Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 227 (2008) 1772–1789
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

Simulation of free dendritic crystal growth in a gravity environment

Minh Do-Quang *, Gustav Amberg

The Linné Flow Centre, Department of Mechanics, Royal Institute of Technology, SE-100 44 Stockholm, Sweden

Received 4 January 2007; received in revised form 10 September 2007; accepted 27 September 2007
Available online 12 October 2007

Abstract

In this paper we simulate the evolution and free particle motion of an individual nucleus that grows into a dendritic crystal. The melt flow and the convective heat transfer around the crystal are simulated as they settle due to gravity. There is an intricate coupling between the settling and the evolution of the crystal. The relative flow induced by the settling enhances the growth at the downward facing parts, which in its turn affects the subsequent settling motion.

Simulations have been done in two dimensions using a semi-sharp phase-field model. The flow was constrained to a rigid body motion by using Lagrange multipliers inside the solidified part. The model was formulated using two different meshes. One is a fixed background mesh, which covers the whole domain. The other is an adaptive mesh, where the node points are also translated and rotated with the movement of the solid particle. In the latter, the dendritic growth is simulated by the semi-sharp phase-field method.

Keywords: Dendritic solidification; Fictitious domain method; Lagrange multiplier

1. Introduction

When solidifying a melt, for instance in casting of metals, the melt is typically cooled until its bulk temperature is below the equilibrium freezing temperature T_m (see Fig. 1). Then any free nuclei in the melt may grow, and often create dendritic "equiaxed crystals". These growing solid particles typically have a slightly different density from the melt, and will thus begin to settle due to gravity. In technological casting situations, this will give rise to a so-called "equiaxed region" at the bottom of the cast, where the crystals that have formed in the bulk, and settled to the bottom accumulate. This is called the "columnar-to-equiaxed transition" (CET) [1–3]. Similar to other materials processing phenomena, this problem involves various spatio-temporal scales. For example, dendritic growth on a microscopic scale has a time scale depending on the growth speed. But on a macroscopic scale, one time scale is determined by the settling speed. Those two length scales may differ by a factor of 5–100. Therefore, in order to predict the micro/macro structures of the "equiaxed region", it

* Corresponding author.
E-mail addresses: minh@mech.kth.se (M. Do-Quang), gustava@mech.kth.se (G. Amberg).

0021-9991/$ - see front matter
doi:10.1016/j.jcp.2007.09.025

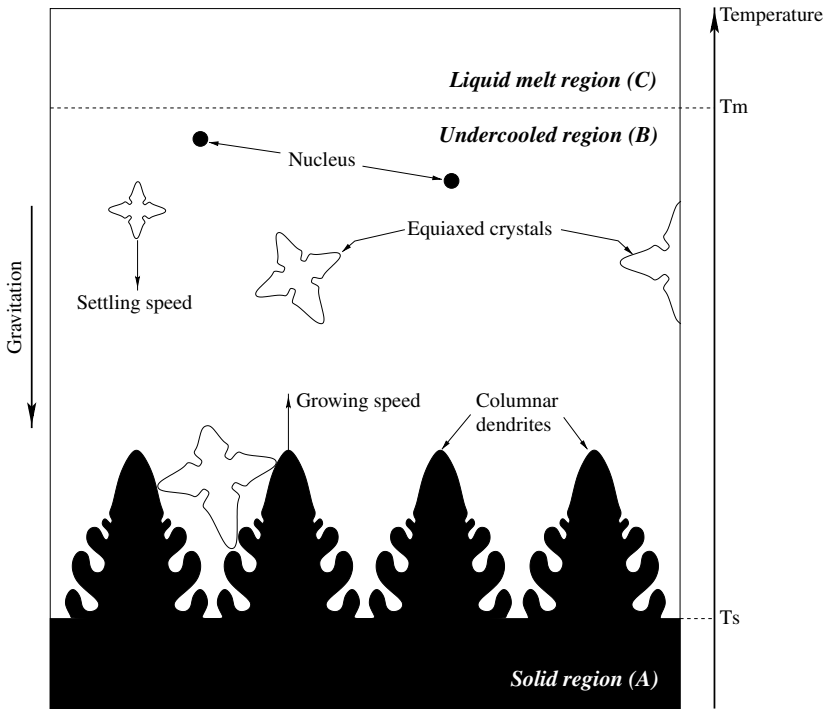


Fig. 1. Schematic of the solidification, "columnar-to-equiaxed transition" (CET).

is important to model the transfer of mass, momentum, and heat on the macroscopic scale. But we cannot neglect the microscopic scale since the micro-structure has a major impact on the properties of the final product. Also, there is a coupling effect between the dendritic growth (in microscopic) with the particle settling due to gravity (in macroscopic) that we have to take into account.

Recently, a group in Nancy successfully recorded the position and size of a single crystal, which settles in an undercooled liquid solution of the model transparent system of Albert and Gérardin [4]. It reveals the necessity to consider the relative movement effect of liquid and crystals on the growth of equiaxed crystals when modelling solidification of a large system.

The micro-structure is significantly altered by the presence of melt flow during solidification [5,6]. In the solidification process, the melt flow is mainly classified either forced convection or natural convection. The forced convection is induced by electromagnetic stirring, rotation, pouring of the melt, etc. The natural convection is caused by the density difference between the solidified and liquid phases or by the temperature dependence of density of the liquid phase. With the presence of a gravitational force, the solidified parts may move and rotate, to erase non-axially symmetric thermal effects.

During the last two decades, significant progress has been made in the computations of dendritic solidification without convection. Simulations have been performed using techniques such as the phase-field method [7–10], the level set method [11–13] and the explicit interface tracking methods [14,15]. The extension of these methods to include the effect of melt flow during the solidification are relatively recent. Tönhardt and Amberg [16] and Beckermann et al. [17] consider the solid phase as rigid and stationary. They used the phase-field method to simulate two-dimensional dendritic growth into an undercooled liquid and set the flow velocity in the solid phase to zero. Beckermann et al. introduced a mixture formulation and an auxiliary interfacial stress term into the momentum equation to ensure the correction of the shear stress at the interface and hold the solid in place. Al-Rawahi and Tryggvason [18] used the front tracking method to simulate the dendritic growth into an undercooled liquid. They used a fixed mesh for the temperature equation in which the temperature boundary condition on the interface is applied explicitly and the heat source is found directly from the temperature gradient near the interface. They used another mesh for velocity and pressure, that exist in the fluid phase only. This leads to remeshing problems.

The level set method [19] is an alternative method to handle interface tracking. Chen et al. [11] was the first to develop a level set method for the Stefan problem that directly solves the diffusion equation. They used implicit finite difference method for solving the heat equation on a Cartesian grid with Dirichlet boundary conditions imposed on the interface and a level set method for capturing the front between the solid and liquid phases of a pure substance. However the treatment at the boundary produces a nonsymmetric linear system that is slow to invert on each implicit time stepping. Zhang et al. [20] used a similar model like [11] to simulate the solidification of molten droplets on a cold substrate. Later, Kim et al. [12] and Gibou et al. [13] applied the level set method to simulate dendritic growth. They used the level set method to keep track of the front and solved the diffusion field using an implicit time discretization method. Kim et al. [12] improved the algorithm presented in [11] in order to have a more accurate computation of the interface velocity and showed that it compares favorably to phase-field methods. Gibou et al. [13] proposed a level set approach that, while preserving the second-order accuracy of [21], produce symmetric linear systems that are fast to invert, hence a much more efficient method than [11]. These authors then applied their algorithm to dendritic solidification in [13] and showed that their results agree with solvability theory are are more efficient than [11,12]. [22] developed high order accurate level set method for the Stefan problem and analyzed the influence of the definition of ghost values on the accuracy of the method.

The main difference between the phase-field method and other methods for simulation of dendritic solidification is that the important physical mechanisms, such as curvature, anisotropy and kinetic effect are implicitly incorporated in the phase-field equations. Also, by solving a diffuse interface on a fixed, or adaptively refined mesh, it avoids the need for applying temperature boundary conditions on the moving interface. It turns out that when we compute the heat fluxes from the temperature nodal values, it shall not have any problem with the discretization error that may otherwise affect the energy solution [23]. The limitation of the phase-field method is the requirement of mesh resolution at the interface and the requirement on the width of the interface. But in this work, it is overcome by using a semisharp method, which was introduced by Amberg [24].

During the phase change, it is usually permissible to consider the volume change to be small and to apply the Boussinesq approximation, effectively assuming the melt and solid to be incompressible while keeping a density difference between solid and liquid. It is also necessary to take into account the fact that the solid is rigid and it will move due to hydrodynamic, gravity and collision forces.

There are two main classes of methods for direct simulation of rigid particulate flows. The first type are the methods which discretize the Navier–Stokes equations on a fixed/Eulerian mesh that covers the whole domain where the fluid may be present. In this method, the flow in the particles is constrained to be a rigid body motion by using Distributed Lagrange Multipliers method (DLM) (see [25,26] for the original method and [27] for the modified method). The second type of methods uses a moving mesh following the motion of the boundary of the particles in the fluid. This method requires remeshing and re-interpolation and is usually referred to as Arbitrary Lagrangian Eulerian (ALE) (see [28–31]). The reduced computational effort due to saving the expensive mesh generation makes a big advantage for the DLM method over the ALE method.

The method presented in this study combines features of the semi-sharp phase-field method for the simulation of a dendritic growth and the fictitious domain method for the simulation of the rigid motion of the dendrite. The semi-sharp phase-field method is implemented to allow a direct calculation of the growth and shape of dendritic solidification. This method has also included the melt flow and its effect on the dendritic growth. The fictitious domain method is used here to allow the dendrite to move and rotate freely in the melt.

2. Numerical modelling

Consider a two-dimensional rectangular domain that contains a solid phase and a liquid phase of a pure substance in an undercooled region (region B in Fig. 1). The liquid is assumed initially to have a temperature below the equilibrium freezing temperature T_m. The initial setup for dendritic growth is a seed placed inside the domain. The dendrite grows into the undercooling melt. Assuming that the densities are different between phases but constant in each phase, the heavier solid parts will settle down due to gravity and accumulate at the bottom of the domain. In this system, the dendritic growth and settling problems are different in time and length scales. Therefore, they will be simulated as two coupled physical problems as follows, see Fig. 2.

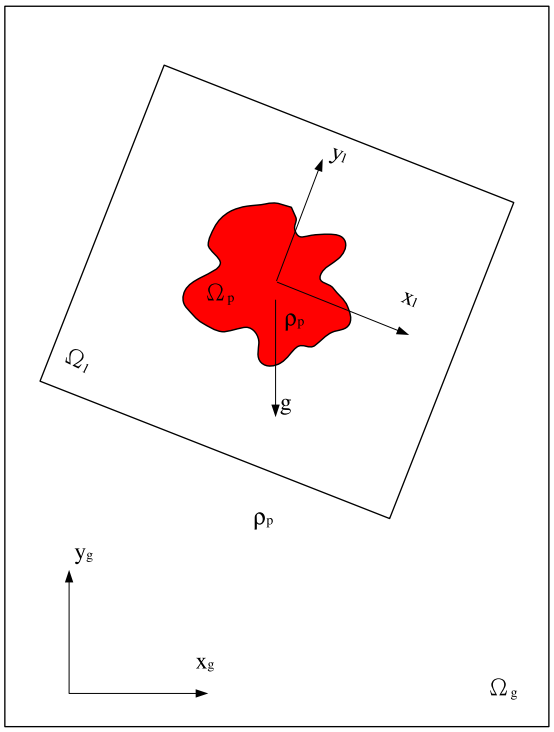


Fig. 2. Computational domains for equiaxed solidification driven by undercooling and gravity forces.

The computational domain is divided into two parts. The local part Ω_l around the dendrite will overlap the global part Ω_g, Fig. 2. The local part, Ω_l, is allowed to move and rotate together with the solidified substance by using the coordination transformation $(x_g, y_g) \rightarrow (x_l, y_l)$. According to this assumption, the velocity in the solid phase of local domain Ω_l is zero. Similar to Amberg [32] and Beckermann [17], we solve the Navier–Stokes with a model such that the velocity is extinguished in the solid phase, and ensure that the interface shear stress is correctly represented.

2.1. Simulation of the dendritic growth in the local domain

In the local domain, Ω_l, a single set of equations is written for all phases involved i.e., the solid and liquid phases. The melt here is treated as an incompressible liquid in a translating and rotating coordinate system. The Navier–Stokes equations of a viscous incompressible fluid are, therefore:

$$\frac{\partial \mathbf{u_l}}{\partial t} + \mathbf{u_l} \cdot \nabla \mathbf{u_l} + \frac{\partial \mathbf{U}}{\partial t} - 2\omega \times \mathbf{u_l} - \mathbf{r} \times \frac{\partial \omega}{\partial t} = -\frac{1}{\rho_f} \nabla p + \nabla \cdot \nu \left(\nabla \mathbf{u_l} + \nabla \mathbf{u_l}^T \right) + \frac{\nu}{H(\phi)} \mathbf{u_l}, \tag{1}$$

$$\nabla \cdot \mathbf{u_l} = 0 \quad \text{in } \Omega_l, \tag{2}$$

where $\mathbf{u_l}$ is the liquid velocity in the local domain; $\mathbf{U}$ and ω are the translational and angular velocities of the solid particle, respectively; $\mathbf{r}$ is the position vector of the point with respect to the centre of mass of the solid particle; p is the pressure and ρ_f is the liquid density. The last term on the right hand-side of Eq. (1) originates from a consideration that the morphology of the phase-change domain could be treated as an equivalent porous medium at the solid–liquid interface [32]. The $H(\phi)$ is the permeability of the mush, $H(\phi)$ is infinite in the liquid region and very rapidly decreases to a small value (i.e. 10^{-8}) in the solid region. This term forces the velocity to vanish as $\phi \rightarrow 1$ (solid region).

The local velocity at the boundary Γ_1 is mapped from the global velocity by using the following equations:

$$\mathbf{u_l} = \mathbf{u_g} - (\mathbf{U} + \mathbf{r} \times \omega), \tag{3}$$

$$\mathbf{x_l} = \mathbf{x_g} - \int_t (\mathbf{U}\,\mathrm{d}t + (\mathbf{r} \times \omega\,\mathrm{d}t)). \tag{4}$$

The consistency method used in this simulation is based on the semisharp phase-field method developed by Amberg [24]. It also includes the model of fluid flow in the melt along with solidification based on [16,17]. In this work, we make a further modification of the phase-field model, including the translation and rotation of an orthonormal coordinate system. The conservation of heat on the local domain takes the form

$$\frac{\partial \theta}{\partial t} + (\mathbf{u_l} \cdot \nabla)\theta = \nabla^2 \theta + \frac{\partial g_\delta(\phi)}{\partial t}, \tag{5}$$

where ϕ is the phase field variable which is $+1$ in the solid and -1 in the liquid, $g_\delta(\phi)$ accounts for the change in internal energy on phase change and should increase from 0 to 1 as ϕ goes from -1 to $+1$.

The phase field evolution equation is written as follows:

$$\tau \frac{\partial \phi}{\partial t} - \tau(\mathbf{u_l} \cdot \nabla)\phi = W^2 \hat{\nabla}^2 \phi - \frac{\partial f(\phi)}{\partial \phi} - \frac{\partial g_\delta(\phi)}{\partial \phi} h(\lambda\theta), \tag{6}$$

where W denotes the interface width parameter, τ links to the kinetic undercooling, and $f(\phi)$ accounts for the entropy densities, and $h(\lambda\theta)$ are functions to be specified below.

The anisotropy is included in Eq. (6) by writing the Laplacian $\hat{\nabla}^2\phi$ as a function of the local normal vector $\mathbf{n}$,

$$\hat{\nabla}^2 \phi = \nabla \cdot (\eta^2 \nabla \phi) - \frac{\partial}{\partial x}\left(\eta \dot{\eta} \frac{\partial \phi}{\partial y}\right) + \frac{\partial}{\partial y}\left(\eta \dot{\eta} \frac{\partial \phi}{\partial x}\right), \tag{7}$$

$$\eta(\mathbf{n}) = 1 + \Gamma \cos(4\beta), \tag{8}$$

where $\beta = \arctan(\frac{\partial \phi}{\partial y} / \frac{\partial \phi}{\partial x})$ is an angle between the interface and orientation of the contours and Γ is the strength of anisotropy.

In order to evaluate the kinetics of the model precisely, we follow the semi-sharp method of Amberg [24] by choosing

$$f(\phi) = \begin{cases} (\phi - 1)^2 & \text{for } \phi > 0, \\ (\phi + 1)^2 & \text{for } \phi < 0, \end{cases} \tag{9}$$

$$g_\delta(\phi) = \frac{1}{2}\left(1 + \phi \sqrt{\frac{1 + \delta^2}{\phi^2 + \delta^2}}\right), \tag{10}$$

where $g_\delta(\phi)$ is a slightly smoothed step function which has a value of $\delta = 0.05$. Amberg also pointed out that the jump in function g_δ generates a jump in the gradient of ϕ, which can be shown to satisfy the following expression:

$$h(\lambda\theta_i) = \frac{W^2}{2}\left[\left(\frac{\mathrm{d}\phi^+}{\mathrm{d}z}\right)^2 - \left(\frac{\mathrm{d}\phi^-}{\mathrm{d}z}\right)^2\right], \tag{11}$$

where the interface contour between solid/liquid interface in the semisharp model is defined where $\phi = 0$ precisely. θ_i is the value of the temperature at the interface $\phi = 0$, and the z axis is chosen normal to the interface. Solving Eq. (6) locally in the interface and inserting this solution (see [24] for details) in Eq. (11), gives

$$h(\lambda\theta_i) = -\frac{1}{2}\frac{\tau U}{W}\sqrt{8 + \left(\frac{\tau U}{W}\right)^2}, \tag{12}$$

where the symbol U includes both the kinetic and the curvature terms, V denotes the local speed of the interface, R the local radius of curvature, and d_0 the capillary length,

$$U = V + \frac{d_0}{R}\frac{W}{\tau}. \tag{13}$$

Eq. (12) guide us to a specific form for the function $h(\lambda\theta_i)$. If this is taken as $h(\lambda\theta_i) = 0.5\lambda\theta_i\sqrt{8 + (\lambda\theta_i)^2}$, Eq. (12) will guarantee that, the linear kinetics $-\lambda\theta_i = \tau U/W$ will be satisfied.

The discontinuous condition on the square gradient of ϕ in Eq. (11) can be implemented in finite elements by rewriting as

$$\frac{\mathrm{d}\phi^+}{\mathrm{d}z} - \frac{\mathrm{d}\phi^-}{\mathrm{d}z} = \frac{2h(\lambda\theta_i)}{W^2} \Big/ \left(\frac{\mathrm{d}\phi^+}{\mathrm{d}z} + \frac{\mathrm{d}\phi^-}{\mathrm{d}z}\right). \tag{14}$$

Introducing the solutions for ϕ^+ and ϕ^- in Eq. (14) as above, we have a simple form that can be used as a source for a line integral along the interface in the two dimensional case. In the 3D simulation the line integral is replaced by an integration over the surface of the interface

$$\int_{\phi=0} (\nabla\phi \cdot v)\mathrm{d}\Gamma \sim \frac{\mathrm{d}\phi^+}{\mathrm{d}z} - \frac{\mathrm{d}\phi^-}{\mathrm{d}z} = -\frac{\tau U}{W^2} = \frac{\lambda\theta}{W}. \tag{15}$$

2.2. Simulation of settling dendrites in the global/coarse domain

The global domain, Ω_g, is the whole domain where the dendrite is growing and settling down by gravity forces. In this domain, we will not re-simulate the growth of dendrite. The shape and size of the dendrite are taken from the local domain to its correct position in the global domain. Here, the solid particle is considered as a rigid body and the melt fluid, which is driven by the hydrodynamic and gravity forces, will drive the solidified part as a rigid body. The method considered in this paper is a fictitious domain method, which was introduced by Glowinski [25] and improved by Patankar [27]. In this model, the flow in the solidified part is constrained to be a rigid body motion using a field of Lagrange multipliers.

Let $\mathcal{F}$ be the fluid domain and $\mathcal{S}$ be the solid domain in Ω_g. The constraint of rigid body motion is represented by

$$\mathbf{u} = \mathbf{U} + \bar{\omega} \times \mathbf{r} \quad \text{in } \mathcal{S}, \tag{16}$$

where $\mathbf{u}$ is the velocity of the fluid at a point in the solidified part; $\mathbf{U}$ and ω are the translational and angular velocities of the solid particle, respectively; and $\mathbf{r}$ is the position vector of the point with respect to the centre of mass of the solid particle.

The conservation of momentum equations in $\mathcal{F}$ are

$$\frac{\partial\mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u} = -\frac{1}{\rho_f}\nabla p + \frac{1}{\rho_f}\nabla\cdot\mu\left(\nabla\mathbf{u} + \nabla\mathbf{u}^T\right) + \mathbf{g} \quad \text{in } \mathcal{F}, \tag{17}$$

$$\nabla\cdot\mathbf{u} = 0 \quad \text{in } \mathcal{F}, \tag{18}$$

where ρ_f is the fluid density, $\mathbf{u}$ is the fluid velocity and $\mathbf{g}$ is the acceleration due to gravity.

The conservation of momentum equations in the solidified domain $\mathcal{S}$ is

$$\frac{\partial\mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u} = -\frac{1}{\rho_s}\nabla p + \frac{1}{\rho_s}\nabla\cdot\mathbf{D}[\Lambda] + \mathbf{g} \quad \text{in } \mathcal{S}, \tag{19}$$

$$\nabla\cdot\mathbf{u} = 0 \quad \text{in } \mathcal{S}, \tag{20}$$

$$\mathbf{D}[\mathbf{u}] = \frac{1}{2}\left(\nabla\mathbf{u} + \nabla\mathbf{u}^{\mathrm{T}}\right) = 0 \quad \text{in } \mathcal{S}, \tag{21}$$

where ρ_s is the density of solid phase, Λ is the Lagrange multiplier due to rigid constraint and $\mathbf{D}[\Lambda]$ is an extra term due to the stress inside the solid which is required to maintain rigidity. In Eq. (19) the viscous diffusion term is absent since the deformation rate is constrained to be zero inside the particle domain.

The no-slip boundary and hydrodynamic force on the solid–liquid interface are implicit in the combined system (fictitious domain method) since it becomes an internal force in the combined system. More about the combined system can be found in Glowinski et al. [25,26] and Patankar et al. [27,33]. In the following section of our paper, we shall have a cursory review of the implementation of this method.

3. Numerical implementation

The governing equations for the free moving dendritic growth are solved using the femLego tool [34]. The dendritic growth is simulated on an adaptive mesh with refinement/derefinement. The domain size of this mesh is small, just sufficient to cover a fully developed dendrite. The other much larger mesh, the global mesh, is used to simulate the sedimentation problem. We use a uniform mesh for this domain. The solution procedure is as follows.

Step 1: Simulate the dendritic growth.

- Given $\mathbf{u}_g^n$, θ_g^n and $\mathcal{S}(t)$, find the boundary condition $\mathbf{u}_l^n$ and θ_l^n for the local domain Γ_l.
- Solve the phase field Eqs. (5) and (6) for the local domain.
- Solve the Navier–Stokes Eqs. (1) and (2) with a permeability term in order to ensure the correction of the shear stress at the interface and hold the solid stationary.

Step 2: Simulate the dendritic sedimentation.

- Updating the solid phase in the global mesh by interpolating from the phase field in the local mesh. Since the global mesh is uniform, the work is fairly fast.
- Solve Eqs. (17)–(19) for the entire domain $\mathcal{C} = \mathcal{S} + \mathcal{F}$.
- Calculate the particle velocities: Given $\mathbf{u}_g^*$ and $\mathcal{S}(t)$, find the translational velocity $\mathbf{U}^{n+1}$ and angular velocity ω^{n+1}.
- Explicit update of the particle position.

$$\mathbf{X}^{n+1} = \mathbf{X}^n + \left(\frac{\mathbf{U}^{n+1} + \mathbf{U}^n}{2}\right)\Delta t. \tag{22}$$

- Enforce rigid motion for the velocities at these grid locations inside the solid phase to give the correct velocity $\mathbf{u}_g^{n+1}$.
- Solve the heat equation in global domain, θ_g^{n+1}.

Step 3: Loop to step 1 until the end of simulation.

3.1. Calculate particle velocity

Given $\mathbf{u}^n$ and the position of solid particle $\mathcal{S}(t)$, we can find the translational velocity, $\mathbf{U}^n$, and angular velocity, ω^n, of the particle:

$$\mathbf{U}^n = \frac{1}{M(t)} \int_{\mathcal{S}(t)} \rho_s \mathbf{u}^n \, dx, \tag{23}$$

$$\mathbf{I}(t)\omega^n = \int_{\mathcal{S}(t)} \mathbf{r} \times \rho_s \mathbf{u}^n \, dx, \tag{24}$$

where $M(t)$ is the mass of the particle, $\mathbf{I}(t)$ is the moment of inertia and ρ_s is the density of solid particle. In two-dimensions the moment of inertia $\mathbf{I}(t)$ is

$$\mathbf{I}(t) = \int_{\mathcal{S}(t)} \rho_s (r_x^2 + r_y^2) \, dx, \tag{25}$$

where $\mathbf{r}$ is the position relative to the centre of mass of the particle, $\mathbf{r} = \mathbf{x} - \mathbf{X}$.

3.2. Implementation of DLM

Eqs. (17)–(19) are the governing equations for both solid and liquid phase. First, we solve (17) for the entire domain $\mathcal{C} = \mathcal{S} + \mathcal{F}$. During this step, the solid object is treated the same as the fluid. The force term due to the gravity in Eq. (17) is written as $\mathbf{g}(\rho_s - \rho_f)/\rho_f$, and is only applied in the solid region $\mathcal{S}$.

Second, we follow the projection scheme that was introduced by Patankar [27], by enforcing rigid motion as written in two steps

$$\rho_s \frac{\hat{\mathbf{u}} - \mathbf{u}^*}{\Delta t} = \phi_g \mathbf{S}, \tag{26}$$

$$\rho_s \frac{\mathbf{u}^{n+1} - \hat{\mathbf{u}}}{\Delta t} = \mathbf{R}, \tag{27}$$

where $\hat{\mathbf{u}}$ is the intermediate velocity field in the particle domain, $\mathbf{R}$ is the unknown force that maintains rigidity and $\mathbf{S}$ is the source term that it includes the forces that arise from the relative density and from the collision forces for the general case where it has more than one solid particle. $\mathbf{S}$ can be calculated explicitly as follows:

$$\mathbf{S} = \rho_s \mathbf{A}_c - (\rho_s - \rho_f)\left[\frac{\mathbf{u}^* - \mathbf{u}^n}{\Delta t} + (\mathbf{u}^* \cdot \nabla)\mathbf{u}^*\right], \tag{28}$$

where ϕ_g represents the fraction of volume of solid phase, $\phi_g = 1$ in solid and $\phi_g = 0$ in liquid phase. $\mathbf{A}_c$ is the acceleration of the particle due to collision [27].

The Eq. (27) is a projection equation. It maintains the rigid motion in a similar way as the pressure projection. So, to find an equation for the Lagrange multiplier, Patankar et al. took the divergence of (21) and substituted (27) into that, which yields an equation for $\mathbf{R}$,

$$\mathbf{D}\left[\mathbf{u}^{n+1}\right] = \mathbf{D}\left[\hat{\mathbf{u}} + \frac{\Delta t}{\rho_s}\mathbf{R}\right] = 0. \tag{29}$$

The above equation implies that $\hat{\mathbf{u}} + (\Delta t \mathbf{R})/\rho_s$ is a rigid body motion, but it gives no information about what this rigid motion should be. To simplify, Patankar imposes the condition that, in the projection step, Eq. (27), the total linear and angular momenta in the solid region $\mathcal{S}$ should be conserved. Then, the solution for $\mathbf{u}^{n+1}$ in solid region $\mathcal{S}$ is $\mathbf{u}^{n+1} = \hat{\mathbf{u}}_R$. Where, $\hat{\mathbf{u}}_R$ is the velocity field of rigid motion given by

$$\hat{\mathbf{u}}_R = \mathbf{U} + \bar{\omega} \times \mathbf{r}. \tag{30}$$

The rigid body velocity shall now distributed over the entire global domains to get the final velocity as

$$\mathbf{u}^{n+1} = (1 - \phi_g)\hat{\mathbf{u}} + \phi_g \hat{\mathbf{u}}_R. \tag{31}$$

4. Validation

A first test case is a cylindrical solid placed at the centre of a cylinder of a radius 1 with prescribed unit tangential velocity all around its boundary. This is chosen to verify the accuracy of the angular velocity computation. This simulation is run for different mesh spacings. $h = 0.05$, $h = 0.025$ and $h = 0.0125$. The solid particle is placed at the centre with diameter $d = 0.5$. The flow is simulated with Reynolds number $Re = 1$ and it was at rest in the beginning.

Fig. 3 shows the velocity field of a rotating cylinder. After a short time of keeping still, the particle begins to rotate and quickly reaches to terminal angular velocity of $\omega \approx 1$. Fig. 4 shows the evolution of the angular velocity of the cylindrical particle with different mesh sizes $h = 0.05$, 0.025 and 0.0125. Where it shows that the computed solutions converged with respect to the mesh size. A small error on the computation comes from the quality of mesh. It makes the centre of mass of the solid particle vary in time and then slows the rotational speed.

The second benchmark problem is the sedimentation of a single particle. As shown in Fig. 5, the positions of two kinds of rigid bodies are shown in sequential times of 0.1 interval from $t = 0$ to 1. The computational domain is a rectangle of 4×8. The diameter of the circular particle and the longest axis of ellipse is 0.5. The

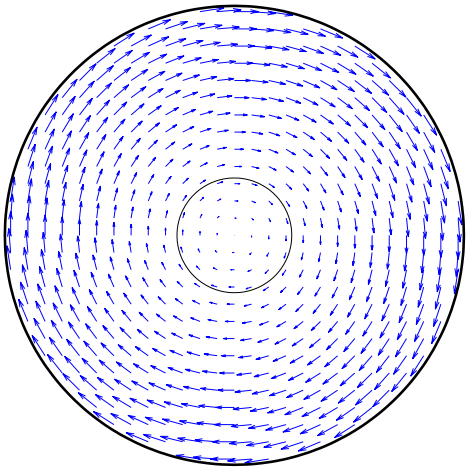


Fig. 3. Velocity field of a Couette flow with a particle inside.

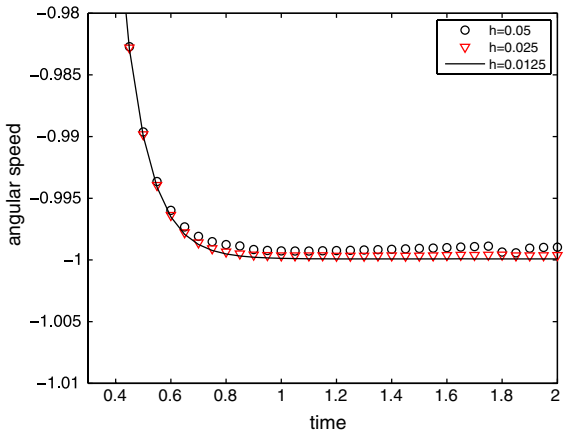


Fig. 4. Evolution of the angular velocity of solid particle.

material properties are $\rho_s = 1.25$, $\rho_l = 1.0$, $v = 0.1$. Initially, two objects are placed at $[0,0]$. Due to gravity forces and hydrodynamic forces, the objects start to sediment to the bottom of the domain. Fig. 6 shows the vorticity of two objects at $t = 1$. The elliptical object is falling down with a significantly reduced speed due to the tumbling of an asymmetric object. It is observed that, there is a *symmetry breaking* problem that takes place in the circular particle case. It falls along a trajectory that deviates slightly from a straight line. The same phenomenon was also pointed out by Glowinski [26].

The movement speed of the solid particle in the case of a circular object is simulated for different mesh sizes and compared well with the published solution in [26]. Fig. 7 shows the histories of translation velocity of the disk with a diameter $d = 0.25$. The material properties are $\rho_s = 1.25$, $\rho_l = 1$ and $\mu = 0.1$. The figure shows a converged solution and agrees well with the simulation data of Glowinski [26].

5. Dendritic growth in a gravity environment

For a dendritic growth into an undercooled pure metal, we consider the growth of a single dendrite and allow it to move freely during the growth. The convection is simulated as it settles due to gravity. The global domain is a rectangle of 8 by 16 non-dimensional units. A uniform mesh is used here with a spacing of 0.04. The local mesh is another rectangle of 5 by 5 with an adaptive mesh refinement and derefinement. The

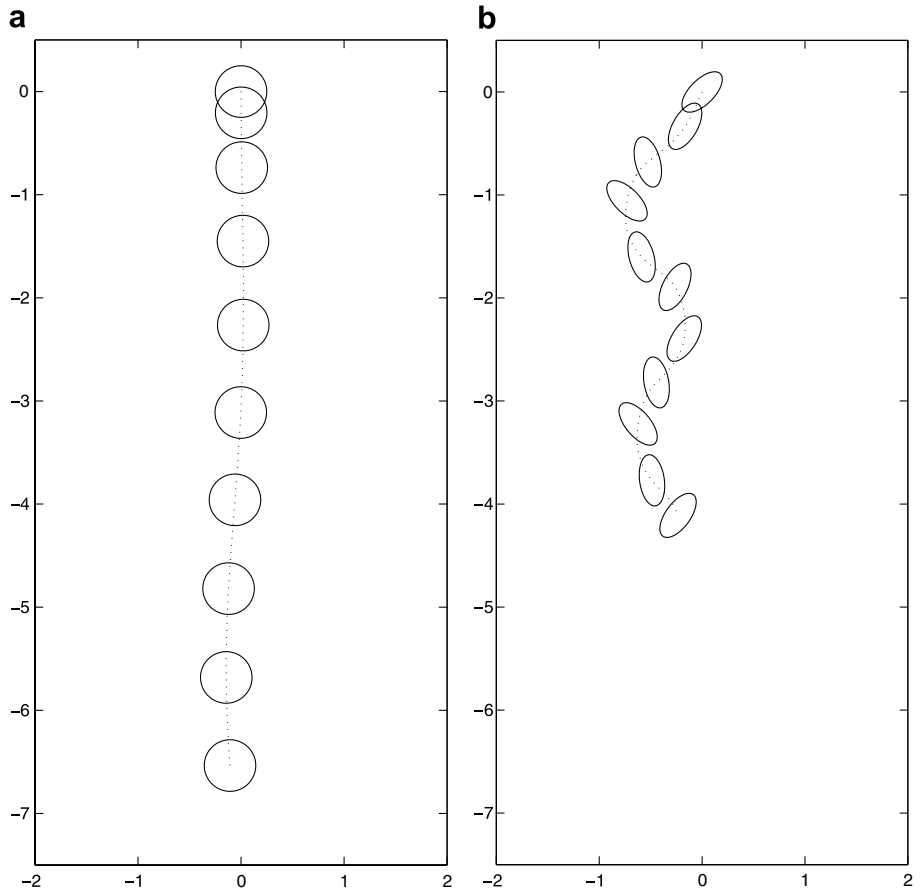


Fig. 5. The tracking position of a (a) circle; (b) ellipse; time of intervals $t = 0.1$.

minimum mesh size in this simulation is set to $h = 510^{-4}$. The time-step is therefore chosen to be $\mathrm{d}t = 810^{-4}$ to assure the stability of the scheme that is used to solve the phase-field equation in the local mesh. This implies that a local Courant number is $Co \approx 1.3$.

Fig. 8(a) shows the dendritic growth problem under anisotropy with four-fold symmetry. The dynamical evolution of the morphology and position of the dendrite is shown from $t = 0.1$ to $t = 1.3$ with time intervals of 0.2. At time zero, we consider a small initial seed as a circle of radius 0.035 with initial temperature 0 and located at position $[0, 0]$. The non-dimensional parameters used in this simulation are $Pr = 1$, $Pe = 0.5$. The density of the solidified part is $\rho_s = 1.06$ to compare with the density of liquid part which is $\rho_l = 1.0$. The temperature in the liquid phase is $\theta_\infty = -0.5$, $W = 0.02$, $\tau = 0.002858$ and $\lambda = 12.858$, and the initial orientation of dendrite is $\beta_0 = 0$. As can be observed in Fig. 8, the dendrite is growing and does not break the symmetry in the horizontal axis. The growth speed of upstream and downstream branches are different due to the downward motion of the dendrite. With the low density difference and high viscosity of liquid melt, the rotation is negligible and the shape of the dendrite is symmetric. At $t = 1.3$, the translational velocity is $\mathbf{U} = 6.03$ and the angular velocity is $\omega = 0.03$.

Fig. 8(b) shows the temperature contours at $t = 1.3$ in the global domain, which are valid outside the local mesh, i.e. outside the crystal. Obviously, the gradient of temperature on the downward branch is higher than the vertical and upward branches due to the compression of isotherms there. Consequently, the growth velocity of the upstream and vertical branches is faster than the growth speed of the downstream branch. The velocity and vorticity of the flow in the global domain is shown in Fig. 9. There is a small vortex pair created behind the dendrite. It is not clear in the simulation on the global domain. But it is clear and more accurate in the local simulation that will appear later in Fig. 11.

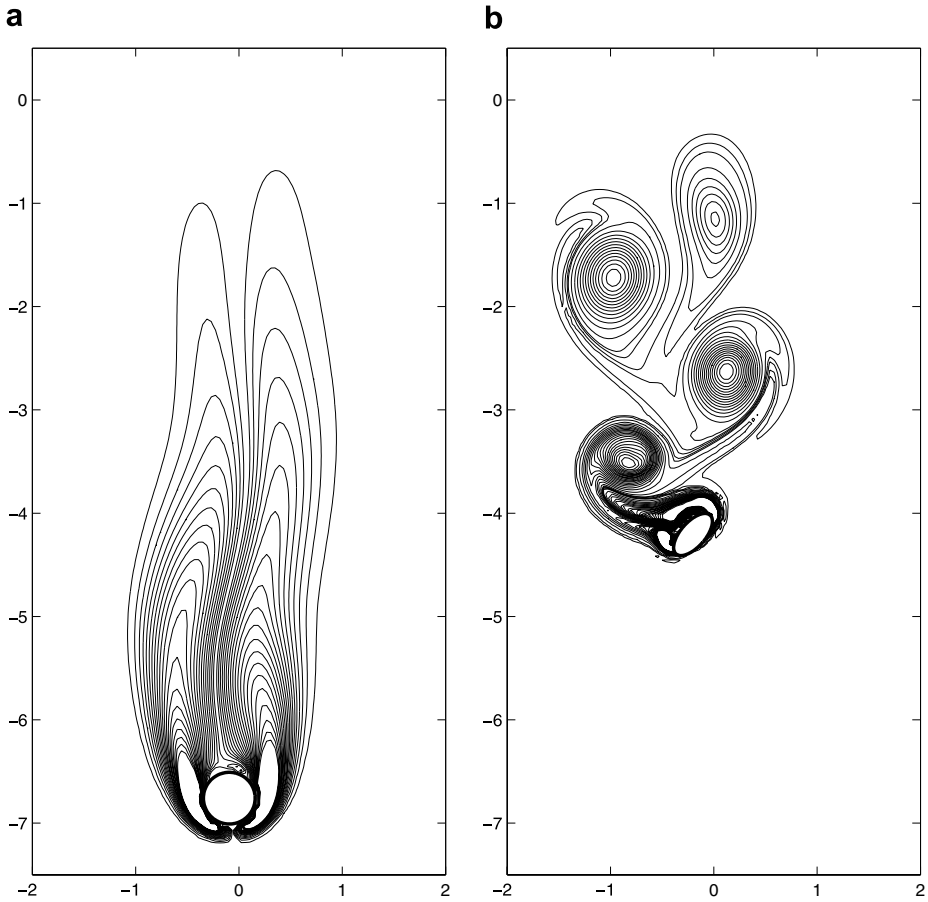


Fig. 6. Vorticity of a (a) circle; (b) ellipse.

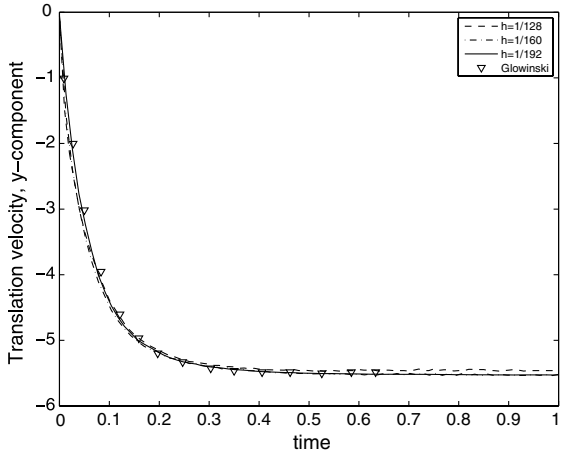


Fig. 7. Histories of translation velocity of the disk $d = 0.25$ for $\rho_s = 1.25$, $\rho_l = 1$, $\mu = 0.1$.

The time history of the tip velocity of the different branches of the dendrite is shown in Fig. 10(a). Obviously, the tip velocity quickly reaches a quasi-steady state at $t > 0.2$. When the dendrite grows, the effect of gravity increases. Therefore, the settling velocity of this dendrite is also increased. Fig. 10(b) shows that the

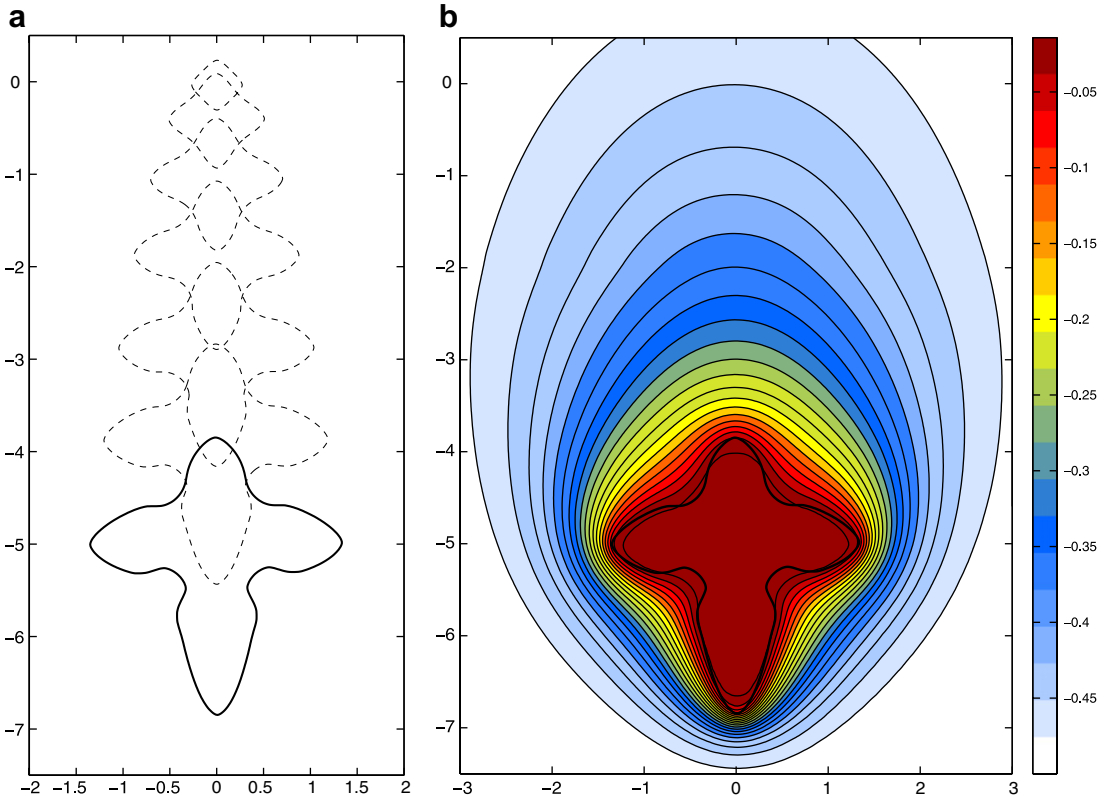


Fig. 8. The dendritic settling under gravity.

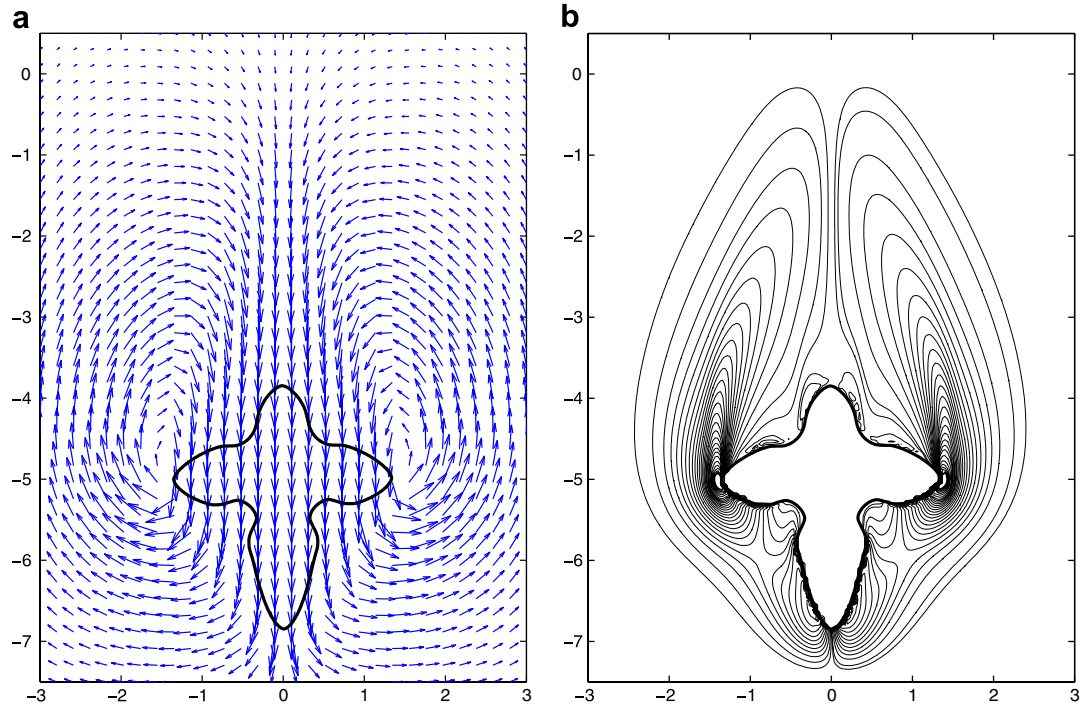


Fig. 9. The velocity and vorticity in global domain at $t = 1.3$.

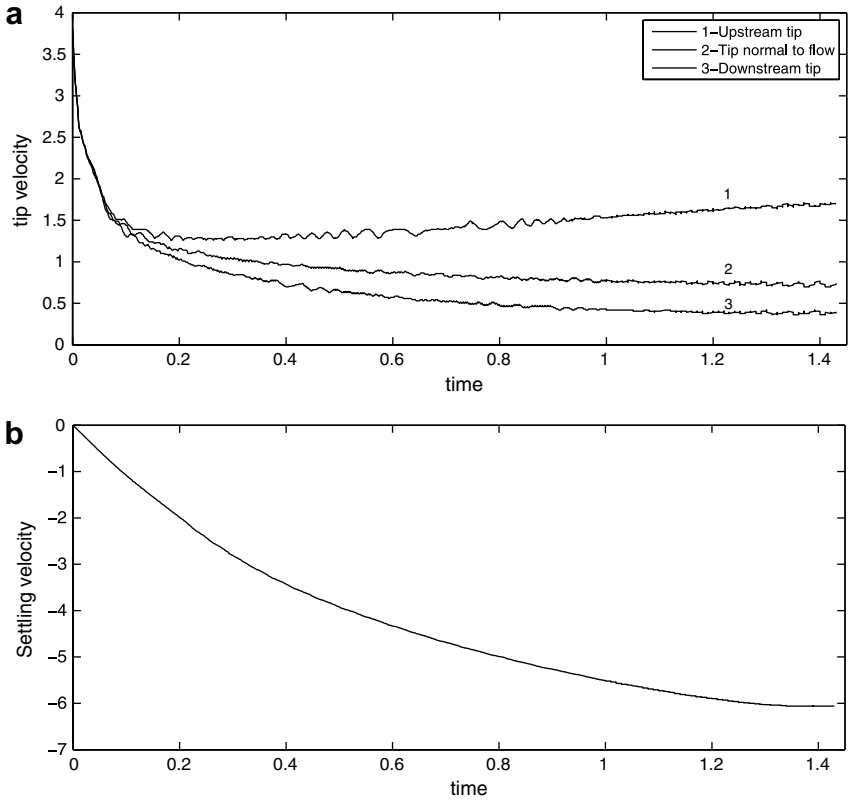


Fig. 10. The tip velocity and settling velocity.

settling velocity of the dendrite increases from 0 in the beginning to 6 at $t = 1.3$. At that time, the settling velocity is approximately six times faster than the tip velocity of the horizontal dendrite branches.

The temperature and velocity field around a tip of a dendrite are shown in Fig. 11. This solution is the local solution, that is attached to the movement of the dendrite and refined to resolve the solution close to the interface. The method of solving the dendritic growth and settling clearly shows a high accuracy of temperature and phase field around and in the interface. At the same time, we do not have to resolve whole the global domain of simulation with the same extent.

The local and global meshes are shown in Fig. 12. In the local mesh, to ensure mesh resolution along the vicinity of the interface, an adaptively refined and derefined mesh is used with an ad hoc error criterion function

$$\epsilon_1 \int_{\Omega_k} \nabla^2 \phi + \epsilon_2 \int_{\Omega_k} \nabla^2 \theta \leqslant \text{ tol.} \tag{32}$$

The implementation of the mesh adaptivity can be described as follows: at each mesh refinement step, an element Ω_k is marked for refinement if the element size is still larger than the minimum mesh size allowed, $h > h_{\min}$, and it does not meet the error criterion (32). Where ϵ_1 and ϵ_2 is an ad hoc parameters (in this simulation $\epsilon_1 = 3$ and $\epsilon_2 = 50$). In case an element meets the error criterion, it is marked for derefinement unless it is an original element. At the next refinement step, elements containing hanging nodes are marked for refinement. The refinement/derefinement stops if and only if no element is marked for refinement/derefinement. More details about this scheme can be found in [35].

The global mesh, having a uniform mesh, is useful for the implementation of the Lagrange multiplier method. The results as we can see in Fig. 8(b) are a global view of the temperature field. There is not enough accuracy in the interface but it is correct outside the interface area, where only the effects of heat diffusion and

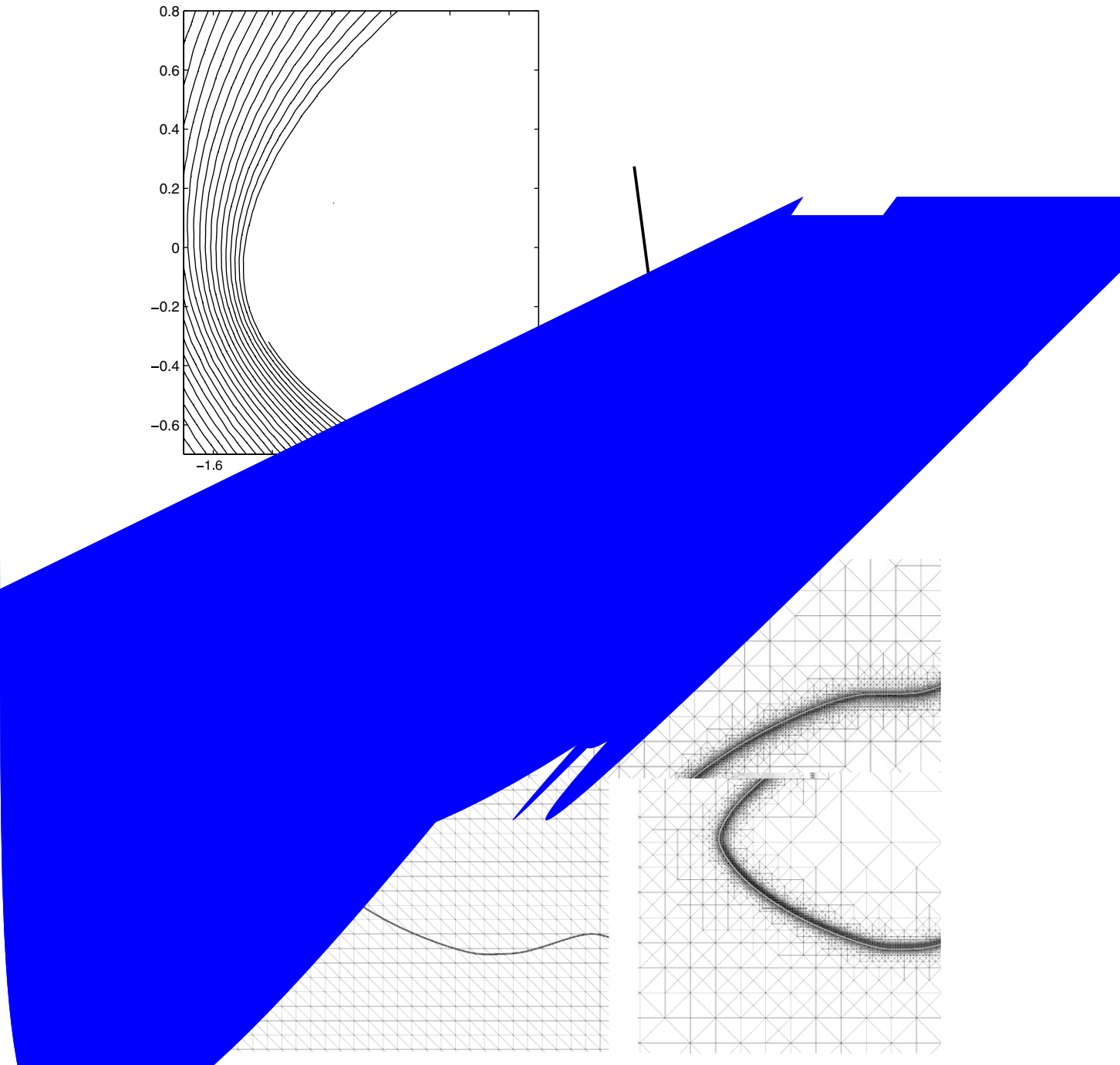
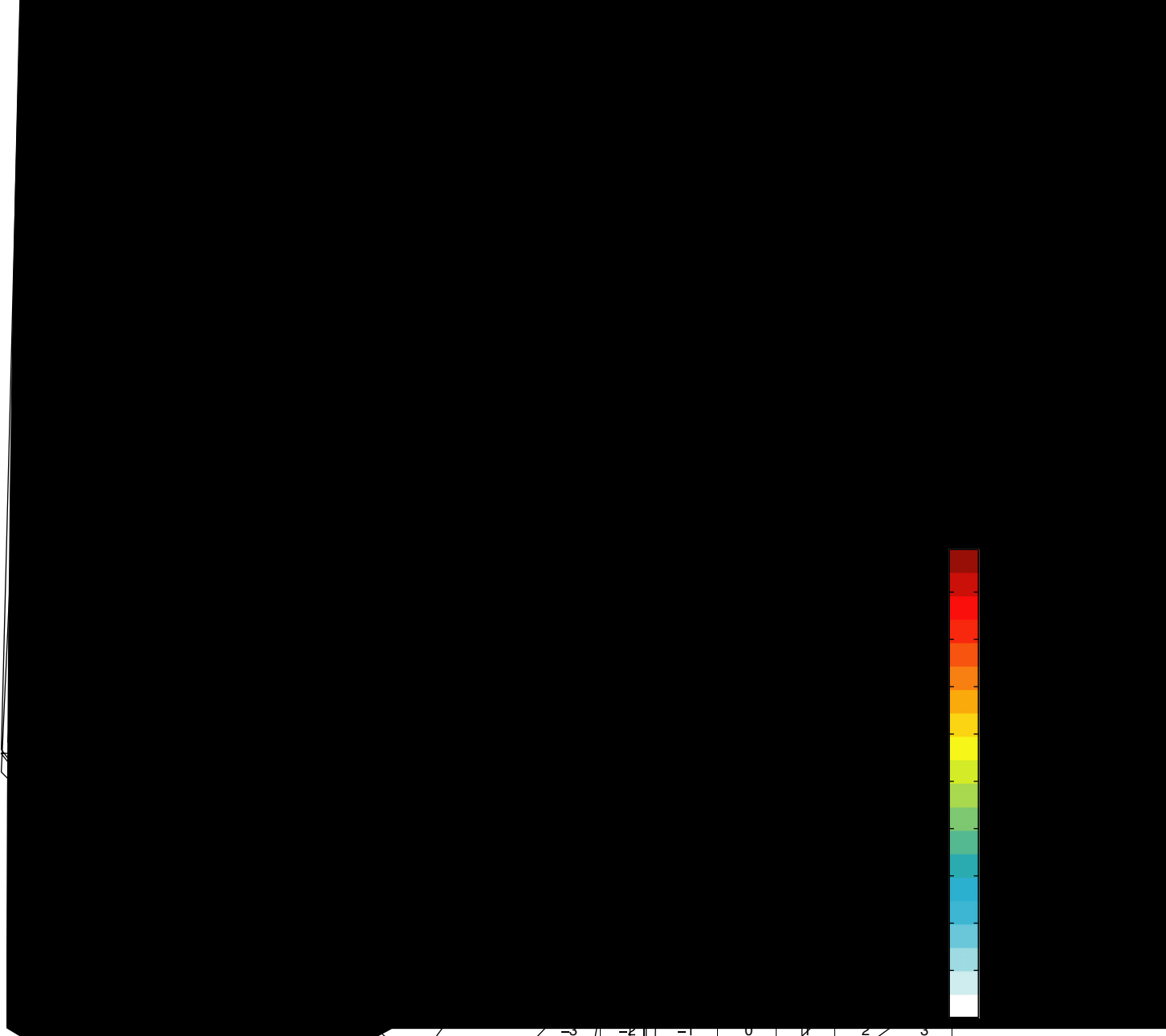


Fig. 12. The local and global mesh around a tip.

at convection are important. Meanwhile, in the local result, Fig. 11, with a refined mesh and using the semi-sharp method [24], the temperature around and in the interface is correct and yields the Gibbs–Thomson kinetics.

The solutions above assumed an initial orientation of the dendrite nucleus of $\beta_0 = 0$. That is, one of the axis of preferred growth of the dendrite is in the vertical direction, which results in a stable settling. This explains why the rotation of the dendrite is small in this case. In the next simulation, assuming that $\beta_0 = -0.34$ rad in



ne vorticity and temperature of a dendrite at $t = 1.3$, with a starting $\beta_0 = -0.34$ rad.

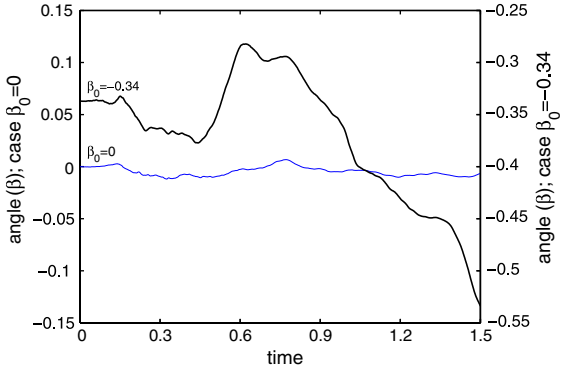


Fig. 14. The history of the orientation of a dendrite.

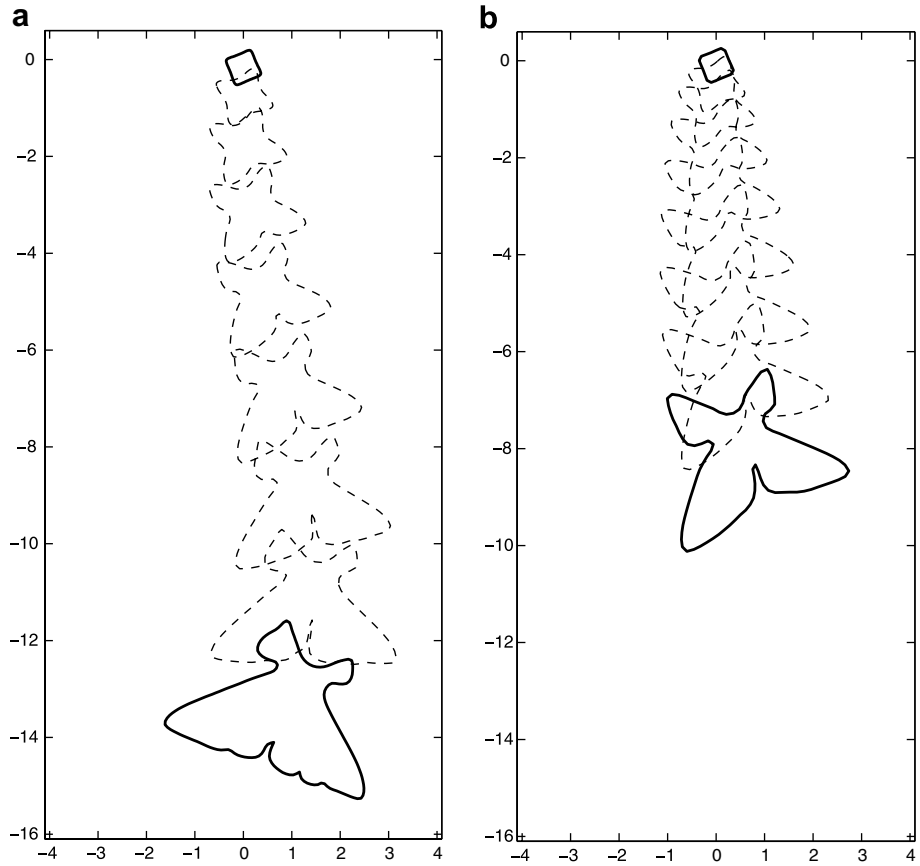


Fig. 15. The dendritic settling under gravity (a) $Pr = 0.1$; (b) $Pr = 1$; with $\beta_0 = \pi/8$.

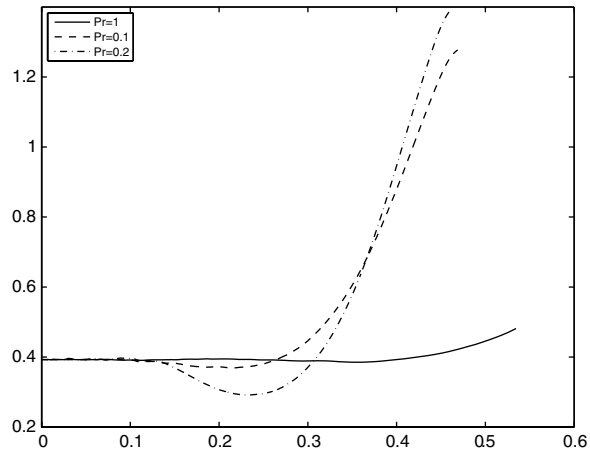


Fig. 16. The orientation angle of the dendrite.

6. Conclusion

A combination of the semisharp phase-field method and the fictitious domain method is presented to model the dendritic solidification of a pure material. The solid phase in this method is solved by a rigid constraint

equation to the fictitious fluid flow and it can be rigid, growing in time, and move freely under gravity and hydrodynamic forces. The hydrodynamic forces on the solid–liquid interface are implicit in this method. This is the advantage of the fictitious domain method (FDM) of distributed Lagrange multiplier method (DLM). The use of two domains for the growth and settling problems provides the advantage that simulations can be performed in a problem involving different scales in time and space. In those problems, a large and coarse mesh is used to simulate the settling problem, where we need to see the evolution of the dendrite settling under the gravitational force. The other mesh, a small and adaptively refined mesh, is used for the simulation of dendritic growth, where the accuracy is very important.

Numerical investigations of dendritic solidification in a gravity environment are presented. In those cases, the growth of a single dendrite freely moving through the melt is studied with different Prandtl numbers and the starting angle of the growing orientation. With a slightly different density from that of the melt, the dendrite is growing and settling under the influence of gravity. The shape and the symmetry of the dendritic solidification depends on the growth orientation of the dendrite and the Prandtl numbers.

References

[1] W.C. Winegard, B. Chalmers, Supercooling and dendritic freezing in alloys, Trans. ASM 46 (1954) 1214–1224.

[2] W. Kurz, D.J. Fisher, Fundamentals of Solidification, third ed., Trans. Tech. Publication, 1986.

[3] A. Badillo, C. Beckermann, Phase-field simulation of the columnar-to-equiaxed transition in alloy solidification, Acta Mater. 54 (2006) 2015–2026.

[4] G. Lesoult, Macrosegregation in steel strands and ingots: Characterisation, formation and consequences, Mater. Sci. Eng. A 413 (2005) 19–29.

[5] S.H. Davis, Theory er, Fundamentals oation Theory(w4d8teel

[27] N.A. Patankar, P. Singh, D.D. Joseph, R. Glowinski, T.W. Pan, A new formulation of the distributed lagrange multiplier/fictitious domain method for particulate flows, Metall. Mater. Trans. B 26 (2000) 1509–1524.
[28] H.H. Hu, D.D. Joseph, M.J. Crochet, Direct simulation of fluid particle motions, Theor. Comp. Fluid Dyn. 3 (1992) 285–306.
[29] H.H. Hu, Direct simulation of flows of solid–liquid mixtures, Int. J. Multiph. Flow 22 (1996) 335–352.
[30] A. Johnson, T. Tzeduyar, Simulation of multiple spheres falling in a liquid-filled tube, Compt. Method Appl. Mech. Eng. 134 (1996) 351–373.
[31] A. Johnson, T. Tzeduyar, 3d simulation of fluid–particle interactions with the number of particles reaching 100, Compt. Method Appl. Mech. Eng. 145 (1997) 301–321.
[32] G. Amberg, Computation of macrosergretation in an iron–carbon cast, Int. J. Heat Mass Transfer 34 (1) (1991) 17–227.
[33] N.A. Patankar, N. Sharma, A fast projection scheme for the direct numerical simulation of rigid particulate flow, Commun. Numer. Meth. Eng. 21 (2005) 419–432.
[34] G. Amberg, R. Tönhardt, C. Winkler, Finite element simulations using symbolic computing, Math. Comput. Simul. 49 (1999) 149–165.
[35] M. Do-Quang, W. Villanueva, I. Singer-Loginova, G. Amberg, Parallel adaptive computation of some time-dependent materials-related microstructural problems, Bull. Polish Acad. Sci. 55 (2) (2007) 229.
[36] R. Tönhardt, G. Amberg, Dendritic growth of randomly oriented nuclei in a shear flow, J. Crystal Growth 213 (1–2) (2000) 161–187.