Study of dendritic growth and coarsening using a 3-D phase field model: Implementation of the Para-AMR algorithm

Z Guo1*, S M Xiong1,2

1 School of Materials Science and Engineering, Tsinghua University, Beijing, China
2 Key Laboratory for Advanced Materials Processing Technology, Ministry of Education, Beijing, China

E-mail: zhipeng_guo@mail.tsinghua.edu.cn

Abstract. To efficiently solve the coupled phase field equations in 3-D, an algorithm comprising of adaptive mesh refinement (AMR) and parallel (Para-) computing capabilities was developed. Dendritic growth and subsequent coarsening were studied by employing the model to simulate multi-dendrite growth under isothermal conditions. Quantitative comparison including decrease of interface area ($S$) and nonlinear growing of the characteristic length (ratio between solid volume $V$ and surface area $S$ i.e. $V/S$) as time was performed between the simulation results and these predicted by the existing theories. In particular, various mechanisms including growth of lower curvature area in expense of higher curvature one, coalescence of neighbouring dendrite arms and groove advancement at the root of higher order arms for dendritic coarsening were identified and successfully revealed via the 3-D phase field simulation. In addition, results showed that the proposed algorithm could greatly shorten the computing time for 3-D phase field simulation and enable one to gain much more insight in understanding the underlying physics during dendrite growth in solidification.

1. Introduction

In solidification of metallic alloys, growth of solid dendrites is always accompanied by coarsening. To reduce the free energy of the system, the coarsening always occurs spontaneously by decreasing the overall curvature or solid/liquid interface area [1,2].

The phase field method is becoming a standard approach to simulate phase transitions, especially for those with complex geometries, e.g. dendrites during solidification [3]. In the phase field approach, an extra equation (the phase field equation) is introduced to characterize the transition between the solid phase and the liquid phase through the variable $\phi$, termed the phase field, which is constant in bulk phases (-1 in liquid and 1 in solid) but changes steeply across the solid-liquid interface. The most beneficial feature for phase field methodology is that explicit tracking of the interface can be avoided, which greatly simplifies the computational procedure.

However, to accurately characterize the underlying physics during phase transition, the grid size must be very small in order to have enough resolution to describe the interface. The consequence is then enormous computing overhead must be dealt with during the phase field simulation especially in the study of coarsening where simulation has to be performed in much longer time before the underlying physics could be revealed. Because of such limitation, current literature in studying dendritic coarsening is rather limited and most existing studies were performed in 2-D [4]. Even
though certain qualitative aspects of physics in dendritic coarsening can be revealed in 2-D, quantitative characterization has to be found through 3-D simulations.

In this paper, based on the block-structured grid configuration as introduced by Berger and co-workers [5] we will first develop an algorithm with both adaptive mesh refinement and parallel computing ability i.e. a Para-AMR approach and then employ this approach to solve the coupled phase field equations in 3-D. Different phase field simulations will then be performed to study the dendritic coarsening.

2. The phase field model and numerical approach

2.1. The phase field model

The 3-D phase field model was derived based on our previous work [6] and the governing equations read:

\[
\tau \frac{\partial \phi}{\partial t} = \nabla \cdot \left( W \left( \frac{r}{\hat{n}} \right)^2 \nabla \phi \right) + \phi (1 - \phi^3) - \lambda (1 - \phi^2)^2 (\theta + kU)
\]

\[
+ \frac{\partial}{\partial x} \left( \nabla \phi \left| W \left( \frac{r}{\hat{n}} \right) \frac{\partial W \left( \frac{r}{\hat{n}} \right)}{\partial \phi_x} \right| \right)
+ \frac{\partial}{\partial y} \left( \nabla \phi \left| W \left( \frac{r}{\hat{n}} \right) \frac{\partial W \left( \frac{r}{\hat{n}} \right)}{\partial \phi_y} \right| \right)
+ \frac{\partial}{\partial z} \left( \nabla \phi \left| W \left( \frac{r}{\hat{n}} \right) \frac{\partial W \left( \frac{r}{\hat{n}} \right)}{\partial \phi_z} \right| \right)
\]

\[
= \nabla \cdot \left( D \frac{1 - \phi}{2} \nabla U - J_{att} \right) + \frac{1}{2} \left[ 1 + (1 - k)U \right] \frac{\partial \phi}{\partial t}
\]

\[
\frac{\partial \theta}{\partial t} = \alpha \nabla^2 \theta + \frac{1}{2} \frac{L / c_p}{\Delta T_0} \frac{\partial \phi}{\partial t}
\]

where \( \tau \) is the relaxation time, \( \phi \) is phase field, \( W \left( \frac{r}{\hat{n}} \right) \) is the anisotropic width of the diffuse interface with \( \hat{n} \) the unit normal vector pointing out into the liquid, \( k \) is the partition coefficient for solute, \( D \) and \( \alpha \) are solute and thermal diffusivities respectively, \( L \) is the heat of fusion per unit volume, \( c_p \) is the specific heat of alloy, \( \lambda \) is the scaling parameter defined as

\[
\lambda = \frac{15}{16} \frac{RT_m (1 - k)}{v_0 H m} \Delta T_0
\]

where \( R \) is the gas constant, \( T_m \) is the melting temperature of the solvent, \( v_0 \) is the molar volume, \( H \) is the energy barrier of the double well potential, and \( m \) is the liquidus slope in the phase diagram.

\[
J_{att} = -\frac{W}{\sqrt{2 \left[ 1 + k - (1 - k) \phi \right]}} \frac{\partial \phi}{\partial t} \nabla \phi
\]

is the so-called “anti-trapping” current developed by Karma [7] to counterbalance spurious effects during the phase field simulation.

\[
U = \frac{2c / c_\infty}{1 + k - (1 - k) \phi} - \frac{1}{1 - k}
\]

and

\[
\theta = \frac{T - T_m - mc_\infty}{\Delta T_0}
\]

are dimensionless solute concentration and temperature respectively, where \( c \) is solute concentration, \( c_\infty \) is the initial solute concentration and

\[
\Delta T_0 = \frac{|m| c_\infty (1 - k)}{k}
\]
is the equilibrium freezing temperature range according to $c_\infty$.

The crystal anisotropy was introduced by taking

$$\tau = \tau_0 A(n) + W = W_0 A(n)$$

and

$$A(n) = 1 + \epsilon_1 (Q - 3/5) + \epsilon_2 (3Q + 66S - 17/7)$$

(9)

$$Q = \left( \phi^i + \phi^j + \phi^k \right)/|\mathbf{P}|^4$$

(10)

$$S = \phi^i \phi^j \phi^k$$

(11)

where $\phi_i = \partial \phi / \partial x$, $\phi_j = \partial \phi / \partial y$, $\phi_k = \partial \phi / \partial z$, $\epsilon_1$ and $\epsilon_2$ weight the magnitude of the anisotropy strength favoring the crystalline orientation of [100] and [110], respectively [3].

According to the thin interface analysis in [5], the kinetic effect can be made to vanish by taking

$$\tau = \tau_0 \left[ 1 + \frac{1}{Le} + k \left( 1 + (1-k)U \right) \right] A(n)^2$$

(12)

where $Le = \alpha / D$ is the Lewis number. The length and time in equations (1) – (3) were further scaled by $W_0 = \lambda d_0 / a$ and $\tau_0 = d^2 a / (D a^2)$ to be fully non-dimensional, where $d_0 = \Gamma / \Delta T$ is the chemical capillary length, and $\Gamma$ is the Gibbs-Thomson coefficient. According to the thin interface analysis [7], $a_1 = 0.8839$ and $a_2 = 0.6267$ respectively.

In this work a simple method was applied to characterize the multi-orientation of the dendrites, which has been employed successfully in our previous work [6] for the 2-D case. The orientation of each of the dendrites is specified with a pre-existing solid seed introduced in the computational domain. As the dendrite grows, very small volumes of liquid are added/transformed to solid; the crystal orientation for this new increment in solid is assumed the same as that of the local crystal orientation of the dendrite. In this way, each dendrite acts like an orientation attracting sink which transforms its closest external layer of liquid into solid by imposing its orientation to it.

### 2.2. The Para-AMR algorithm

To solve equations (1) – (3), we have developed a Para-AMR algorithm which is in fact an implementation of the adaptive mesh refinement on a parallel computing architecture.

#### 2.2.1. The adaptive mesh refinement (AMR)

In this study, the AMR was realized based on a block-structured grids, which was first introduced by Berger and co-workers [5]. Accordingly, the AMR was performed in three sequential steps including (1) tagging the target cells that need refined, (2) clustering the tagged cells into patch-boxes, and (3) ensuring the coarse/refined grids are properly nested.

To refine the target grids the first step is to tag the cells that need refined. For dendritic microstructure the solid-liquid interface is the location where the phase changes the fastest and thus the position that needs refined. In this work, we employed a gradient criterion to perform the tagging i.e.

$$|\nabla \phi| + \beta_s |\nabla U| + \beta_\theta |\nabla \theta| \geq \xi$$

(13)

where $\beta_s$ and $\beta_\theta$ are weighted parameters for solute and temperature respectively. $\xi$ is a threshold value (always small in magnitude) which has to be retrieved via numerical tests according to specific simulation cases.

After certain grids/points have been tagged, the grid refinement would start from the coarsest grid level. A cluster algorithm developed by Berger and co-workers [8] was adopted to separate the tagged points into clusters. The clusters are also called patch-boxes, which in 2-D are rectangles but in 3-D they are boxes with interior boundaries. After the patch-box was generated, grid points/cells would be filled inside and used for data computation. The cluster algorithm would be performed repeatedly until reaching the top grid level, after which the multi-level grid with a hierarchical structure was created.
An extra layer of cells or two will always be added to the coarse grid level to ensure the multi-level grids are properly nested.

![Figure 1](image)

**Figure 1.** (a) The configuration of the patch-boxes on different grid levels generated during the adaptive mesh refinement and (b) the dispatching of computing work to the parallel processors. The dendrite profile i.e. the solid/liquid interface was indicated in the middle of each figure.

A typical result of the grid structure according to a dendrite profile was shown in figure 1a, in which the location of the patch-boxes on different grid levels \(l_0\) was shown. The simulation was according to a single dendrite growth in an isothermal domain i.e. \(\theta = -0.2\). Other computation parameters were \(k = 0.15, \lambda = 10, \beta_U = 0.3\) and \(\beta_\theta = 0\) (isothermal). A solid seed was initially planted in the middle of the domain and then growing for 6000 time steps. The phase field profile according to \(\phi = 0\) was then extracted and shown in the middle of figure 1a.

As shown, the multi-level grid was properly nested i.e. the patch-boxes belonging to a finer grid always lay inside that of a coarser grid. Besides, the layout of the patch-boxes could perfectly describe the outline profile of the dendrite, especially on the top grid level.

### 2.2.2. The parallel computing

After the grid was generated, the layout of the patch-boxes including all information regarding to the configuration of geometry was then broadcasted to all processors, based on which the computing work on each grid level was constructed, partitioned and dispatched. Two schemes were applied in this approach to avoid load imbalance. If the number of the patch-boxes was smaller than that of the parallel processors, the algorithm will dispatch the computing work uniformly starting from the root process. On the contrary, if the number of the patch-boxes was larger the computing work would be dispatched by employing a method based on space filling curve [9].

A typical result about the parallel dispatching of the computing work is shown in figure 1b, in which four processors were used to demonstrate the parallelization on grid level 4. Because the number of the patch-boxes was larger than that of the parallel processors the dispatching of the computing work was realized using the space filling curve principle and as seen from figure 1b the patch-boxes were distributed rather uniformly into the four processors (as indexed by 0 - 3).

At each time step after updating the variables, related data has to be communicated, which primarily comprised two steps. The first step was to communicate and update the data at the external boundaries of each patch-box. This could happen either within one processor (e.g. between patch-boxes \(P_1\) and \(P_2\) in figure 1b) or between different processors (e.g. between \(P_1\) and \(P_3\)). To simplify this,
a layer of cells (also known as ghost cells) were added to the external boundaries of each patch-box to receive the data from its nearest neighbours. Secondly, certain amount of data between different grid levels must be communicated to perform the restriction (i.e., updating values on the coarse grid using those from the fine grid) and interpolation (i.e., updating values on the fine grid using those from the coarse grid) at the interior boundaries.

The code for solving the phase field equations was built on the BoxLib [10] software framework for block-structured adaptive mesh methods, which currently supports a hierarchical programming approach for multi core architectures based on both MPI [11] and OpenMP [12].

3. Numerical tests

To test the accuracy and efficiency of the algorithm, simulation was performed in 3-D on a single dendrite growth scenario with key parameters including \( k = 0.15, \lambda = 30, \theta = -0.12, \beta_{0.3} = 0.3 \) and \( \theta_{0.3} = 0 \). The computing domain was \( 204.8 \times 204.8 \times 204.8 \), i.e. equivalent to \( 256 \times 256 \times 256 \) grid cells if a uniform grid of \( dx = 0.8 \) was employed. During the test, the number of the grid levels i.e. \( N_{amr} \) and the number of the parallel processes i.e. \( N_p \) were varied and attention was focused on their effects on the simulation elapsed time. For each simulation, a total of 6000 time steps were calculated.

![Figure 2](image)

**Figure 2.** Numerical test results on (a) accuracy and (b) parallel efficiency. The dendrite profiles extracted using the explicit algorithm (EM) and the adaptive mesh refinement (AMR) were compared in detail in the insert “Z1”, together with “Z2” showing the dendrite morphology in 3-D and the configuration of the patch-boxes on the top grid level.

Figure 2 shows the related result for the numerical test. In figure 2a, the phase field profiles according to \( \phi = 0 \) were extracted for the case when the explicit method (EM) was applied i.e. \( N_{amr} = 1 \) and \( N_p = 1 \) and the case when the AMR was applied i.e. \( N_{amr} = 5 \) and \( N_p = 1 \). Because the two profiles were almost overlapping a zoom-in insert “Z1” was used to show the difference. According to the simulation result, the difference on the tip position of the two profiles was about 0.05%, clearly indicating the accuracy of the algorithm. The morphology of the dendrite in 3-D and the related configuration of the patch-boxes on the top grid level were shown in the insert designated by “Z2”. Figure 2b shows the elapsed time as a function of \( N_p \) and as shown, without the adaptive mesh refinement, the total elapsed time was about 9000 seconds when \( N_p = 1 \) i.e. when the explicit algorithm was applied. Increasing the process number to 48 decreased the elapsed time to 834 seconds, i.e. the simulation was speeded up for about 10.8 times. It is clear that the gained speed-up here is
lower than the theoretical value i.e. 48 and data communication between patch-boxes and grid levels should be responsible for this.

The application of the AMR also significantly reduced the elapsed time. As shown by the solid spheres, the elapsed time for the case when \( N_{\text{core}} = 5 \) and \( N_p = 1 \) was about 1570 seconds i.e. less than 18\% of that when the explicit algorithm was applied. Further increasing the process number to \( N_p = 12 \) led to another speed-up of about 86\%, i.e. 40 times speed-up overall.

From figure 2b, it can be seen that the elapsed time (\( t_{\text{simu}} \)) changed as a function of the process number in the following manner:

\[
\log t_{\text{simu}} = k_c \log N_p + \log t_{\text{simu}}^{N_p=1}
\]

where \( t_{\text{simu}}^{N_p=1} \) is the elapsed time for the simulation when \( N_p = 1 \). This relationship i.e. equation (14) was best described by the linear fits as designated by “L1” and “L2” for \( N_{\text{core}} = 1 \) and \( N_{\text{core}} \neq 1 \) respectively. And according to figure 2b, the slope for these two cases was about -0.64 (L1) and -0.78 (L2) respectively, i.e. the algorithm worked better if the AMR was applied. The slope values for both cases are lower than the theoretical magnitude i.e. -1 but comparing with the explicit algorithm, it is clear that with the application of the Para-AMR algorithm the computing efficiency for the 3-D phase field simulation could be increased by two orders of magnitude i.e. simulation time could be reduced from an order of magnitude of 10^4 seconds to 10^2 seconds.

![Figure 3](image1.png)

**Figure 3.** Phase field simulations of the dendritic morphology at time (a) 2000 dt, (b) 8000 dt and (c) 20,000 dt. (d) shows the coarsening of dendrite D1 from \( t = 8000 \) dt to 20,000 dt. The three coarsening mechanisms including small arm melting, groove advancement and coalescence were highlighted using examples designated by “1”, “2” and “3” respectively.
4. Simulation of dendrite growth and coarsening

Simulations were performed to study the multi-dendrite growth and coarsening in 3-D. Similar as the numerical test, five levels of grid were applied during the simulation and 192 cores were implemented for the parallel computation. The system domain was $409.6 \times 409.6 \times 409.6$, i.e. equivalent to $512 \times 512 \times 512$ cells if an uniform grid of $dx = 0.8$ was applied. The simulation was initialized by planting 10 solid seeds with random orientations in an isothermal domain with temperature of $\theta = -0.12$. Other calculation parameters stayed the same as those in the numerical test until state otherwise. Periodic boundaries were applied at each wall of the domain. The simulation was kept on going for a total of 140,000 time steps to enable the dendrites to reach a rather mature growing stage i.e. coarsening.

4.1. Multi-dendrite growth and coarsening

Figure 3 shows the phase field simulation of a multi-dendrite growth case. The dendritic morphologies at $t = 2000 \, dt$, $8000 \, dt$ and $20000 \, dt$ were shown in figures 3a, 3b and 3c respectively. As time evolved, dendrites started to grow and then impinge together. Detailed features including side-branching of the high order arms could also be observed. Taking dendrite D1 for example, the three coarsening mechanisms including small arm melting (“1”), groove advancement at the roots (“2”) and coalescence between neighboring arms (“3”) could be clearly revealed through the simulation, which strikes great similarity to that reported in literature [13].

4.2. The characteristic length of coarsening

To quantify the coarsening behavior of the dendrite growth, the characteristic length $R$ i.e. the ratio between solid volume ($0 < \phi < 1$) and the surface area was calculated. Figure 4 shows the change of the characteristic length as solidification time evolved. To achieve a better illustration and comparison with existing theories, $R^n$ was plotted as a function of time ($t$) instead of $R$.

![Figure 4](image)

According to the LSW theory [14], the characteristic length changes in the following manner:

$$ R_t^n - R_0^n = k_c t $$  \hspace{1cm} (15)

where $R_t^n$ is the characteristic length at time $t$ and $k_c$ is a rate constant related to alloy thermodynamic properties. Through numerical tests, it was found that the best fit for the current simulation occurred...
when \( n = 4.3 \) rather than 3, i.e. the commonly accepted value for the LSW theory. However, \( n = 4.3 \) was quite close to that obtained by Terzi and co-workers [15] through the recent high brilliance X-tomography experiment (in which \( n = 4.4 \)). More evidence is thus required from both modeling and experiment to clarify this difference.

5. Conclusion

(1) A computing scheme with the ability of adaptive mesh refinement and parallel computing (i.e. Para-AMR) has been developed to solve the coupled phase field equations in 3-D. The whole algorithm was realized based on (a) a gradient criterion for grid/point tagging, (b) the automatic clustering algorithm for grid generation, and (c) parallelization i.e. dispatching of computing work based on the principle of space filling curve. Numerical tests revealed that the Para-AMR algorithm, without compromising any accuracy of the phase field simulation, can significantly reduce the computing time in a degree of about two orders of magnitude.

(2) 3-D phase field simulations of multi-dendrite growth were performed and dendrite growth and coarsening were investigated. The three coarsening mechanism including small arm melting, interdendritic groove advancement and coalescence between neighboring arms can be successfully revealed through the simulation. By fitting the growing law during dendritic coarsening in particular the formula

\[
R_t^n - R_0^n = k_t t
\]

we found that the best fit was achieved when \( n = 4.3 \) which was higher than that predicted by the classic LSW theory but very similar to that determined through the X-tomography experiments.

Acknowledgments

The authors would like to thank the Natural Science Foundation of China (Project No. 51205229), the Tsinghua University Initiative Scientific Research Program (20121087918) and the Newton International Fellowship Scheme for financial support, and the National Laboratory for Information Science and Technology in Tsinghua University for access to supercomputing facilities. The authors would also like to thank the open source software Visit, based on which the visualization of the simulation results were performed.

References

[1] Rowenhorst D J and Voorhees P W 2012 Annu. Rev. Mater. Sci. 42 105
[2] Bray A J 2002 Adv. Phys. 51 481
[3] Haxhimali T, Karma A, Gonzales F and Rappaz M 2006 Nat. Mater. 5 660
[4] Wang J and Yang G 2008 Acta Mater. 56 4585
[5] Berger M and Colella P 1989 J. Comput. Phys. 82 64
[6] Guo Z, Mi J, and Grant P S 2012 J. Comput. Phys. 231 1781
[7] Karma A 2001 Phys. Rev. Lett. 87 115701
[8] Berger M and Rigoutsos I 1991 IEEE Trans. Syst. Man Cybern. 21 1278
[9] Bader M 2013 Space-Filling Curves: An introduction with applications in scientific computing (Berlin: Springer)
[10] BoxLib https://ccse.lbl.gov/BoxLib/index.html
[11] Gropp W, Lusk E and Skjellum A 1999 Using MPI: Portable Parallel Programming with the Message-Passing Interface (Cambridge: MIT Press)
[12] Chapman B, Jost G and van der Pas R 2007 Using OpenMP Portable Shared Memory Parallel Programming (Cambridge: MIT Press)
[13] Limodin N, Salvo L, Boller E, Suery M, Felberbaum M, Gailliegue S and Madi K 2009 Acta Mater. 57 2300
[14] Ardell A J and Ozolins V 2005 Nature Mater. 4 309
[15] Terzi L, Salvo L, Suery M, Dahle A K and Boller E 2010 Acta Mater. 58 20