CS 294-73 Final Project: Two-Dimensional Dendritic Growth Using Phase-Field Model

Due on Friday, Dec 18, 2015

University of California, Berkeley
Fall 2015

Group H

Xingjie Pan
Xian Shi
Letian Wang

1 Problem Description

Dendritic crystal growth is a common phenomena during the casting of metals and alloys. The solid-liquid interface grows from a solid seed and evolve as the total thermal energy dissipates. Figure 1 shows the shadowgraphic images from a standard dendritic growth. The study of size, shape and growth velocity of dendritics is very important as they largely influence the quality of desired materials.

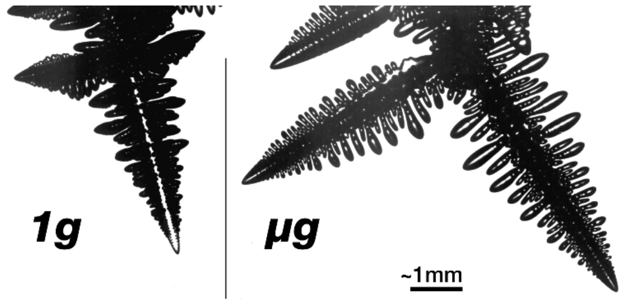


Figure 1: Isothermal dendritic growth experiment of succinonitrile (SCN) dendrites, by Rensselaer Polytechnic Institute (RPI) and NASA/ Glenn Research Center(GRC). NASA MIX No.0003748.

A major challenge of dentritic growth studies is to define the interface. In general, there are two distinct approaches: phase-field model and sharp interface model, as shown in Figure 2. In the phase-field method, the jump discontinuity of phase field is replaced by a smooth and continuous interface of very small width. Threrefore, the singularities involved in the sharp interface are removed.

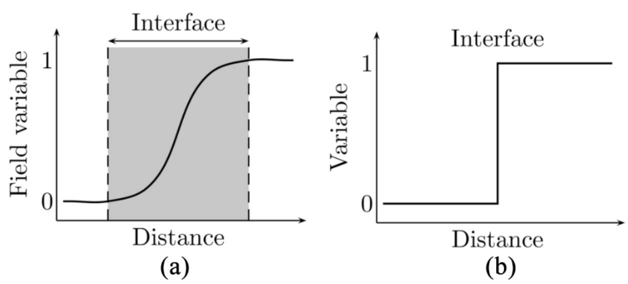


Figure 2: Interface representation: (a) phase-field model, (b) sharp interface model

In order to implement the phase-field model, two parabolic non-linear PDEs are to be solved: phase-field equation of motion with respect to phase-field parameter ϕ, and thermal diffusion equation with respect to the dimensionless temperautre u. The concrete forms of these two equations are given as follows. More details can be found in the math description document.

$$
\begin{cases}
\frac{\partial u}{\partial t} = & D\nabla^2 u + L\frac{\partial \phi}{\partial t} \\
\tau\frac{\partial \phi}{\partial t} = & \phi(1-\phi)(\phi - \frac{1}{2} + \tilde{n}(u)) - \frac{\partial}{\partial x}(WW'\frac{\partial \phi}{\partial y}) \\
 & +\frac{\partial}{\partial y}(WW'\frac{\partial \phi}{\partial x}) + \nabla(W^2)\cdot\nabla\phi + W^2\nabla^2\phi \\
W = & W_0(1+\mu cos(a_0(\theta-\theta_0)) \\
\theta = & tan^{-1}(\frac{\partial \phi}{\partial y}/\frac{\partial \phi}{\partial x}) + \pi(1 - sign(\frac{\partial \phi}{\partial x}))
\end{cases}
\tag{1}
$$

where the parameters are listed below:

D: thermal diffusion constant	τ: relaxiation time	β: material parameter
η: material parameter	u_m: melting temperature	W_0: initial interfacial width
μ: modulation of interfacial width	a_0: anisotropic mode number	θ_0: orientation angle
L: latent heat		

2 Program Discription

2.1 Layers of the program

The program is cascaded into three layers, namely utilities, phase field method and user interface. Functions and classes of each layer depend on codes of lower layers.

2.1.1 Layer 1: utilities

The utilities layer is made of classes and functions that are basic mathematical abstractions and output handlers that are independent to the phase field method. Classes in this layer are **Point, Box, RectMDArray, RK4, WriteRectMDArray** and **VisitWriter** from previous assignments, as well as newly developed **RectMDOperators** including gradient and Laplace operators.

2.1.2 Layer 2: phase field method

The phase field method layer is made of classes that specify the phase field description of dentritic growth. Classes in this layer are **Dendritic, DendriticShift** and **DendriticGrowth**. The Dendritic class saves field variables and all parameters of a dendritic growth problem. The DendriticShift class saves shifts of field variables of each time step. The DendriticGrowth class defines the formula of RHS of the PDE (1).

2.1.3 Layer 3: user interface

The user interface layer only contains the **main** function which handles inputs from users, constructs objects for time advancing scheme, updates the state of the system with RK4 and writes data into .vtk files.

2.2 Pipeline of the program

./main.exe:
Get and parse user's inputs;
Construct objects of class Dendritic;
while *time* < *required time* **do**
  Advance state with RK4:
  {
    Calculate RHS with class DendritcGrowth;
    Save shift in an object of class DendriticShift;
    Update current state;
  }
  Output current state;
end

2.3 How to use the program

2.3.1 Inputs

The program takes no command line arguments. After the running the program, it will interactively ask the user to input the size of the mesh, the case number of the problem and length of simulation time.

2.3.2 Outputs

Outputs of the program are phase field in .vtk format at each time step.

3 Results and Discussions

Numerical simulations shown in this session are conducted on a 256*256 grid domain with $h = 0.03$. The time step is set to $dt = 0.0001$ to ensure numerical stability. Other parameters include thermal diffusion coefficient $D = 1$, relaxation time $\tau = 0.03$, orientation angle $\theta = 0$, interfacial width $\delta = 0.01$ and modulation of interfacial width $\mu = 0.02$. Dirichlet boundary conditions are implemented for both phase field and dimensionless temperature filed. Three test cases are set up in the main function with different values of anisotropy number and latent heat. Time advancing of the phase field, effects of anisotropy and latent heat are discussed in later sessions.

3.1 Transient Dendritic Growth

Transient dendritic growth of test case 1 is shown in Figure 3. At time $t = 0$, a central nucleus is initialized at the center of the domain. The branches and side branches gradually developed, with both tip radius and tip velocity as functions of time.

3.2 Parametric Study

3.2.1 Effect of Anisotropy

Anisotropy has a direct effect on the shape of the dendritic crystal. Figure 4 shows dendritic growth with different anisotropy numbers. Larger anisotropy number not only result in a corresponding number of main branches, but also lead to more growth of small side branches. The tip velocity of main branches is higher with smaller anisotropy, as the tip in $a_0 = 4$ case is closer to boundaries compared to the other two cases.

3.2.2 Effect of Latent Heat

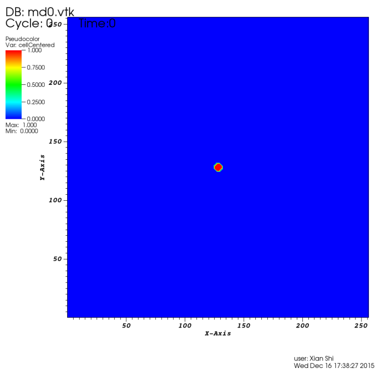
Figure 5 shows the dendritic growth with different values of latent heat. In Case (a) where latent heat is the smallest, the main branches grow faster and occupy more in the solid phase. Therefore as latent heat k increases, the growth rate decreases and the front becomes less stable and develops dendritic branching. As higher latent heat favors the local heat release, the local temperature rises and slows down the thermal energy removal process, leading to a slower dentritic crystal growth.

4 Work Distribution

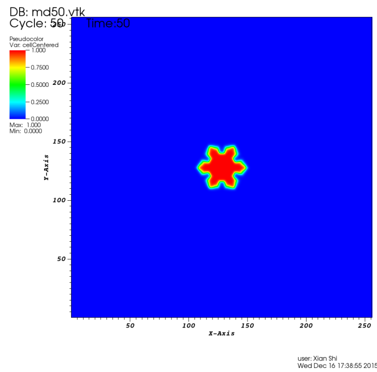
It has been a fantastic experience working with each team member in this group. Everyone has been dedicated to the project and made their indispensable contributions. The following table shows specific work distribution.

	Xingjie Pan	Xian Shi	Letian Wang
Algorithm	+++	+++	+++
Code development	+++	+++	+++
Test development	++	++	+++
Documents	+++	++	++
Final Report	++	+++	++
Overall	+++	+++	+++

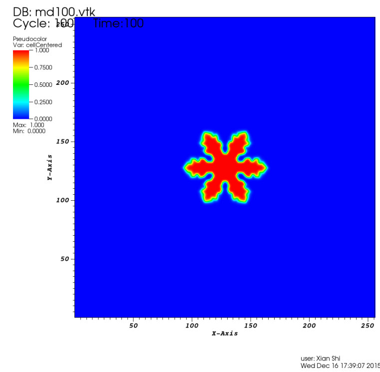
Table 1: Work distribution



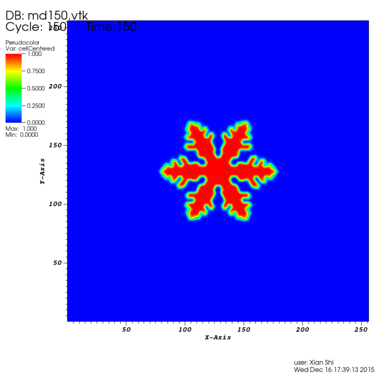
(a) t = 0



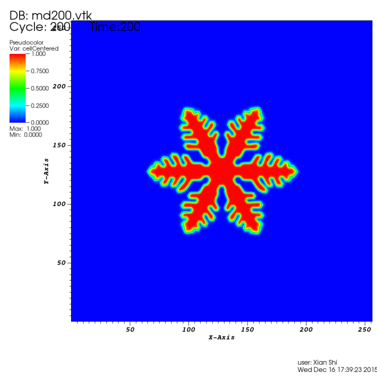
(b) t = 0.05



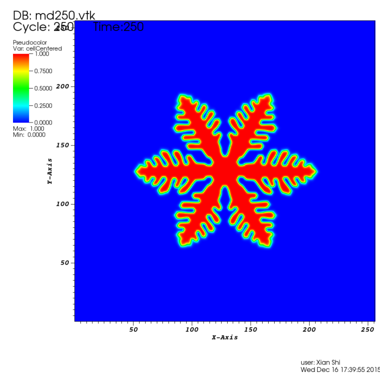
(c) t = 0.10



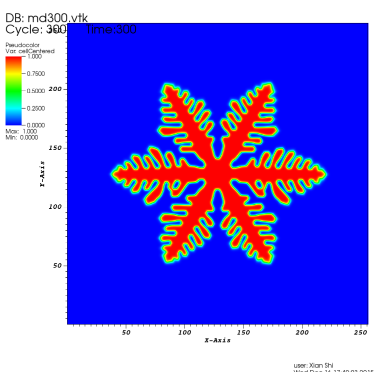
(d) t = 0.15



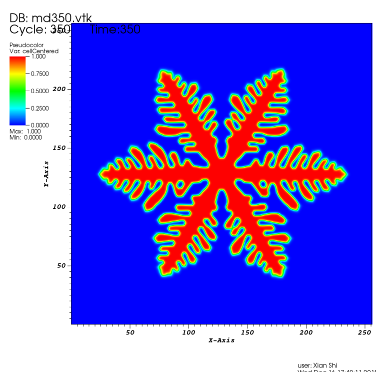
(e) t = 0.20



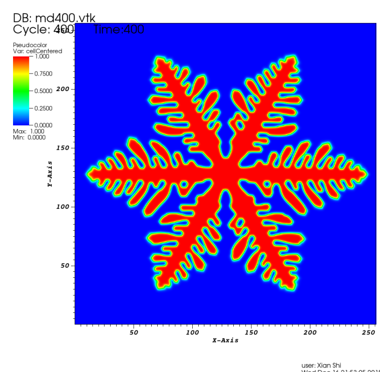
(f) t = 0.25



(g) t = 0.30

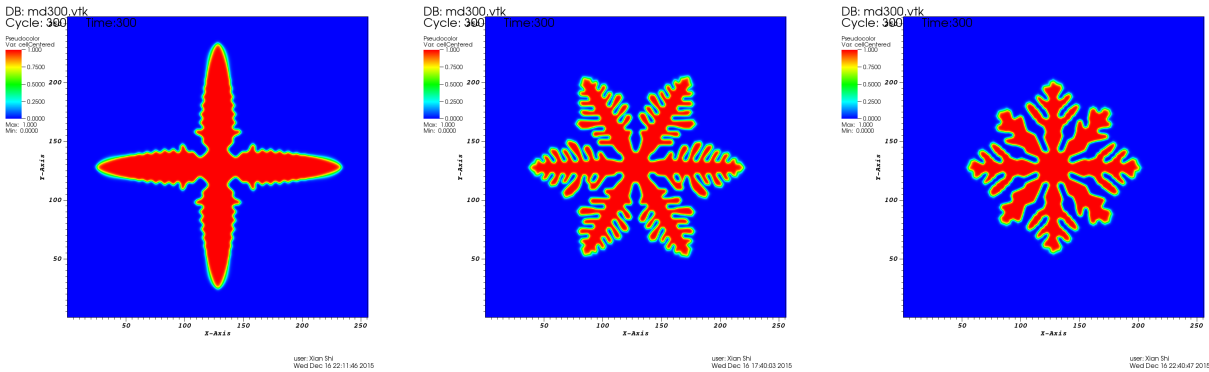


(h) t = 0.35



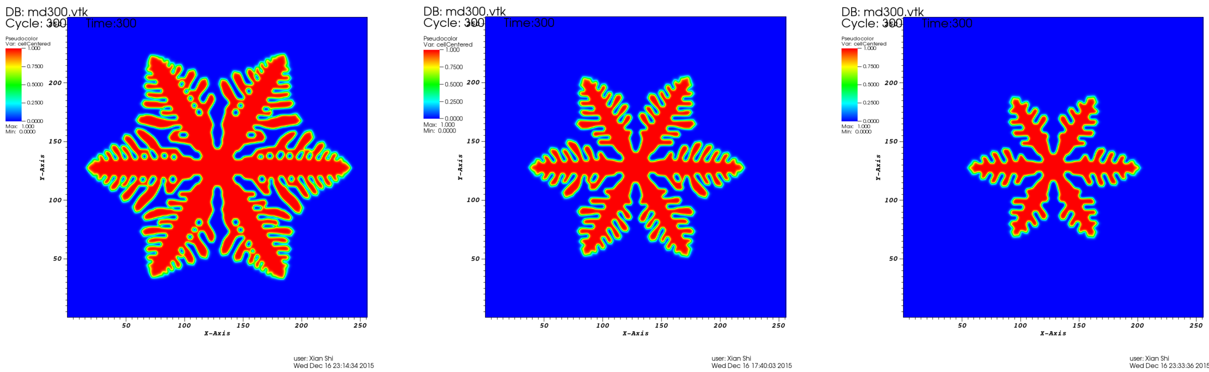
(i) t = 0.40

Figure 3: Time series of dendritic growth, anisotropy $a_0 = 4$, Latent heat $L = 2$.



(a) $a_0 = 4$ (b) $a_0 = 6$ (c) $a_0 = 8$

Figure 4: Dendritic growth with different anisotropy numbers, $t = 0.30$.



(a) latent heat = 1.7 (b) latent heat = 2.0 (c) latent heat = 2.3

Figure 5: Dendritic growth with different latent heat values, $t = 0.30$.