient method for phase-field model with finite interface dissipation



Geng Zhang*, Dan Cai, Yong Du

State Key Lab of Powder Metallurgy, Central South University, 410083 Changsha, China

ARTICLE INFO

Article history:

Received 20 November 2015
Received in revised form 2 March 2016
Accepted 3 March 2016

Keywords:

Phase-field
Interface dissipation
Efficiency
Numerical stability

ABSTRACT

In order to increase the time step and ensure the stability of the numerical solution, the error balancing method (EBM) is presented to solve the phase-field model with finite interface dissipation. An error is introduced to the phase-field to balance the error introduced to its gradient term in discretization. For solving this model, the EBM in the explicit scheme improves numerical stability and simulation efficiency with the same accuracy as the traditional scheme. Both theory and realistic tests demonstrate that the EBM keeps the solute conservation effectively. An intermediate variable that denotes concentration is introduced in order to ensure the phase concentration equations have the same form within both the interface and the bulk. Then, the EBM is extended to the implicit scheme. Two cases presented in this paper demonstrate that the EBM in the explicit scheme can enhance efficiency by 120 times compared with the traditional scheme; the EBM in the implicit scheme can enhance efficiency by 5 times compared with it in the explicit scheme.

© 2016 Elsevier B.V. All rights reserved.

1. Introduction

For the past few decades, the phase-field approach has played a useful role in various evolution processes of materials, including nucleation, precipitation, crystal growth, deformation, elasticity, fracture [1–9]. Recently, the phase-field model with finite interface dissipation [10,11] has been developed in the framework of the multiphase-field (MPF) formalism [12,13]. Each phase concentration is assigned by a kinetic equation to account for interface dissipation, rendering it novel. This differs from simply applying an extra condition for solute partition among the phases as in the traditional models: the condition of a given partition or the condition of equal diffusion potentials. Thus, the extra calculation for solute partition in the interface can be avoided in numerical simulations. There are two features for the phase-field model with finite interface dissipation: (i) various kinetic processes from the chemical equilibrium to strong non-equilibrium phase transformations can be nicely described, and (ii) the CALPHAD (CALCulation of PHASE Diagram) thermodynamic database can be incorporated directly into the phase-field simulation [14].

Convergence, stability, accuracy, and efficiency should be considered comprehensively for numerical simulations to make them reach an acceptable balance. For solving this model, the stability condition is more rigorous than the convergence condition (for details, please see Section 2). Practical numerical calculation

demonstrates that the time step ensuring stability is much less than the time step ensuring convergence. In order to guarantee both convergence and numerical stability, we must set sufficiently small time step and space step, which greatly influences simulation efficiency. The time consumption of numerical simulations mainly depends on the number of main loops. The smaller the time step is, the more the main loops are and the lower the efficiency becomes; the reverse applies as well. Therefore, increasing the time step is crucial for improving efficiency in numerical simulations.

In order to increase the time step and reduce the number of main loops, the error balancing method (EBM) is presented to solve this model, which replaces $\phi_x(G_0)$ (G_0 is a grid point in the interface) with an weighted average of ϕ_x in the phase concentration equations to discretize the gradient term. This method balances the errors in discretization, increases the time step ensuring stability, as well as decreases the numbers of main loops. This results in a notable improvement in simulation efficiency. Two cases simulated by this model are employed to test the EBM. The solidification of Al–Cu alloy is used to test the EBM in the explicit scheme; the coarsening and ripening process in Si-0.135As is used to test the EBM in the implicit scheme. The simulation results show that the EBM greatly improved simulation efficiency when solving this model. This paper focuses on improving simulation efficiency under the premise of ensuring numerical stability; it establishes an important foundation for the wide application of this model in various phase transition processes.

* Corresponding author. Tel.: +86 731 88836213; fax: +86 731 8710855.

E-mail address: zhangziying04@163.com (G. Zhang).

2. Error balancing method (EBM)

For a binary system with dual phases, the evolution equations of the phase-field model with finite interface dissipation can be written as:

$$\frac{\partial \phi_\alpha}{\partial t} = K_{\alpha\beta} \left\{ \sigma_{\alpha\beta} \left[\nabla^2 \phi_\alpha + \frac{\pi^2}{\eta_{\alpha\beta}^2} \left(\phi_\alpha - \frac{1}{2} \right) \right] + \frac{\pi}{\eta_{\alpha\beta}} \sqrt{\phi_\alpha(1-\phi_\alpha)} \Delta g_{\alpha\beta}^{phi} \right\}, \quad (1a)$$

$$\phi_\alpha \frac{\partial c_\alpha}{\partial t} = \nabla \bullet (\phi_\alpha D_\alpha \nabla c_\alpha) + P_{\alpha\beta} \phi_\alpha \phi_\beta (\tilde{\mu}_\beta - \tilde{\mu}_\alpha) + \phi_\alpha \dot{\phi}_\alpha (c_\beta - c_\alpha), \quad (1b)$$

$$\phi_\beta \frac{\partial c_\beta}{\partial t} = \nabla \bullet (\phi_\beta D_\beta \nabla c_\beta) + P_{\alpha\beta} \phi_\alpha \phi_\beta (\tilde{\mu}_\alpha - \tilde{\mu}_\beta) + \phi_\beta \dot{\phi}_\beta (c_\alpha - c_\beta), \quad (1c)$$

where $K_{\alpha\beta} = \left[1 + \frac{\pi^2 \mu_{\alpha\beta}}{8P_{\alpha\beta} \eta_{\alpha\beta}} (c_\alpha - c_\beta)^2 \right]^{-1} \mu_{\alpha\beta}$, $\Delta g_{\alpha\beta}^{phi} = f_\beta - f_\alpha - (\phi_\alpha \mu_\alpha + \phi_\beta \mu_\beta)(c_\beta - c_\alpha)$, and $\tilde{\mu}_\alpha = \frac{\partial f_\alpha}{\partial c_\alpha}$ are the kinetic coefficient, the driving force, and the diffusion potential (different from the chemical potential), respectively; $\sigma_{\alpha\beta}$, $\eta_{\alpha\beta}$, D_α , and $P_{\alpha\beta}$ are the interface energy (surface density), the interface width, the diffusivity, and the permeability (i.e. the dissipation coefficient), respectively.

The advantage of this model is that arbitrary phase concentration can be obtained by solving evolution equations rather than solving Fick's diffusion equation and partition condition in the previous research [15]. From the phase concentration equations, it is known that $\nabla \phi_\alpha$ tends to 0 when ϕ_α tends to 0, theoretically. If we set sufficiently small space step and time step, the numerical solution of Eqs. (1a)–(1c) behaves well, but this greatly increases the amount of calculations. If the grid points in the interface are not sufficient, the simulation results will lose stability because ϕ_α at a grid point near phase β in the interface is too small. Thus, when the traditional scheme is used to solve this model, it is very difficult to make numerical stability and simulation efficiency reach an expected balance.

In order to overcome the difficulties above, we present the EBM to discretize the phase concentration equations, i.e., we replace $\phi_\alpha(G_0)$ with $M_{n+1}(\phi_\alpha(G_0), \dots, \phi_\alpha(G_n))$, which is an weighted average of ϕ_α . The EBM will balance the errors of $\phi_\alpha(G_0)$ and $\nabla \bullet (\phi_\alpha D_\alpha \nabla c_\alpha)(G_0)$, due to discretization.

The grid points that are used to discretize the gradient term will also be used to calculate the weighted average M_{n+1} . Let $\Delta x = \Delta y = \Delta z = h$. For 1-D case:

$$\nabla \bullet (\phi D \nabla c)|_i = \frac{1}{2h^2} [(\phi_{i+1} D_{i+1} + \phi_i D_i)(c_{i+1} - c_i) - (\phi_i D_i + \phi_{i-1} D_{i-1})(c_i - c_{i-1})]; \quad (2)$$

$\phi(G_0)$ will be replaced with:

$$\phi(G_0) \rightarrow \lambda \phi_i + \frac{1-\lambda}{2} (\phi_{i+1} + \phi_{i-1}). \quad (3)$$

For 2-D case, if we apply five-point scheme to discretize the gradient term:

$$\nabla \bullet (\phi D \nabla c)|_{ij} = \frac{1}{2h^2} \begin{bmatrix} (\phi_{i+1,j} D_{i+1,j} + \phi_{i,j} D_{i,j})(c_{i+1,j} - c_{i,j}) \\ -(\phi_{i,j} D_{i,j} + \phi_{i-1,j} D_{i-1,j})(c_{i,j} - c_{i-1,j}) \\ +(\phi_{i,j+1} D_{i,j+1} + \phi_{i,j} D_{i,j})(c_{i,j+1} - c_{i,j}) \\ -(\phi_{i,j} D_{i,j} + \phi_{i,j-1} D_{i,j-1})(c_{i,j} - c_{i,j-1}) \end{bmatrix}; \quad (4)$$

$\phi(G_0)$ will be replaced with:

$$\phi(G_0) \rightarrow \lambda \phi_{ij} + \frac{1-\lambda}{4} (\phi_{i+1,j} + \phi_{i-1,j} + \phi_{i,j+1} + \phi_{i,j-1}). \quad (5)$$

If we apply 9-point scheme to discretize the gradient term:

$$\nabla \bullet (\phi D \nabla c)|_{ij} = \frac{2}{3} \frac{1}{2h^2} \begin{bmatrix} (\phi_{i+1,j} D_{i+1,j} + \phi_{i,j} D_{i,j})(c_{i+1,j} - c_{i,j}) \\ -(\phi_{i,j} D_{i,j} + \phi_{i-1,j} D_{i-1,j})(c_{i,j} - c_{i-1,j}) \\ +(\phi_{i,j+1} D_{i,j+1} + \phi_{i,j} D_{i,j})(c_{i,j+1} - c_{i,j}) \\ -(\phi_{i,j} D_{i,j} + \phi_{i,j-1} D_{i,j-1})(c_{i,j} - c_{i,j-1}) \end{bmatrix} + \frac{1}{3} \frac{1}{4h^2} \begin{bmatrix} (\phi_{i+1,j+1} D_{i+1,j+1} + \phi_{i,j} D_{i,j})(c_{i+1,j+1} - c_{i,j}) \\ -(\phi_{i,j} D_{i,j} + \phi_{i-1,j-1} D_{i-1,j-1})(c_{i,j} - c_{i-1,j-1}) \\ +(\phi_{i+1,j-1} D_{i+1,j-1} + \phi_{i,j} D_{i,j})(c_{i+1,j-1} - c_{i,j}) \\ -(\phi_{i,j} D_{i,j} + \phi_{i-1,j+1} D_{i-1,j+1})(c_{i,j} - c_{i-1,j+1}) \end{bmatrix}; \quad (6)$$

$\phi(G_0)$ will be replaced with:

$$\phi(G_0) \rightarrow \lambda \phi_{ij} + \frac{1-\lambda}{8} (\phi_{i+1,j} + \phi_{i-1,j} + \phi_{i,j+1} + \phi_{i,j-1} + \phi_{i+1,j+1} + \phi_{i-1,j-1} + \phi_{i+1,j-1} + \phi_{i-1,j+1}). \quad (7)$$

For 3-D case, if we apply 7-point scheme to discretize the gradient term:

$$\nabla \bullet (\phi D \nabla c)|_{ijk} = \frac{1}{h^2} \begin{bmatrix} (\phi_{i+1,j,k} D_{i+1,j,k} + \phi_{i,j,k} D_{i,j,k})(c_{i+1,j,k} - c_{i,j,k}) \\ -(\phi_{i,j,k} D_{i,j,k} + \phi_{i-1,j,k} D_{i-1,j,k})(c_{i,j,k} - c_{i-1,j,k}) \\ +(\phi_{i,j+1,k} D_{i,j+1,k} + \phi_{i,j,k} D_{i,j,k})(c_{i,j+1,k} - c_{i,j,k}) \\ -(\phi_{i,j,k} D_{i,j,k} + \phi_{i,j-1,k} D_{i,j-1,k})(c_{i,j,k} - c_{i,j-1,k}) \\ +(\phi_{i,j,k+1} D_{i,j,k+1} + \phi_{i,j,k} D_{i,j,k})(c_{i,j,k+1} - c_{i,j,k}) \\ -(\phi_{i,j,k} D_{i,j,k} + \phi_{i,j,k-1} D_{i,j,k-1})(c_{i,j,k} - c_{i,j,k-1}) \end{bmatrix}; \quad (8)$$

$\phi(G_0)$ will be replaced with:

$$\phi(G_0) \rightarrow \lambda \phi_{ijk} + \frac{1-\lambda}{6} (\phi_{i+1,j,k} + \phi_{i-1,j,k} + \phi_{i,j+1,k} + \phi_{i,j-1,k} + \phi_{i,j,k+1} + \phi_{i,j,k-1}). \quad (9)$$

If we apply 19-point scheme to discretize the gradient term:

$$\nabla \bullet (\phi D \nabla c)|_{ijk} = \frac{2}{3} \frac{1}{h^2} \begin{bmatrix} (\phi_{i+1,j,k} D_{i+1,j,k} + \phi_{i,j,k} D_{i,j,k})(c_{i+1,j,k} - c_{i,j,k}) \\ -(\phi_{i,j,k} D_{i,j,k} + \phi_{i-1,j,k} D_{i-1,j,k})(c_{i,j,k} - c_{i-1,j,k}) \\ +(\phi_{i,j+1,k} D_{i,j+1,k} + \phi_{i,j,k} D_{i,j,k})(c_{i,j+1,k} - c_{i,j,k}) \\ -(\phi_{i,j,k} D_{i,j,k} + \phi_{i,j-1,k} D_{i,j-1,k})(c_{i,j,k} - c_{i,j-1,k}) \\ +(\phi_{i,j,k+1} D_{i,j,k+1} + \phi_{i,j,k} D_{i,j,k})(c_{i,j,k+1} - c_{i,j,k}) \\ -(\phi_{i,j,k} D_{i,j,k} + \phi_{i,j,k-1} D_{i,j,k-1})(c_{i,j,k} - c_{i,j,k-1}) \end{bmatrix} + \frac{1}{3} \frac{1}{8h^2} \begin{bmatrix} (\phi_{i+1,j+1,k} D_{i+1,j+1,k} + \phi_{i,j,k} D_{i,j,k})(c_{i+1,j+1,k} - c_{i,j,k}) \\ -(\phi_{i,j,k} D_{i,j,k} + \phi_{i-1,j-1,k} D_{i-1,j-1,k})(c_{i,j,k} - c_{i-1,j-1,k}) \\ +(\phi_{i+1,j-1,k} D_{i+1,j-1,k} + \phi_{i,j,k} D_{i,j,k})(c_{i+1,j-1,k} - c_{i,j,k}) \\ -(\phi_{i,j,k} D_{i,j,k} + \phi_{i-1,j+1,k} D_{i-1,j+1,k})(c_{i,j,k} - c_{i-1,j+1,k}) \\ +(\phi_{i,j,k+1} D_{i,j,k+1} + \phi_{i,j,k} D_{i,j,k})(c_{i,j,k+1} - c_{i,j,k}) \\ -(\phi_{i,j,k} D_{i,j,k} + \phi_{i-1,j,k-1} D_{i-1,j,k-1})(c_{i,j,k} - c_{i-1,j,k-1}) \\ +(\phi_{i-1,k+1} D_{i-1,j,k+1} + \phi_{i,j,k} D_{i,j,k})(c_{i-1,j,k+1} - c_{i,j,k}) \\ -(\phi_{i,j,k} D_{i,j,k} + \phi_{i+1,k-1} D_{i+1,j,k-1})(c_{i,j,k} - c_{i+1,j,k-1}) \\ +(\phi_{i+1,j,k+1} D_{i+1,j,k+1} + \phi_{i,j,k} D_{i,j,k})(c_{i+1,j,k+1} - c_{i,j,k}) \\ -(\phi_{i,j,k} D_{i,j,k} + \phi_{i-1,j,k-1} D_{i-1,j,k-1})(c_{i,j,k} - c_{i-1,j,k-1}) \\ +(\phi_{i+1,j,k-1} D_{i+1,j,k-1} + \phi_{i,j,k} D_{i,j,k})(c_{i+1,j,k-1} - c_{i,j,k}) \\ -(\phi_{i,j,k} D_{i,j,k} + \phi_{i-1,j,k+1} D_{i-1,j,k+1})(c_{i,j,k} - c_{i-1,j,k+1}) \end{bmatrix}; \quad (10)$$

$\phi(G_0)$ will be replaced with:

$$\phi(G_0) \rightarrow \lambda \phi_{ijk} + \frac{1-\lambda}{18} \begin{bmatrix} \phi_{i+1,j,k} + \phi_{i-1,j,k} + \phi_{i,j+1,k} + \phi_{i,j-1,k} + \phi_{i,j,k+1} + \phi_{i,j,k-1} \\ +\phi_{i+1,j+1,k} + \phi_{i-1,j-1,k} + \phi_{i-1,j-1,k} + \phi_{i+1,j-1,k} \\ +\phi_{i+1,k+1} + \phi_{i-1,k+1} + \phi_{i-1,k-1} + \phi_{i+1,k-1} \\ +\phi_{i+1,j,k+1} + \phi_{i-1,j,k+1} + \phi_{i-1,j,k-1} + \phi_{i+1,j,k-1} \end{bmatrix}. \quad (11)$$

Next, we will investigate why the EBM increases the time step. Considering that $\phi(\vec{x})$ is 2-order differentiable with respect to spatial variable \vec{x} , $\phi(\vec{x})$ is convex along any direction when $\phi(\vec{x}) \rightarrow 0$.

Download English Version:

<https://daneshyari.com/en/article/7958800>

Download Persian Version:

<https://daneshyari.com/article/7958800>

[Daneshyari.com](https://daneshyari.com)