Study of Coherent Solid Dendritic Precipitate Transformation Using a Phase-Field Model: Implementation of a Parallel Multigrid Scheme

Jun Wu¹, Zhipeng Guo², Xiaopeng Zhang¹, Bin Su¹, Zhenhong Wang¹, Chao Luo¹, *

¹Institute of Materials, China Academy of Engineering Physics, Jiangyou 621907, China
²School of Materials Science and Engineering, Tsinghua University, Beijing 100084, China

* Corresponding author: luochaoboss@sohu.com

Abstract. To study the behavior of solid dendritic growth involving a cubic-to-cubic coherent precipitate transformation, a phase-field model incorporating the coherent elastic strain energy based on a solidification model, is proposed and a parallel multigrid scheme is developed to solve the elasticity problem in the model. Numerical tests in 2D demonstrate that the algorithm could shorten the elapsed time by two order of magnitude, making it feasible to carry out investigation on the underlying physics of the phase transformation. The simulation results show that the appearance of the elastic energy promotes precipitate’s dendritic growth with preferential directions of <11>. With the interfacial energy anisotropy is considered, the ultimate morphology of precipitate is highly dependent on competition between anisotropy of the elastic energy and the interface energy.

1. Introduction
For the coherent solid-solid transformation, it is well known that the lattice mismatch between the parent and product phases generates elastic strain energy, which may have a significant effect on the morphology of the system. For Cu-Zn alloys in which parent and product phases have the similar lattice structure, solid dendrite precipitate was observed, due to the appearance of the elastic strain energy [1].

In recent years, the phase-field method has been developed into one of the most powerful tools for the simulation of the microstructure evolution. It is widely used to simulate the dendritic growth during solidification [2-3] and coherent precipitate growth in solid-state phase transformation [4-5]. The driving force of the microstructure is the minimization of the free energy of the system. As coherent precipitate growth as concerned, not only reduction in bulk energy and interfacial energy but also reduction in the long-range interaction i.e. elastic strain energy need to be considered as the driving force. Accordingly, some accurate and efficient models have been developed for calculating the elastic energy. In the case of elasticity homogeneity, the elastic equilibrium equation could be solved efficiently by the Fourier spectral method which was developed by Khachaturyan [6] and widely used by other researchers [4-5]. Meanwhile, for the elasticity inhomogeneity systems whose
solution becomes more complicated for Khachaturyan's method, a conjugate gradient scheme [7] and a iterative-perturbation scheme [8] based on the Fourier spectral method and a Gauss-Seidal iterative scheme based on the finite difference adaptive mesh technique [9] were developed.

Multigrid is one of the most efficient solvers for PDE and has a complexity of $O(N \log N)$ while the commonly used FFT's complexity is $O(N \log N)$, where $N$ is the total number of grid points [10]. In this way, multigrid works efficiently and is successful in solving many mathematical and physical problems [11-13]. In this paper, based on the Karma’s solidification model we propose a phase-field model incorporating the elastic energy into the total free energy. We implement the multigrid method with a parallel computing strategy to solve the elastic displacement equation in real space and examine validity of the algorithm in terms of computing efficiency and accuracy. And simulations are performed to investigate the effect of elastic energy on the dendrite-like precipitate's growth.

2. Phase-field Model

2.1. Governing equations

The phase-field model for coherent precipitate growth proposed here is based on the model for solidification of binary dilute alloy developed by Karma and co-workers [14] in which parameters of materials is related by the thin-interface limit analysis. We need to begin with the construction of the total free energy for the system.

The bulk free energy density of binary alloy of A and B atoms can be expressed by

$$F(\varphi, c, T) = \int \left[ \frac{\sigma}{2} \nabla \varphi \cdot \nabla \varphi + f_{ab}(\varphi, c, T) + f^{el}(\varphi) \right] dV$$

(1)

Where $\frac{\sigma}{2} |\nabla \varphi|^2$ represents the interfacial gradient energy density, and $f^{el}(\varphi)$, a function of the phase variable, denotes the elastic stain energy density. $\varphi$, $c$ and $T$ is the phase variable, solute concentration, and temperature, respectively.

For simplification, the temperature throughout the simulation domain is set to be a constant, i.e. a constant undercooling is employed. According to Ref. [14], governing equations for phase field and solute field when coupling the elastic strain energy can be expressed in the original form of:

$$\frac{\partial \varphi}{\partial t} = -K_e \frac{\partial F}{\partial \varphi}$$

(2)

$$\frac{\partial c}{\partial t} = \nabla \cdot (K_e \nabla \frac{\partial F}{\partial c} - \bar{j}_{at})$$

(3)

Where $\bar{j}_{at}$ is the anti-trapping current term.

Define the dimensionless definition of solute concentration and temperature as follows:

$$U = \left[ \frac{2c / c_0}{1 + k - (1 - k) \varphi} \right] \frac{1}{1 - k}$$

$$\theta = \frac{m c_0^0 - (T - T_m)}{m c_0^0 (1 - k)}$$

(4)

(5)
Where \( c_q^0 \) defines the equilibrium concentration in the parent phase side and \( mc_q^0(1-k) \) is the scaling temperature \( \Delta T_0 \).

Substitute \( F \), \( U \) and \( \theta \) into Eqs. (2) And (3) with dimensionless space and time variables, and we have

\[
\frac{1+(1-k)U}{2} \frac{\partial \varphi}{\partial t} = \bar{\nabla} \left[ A(\phi)^2 \bar{\nabla} \varphi \right] - \frac{\partial}{\partial \bar{x}} \left[ A(\phi)A'(\phi) \frac{\partial \varphi}{\partial \bar{y}} \right] + \frac{\partial}{\partial \bar{y}} \left[ A(\phi)A'(\phi) \frac{\partial \varphi}{\partial \bar{x}} \right] + \varphi - \phi^\alpha - \lambda(1-\varphi^\alpha)(\theta + U) - \frac{1}{H} \frac{\partial f^\alpha(\varphi)}{\partial \varphi} \tag{6}
\]

And

\[
\frac{1+k}{2} \frac{\partial U}{\partial t} = \bar{\nabla} \left[ \bar{D} \frac{1-\varphi}{2} \bar{\nabla} U + \frac{1}{\sqrt{2}} \left[ 1+(1-k)U \right] \frac{\partial \varphi}{\partial \bar{t}} \bar{\nabla} \varphi \right] + \frac{1}{2} \bar{D} \left\{ \varphi \left[ 1+(1-k)U \right] \right\} \tag{7}
\]

respectively, where the scaling parameter is given by \( \lambda = -\frac{15}{16} \frac{Lmc_q^0(1-k)}{HT_0} \). \( A(\phi) \) is the anisotropy function of the interfacial energy. \( k \) is the equilibrium solute concentration partition coefficient according to the phase diagram, and \( \bar{D} \), \( \bar{x} \) and \( \bar{t} \) is the dimensionless solute diffusivity, space and time respectively. \( L \) is the latent heat of phase transformation for the solvent and \( m \) is the slope of solubility in the phase diagram.

2.2 Elastic strain energy in the model

The elastic strain energy density \( f^{el}(\varphi) \) in Eq. (6) is calculated according to micro-elasticity theory for coherent precipitates developed by Khachaturyan [6] with details given in the following. The local stress free transformation strain \( \varepsilon_q^0(r) \) is phase-dependent and introduced in form of

\[
\varepsilon_q^0(r) = \varepsilon_q^0(p) \frac{1+g(\varphi)}{2} \tag{8}
\]

Where \( \varepsilon_q^0(p) \) is the strain tensor dependent on the lattice misfit between the precipitate and parent phases. For a cubic system, \( \varepsilon_q^0(p) = \varepsilon^{00}_q \delta_q \) where \( \varepsilon^{00}_q \) is the lattice misfit. And \( g(\varphi) \) interpolates between -1 and 1 in parent and precipitate phases, respectively.

The elastic strain \( \varepsilon_q^{el} \) can be expressed by the difference between the total strain \( \varepsilon_q^{tot} \) and the stress-free strain \( \varepsilon_q^0 \):

\[
\varepsilon_q^{el} = \varepsilon_q^{tot} - \varepsilon_q^0 \tag{9}
\]

Where \( \varepsilon_q^{tot} \) is the sum of the macro homogeneous strain and the local heterogeneous strain \( \varepsilon_q^{tot} = \bar{\varepsilon}_q + \delta \varepsilon_q \). The macro homogeneous strain is assumed to zero in this study. And the local heterogeneous is related to the heterogeneous displacement according the elastic theory, that is
Where $u_i$ defines the displacement in $x_i$ direction.

According to Hooke's law, the elastic stress and elastic strain energy density can be given by

$$\sigma_{ij}^{el} = C_{ijkl} \varepsilon_{ijkl}^{el} = C_{ijkl} (\varepsilon_{ij}^0 + \delta \varepsilon_{ij} - \varepsilon_{ij}^0)$$

And

$$f_{ij}^{el} = \frac{1}{2} C_{ijkl} \varepsilon_{ijkl}^{el} = \frac{1}{2} C_{ijkl} (\varepsilon_{ij}^0 + \delta \varepsilon_{ij} - \varepsilon_{ij}^0) (\varepsilon_{ij}^0 + \delta \varepsilon_{ij} - \varepsilon_{ij}^0)$$

Respectively, where $C_{ijkl} = \frac{1+\tilde{g}(\phi)}{2} C_{ijkl}^p + \frac{1-\tilde{g}(\phi)}{2} C_{ijkl}^q$ is the elastic modulus tensor. $C_{ijkl}^p$ and $C_{ijkl}^q$ are the elastic constants of the precipitate and parent phases, respectively.

During the phase transformation, the mechanical equilibrium is achieved quickly and gives

$$\frac{\partial \varepsilon_{ij}^{el}}{\partial x_j} = 0$$

Substituting Eqs. (8), (10) and (11) into (13) gives the displacement equation

$$\frac{1}{2} \frac{\partial}{\partial x_j} \left[ C_{ijkl} \left( \frac{\partial u_k}{\partial x_i} + \frac{\partial u_i}{\partial x_k} \right) \right] = \frac{\partial}{\partial x_j} \left[ C_{ijkl} \varepsilon_{ijkl}^0 (p \frac{1+g(\phi)}{2}) \right]$$

At a given time step, the phase variable $\phi$ is determined and the displacement which is only dependent on the space variable $x$ can be solved under a boundary condition. Therefore, we can obtain $\delta \varepsilon_{ij}$ by Eq. (10), and then the elastic energy density can be calculated by Eq. (12).

### 3. Numerical scheme

For discretization of the phase-field equations i.e. Eqs. (6) An (7), the employed scheme is the same with the one used in Ref. [15]. That is, explicit time stepping of the forward Euler method with a constrain limit is employed, the first derivative with respect to space is discretized by the center difference scheme, and finite volume method is used to discretize the divergence operator, which is also applied to the Eq. (14) in each direction of $u$ vector.

A standard multigrid method for solving the linear equation $A_n X_n = F_n$, combined with a Gauss-Seidal type iterative smoother, is applied to solve the discretized form of Eq. (14), with details described in Ref. [16]. In the implementation of the algorithm, a recursive V-cycle function is employed with the solving procedure that starts on the finest level, goes down to the coarsest level and then goes back to the finest level. A four-point average restriction and piecewise constant interpolation is used for grid transferring which is an important ingredient in the method. In order to evaluate convergence for the multigrid method, the residual $R = \|F - AX\|_F$ for Eq. (14) in the domain is calculated. A convergence criterion $R < \text{eps}$ is employed where $\text{eps}$ is an accuracy tolerance with a small value, typically $1e-4$. 


The parallel scheme is implemented by dividing the discrete problem domain into a series of square sub-domains on each grid level. Each sub-domain contains an interior ghost-cell layer which can receive data of the neighboring grids at the same grid level. In order to obtain a good parallel efficiency and reduce the data communication, each domain is distributed on one core, which means that the grid size of sub-domain is selected carefully according to the number of applied parallel cores. The parallel implementation on the coarsest grid level requires particular attention to ensure at least one grid point for each core. Here, the selected coarsest grid level is the one on which 2×2 grid points are distributed for each sub-domain and core. It should be noted that all the multigrid procedures such as relaxation, restriction, interpolation and calculation of residual are implemented in parallel for all relevant data.

During the relaxation at each grid level, a Gauss-Seidel-type smoother is employed, due to its most popular use in the context of multigrid. Since the frequently used Gauss-Seidel red-black (GS-RB) smoother cannot be carried out fully in parallel when the nine-point stencils are involved during the discretization, we extend the solver to the four-color pointwise Gauss-Seidel (GS-FC) relaxation described in Ref. [10]. There are four quarter-steps for relaxation and only one type of points are smoothed for the discretized form of Eq. (14) using the updated value in the neighbor points at each step. In this way, for each type of points, there is no data dependencies and each quarter-step relaxation can be executed fully in parallel. Data communication takes place after one iteration via the message-passing interface (MPI) libraries.

After solving the displacement equations, the variables at a new time step such as the elastic strain, elastic driving force, the phase variables and solute variables are calculated in parallel only on the finest grid level using the same grid partition strategy with the multigrid solving procedure. After the calculation, data at the ghost cells are updated by neighboring core.

The algorithm is coded on the block-structured library BoxLib and results are visualized using the software VisIt. The parallel computing is implemented on the supercomputer named Landau in Tsinghua University.

4. Numerical results and discussion

4.1. Convergence test of the multigrid solver on the elasticity problem

The problem domain \( \Omega \) is chosen as 256×256 cells and periodic boundary condition is applied for \( \phi \), \( U \), \( u \), and \( u_2 \). An round precipitate phase with a radius of \( R = 1.5\Delta x \) is seeded initially at the center of the domain. The dimensionless elastic constants for a coherent cubic-to-cubic phase transformation are chosen as \( C_{11}^c/GPa = 200 \cdot C_{12}^c/GPa = 100 \cdot C_{44}^c/GPa = 100 \cdot C_{11}^q/GPa = 250 \cdot C_{12}^q/GPa = 180 \) and \( C_{44}^q/GPa = 130 \), and the eigenstrain is chosen as \( \epsilon^{\text{en}} = 0.005 \). The initial seed evolves for \( 100\Delta t \) by solving the phase field equation and solute field equation without the effect of the elasticity using parameters including \( \Delta t = 0.6 \), \( \lambda = 10 \), \( \epsilon_s = 0 \), \( k = 0.15 \) and \( \theta = 0.75 \). The time step, \( \Delta t \), is set to be eighty percent of the maximum value required by the stable condition of Eq. (7). The anisotropy function is employed in the form of \( A(\omega) = 1 + \epsilon_s \cos(4\omega) \) and the anisotropy strength \( \epsilon_s = 0 \) is assumed. Therefore, the precipitate phase in a nearly round shape with diffuse interface is obtained (see Figure 1). The elastic effect is calculated for the resulting phase profile by the multigrid method.
The stress distribution of three components induced by coherent precipitation is presented in Figure 2. Take the stress component $\sigma_{i1}$ as an example to illustrate the stress distribution. Since lattice constant of the precipitate phase is larger, the precipitate phase is compressed in the center and the parent phase is forced to expand during the precipitation, resulting in a passive $\sigma_{i1}$, i.e. a compressible stress in the precipitate phase, and a tensile stress and compressible one in the parent phase along the y-direction and x-direction from the center, respectively. The result is as expected. Furthermore, the stress distribution shown in Figure 2 is very similar with the one in Ref. [17], demonstrating that the elastic effect solved by the multigrid method is believable.

To examine the convergence of the solution to the displacement equations, the stress components $\sigma_{i1}$ and $\sigma_{i2}$ in Figure 2 along x-direction with different solution tolerances varying from $10^{-1}$ to $10^{-8}$ is shown in Figure 3. The elastic stress converges when the residual is $10^{-4}$. When $\epsilon_{ps}$ is reduce to lower than $10^{-3}$, the V-cycles number increased significantly, indicating that much more computing time is needed to solve the equation. Therefore, in order to obtain a good combination of computing efficiency and accuracy, the solution tolerance is selected as $10^{-4}$ in the following kinetic case.
Figure 3. Stress distribution along x-direction using the different residuals: (a) for $\sigma_{11}$ and (b) for $\sigma_{22}$.

It should be noted that the convergence of the multigrid method is also dependent on V-cycle parameters including the number of the pre- and post-smooths, and bottom-relaxation. For the phase profile in Figure 1, Table 1 shows the employed V-cycle parameters in the form of $(a, b, c)$ where $a$, $b$, $c$ are the pre-smoothing, post-smoothing and bottom-relaxation iterations, respectively, and the corresponding required number of V-cycles and elapsed time for a solution tolerance of $\varepsilon_p=10^{-4}$ using a single core. Case 2$\#$ with a typical V-cycle parameter of 2 pre-smoothing, 2 post-smoothing and 8 bottom-relaxation iterations costs the least computing time. The increase of the number of the pre-smoothing or post-smoothing iterations will reduce the V-cycle number but introduce much more computing effort on the finest grid level, and therefore may require more computing time (seeing Case 6$\#$). Meanwhile, just increasing the number of bottom-relaxation iterations will introduce more V-cycles (seeing Case 8$\#$). Accordingly, the V-cycle best performed parameter of $(2, 2, 8)$ is chosen for the following kinetic case in terms of the computing efficiency.

Table 1. The effect of the parameter used in V-cycle on the convergence

| Case | V-cycle parameters | No. of V-cycles | Elapsed time (s) |
|------|--------------------|-----------------|------------------|
| 1$\#$ | (2,1,8)           | 60              | 4.4569 s         |
| 2$\#$ | (2,2,8)           | 48              | 4.2462 s         |
| 3$\#$ | (2,2,20)          | 48              | 4.2602 s         |
| 4$\#$ | (2,5,8)           | 35              | 4.6383 s         |
| 5$\#$ | (5,2,8)           | 35              | 4.6368 s         |
| 6$\#$ | (5,5,8)           | 29              | 5.1003 s         |
| 7$\#$ | (1,1,20)          | 80              | 4.7205 s         |
| 8$\#$ | (1,1,40)          | 80              | 4.7289 s         |
| 9$\#$ | (3,3,8)           | 38              | 4.5044 s         |

The GS-FC method directly on the uniform grid level (named as GS-FC-U) is also applied to the same problem, and the number of iterations required is up to 3612 with elapsed time increased by more than 10 times. Although the high-frequency error can be reduced rapidly by the standard iterative scheme Gauss-Seidal method, the low-frequency error whose length scale is larger than the grid size is reduced inefficiently. The multigrid scheme using levels with different grid sizes and combined with the standard iterative method GS-FC, can reduce the high-frequency error on the fine grid level and the low-frequency one by the coarse grid correction. Accordingly, the implementation of the multigrid scheme to the elastic problem can reduce the computing efforts and shorten the execution time effectively compared with the GS-FC-U method.
4.2. Parallel performance

In this section, kinetic phase-field simulations for coherent precipitate growth are implemented to evaluate the parallel performance and scalability of the proposed algorithm. A coupling constant for the contribution of the elastic effect to the driving force of phase evolution, is employed in a dimensionless form of \( \frac{2T_u}{\Delta t} GPa \) and properly set to be 10. The elastic constants, eigenstrain and other key parameters in the model are the same as that in Section 4.1. The phase profile in Figure 1 is used as the initial condition, and all tests are terminated when the simulation time reaches \(100\Delta \tau\). All simulations with different problem domain sizes and varying the number of cores employed \((N_p)\) from 1 to 256 are performed.

In Section 4.1, zero is used as the initial value for iteration to solve the elastic problem. Here for the kinetic case, efficiency is improved significantly by using the solution of the displacement equation at last time step as the approximation. Because there is a very small difference of phase field and solute filed between the two adjacent time steps, the solution resulting from the comparatively accurate approximation will converge quickly. Tests show that the number of V-cycles required can decrease by three quarters.

The execution time \((T_p)\) as a function of the number of the cores is shown in Figure 4 where the black lines show the ideal trend according to the linear scalability theory. It should be noted that the minimum number of cores required grows up to 4 for the case of 8196×8196 cells, since the memory requirement increases with the problem domain size. As expected, the calculation time decrease as the number of cores increase for all cases. However, the larger the problem domain size is, the less deviation from the ideal line with the increase of the core number is. For example, there is a significant deviation at \(pN = 256\) in the case of 8196×8196 cells but only the same situation takes place at \(pN = 16\) in the case of 256×256 cells. This can be explained by the fact that the ratio of computation and communication is higher for the case of a large problem domain size.

![Figure 4. Execution time versus the number of cores with different problem domain sizes](image)

Table 2 shows an alternative assessment of the parallel performance for some typical cases of a constant domain size using the speed-up \((S_p)\) defined as the ratio of the execution time on single core and on \(N_p\) parallel cores and efficiency \((E_p)\) defined as \(E_p = S_p/N_p\). As seen, in each case, the speed-up and efficiency increases and decreases with the increase of the number of the cores, respectively. The speed-up and efficiency get closer to the theoretic value i.e. \(T^i/T^p\) (the superscript denoting the number of core) and 1, respectively, for the case of a relative small \(N_p\) used at a fixed domain size or a relative large domain size at a fixed \(N_p\). For the case of 4096×4096 cells, when \(N_p = 64\), the parallel
computing still has a good performance with an efficiency of 85%, and even though the efficiency is only 56% when the number of cores increasing up to 256, the total execution time can be shortened to one third of that spent at $N_p = 64$. Therefore, the goal of parallel implementation to solve large scale problems in an acceptable execution time can be achieved via the proposed algorithm.

### Table 2. Parallel performance assessment for three typical cases of a fixed size

| $N_p$ | $T_p$ (s) | $S_p$ | $E_p$ | $T_p$ (s) | $S_p$ | $E_p$ | $T_p$ (s) | $S_p$ | $E_p$ |
|-------|-----------|-------|-------|-----------|-------|-------|-----------|-------|-------|
|       | (by GS-FC-U) |       |       | (by MG)   |       |       | (by MG)   |       |       |
| 1     | 1394.1    | 1     | 1     | 127.3     | 1     | 1     | 2056.6    | 1     | 1     |
| 4     | 351.1     | 3.97  | 0.99  | 33.2      | 3.83  | 0.96  | 529.0     | 3.89  | 0.97  |
| 16    | 96.0      | 14.52 | 0.91  | 11.1      | 11.47 | 0.72  | 143.6     | 14.32 | 0.90  |
| 64    | 36.8      | 37.9  | 0.59  | 6.9       | 18.45 | 0.29  | 43.8      | 46.95 | 0.73  |
| 256   | —         | —     | —     | —         | —     | —     | —         | —     | —     |

The result of the parallel performance using the GS-FC-U iteration directly for the case of 256×256 cells is also shown in the Table 3. Obviously, although the overall execution time of the direct GS-FC-U iteration is nearly one-order higher than the multigrid method’s, the speed-up and efficiency are much better than that obtained by the multigrid method, at a given core number. The reason is that there is only communication effort on the fine grid level in the direct GS-FC-U iteration while wasteful communications are not avoided on the coarse grid level in the multigrid method. However, it is predicted that the difference of the parallel performance between the two methods decreases with the increase of the problem domain size, which means that the combination of the parallel and multigrid scheme is very suitable for a relative large scale problem.

5. Simulations of morphological evolution effected by the elasticity energy

In this section, microstructure evolution of a single coherent precipitate is simulated using the proposed scheme and the effect of the elasticity energy on the shape of the precipitate is examined specially. A typical temporal evolution of the morphological pattern for a single coherent precipitate is shown in Figure 5 where the problem domain is 512×512 cells, the evolution time is set to be $t = 5000\Delta t$ and other parameters used are the same as that used in Section 4.1. With evolution time increasing, the precipitate with fourfold arms grows preferentially in the directions of $\vec{e}_{<11>}$, having a good agreement with results in Refs. [18-19]. The distribution of the solute concentration field is shown in Figure 5 (f), which is similar with that of a solidified dendrite.

To understand the formation of the precipitate pattern, two main resources of anisotropy which is considered as genes controlling microstructure evolution are analyzed here: (a) the anisotropy of the cubic elastic constants and (b) the surface energy anisotropy. The anisotropy of the cubic elastic constants can be defined by $\delta = \frac{2C_{44}}{C_{11}-C_{12}}$. The four-fold surface anisotropy function used is given in Section 4.1. In this case of Figure 5, the surface energy applied is isotropic i.e. $\epsilon_i = 0$, so the anisotropic precipitate growth results completely from the elasticity energy anisotropy.
Figure 5. Temporal evolution of the morphological pattern of a single coherent precipitate: (a) $40\Delta\tau$; (b) $500\Delta\tau$; (c) $1500\Delta\tau$; (d) $2500\Delta\tau$; (e) $5000\Delta\tau$; (f) the dimensionless solute concentration $\frac{c}{c^0}$.

Table 3. Elastic constants ($C_{ij} = C_{ij}^p = C_{ij}^c$) and surface energy anisotropy used in the simulated cases

| Case | $C_{11}$ (GPa) | $C_{12}$ (GPa) | $C_{44}$ (GPa) | $\delta$ | $\varepsilon_4$ |
|------|----------------|----------------|----------------|--------|---------------|
| Case 10# | 200           | 100            | 100            | 2      | 0             |
| Case 11# | 200           | 100            | 200            | 4      | 0             |
| Case 12# | 200           | 100            | 250            | 5      | 0             |
| Case 13# | 200           | 100            | 200            | 4      | 0.005         |
| Case 14# | 200           | 100            | 200            | 4      | 0.01          |
| Case 15# | 200           | 100            | 200            | 4      | 0.02          |

In the following, the effect of each anisotropic factor on the precipitate shape will be examined independently by the simulation cases listed in Table 3. In these cases, a single coherent precipitate growth is simulated under the condition that one factor is held constant and the other is varied. The problem domain and evolution time is chosen to be $1024\times1024$ cells and $10000\Delta\tau$, respectively. And other parameters keep the same.

In Case 10#–12#, the cubic anisotropy factor $\delta$ is increased from 2 to 5 by holding $C_{11}$ and $C_{12}$ constant and varying $C_{44}$, while the elastic constants of both phases are set to be the same and the surface energy is isotropic. The morphological variation with the cubic anisotropy is shown in Figure 6. For the three cases, under the domination of the cubic anisotropy, all the preferential growth directions of the precipitate are along $<11>$. And the degree of elongation along the preferential growth direction is strongly dependent on the cubic anisotropy. With the increase of the cubic anisotropy, there is a distinct decrease and increase for the tip radius and the length of the primary arms, respectively.
Figure 6. The profile of the precipitate as a function of $\delta$ under isotropic surface: (a) $\delta = 2$; (b) $\delta = 4$; (c) $\delta = 5$

Figure 7. The effect of combination of elasticity energy anisotropy and surface energy anisotropy on the morphological pattern of the precipitate: (a) $\varepsilon_1 = 0.005$; (b) $\varepsilon_1 = 0.01$; (c) $\varepsilon_1 = 0.02$

In Case 13#~15#, a relative large cubic anisotropy i.e. $\delta = 4$ and homogeneous elasticity are applied constantly, while the surface energy anisotropy $\varepsilon_1$ which causes the precipitate to grow preferentially along $<10>$ directions is varied from 0.005 to 0.02. Accordingly, the effect of the surface energy anisotropy on morphological pattern of precipitate is shown in Figure 7. For the case of $\varepsilon_1 = 0.005$, the dendritic pattern with preferential growth directions of $<11>$ indicates that the precipitate growth is controlled by the cubic elastic anisotropy as discussed above. When $\varepsilon_1$ is increased to 0.01, a flower-like structure is produced. A relative large surface energy anisotropy i.e. $\varepsilon_1 = 0.02$ results a strongly preferential growth along $<10>$ directions. Similar results can be found in Greenwood's work [19]. Since the precipitate growth is corresponding to minimization procedure of the total free energy, the transformation of the precipitate morphology can be interpreted by the competition between anisotropies of the surface energy and the elasticity energy. A critical value for surface energy anisotropy can be found to counteract the cubic anisotropy, producing a isotropic pattern. The surface energy anisotropy plays a major role in the precipitate growth when $\varepsilon_1$ is above the critical value. Otherwise, the cubic anisotropy will dominate.

6. Conclusion
In this paper, a parallel multigrid algorithm is proposed to solve the elasticity problem in the phase-field model for cubic-to cubic coherent precipitation. Numerical tests confirm the efficiency and scalability of the parallel multigrid algorithm combined with GS-FC smoother. The parallel multigrid method can reduce the total simulation time by two orders and is suitable for a relative large scale
problem. The morphological evolution of the precipitate is simulated by the elastic strain energy using the proposed algorithm. The anisotropy introduced by the elasticity energy has a significant effect on the precipitate pattern and competes with the surface energy anisotropy. Our future work will focus on extending the parallel multigrid scheme to three-dimensional simulation for the coherent precipitate phase transformation.

Acknowledgments
The authors would like to thank the SCIENCE CHALLENGE PROJECT (TZ2016004) for financial support.

References
[1] S. W. Husain, M. S. Ahmed, I. Qamar, Dendritic morphology observed in the solid-state precipitation in binary alloys, Metall. Mater. Trans. A 30A (1999) 1529-1534.
[2] T. Takaki, M. Ohno, Y. Shibuta, S. Sakane, T. Shimokawabe, T. Aoki, Two-dimensional phase-field study of competitive grain growth during directional solidification of polycrystalline binary alloy, J. Cryst. Growth 442 (2016) 14-24.
[3] M. Yang, S. M. Xiong, Z. Guo, Characterisation of the 3-D dendrite morphology of magnesium alloys using synchrotron X-ray tomography and 3-D phase-field modelling, Acta Mater. 92 (2015) 8-17.
[4] Y. H. Wen, Y. Wang, and L. Q. Chen, Phase-field simulation of domain structure evolution during a coherent hexagonal-to-orthorhombic transformation, Philos. Mag. A 80 (2000) 1967-1982.
[5] G. M. Han, Z. Q. Han, A. A. Luo, A. K. Sachdev, B. C. Liu, A phase field model for simulating the precipitation multi-variant β-Mg17Al12 in Mg-Al based alloys, Scripta Mater. 68 (2013) 691-694.
[6] A. G. Khachaturyan, Theory of Structural Transformations in Solids, Wiley, New York, 1983.
[7] J. Zhu, L. Q. Chen, J. Shen, Morphological evolution during phase separation and coarsening with strong inhomogeneous elasticity, Model. Simul. Mater. Sc. 9 (2001) 499-511.
[8] S. Y. Hu and L. Q. Chen, A phase-field model for evolving microstructures with strong elastic inhomogeneity, Acta Mater. 49 (2001) 1879-1890.
[9] M. Greenwood, Phase-field modeling of solid state precipitates, PhD Thesis, McMaster University, 2008.
[10] U. Trottenberg, C. Oosterlee, A. Schuller, Multigrid, Academic Press, London, 2001.
[11] V. T. Zhukov, N. D. Novikova, and O. B. Fedoritova, Parallel Multigrid Method for Solving Elliptic Equations, Math. Model. Comput. Simul. 6 (2014) 425-434.
[12] Z. Guo, J. Mi, P.S. Grant, An implicit parallel multigrid computing scheme to solve coupled thermal-solute phase-field equations for dendrite evolution, J. Comput. Phys. 321 (2012) 1781-1796.
[13] J. Shin, S. Kim, D. Lee, J. Kim, A parallel multigrid method of the Cahn-Hilliard equation, Comp. Mater. Sci., 71 (2013) 89-96.
[14] B. Echebarria, R. Folch, A. Karma, M. Plapp, Quantitative phase-field model of alloy solidification, Phys. Rev. E 70 (2004) 061604.
[15] Z. Guo, S.M. Xiong, On solving the 3-D phase field equations by employing a parallel-adaptive mesh refinement (Para-AMR) algorithm, Comput. Phys. Commun. 190, (2015) 89-97.
[16] Alfio Borzi, Introduction to multigrid methods, technical report, 2003.
[17] T. W. Heo,Phase-field modeling of microstructure evolution in elastically inhomogeneous polycrystalline materials, PhD thesis, the Pennsylvania State University, 2012.
[18] H. J. Jou, P. H. Leo, J. S. Lowengrub, Microstructural evolution in inhomogeneous elastic media, J. Comput. Phys. 131 (1997) 109-148.
[19] M. Greenwood, J. J. Hoyt, N. Provatas, Competition between surface energy and elastic anisotropies in the growth of coherent solid-state dendrites, Acta Mater. 57 (2009) 2613-
2623.