

UC Santa Barbara

UC Santa Barbara Electronic Theses and Dissertations

Title

Sharp numerical methods for the solution of Stefan-type problems on adaptive quadtree grids

Permalink

<https://escholarship.org/uc/item/4931f54c>

Author

Bayat, Elyce

Publication Date

2023

Peer reviewed|Thesis/dissertation

University of California
Santa Barbara

Sharp numerical methods for the solution of Stefan-type problems on adaptive quadtree grids

A dissertation submitted in partial satisfaction
of the requirements for the degree

Doctor of Philosophy
in
Mechanical Engineering

by

Elyce Bayat

Committee in charge:

Professor Frederic Gibou, Chair
Professor Alban Sauret
Professor Emilie Dressaire
Professor Tresa Pollock

December 2023

The Dissertation of Elyce Bayat is approved.

Professor Alban Sauret

Professor Emilie Dressaire

Professor Tresa Pollock

Professor Frederic Gibou, Committee Chair

September 2023

Sharp numerical methods for the solution of Stefan-type problems on adaptive quadtree
grids

Copyright © 2023

by

Elyce Bayat

Acknowledgements

I would first like to thank my advisor, Frederic Gibou, for his kindness and support throughout the process of working on my PhD. I appreciate all your flexibility and understanding through all the different phases of the graduate school process.

I would also like to thank the members of my committee – Alban Sauret, Emilie Dressaire, and Tresa Pollock, for their time and energy in reviewing all of my materials for both my candidacy exam and my dissertation and defense. In particular, I'd like to thank Alban Sauret, who has often served as an informal co-advisor alongside Frederic during different phases of my work, which is very much appreciated.

Next I'd like to thank my colleague and good friend, Rochishnu Chowdhury, for being an excellent collaborator on the multialloy project – working together made the process all the more enjoyable. Additionally, a thanks is in order to the former lab members Raphael Egan and Daniil Bochkov, for helping me to get started out working the group and always answering my questions with a lot of patience.

To all the healthcare professionals who I've worked with throughout my graduate studies – thank you. My quality of life has been so much better than it would have been, and I feel it is important that is recognized.

And last but of course not least, I am grateful to my friends and family who have seen me through this process – thanks to Annie, Zoe, and Kerry for always being good friends with good perspectives. A special shout out of course goes to the cousin squad - Henna, Sam, Sophia, and Helena; you all are excellent. To Buddy and Marcy for always offering their unwavering support and admirable attitudes on life throughout this process. And of course to Jackie, for all of the love and support, I am every day grateful that we are a team.

Curriculum Vitæ

Elyce Bayat

Education

- 2023 Ph.D. in Mechanical Engineering (Expected), University of California, Santa Barbara.
- 2017 B.S. in Aerospace Engineering, University of California, San Diego.

Publications

Bayat, E., Egan, R., Bochkov, D., Sauret, A., Gibou, F. (2022). A sharp numerical method for the simulation of Stefan problems with convective effects. *Journal of Computational Physics*, 471, 111627.

Abstract

Sharp numerical methods for the solution of Stefan-type problems on adaptive quadtree
grids

by

Elyce Bayat

Interfacial evolution such as solidification/melting and precipitation/dissolution is governed mathematically by Stefan-type problems, free boundary problems which relate the interface evolution to the transport of temperature and/or concentration and their gradients at the interface. Typically, the classical Stefan problem considers the diffusive transport of the temperature and/or species in each phase. However, coupling the Stefan problem with fluid flow allows us to consider convective transport effects, which lead to a complex coupling between the interface morphology and corresponding flow dynamics. This type of phenomena is found in a wide variety of settings, from formation of natural landscapes, to engineering systems, to manufacturing processes.

The presented dissertation covers numerical approaches for the simulation of such Stefan-type problems coupled with incompressible fluid flow, which are shown to achieve reasonable numerical convergence, reproduce quantitative experimental results, and capture previously observed qualitative phenomena. The first chapter of the dissertation covers a foundational numerical approach to solve this type of problem in the context of solidification/melting of a pure substance. The numerical method utilizes sharp interface techniques that accurately capture discontinuities and gradients at the interface, and adaptive grids which provide a computationally efficient solution. Then, the subsequent chapters will discuss the extension and application of the foundational approach to two different problems of interest. First, Chapter 2 will discuss the extension of the founda-

tional method to the application of mineral precipitation/dissolution in porous media, and will introduce an augmentation to the numerical method in order to accommodate large disparities in the time scales governing the fluid flow and interface evolution. Then, Chapter 3 will discuss the simulation of solidification of multi-component alloys with convective effects, which is achieved by combining the strategies developed in this dissertation with an existing technique for simulating purely diffusion-driven multi-component alloy solidification.

Contents

Curriculum Vitae	v
Abstract	vi
0.1 Permissions and Attributions	1
1 Solution of the temperature driven solidification/melting problem governed by one scalar transport field – foundational numerical approach, and validation	2
1.1 Introduction	2
1.2 Problem Description	5
1.3 Numerical Approach	8
1.4 Convergence test	37
1.5 Ice growth on a cooled cylinder in crossflow – Quantitative benchmark .	40
1.6 Shape dynamics and heat transfer for varying Re and St numbers	48
1.7 Conclusion	66
2 Adaptation to the concentration driven problem governed by one scalar transport field in the context of mineral precipitation/dissolution	68
2.1 Introduction	68
2.2 Physical model	71
2.3 Numerical methods	73
2.4 Validation test – comparison with Molins et al.(1)	78
2.5 Assessing appropriate values of CFL_{QS}	82
2.6 Conclusion	90
3 Extension to multi-component alloy solidification – temperature and concentration driven solidification with multiple scalar transport fields	92
3.1 Introduction	92
3.2 Problem Description	96
3.3 Numerical approach	100
3.4 Results	106

3.5 Conclusion 119

0.1 Permissions and Attributions

1. The content of Chapter 1 has appeared previously in the Journal of Computational Physics as “A sharp numerical method for the simulation of Stefan problems with convective effects” (2). It is reproduced here in accordance with the publisher’s copyright policy, which allows authors to include their articles in a thesis or dissertation, provided that this is not to be published commercially: <https://www.elsevier.com/about/policies/copyright> Author-rights .
2. The content of Chapter 3 is the result of a direct collaboration with Rochishnu Chowdhury, as advised by Frederic Gibou.

Chapter 1

Solution of the temperature driven solidification/melting problem governed by one scalar transport field – foundational numerical approach, and validation

1.1 Introduction

Interfacial evolution driven by heat and mass transfer is governed by the classical Stefan problem, which describes such evolution driven by diffusive processes. An area of interest is the extension of the Stefan problem in the presence of convective processes, and the effect that such processes may have on the evolution of the interface. The consideration of convective effects leads to a complex coupling between the interface shape and fluid flow, giving rise to a fascinating coevolution of interface morphology and

corresponding flow dynamics (3; 4; 5; 6; 7). The advective transport of either energy or species can have a unique effect on the interface shape and vice versa; the changes in interface can lead to changes in flow and heat transfer behavior. This type of phenomena is present in a wide variety of applications – from water, energy (8; 9), and metallurgical systems (10; 11) to the formation of natural landscapes (3; 4; 7). Similar problems relating to this coupling between the time evolution of the geometry and the flow dynamics (12; 13) may be found in erosion/deposition (3), flow through porous media (14), and even in biofilm growth (15).

Regimes where diffusion and convection both play comparable roles in the interfacial growth and heat or mass transfer behavior are not well-suited for analytical approaches and demand computational solutions. Numerical solution of the classical Stefan problem (purely diffusion-driven) has been tackled using various approaches, including phase-field (16; 17; 18; 19; 20; 21), Volume of Fluids (22; 23; 24; 25; 26), Front Tracking (27), integral (28), moving grids (29; 30), finite element (31; 32), and level-set (33; 34; 30; 35; 36; 37) methods. For solution of the coupled Stefan problem with fluid dynamics, some recent approaches include finite element (38), immersed boundary (6), level-set (39), and front-tracking methods (40; 41; 42).

In this study, we utilize sharp interface numerical methods, *i.e.* methods that numerically preserve the discontinuity in discontinuous quantities. We develop an accurate and efficient method which combines three main aspects – adaptive grids for more efficient computation, sharp interface representation with accurate gradients at the interface for calculation of the interfacial velocity, and accurate application of boundary conditions at the interface to best capture the coupled nature of interface motion and flow. Firstly, the problem is multiscale in nature – the interfacial dynamics must be resolved at a relatively fine length scale, and it is beneficial to resolve other physics at varying length scales (ie. fluid recirculation zones, vortices, etc). Indeed, Huang *et al.*(6) notes the need for adap-

tive grids in order to study regimes of interest, often with thin boundary layers, without sacrificing computational cost. Thus, we use adaptive grids in order to achieve modest computational cost whilst still resolving the length scales of interest. Secondly, as the evolution of the interface is governed by gradients at the interface, we seek a method with accurate calculation of such gradients. In addition, we apply boundary conditions at the interface to avoid $O(1)$ error in gradients typical when numerically smearing the solution profile near the boundary. In particular, we make use of and build upon recent elliptic and parabolic solvers developed on Quad-/Oc-tree grids, which provide second-order accurate gradients in the maximum norm, their extension to Stefan-type problems and Navier-Stokes solvers on such grids as well as the level-set method on adaptive grids in parallel (43; 37; 44; 45; 46; 47; 48; 49; 50; 51; 52; 53; 54; 55). Lastly, the ability to apply accurate interfacial boundary conditions enables us to enforce the conservation of mass across the interface, which relates interfacial velocity and fluid velocity, rather than approximating this boundary condition as no-slip as is done in previous work (40; 6). While the no-slip approximation is often a fair one, this allows for a coupling between interfacial velocity and fluid velocity that is truer to the physics and allows for the possibility of tackling regimes in the future where the no-slip approximation may not hold.

This Chapter is organized as follows. First, we describe the physical model used for the Stefan-type problem with convective effects in Sec. 1.2. In Sec. 1.3, we outline the algorithm used in our numerical approach for the solution of this problem, and provide an overview of the numerical methods utilized. Sec. 1.4 details the results from a numerical verification test used to capture the convergence behavior of the numerical approach. Sec. 1.5 describes several numerical experiments conducted to validate the performance of the solver against past experimental results. In Sec. 1.6, we investigate the role of the Reynolds and Stefan numbers on the relationship between the interface morphology, flow dynamics, and heat transfer near the interface, and offer a new scaling law for average

Nusselt number $\overline{Nu_d}$ at the interface as it relates with Re and St . Lastly, we briefly summarize the current state of the proposed numerical method and outline possibilities for future work in Sec. 1.7.

1.2 Problem Description

1.2.1 Governing equations

We consider a physical domain Ω , which may be separated into liquid and solid subdomains denoted Ω_l and Ω_s , respectively. Throughout the paper, we use the subscript l to denote fluid quantities and the subscript s to denote solid quantities. In the present derivation, we focus on the temperature-driven solidification process; however it is noted that an analogous derivation may be done in the case of a concentration-driven dissolution/precipitation process (6).

We model the solidification process at the macroscopic level and thus consider the transition between the liquid and solid regions to be sharp, where the sharp interface between the two regions is denoted as Γ . Assuming a Newtonian, incompressible flow in the liquid domain Ω_l , conservation of mass and momentum yield the incompressible Navier-Stokes equations given by

$$\nabla \cdot \mathbf{v} = 0$$

$$\rho_l \left(\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} \right) = \mu_l \nabla^2 \mathbf{v} - \nabla P + \rho_l \mathbf{f},$$

which govern the fluid velocity and pressure, where ρ_l is the fluid density, μ_l is the fluid dynamic viscosity, and $\mathbf{f}$ may describe the presence of external forces such as gravity.

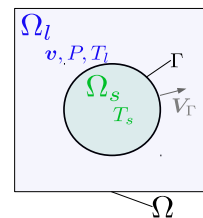


Figure 1.1: Example of notations used. The physical domain Ω , is subdivided into the liquid and solid subdomains Ω_l and Ω_s , respectively. The subdomains are separated by the evolving interface Γ .

Writing the energy conservation for the liquid domain and assuming that the internal energy is a linear function of the temperature in the range of interest yields an advection-diffusion equation in temperature in the fluid, given by

$$\frac{\partial T_l}{\partial t} + \mathbf{v} \cdot \nabla T_l = \alpha_l \nabla^2 T_l + g_l,$$

where α_l is the fluid thermal diffusivity, and g_l may describe external heat sources per unit volume divided by the fluid heat capacity per unit volume $\rho_l c_{v,l}$ in the fluid. We note that for heat transport in the fluid we neglect viscous dissipation, which is typically negligible except in the case of certain high speed flows (56). In the solid domain, the heat transfer is given by the standard heat equation

$$\frac{\partial T_s}{\partial t} = \alpha_s \nabla^2 T_s + g_s,$$

where g_s represents an analogous heat source in the solid and α_s is the solid thermal diffusivity. On the interface Γ , the energy balance reduces to the Stefan condition, given by

$$\mathbf{n} \cdot \mathbf{V}_\Gamma = -\frac{(k_l \nabla T_l - k_s \nabla T_s) \cdot \mathbf{n}}{\rho_s L},$$

where k_l and k_s are the thermal conductivities of the fluid and solid respectively, L is the latent heat of fusion, and $\mathbf{n}$ denotes the unit normal which points outward from the liquid subdomain (56). The sharp interface $\Gamma = \Gamma(t)$ which separates the liquid and the solid subdomains will then evolve with the normal velocity $\mathbf{n} \cdot \mathbf{V}_\Gamma$ governed by this conservation of energy across interface. The interfacial condition for the temperature is derived from the Gibbs-Thomson relation (57), which captures the effects of the interface curvature on the melt temperature by relating T_Γ with the melt temperature of the given species, T_m , the curvature of the interface, κ , the latent heat of fusion L , and the interfacial energy

of the solid-liquid interface, γ_{sl} . This expression is given by

$$T_\Gamma = T_m(1 - \sigma\kappa),$$

where $\sigma = \gamma_{sl}/(\rho_s L)$ (57). To determine the interfacial boundary condition for the fluid velocity, we again consider a mass balance across the interface, which reduces to the condition

$$\mathbf{n} \cdot \mathbf{v} = \mathbf{V}_\Gamma \cdot \mathbf{n} \left(1 - \frac{\rho_s}{\rho_l} \right).$$

1.2.2 Non-dimensionalized form

It is convenient to express the physical model in a non-dimensionalized form, noting the characteristic length, velocity, and time scales as d , u_∞ , and d/u_∞ , respectively. The characteristic temperature difference is defined as $\Delta T = T_\infty - T_0$, where T_∞ and T_0 are characteristic temperatures of the system being modeled. We may therefore rescale temperature as $\hat{T} = (T - T_0)/(\Delta T)$, velocity as $\hat{\mathbf{v}} = \mathbf{v}/u_\infty$, time as $\hat{t} = tu_\infty/d$, length as $\hat{\mathbf{r}} = \mathbf{r}/d$, and curvature as $\hat{\kappa} = d\kappa$. This leads to the following non-dimensional groups: the Reynolds number: $Re = \rho_l u_\infty d / \mu_l$, the Prantdl number: $Pr = \mu_l / (\rho_l \alpha_l)$, the Peclet number: $Pe = u_\infty d / \alpha_l$, the Stefan number: $St = c_{p,s} / (L \Delta T)$. Dropping the $\hat{}$ notation for convenience and taking source terms as zero, we arrive at the following dimensionless system expressed in Eq. (1.2) - (1.8).

$$\nabla \cdot \mathbf{v} = 0, \quad \text{in } \Omega_l \quad (1.2)$$

$$\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} = \frac{1}{Re} \nabla^2 \mathbf{v} - \nabla P, \quad \text{in } \Omega_l \quad (1.3)$$

$$\frac{\partial T_l}{\partial t} + \mathbf{v} \cdot \nabla T_l = \frac{1}{Pe} \nabla^2 T_l, \quad \text{in } \Omega_l \quad (1.4)$$

$$\frac{\partial T_s}{\partial t} = \frac{1}{Pe} \frac{\alpha_s}{\alpha_l} \nabla^2 T_s, \quad \text{in } \Omega_s \quad (1.5)$$

$$\mathbf{n} \cdot \mathbf{V}_\Gamma = \frac{St}{Pe} \frac{\alpha_s}{\alpha_l} \left(\nabla T_s - \frac{k_l}{k_s} \nabla T_l \right) \cdot \mathbf{n}, \quad \text{on } \Gamma \quad (1.6)$$

$$T_l|_\Gamma = T_s|_\Gamma = T_\Gamma = T_m \left(1 - \kappa \frac{\sigma}{d} \right) - \kappa \frac{\sigma}{d} \frac{T_0}{\Delta T}, \quad \text{on } \Gamma \quad (1.7)$$

$$\mathbf{n} \cdot \mathbf{v}|_\Gamma = \mathbf{n} \cdot \mathbf{V}_\Gamma \left(1 - \frac{\rho_s}{\rho_l} \right), \quad \text{on } \Gamma. \quad (1.8)$$

For the remainder of the chapter, we will refer to the problem in its non-dimensionalized form. It is important to note that this model can be applied to an analogous dissolution problem, where the transported scalar field is concentration, and a similar interfacial velocity can be derived based on species transport across the interface rather than energy.

1.3 Numerical Approach

To tackle the solution of this multiphysics coupled problem, a host of challenges arise. To adequately capture the evolution of the free boundary, which may develop to be quite irregular, and in some cases may present changes in topology, we use the level-set method introduced in (58) (see (59) for a recent review on Quad-/Oc-tree). Considering the multiscale nature of the problem, it is advantageous to utilize an adaptive grid approach, which provides a more manageable computational cost while still allowing the capturing of relevant physics in regions of interest like sharp variation of temperature near the interface, evolution of flow structures like vortices, etc. In this work, we use the

parallel level-set framework introduced in (60), which makes use of the scalable **p4est** library (61) enabling the dynamic management of a collection of adaptive quad-/oc-trees. Additionally, the boundary conditions for the temperature and the fluid’s variables are imposed in a sharp manner, *i.e.* at the interface.

The algorithm we introduce is provided in section 1.3.1; the following sections describe the details of the numerical methodologies.

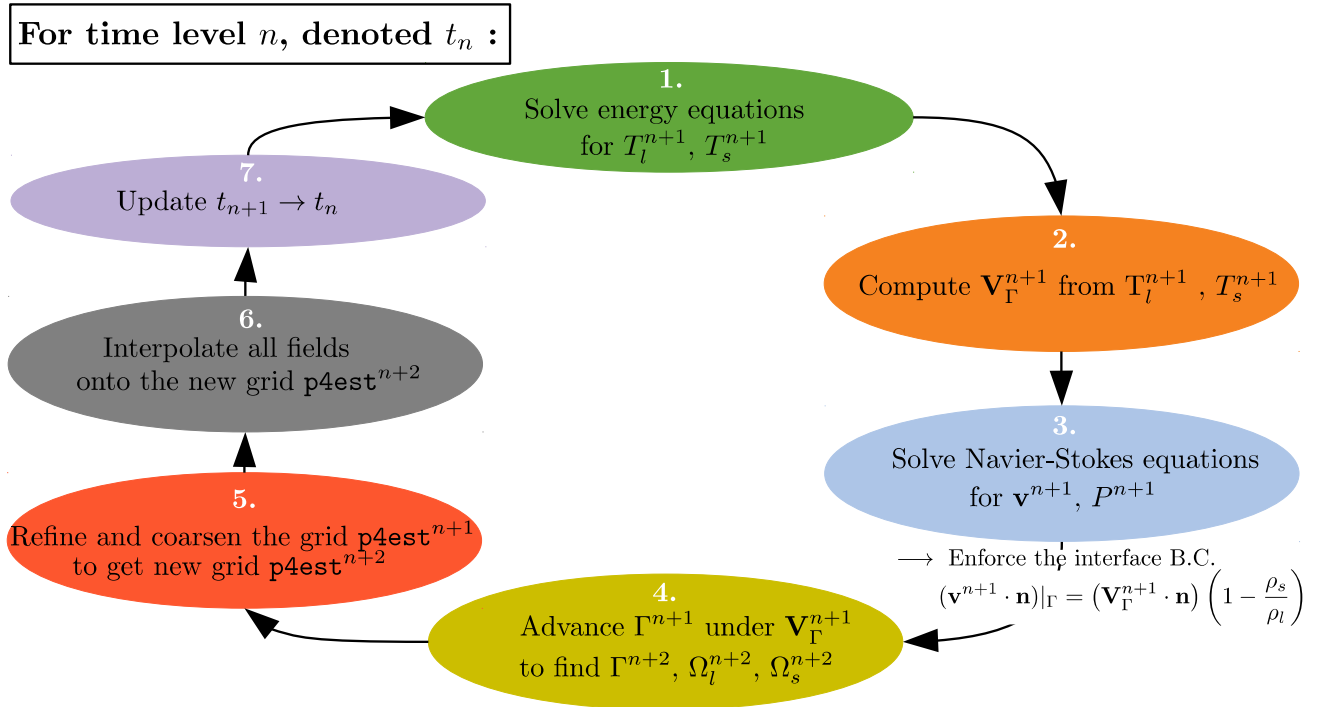


Figure 1.2: Illustration of the solution procedure.

1.3.1 General algorithm to solve the model

Given the initial conditions for temperatures, flow velocity, grids at t_{-1} , t_0 , and an initial interface location, we proceed for each n -th time level (starting with $n = 0$) as

follows:

1. The energy equations (or equivalent) given by equations (1.4) and (1.5) are solved for T_l^{n+1} and T_s^{n+1} . Advective terms are discretized using the semi-Lagrangian formulation detailed in Sec. 1.3.5 and backtrace values are found using the quadratic non-oscillatory interpolation method described in Sec. 1.3.9. The diffusive terms are treated implicitly and discretized spatially using the finite-difference approach described in Sec. 1.3.6.
2. The interfacial velocity $\mathbf{V}_\Gamma^{n+1}$ is computed as per equation (1.6) from the jump in the fields T_l^{n+1} , T_s^{n+1} . Values of T_l^{n+1} and T_s^{n+1} are extended in a narrow band across the interface using a PDE extrapolation method in order to compute the jump.
3. A pressure-free projection method is used to solve the Navier-Stokes equations (1.2),(1.3) for $\mathbf{v}^{n+1}$, P^{n+1} . Advective terms are again discretized with a semi-Lagrangian approach. Diffusive terms are treated implicitly and discretized using finite-volume methods. The interfacial boundary condition, $(\mathbf{v}^{n+1} \cdot \mathbf{n})|_\Gamma = (\mathbf{V}_\Gamma^{n+1} \cdot \mathbf{n})(1 - \rho_s/\rho_l)$, is approximated in a component-wise fashion, which is described in further detail in Sec. 1.3.7.
4. The interface Γ^{n+1} is advanced to find Γ^{n+2} and thus Ω_l^{n+2} and Ω_s^{n+2} by advecting the level-set function ϕ^{n+1} under $\mathbf{V}_\Gamma^{n+1}$ with Δt_{n+1} . The adaptive timestep Δt_{n+1} is computed via the expression $\Delta t_{n+1} = \text{CFL} \times \min(\Delta x)/\max(\mathbf{v}^{n+1}, \mathbf{V}_\Gamma^{n+1})$, where the CFL (Courant–Friedrichs–Lewy) coefficient is selected for accuracy. Further details of the level set method can be found in Sec. 1.3.2.
5. The current grid, $\mathbf{grid}^{n+1}$, is refined and coarsened according to desired criteria, such as a uniform band around the interface Γ^{n+2} , or resolution of vortex structures

defined by certain ranges of vorticity. Further details on the adaptive quadtree grids used are described in Sec. 1.3.3.

6. All fields are transferred to the new grid via interpolation, using the methods detailed in Sec. 1.3.9.

7. Proceed to the next time level

1.3.2 Level-set method for interface representation

To track the interface Γ in a sharp manner, we make use of the level-set method (58), which represents the interface as the zero-contour of a higher dimensional function, $\phi(t, \mathbf{r})$, with signed distance property. Thus, the interface and the liquid and solid subdomains can be represented as

$$\Gamma = [\mathbf{r} \in \Omega | \phi = 0], \quad \Omega_l = [\mathbf{r} \in \Omega | \phi < 0], \quad \Omega_s = [\mathbf{r} \in \Omega | \phi > 0].$$

The level-set function provides a straightforward access to the computation of the normal $\mathbf{n}$ to the interface and the interface curvature κ as

$$\mathbf{n} = \frac{\nabla \phi}{|\nabla \phi|}, \quad \kappa = \nabla \cdot \mathbf{n}.$$

To advance the interface, the level-set function is evolved under the external velocity field, $\mathbf{V}_\Gamma$, by solving the equation

$$\frac{\partial \phi}{\partial t} + \mathbf{V}_\Gamma \cdot \nabla \phi = 0,$$

which is solved using the semi-Lagrangian method detailed in Sec. 1.3.5. The level-set function is reinitialized at each time step to restore the signed distance property from any

degradation that may occur under advection. This is done by solving the reinitialization equation (62) in fictitious time τ :

$$\frac{\partial \phi}{\partial \tau} + \text{sgn}(\phi_{\tau=0}^n)(|\nabla \phi^n| - 1) = 0,$$

for a few time steps. Here, sgn is the signum function. In the present work, the reinitialization step is solved using a TVD-RK2 time-stepping scheme with a Godunov Hamiltonian discretization of $|\nabla \phi|$, and the sub-cell fix of (63), extended to the case of adaptive grids in (64).

1.3.3 Adaptive quadtree grids

In order to accurately capture the various multiscale physical phenomena in the presented problem, *ie.* interfacial dynamics, flow structures, with an accessible level of computational cost, we utilize adaptive Cartesian quadtree grids for discretization of the computational domain.

Initially, the “root” (also considered level 0) of the quadtree data structure is one cell which describes the entire computational domain. The root cell may then be split into four equal-sized “child” cells, yielding the next level of refinement (level 1). Cells may be split in this manner (a) recursively to varying levels of refinement, denoted by l , and (b) selectively according to a set of given criteria, *ie.* distance to the interface, or fluid vorticity. The size of a cell is thus given by $L/2^l$, where L is the size of the root cell, and l is the level of refinement. Limits are placed on the lowest and highest allowable refinement levels of the grid, usually denoted by $l_{\min}$ and $l_{\max}$. This process is illustrated and exemplified in Fig. 1.3.

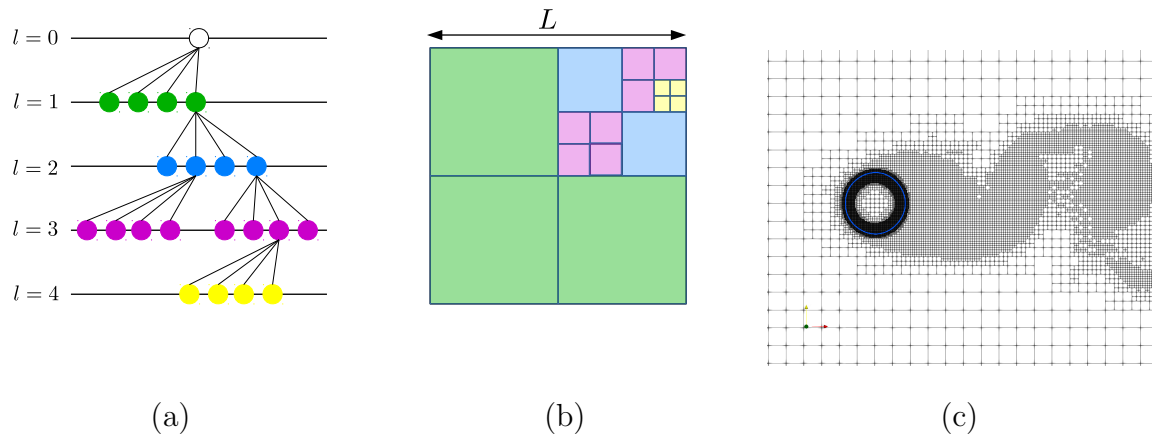


Figure 1.3: Illustration of a quad-tree grid hierarchical structure (a) associated to a computational grid (b). (c) Example of adaptive Cartesian quadtree structure, with custom refinement around the interface, as well as according to areas where properties such as fluid vorticity or gradients of temperature are above user-chosen thresholds.

The solution of the Navier-Stokes equations uses the standard MAC grid arrangement (further described in section 1.3.7) to ensure numerical stability of the projection method. This method stores the velocity components $\mathbf{v}$ at the face-centers of the grid, and the pressure P (and pressure-related Hodge variable Φ) at the cell-centers. However, the temperature fields T_l and T_s and the level-set function ϕ are stored at the nodes, allowing for the use of higher accuracy node-sampled methods which can provide second order accurate results for the solution and its gradients (described in 1.3.4) The interface velocity $\mathbf{V}_\Gamma$, computed from the gradients of temperature, is also node-sampled. An illustration of this data layout is given in Fig. 1.4.

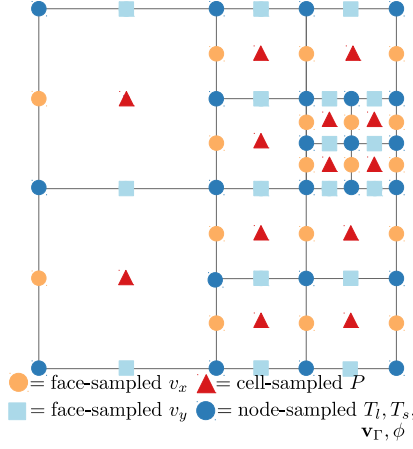


Figure 1.4: Illustration of data layout used on adaptive quadtree grids.

1.3.4 Node-based spatial discretization on quadtrees

When solving for the advection equation on quadtree grids, (65; 66) showed that a node-based data structure is advantageous when using semi-Lagrangian methods. Such an approach was extended to second-order accuracy for the level-set method and the Navier-Stokes equations in Refs. (67; 68; 69). In addition, (70; 64; 71) introduced solvers for the Poisson and heat equations on irregular domains on node-based quad-/oc-trees and showed that both the solution and its gradients are second-order accurate in the L^∞ norm. This is an advantage in the case of Stefan problems since it is the gradient of the solution that drives the overall accuracy. When considering the spatial discretization on a node-based quadtree grid structure, the inevitable case will arise in which the node about which we are discretizing is missing a direct neighbor in one of the Cartesian directions (the case referred to as a T-junction). An example of such a scenario is illustrated in Fig. 1.5, which provides the definition of the notations used in the following.

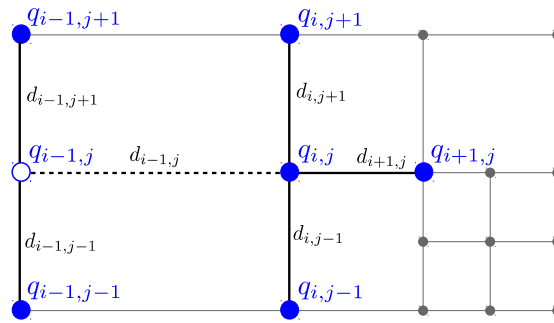


Figure 1.5: Example discretization on quadtree grid in T-junction case. The missing data of a field q at the grid node (x_{i-1}, y_j) is denoted by $q_{i-1,j}$, illustrated by the white colored grid node with blue edge color. Its distance to the node (x_i, y_j) is denoted by $d_{i-1,j}$.

In cases such as these standard formulas may be used in both Cartesian directions, with the exception that the missing value ($q_{i-1,j}$ in this case) is defined using quadratic interpolation, as introduced in (67):

$$q_{i-1,j} = \frac{d_{i-1,j-1} q_{i-1,j+1} + d_{i-1,j+1} q_{i-1,j-1}}{d_{i-1,j-1} + d_{i-1,j+1}} - \left(\frac{d_{i-1,j-1} d_{i-1,j+1}}{d_{i-1,j+1} + d_{i-1,j-1}} \right) \left(\frac{q_{i,j+1} - q_{i,j}}{d_{i,j+1}} + \frac{q_{i,j-1} - q_{i,j}}{d_{i,j-1}} \right).$$

This definition of a ghost value is always possible in a node-based setting and is third-order accurate. It then enables one to approximate the first-order and the second-order derivatives with second-order and first-order accuracy, respectively, using standard finite difference formulas as if the grid did not have T-junction node. For example, considering the layout in Fig. 1.5, the discretizations of the first- and second-order derivatives of q

(denoted as q_x and q_{xx} , respectively) at the node (i, j) are:

$$q_x|_{i,j} \approx \frac{q_{i+1,j} - q_{i,j}}{d_{i+1,j}} \frac{d_{i-1,j}}{d_{i+1,j} + d_{i-1,j}} + \frac{q_{i,j} - q_{i-1,j}}{d_{i-1,j}} \frac{d_{i+1,j}}{d_{i+1,j} + d_{i-1,j}},$$

$$q_{xx}|_{i,j} \approx \left(\frac{q_{i+1,j} - q_{i,j}}{d_{i+1,j}} - \frac{q_{i,j} - q_{i-1,j}}{d_{i-1,j}} \right) \left(\frac{2}{d_{i+1,j} + d_{i-1,j}} \right).$$

Discretizing the partial derivatives in a dimension-by-dimension fashion allows one to write the discretizations needed to approximate the Stefan model, including the implicit treatment of the parabolic parts of the equations and the level-set equations.

1.3.5 Semi-Lagrangian discretizations of advective terms

The need for discretization of advective terms arises multiple times in the given system of equations: once for the advection of the level-set function used to evolve the interface, once to discretize the advective terms in the Navier-Stokes equations, and once to capture the advection of temperature in the liquid subdomain. For each of these cases, we make use of a semi-Lagrangian discretization with a Backward Difference Formula. To illustrate this, let us consider the case of some scalar field q , advected under a given velocity field $\mathbf{u}$:

$$q_t + (\mathbf{u} \cdot \nabla)q = \psi, \tag{1.9}$$

with source term ψ . We may write a discretization for equation (1.9) by tracing the values of q through the characteristic curve of q advected under $\mathbf{u}$. These backtraced values of q , denoted by q_d^n and q_d^{n-1} , are found by evaluating q at departure points $\mathbf{r}_d$ along the characteristic curve that passes through the point $\mathbf{r}(t = t_{n+1})$. Thus,

$$q_d^n = q(t_{n+1}, \mathbf{r}_d^n), \quad q_d^{n-1} = q(t_n, \mathbf{r}_d^{n-1}).$$

The first-order discretization using only one departure point is given by

$$\frac{q^{n+1} - q_d^n}{\Delta t_{n+1}} = \psi^{n+1}.$$

The second-order method, using two departure points, becomes

$$\alpha \frac{q^{n+1} - q_d^n}{\Delta t_{n+1}} + \beta \frac{q_d^n - q_d^{n-1}}{\Delta t_n} = \psi^{n+1},$$

where α and β are Backward Difference Formula coefficients.

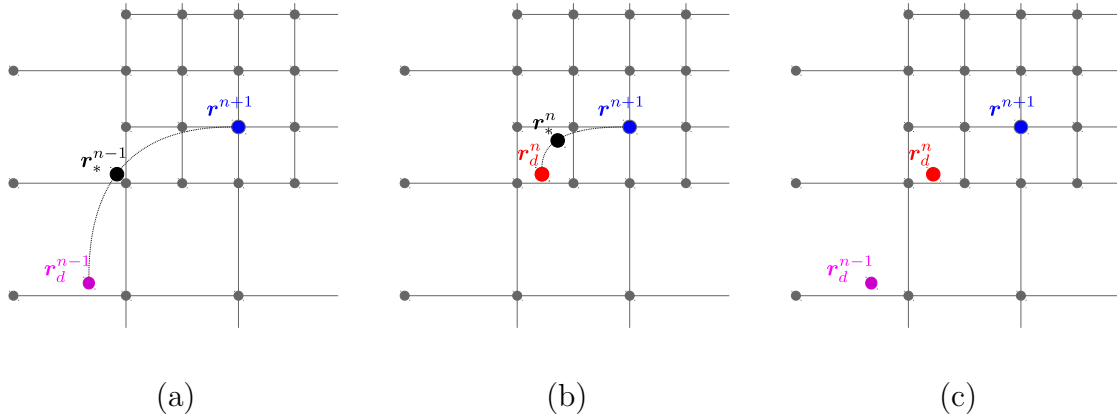


Figure 1.6: Illustration of procedure to acquire departure points for Semi-Lagrangian discretization. (a) Computing departure point r_d^{n-1} , (b) computing departure point r_d^n , and (c) departure points r_d^{n-1} and r_d^n .

To obtain the backtraced values q_d^{n+1} and q_d^n , we first must obtain the departure points (see Fig. 1.6), computed from solving the characteristic equation $d\mathbf{r}/ds = \mathbf{u}$, where s is the direction along the curve. The departure points therefore can be found by tracing this characteristic curve backward in time using the midpoint rule. Given a point r^{n+1} for which we wish to compute departure points, we define intermediate points r_*^n and

$\mathbf{r}_*^{n-1}$ and compute

$$\mathbf{r}_*^n = \mathbf{r}^n - \left(\frac{\Delta t_{n+1}}{2} \right) \mathbf{u}(t_{n+1}, \mathbf{r}^n), \quad \mathbf{r}_d^n = \mathbf{r}^n - (\Delta t_{n+1}) \mathbf{u}(t_{n+\frac{1}{2}}, \mathbf{r}_*^n),$$

and

$$\mathbf{r}_*^{n-1} = \mathbf{r}^n - (\Delta t_{n+1}) \mathbf{u}(t_n, \mathbf{r}^n), \quad \mathbf{r}_d^{n-1} = \mathbf{r}^n - (\Delta t_{n+1} + \Delta t_n) \mathbf{u}(t_n, \mathbf{r}_*^{n-1}).$$

The intermediate velocity $\mathbf{u}(t_{n+\frac{1}{2}}, \mathbf{r}_*^n)$ is evaluated at the location $\mathbf{r}_*^n$ via multilinear interpolations from the velocity fields at times t_n and t_{n-1} , and is given by

$$\mathbf{u}(t_{n+\frac{1}{2}}, \mathbf{r}_*^n) = \left(1 + \frac{\Delta t_{n+1}}{2\Delta t_n} \right) \mathbf{u}(t_{n+1}, \mathbf{r}_*^n) + \left(\frac{\Delta t_{n+1}}{2\Delta t_n} \right) \mathbf{u}(t_n, \mathbf{r}_*^n) + O(\Delta t^2).$$

As the computed departure points rarely ever fall at the precise location of grid nodes, the backtraced values of q are evaluated at the departure points using the quadratic non-oscillatory interpolation method of Sec. 1.3.9. In the first order case, we simply select $\alpha = 1$. For the second order formula, we select α and β as

$$\alpha = \frac{2\Delta t_{n+1} + \Delta t_n}{\Delta t_{n+1} + \Delta t_n}, \quad \beta = -\frac{\Delta t_{n+1}}{\Delta t_{n+1} + \Delta t_n}, \quad (1.10)$$

to ensure second-order accuracy, as demonstrated in (54).

For the three varying cases for which this discretization method is used, we may simply select α , β , Φ , and ψ appropriately. For example:

1. For the level-set function advection under the interfacial velocity field, we select $q = \phi$, $\mathbf{u} = \mathbf{V}_\Gamma$, $\psi = 0$, $\alpha = 1$, which yields the update $\phi^n = \phi_d^{n-1}$
2. For the Navier-Stokes advection, we select q to be each of the components of $\mathbf{v}$,

$\mathbf{u} = \mathbf{v}$, $\psi =$ viscous terms, and α, β as given by equation (1.10)

3. For advection of the scalar field T_l in the fluid, we choose $q = T_l$, $\mathbf{u} = \mathbf{v}$, $\psi =$ diffusion terms, and α, β as given by equation (1.10),

where $\mathbf{V}_\Gamma$, $\mathbf{v}$, and T_l are the interface velocity, fluid velocity, and fluid temperature, respectively. To evaluate the backtraced values for the fluid velocity components, the face-sampled velocity fields are interpolated from the faces to the nodes at the end of each timestep to allow for node-based calculation of semi-Lagrangian discretization terms. This is done because (a) the fluid temperature equation requires the computation of node-based advection terms and (b) for the Navier-Stokes advection, the node-based interpolation method is much less expensive than the face and cell based method.

We also note that, although a backtraced point should follow the velocity field and therefore does not land inside the solid region, numerical approximations may be responsible for this case to occur. However, we are extrapolating the fields inside the solid and those are used in such pathological cases.

1.3.6 Transport equations on node-sampled fields

To write the full discretization for the advection-diffusion equations for the temperature in the liquid (1.4) and solid (1.5) phases, we discretize the diffusive terms implicitly in time and with the approximations of Sec. 1.3.3 in space; the advective terms are discretized using the semi-Lagrangian discretization described in Sec. 1.3.5. Thus, the heat transport equations can be discretized as

$$\alpha \frac{T_{l,s}^{n+1}{}_{i,j} - T_{l,s,d}^n}{\Delta t_{n+1}} + \beta \frac{T_{l,s,d}^n - T_{l,s,d}^{n-1}}{\Delta t_n} = g_{l,s}^{n+1} +$$

$$\gamma \left(\frac{T_{l,s}^{n+1}{}_{i+1,j} - T_{l,s}^{n+1}{}_{i,j}}{d_{i+1,j}} - \frac{T_{l,s}^{n+1}{}_{i,j} - T_{l,s}^{n+1}{}_{i-1,j}}{d_{i-1,j}} \right) \left(\frac{2}{d_{i+1,j} + d_{i-1,j}} \right) +$$

$$\gamma \left(\frac{T_{l,s}^{n+1}{}_{i,j+1} - T_{l,s}^{n+1}{}_{i,j}}{d_{i,j+1}} - \frac{T_{l,s}^{n+1}{}_{i,j} - T_{l,s}^{n+1}{}_{i,j-1}}{d_{i,j-1}} \right) \left(\frac{2}{d_{i,j+1} + d_{i,j-1}} \right),$$

where γ is $\frac{1}{Pe}$ in the fluid case and $\frac{1}{Pe} \frac{\alpha_s}{\alpha_l}$ in the solid case. In the fluid region, the points $T_{l,s,d}^n$ and $T_{l,s,d}^{n-1}$ denote the Semi-Lagrangian backtraced values for which the departure points are computed using the fluid velocities $\mathbf{v}^n$ and $\mathbf{v}^{n-1}$. For the solid region, these points are simply the value at the node, as there is no advective effect in the solid. The coefficients α and β are chosen for the fluid region as discussed in Sec. 1.3.5, and are taken to be 1 and 0 respectively in the solid.

Special care must be taken to impose the Gibbs-Thomson boundary condition (1.7) for the temperature at the solid-liquid interface. We use a sharp treatment based on the Shortley-Weller method that is described in (72; 70; 35), which essentially includes boundary points into the discretization stencil and shortens the corresponding discretization arm length appropriately such that it ends on the interface Γ . For example, in the case below shown in Fig. 1.7, the spatial discretization on T incorporates the

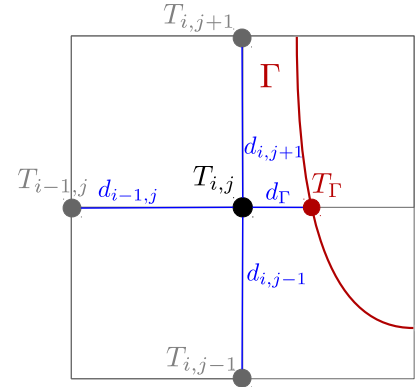


Figure 1.7: Illustration of discretization stencil used to apply Dirichlet boundary conditions at the interface.

interfacial condition $T|_{\Gamma}$, becoming

$$\begin{aligned} \nabla^2 T_{i,j}^{n+1} = & \left(\frac{T_{\Gamma}^{n+1} - T_{i,j}^{n+1}}{d_{\Gamma}} - \frac{T_{i,j}^{n+1} - T_{i-1,j}^{n+1}}{d_{i-1,j}} \right) \left(\frac{2}{d_{\Gamma} + d_{i-1,j}} \right) + \\ & \left(\frac{T_{i,j+1}^{n+1} - T_{i,j}^{n+1}}{d_{i,j+1}} - \frac{T_{i,j}^{n+1} - T_{i,j-1}^{n+1}}{d_{i,j-1}} \right) \left(\frac{2}{d_{i,j+1} + d_{i,j-1}} \right). \end{aligned}$$

The location of the interface is found by finding the zero of a quadratic interpolation of the level-set function in each spatial direction. This method has been proven to result in second-order accurate solutions and their gradients (73).

1.3.7 Solving the Navier-Stokes equations

For solving the incompressible Navier-Stokes equations given by equation (1.2)-(1.3), we take a projection method approach on a MAC sampling. The classical form of the projection method was first introduced by (74), and different variations are well-outlined for the interested reader in (75). In our case, a pressure-free method based on (76) and developed in a parallel adaptive grid environment by (54; 55) is used.

We write the time discrete form of the momentum equation (1.3), making use of the semi-Lagrangian method described in Sec. 1.3.5, giving

$$\alpha \frac{\mathbf{v}^{n+1} - \mathbf{v}_d^n}{\Delta t_{n+1}} + \beta \frac{\mathbf{v}_d^n - \mathbf{v}_d^{n-1}}{\Delta t_n} = \frac{1}{Re} \nabla^2 \mathbf{v}^{n+1} - \nabla P. \quad (1.11)$$

Making use of the Hodge decomposition, $\mathbf{v} = \mathbf{v}^* - \nabla \Phi$ into equation (1.11) yields

$$\frac{\alpha}{\Delta t_{n+1}} (\mathbf{v}^* - \nabla \Phi) + \mathbf{v}_d^n \left(\frac{\beta}{\Delta t_n} - \frac{\alpha}{\Delta t_{n+1}} \right) + \mathbf{v}_d^{n-1} \left(\frac{\beta}{\Delta t_n} \right) = \frac{1}{Re} \nabla^2 (\mathbf{v}^* - \nabla \Phi) - \nabla P.$$

The momentum equation can now be decoupled into two separate equations – a pressure-free momentum equation for the intermediate velocity field $\mathbf{v}^*$ given by equation (1.12a),

and a relation between the fluid pressure P and the Hodge variable Φ given by equation (1.12b).

$$\alpha \frac{\mathbf{v}^* - \mathbf{v}_d^n}{\Delta t_{n+1}} + \beta \frac{\mathbf{v}_d^n - \mathbf{v}_d^{n-1}}{\Delta t_n} = \frac{1}{Re} \nabla^2 \mathbf{v}^*, \quad (1.12a)$$

$$P^{n+1} = \frac{\alpha}{\Delta t_{n+1}} \Phi^{n+1} - \frac{1}{Re} \nabla^2 \Phi^{n+1}. \quad (1.12b)$$

Substituting the Hodge decomposition into the incompressibility condition then gives the equation that must be satisfied by Φ :

$$\nabla^2 \Phi^{n+1} = \nabla \cdot \mathbf{v}^*. \quad (1.13)$$

Lastly, conditions for $\mathbf{v}^*$ and Φ on a given boundary $\partial\Omega$ must be carefully selected to ensure compatibility with the physical conditions imposed on $\mathbf{v}$. First considering $\mathbf{v}^*$, we use the Hodge decomposition to write the Dirichlet case as $\mathbf{v}^*|_{\partial\Omega} = \mathbf{v}^{n+1}|_{BC} + (\nabla \Phi^{n+1})|_{\partial\Omega}$, and the Neumann case as $(\nabla \mathbf{v}^* \cdot \mathbf{n})|_{\partial\Omega} = (\nabla \mathbf{v}^{n+1} \cdot \mathbf{n})|_{BC} + (\nabla \nabla \Phi^n \cdot \mathbf{n})|_{\partial\Omega}$. Now, conditions on the Hodge variable Φ can be selected to best enforce the conditions on $\mathbf{v}$. For boundaries where a Dirichlet condition on $\mathbf{v}$ is applied, we write $(\nabla \Phi \cdot \mathbf{n})|_{\partial\Omega} = 0$. In the Neumann case, we note that because the calculation of the Hodge variable is only second-order accurate, the term $\nabla \nabla \Phi$ cannot be approximated accurately. Instead, we discard it and simply use $(\nabla \mathbf{v}^* \cdot \mathbf{n})|_{\partial\Omega} = (\nabla \mathbf{v}^{n+1} \cdot \mathbf{n})|_{BC}$ as an approximation. This selection enforces equality between $\mathbf{v}$ and $\mathbf{v}^*$ in the normal direction; however, it does not enforce equality in the tangential direction and thus introduction of spurious slip is possible. It is also of note that the conditions imposed on $\mathbf{v}^*$ require the application of a boundary condition on the Hodge variable at time level $n+1$, which is unknown. Instead, we use Φ^n as an approximation, and iterate on the solution of $\mathbf{v}^*$ and Φ until the gradient of the computed Hodge variable is no longer changing within a specified tolerance. In practice, this typically serves as a relatively good approximation and requires very few

iterations, as was demonstrated in (54). The general procedure for solving for the fluid is as follows:

1. Solve equation (1.12a) for the intermediate velocity field $\mathbf{v}^*$,
2. Solve equation (1.13) for the Hodge variable Φ^{n+1} that will be used to enforce incompressibility,
3. Project the intermediate velocity field onto the divergence free space: $\mathbf{v}^{n+1} = \mathbf{v}^* - \nabla \Phi^{n+1}$,
4. Recover the pressure P^{n+1} via equation (1.12b),

where steps 1-2 are iterated until convergence of $\nabla \Phi$ within a given tolerance. The procedure given above relies on the spatial discretizations of $\frac{1}{Re} \nabla^2 \mathbf{v}^*$ for the face-sampled intermediate velocity field $\mathbf{v}^*$ in Step 1, and of $\nabla^2 \Phi$ for the cell-sampled Hodge variable Φ and $\nabla \cdot \mathbf{v}^*$ for the face-sampled intermediate velocity in Step 2, which are described in detail in the following sections.

Discretization of the viscous term

The viscous term of (1.12a), given by $\frac{1}{Re} \nabla^2 \mathbf{v}^*$, is discretized using an implicit finite volume approach which makes use of control volumes constructed from Voronoi cells. We take advantage of the fact that for a fluid with uniform viscosity, the components of the velocity field are decoupled and therefore can be solved for separately.

The edges of the Voronoi cells which join two data points are constructed from the bisector line of the segment which connects the two data points. This construction is convenient in part because it yields a flux that is orthogonal the edge connecting the two points. Additionally, the cells are created such that the edges of the control volume next to the interface lie on the interface. An example of such a cell is shown in Fig. 1.8.

Let us consider the point $\mathbf{r}_0$, the location of a face center about which we wish to build the discretization for the corresponding velocity component of $\mathbf{v}^*$. We construct the Voronoi cell C about the point $\mathbf{r}_0$ by selecting j neighboring points $\mathbf{r}_j$. The distance d_j denotes the distance between points $\mathbf{r}_0$ and $\mathbf{r}_j$, and s_j is the length of the edge created by bisecting d_j . An illustration of this construction is given in Fig. 1.8.

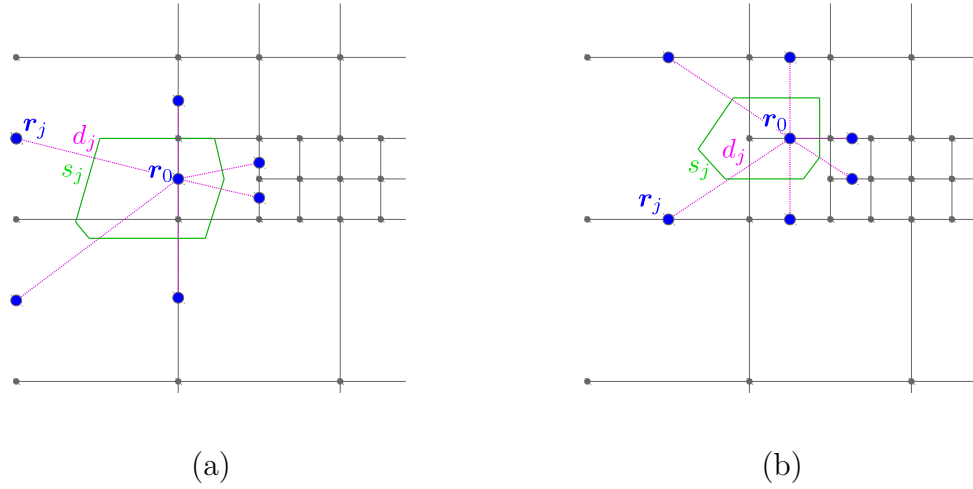


Figure 1.8: Example illustrations of Voronoi cell constructions for face-sampled velocity discretization. (a) A Voronoi cell to discretize x-component of velocity about point $\mathbf{r}_0$. (b) A Voronoi cell to discretize y-component of velocity about point $\mathbf{r}_0$.

Suppose that we are writing the discretization for the x -component of $\mathbf{v}^*(\mathbf{r}_0) = (u^*, v^*)$, then the momentum equation using the semi-Lagrangian approximation for the advection part gives:

$$\int_C \alpha \frac{u^* - u_d^n}{\Delta t_{n+1}} + \beta \frac{u_d^n - u_d^{n-1}}{\Delta t_n} dV = \int_C \frac{1}{\text{Re}} \nabla^2 u^* dV = \frac{1}{\text{Re}} \int_{\partial C} \nabla u^* \cdot \mathbf{n} dS$$

The right-hand side can then be discretized using the Voronoi cell construction, yielding

the final form

$$\text{Vol}(C) \left(\alpha \frac{u^* - u_d^n}{\Delta t_{n+1}} + \beta \frac{u_d^n - u_d^{n-1}}{\Delta t_n} \right) = \frac{1}{Re} \sum_{j \in \text{Voro}(\mathbf{r}_0)} \frac{s_j(u^*(\mathbf{r}_j) - u^*(\mathbf{r}_0))}{d_j},$$

where $\text{Vol}(C)$ refers to the volume of the current Voronoi cell C and $\text{Voro}(\mathbf{r}_0)$ is the set of points that make up the given Voronoi cell.

Neumann boundary conditions on $\mathbf{v}$ are applied by incorporating $\nabla \mathbf{v} \cdot \mathbf{n}|_\Gamma$ naturally into the flux discretization on the edges of cells which are in contact with the boundary. To apply the Dirichlet interface condition on fluid velocity, which relates the fluid velocity to the velocity of the evolving interface given by equation (1.8), a finite-difference approach utilizing discretizations of the form discussed in Sec. 1.3.3 is used. A uniform grid is enforced in a band around the interface, and in the presence of the interface, the discretization collapses to a uniform finite difference discretization which takes into account the boundary condition value via the Shortley-Weller method (72; 70).

Discretizations for projection step

The discretization of equation (1.13) satisfied by the Hodge variable Φ follows the approach introduced in (77), which is a second-order accurate extension of the discretization of (78).

To solve the projection step (1.13), we use a cell-based finite-volume approach with a second order accurate discretization of Cartesian derivatives at the cell's faces. The discretization considers the integration form of equation (1.13) over a finite volume cell C :

$$\int_{\partial C} \nabla \Phi \cdot \mathbf{n}_f \, dS = \int_C \nabla \cdot \mathbf{v}^* \, dV, \quad (1.14)$$

where $\mathbf{n}_f$ is the outward normal to face f of the finite volume cell C . The idea in (77) is to

define the flux on the face of a cell to be that of the largest cell neighboring that face. For example, let us consider a finite volume cell C and neighboring faces f_j , where the Hodge variable is stored at the cell centers. An illustration of such a cell, with corresponding notations, is displayed below in Fig. 1.9 (a) - (b).

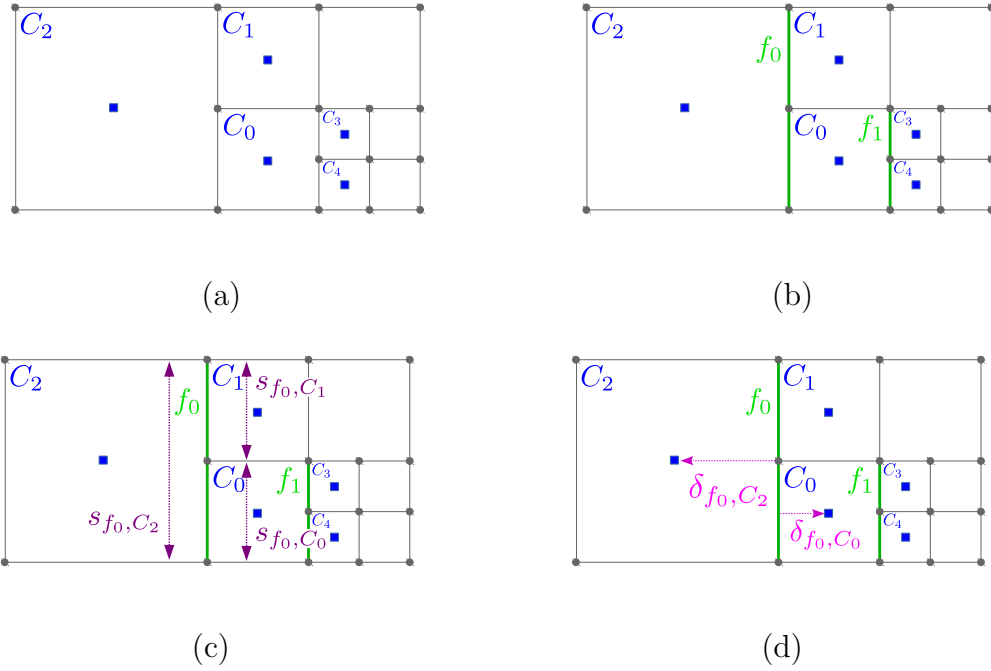


Figure 1.9: Example illustration of finite volume discretization notations for the Hodge variable. (a) Example labeled cells, with cell centers shown as blue squares, (b) with labeled faces displayed, (c) with edge lengths between given faces and cells displayed, (d) with signed distances between faces and cell centers displayed.

For a given face f at which we wish to discretize the flux of Φ , we define C_* as the largest direct neighbor cell of the face f and $N_C(f)$ as the set of cells in contact with C_* and f . We can then define the distances s_{C_i, C_k} as the edge length connecting two cells C_i and C_k , and δ_{f, C_i} as the length from face f to the center of the cell C_i . We can then

write the average discretization distance for discretizing $\nabla\Phi \cdot \mathbf{n}_f$ across f as

$$\Delta = \sum_{C_i \in N_C(f)} \frac{s_{C_i, C_*}}{s_{C_*, C_*}} (\delta_{f, C_*} - \delta_{f, C_i}).$$

The discretization for the flux of Φ is then given by

$$\nabla\Phi \cdot \mathbf{n}_f = \sum_{C_i \in N_C(f)} \left(\frac{s_{C_i, C_*}}{s_{C_*, C_*}} \right) \left(\frac{\Phi_i - \Phi_*}{\Delta} \right)$$

For a given cell C for which we wish to discretize the divergence of fluid velocity at the cell center, we additionally define the lengths $s_{f_j, C}$ as the length of the edge connecting cell C and a given face f_j . We denote $N_f(C)$ as the set of all faces in contact with cell C .

The average size of a face f_j is defined as

$$A_{f_j} = \frac{1}{2} \sum_{C_i \in N_C(f_j)} s_{C_i, C_*},$$

which is analogous with the length s_{C_*, C_*} as defined previously for a given face with corresponding cell C_* .

The right-hand side of (1.14), aka the divergence of the intermediate velocity field, can then be discretized for a given cell C as

$$\nabla \cdot \mathbf{v}^*|_C = \frac{1}{\Delta_C} \sum_{f_j \in N_f(C)} \left(\frac{\delta_{f_j, C}}{|\delta_{f_j, C}|} \right) \left(\frac{s_{f_j, C} \mathbf{v}_{f_j}^*}{A_{f_j}} \right),$$

where Δ_C is the size of cell C .

An example of notations for the x -direction discretizations is shown above in Fig. 1.9(a)-(d). For example, when considering the flux across f_0 , we have $C_* = C_2$, $N_C(f_0) = \{C_0, C_1\}$, and the corresponding edge lengths and signed distances shown in Fig. 1.9 (c) -

(d). Considering the divergence of the intermediate velocity discretized for C_0 , we would have $N_f(C_0) = \{f_0, f_1\}$.

Neumann boundary conditions are applied by including the boundary condition flux term naturally into the finite volume discretization. Dirichlet boundary conditions are again applied using a similar finite difference approach as (72; 70), enforcing a uniform grid near the interface.

A note on the boundary conditions

It is worth noting that because we solve (1.12a) in a component-wise fashion, we must correspondingly prescribe the boundary condition on fluid velocity given by (1.8) in a component-wise manner. As such, we prescribe the conditions

$$u|_{\Gamma} = (\mathbf{V}_{\Gamma} \cdot \mathbf{n}) \left(1 - \frac{\rho_s}{\rho_l}\right) n_x, \quad (1.15)$$

$$v|_{\Gamma} = (\mathbf{V}_{\Gamma} \cdot \mathbf{n}) \left(1 - \frac{\rho_s}{\rho_l}\right) n_y, \quad (1.16)$$

where u is the x -component of the fluid velocity sampled on the vertical face centers, v is the y -component of the fluid velocity sampled on the horizontal face centers, and n_x and n_y are the x and y -components of the unit normal vector $\mathbf{n}$, respectively. The conditions (1.15) and (1.16) satisfy equality when substituted back into (1.8) and serve to approximate the condition. We additionally recall that the condition which is actually imposed is on the intermediate velocity field $\mathbf{v}^*$ rather than $\mathbf{v}^{n+1}$, and so the form of the boundary condition is modified in order to account for the Hodge decomposition.

Therefore, the conditions applied to the intermediate velocity field may be expressed as

$$v_x^*|_{\Gamma} = (\mathbf{V}_{\Gamma} \cdot \mathbf{n}) \left(1 - \frac{\rho_s}{\rho_l} \right) n_x + \frac{\partial \Phi}{\partial x}|_{\Gamma}, \quad (1.17)$$

$$v_y^*|_{\Gamma} = (\mathbf{V}_{\Gamma} \cdot \mathbf{n}) \left(1 - \frac{\rho_s}{\rho_l} \right) n_y + \frac{\partial \Phi}{\partial y}|_{\Gamma}, \quad (1.18)$$

where Φ is the Hodge variable.

The interfacial velocity $\mathbf{V}_{\Gamma}$, unit normals $\mathbf{n} = (n_x \ n_y)^T$, and scalar normal velocity $\mathbf{V}_{\Gamma} \cdot \mathbf{n}$ are stored at the grid nodes. The unit normals are calculated using the level-set function as described in Sec. 1.3.2, using the node-based spatial discretizations described in Sec. 1.3.3 to calculate the gradient of the level-set function. The gradient of the Hodge variable $\nabla \Phi = \left(\frac{\partial \Phi}{\partial x} \ \frac{\partial \Phi}{\partial y} \right)^T$ is discretized at the face centers using the approach described in Sec. 1.3.7.

To apply the boundary conditions, we utilize the Shortley-Weller approach (72; 70) similar to that described previously in Sec. 1.3.6. For a given face whose neighbor lies across the interface, the boundary condition value is incorporated into the discretization instead, and the corresponding discretization distance is shortened such that reflects the distance to the interface. In this case, the boundary condition value is computed as given by either (1.17) or (1.18), where the values of $\mathbf{V}_{\Gamma} \cdot \mathbf{n}$, n_x , n_y , $\partial \Phi / \partial x$, and $\partial \Phi / \partial y$ are all bilinearly interpolated to the relevant interface location required by the discretization. Illustrations of the data layout and examples of Voronoi cell discretizations close to the interface are depicted in Fig. 1.10.

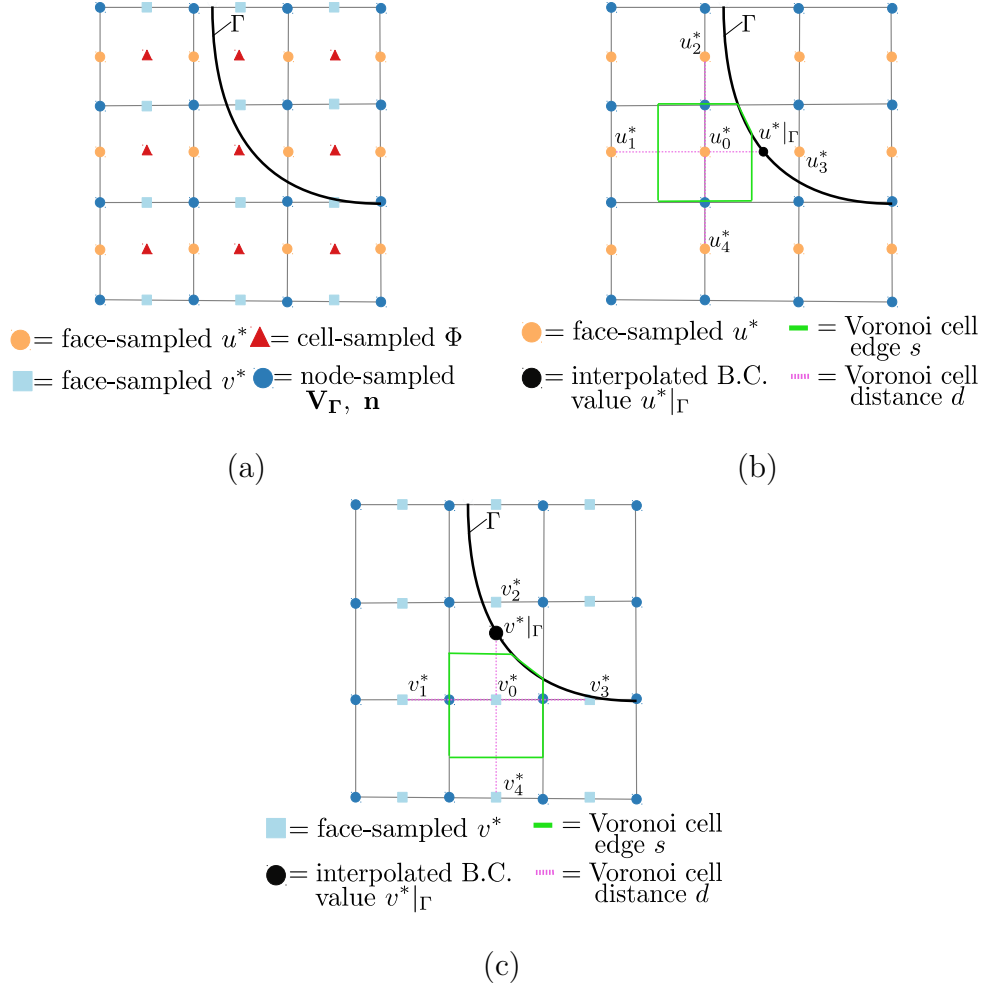


Figure 1.10: Illustrations of the application of component-wise interfacial boundary conditions on the fluid velocity given by eqs. (1.15) - (1.16). (a) Data layout for relevant fields that enter the boundary condition. The interfacial velocity $\mathbf{V}_\Gamma$ and unit normals $\mathbf{n}$ are sampled at the nodes, and interpolated bilinearly from the nodes to relevant locations for calculating the boundary condition values u_Γ and v_Γ as depicted in (b, c). (b,c) Examples of a Voronoi cells for near-interface discretizations of the (b) the x -component of the fluid velocity, u , at the vertical face centers and (c) the y -component of the fluid velocity, v , at the horizontal face centers.

1.3.8 Extension of fields

It is important to define valid values for the solutions on each side of the interface in order to perform discretizations that avoid discontinuities in the solution, or simply to define values that do not exist. This is the case for example when defining valid values for the solid after we advance the interface from t^n to t^{n+1} or when discretizing an equation at a node that belongs to the liquid region but which requires neighboring values that are not necessarily in the liquid region. We make use of different extrapolation schemes depending on where the data is stored.

PDE-based extrapolation of node-sampled fields

For the extension of node-sampled fields across the interface, we use the PDE-based quadratic extrapolation approach of (79), adapted from (80). Considering a scalar field q that we want to extend from the region defined by $\phi \leq 0$ to the region given by $\phi > 0$, we first define the unit normal, $\mathbf{n}$, pointing outwards from the domain $\phi \leq 0$, and compute $q_{\mathbf{nn}} = \nabla(\nabla q \cdot \mathbf{n}) \cdot \mathbf{n}$ and $q_{\mathbf{n}} = \nabla q \cdot \mathbf{n}$ with central differencing given in section 1.3.4. Then, we successively solve the following set of partial differential equations for about 20 iterations each, in fictitious time τ :

$$\begin{aligned} \frac{\partial q_{\mathbf{nn}}}{\partial \tau} + H(\phi)(\mathbf{n} \cdot \nabla q_{\mathbf{nn}}) &= 0, \\ \frac{\partial q_{\mathbf{n}}}{\partial \tau} + H(\phi)(\mathbf{n} \cdot \nabla q_{\mathbf{n}} - q_{\mathbf{nn}}) &= 0, \\ \frac{\partial q}{\partial \tau} + H(\phi)(\mathbf{n} \cdot \nabla q - q_{\mathbf{n}}) &= 0, \end{aligned}$$

where the Heaviside $H(\phi)$ is defined as:

$$H(\phi) = \begin{cases} 0, & \text{if } \phi \leq 0 \\ 1, & \text{if } \phi > 0 \end{cases}.$$

Geometric extrapolation of face and cell-sampled fields

For extension of face and cell-sampled fields, the data layout of the face and cell centers present a challenge in implementation compared with node-sampled fields, and thus a geometric quadratic least-squares extrapolation approach is used (54). Again, we consider a scalar field q which we want to extend from the region $\phi \leq 0$ to the region $\phi > 0$. Let us first define a point $\mathbf{r}_+$ in the positive subdomain, as shown in Fig. 1.11, at which we want an extended value of q , denoted $q(\mathbf{r}_+)$. This time, we define $\mathbf{n}_+$ as the unit normal pointing outward from the positive subdomain defined by $\phi > 0$. To evaluate q at this point, we build a second degree Newton polynomial along the normal direction using the interface boundary condition and two values from the known region $\phi \leq 0$.

1.3.9 Interpolation methods

Interpolation procedures depend on the location where the data is stored for the given field of interest. For all node-sampled fields except fluid velocity components, a quadratic non-oscillatory method is used, whereas a standard quadratic approach is used for fluid velocity components, and a geometric weighted least-squares method is used for cell and face - sampled fields.

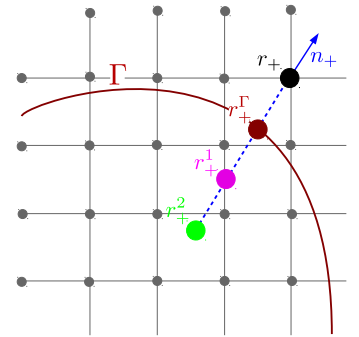


Figure 1.11: Illustration of points defined for geometric extrapolation procedure.

Quadratic non-oscillatory interpolation of node-sampled fields

To perform the interpolation of node-sampled fields we utilize a quadratic non-oscillatory method, which defines the interpolant using a standard bi-linear interpolation formula with added second-derivative correction terms. The second-derivatives used to apply the correction are selected using the `minmod` slope limiter, as is done in (67), in order to achieve better stability in regions where the field may have sharp changes. However, we make a slight modification to the definition of the non-oscillatory second derivatives - rather than apply the `minmod` slope limiter to the second derivatives defined at all the vertices of the cell, we first interpolate the second derivatives along the faces of the cell, and then apply the `minmod` slope limiter. This modification ensures continuity of the interpolant across cells in uniform regions.

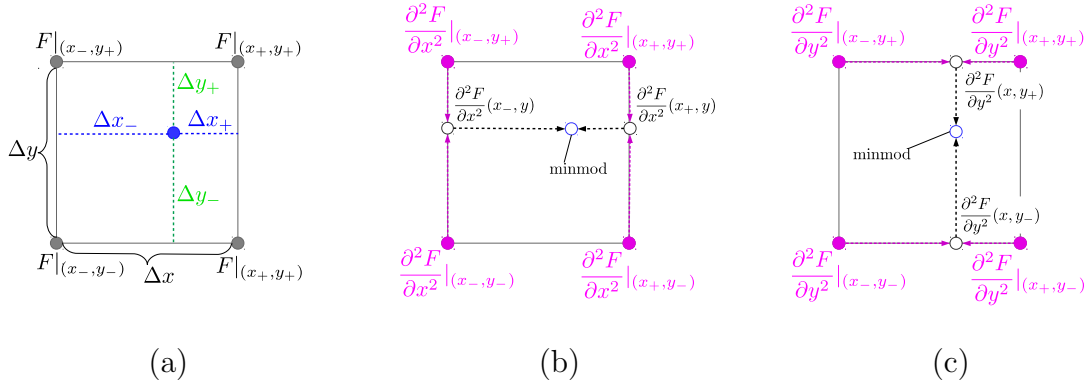


Figure 1.12: Illustration of quadratic non-oscillatory interpolation process with (a) grid cell with interpolation notations, (b) non-oscillatory approximation of the second-order derivative in x , (c) non-oscillatory approximation of the second-order derivative in y .

Referring to Fig. 1.12, the interpolation of a scalar quantity $q(\mathbf{r})$ to a location $\mathbf{r} =$

(x, y) takes the form

$$q(\mathbf{r}) = q(x_-, y_-)w_{--} + q(x_+, y_-)w_{+-} + q(x_-, y_+)w_{-+} + q(x_+, y_+)w_{++} - \\ \left(\frac{1}{2}\Delta x_+ \Delta x_-\right)q_{xx}|_{(\text{NO})} - \left(\frac{1}{2}\Delta y_+ \Delta y_-\right)q_{yy}|_{(\text{NO})},$$

where $q_{xx}|_{(\text{NO})}$ and $q_{yy}|_{(\text{NO})}$ are the selected “non-oscillatory” second-order derivatives of the function q , and where the standard bilinear interpolation weights are given by

$$w_{--} = \left(\frac{\Delta x_+}{\Delta x}\right) \left(\frac{\Delta y_+}{\Delta y}\right), \quad w_{++} = \left(\frac{\Delta x_-}{\Delta x}\right) \left(\frac{\Delta y_-}{\Delta y}\right), \\ w_{-+} = \left(\frac{\Delta x_+}{\Delta x}\right) \left(\frac{\Delta y_-}{\Delta y}\right), \quad w_{+-} = \left(\frac{\Delta x_-}{\Delta x}\right) \left(\frac{\Delta y_+}{\Delta y}\right).$$

To select the non-oscillatory second-order derivative values, we first interpolate the derivatives to locations on the grid cell faces that correspond to the point of interest, and then applying the `minmod` limiter. This is expressed as

$$q_{xx}|_{NO} = \text{minmod}(q_{xx}(x_-, y), q_{xx}(x_+, y)), \\ q_{yy}|_{NO} = \text{minmod}(q_{yy}(x, y_-), q_{yy}(x, y_+)),$$

where the second-order derivative values are interpolated along the cell faces as

$$q_{xx}(x_-, y) = \frac{\Delta y_+}{\Delta y}(q_{xx})|_{(x_-, y_-)} + \frac{\Delta y_-}{\Delta y}(q_{xx})|_{(x_-, y_+)}, \\ q_{xx}(x_+, y) = \frac{\Delta y_+}{\Delta y}(q_{xx})|_{(x_+, y_-)} + \frac{\Delta y_-}{\Delta y}(q_{xx})|_{(x_+, y_+)}, \\ q_{yy}(x, y_-) = \frac{\Delta x_+}{\Delta x}(q_{yy})|_{(x_-, y_-)} + \frac{\Delta x_-}{\Delta x}(q_{yy})|_{(x_+, y_-)}, \\ q_{yy}(x, y_+) = \frac{\Delta x_+}{\Delta x}(q_{yy})|_{(x_-, y_+)} + \frac{\Delta x_-}{\Delta x}(q_{yy})|_{(x_+, y_+)},$$

and the `minmod` slope limiter is given by

$$\text{minmod}(a, b) = \begin{cases} a, & |a| < |b| \text{ and } ab > 0 \\ b, & |a| > |b| \text{ and } ab > 0 \\ 0, & ab < 0 \end{cases}.$$

Quadratic weighted least-squares interpolation of face and cell-sampled fields

In the case of the Navier-Stokes fields, it may not always be possible to define a square using only face and cell centers on an adaptive grid, as it is in the case of node grid points. Therefore, it may not always be possible to write second-order accurate approximations of face and cell-sampled values, as is done for node-sampled ones. Due to this challenge in data layout, we utilize a weighted least-squares approach to interpolation of face and cell-sampled fields, as is done in (54) and summarized below.

Considering a field q sampled at the cells or the faces of the computational grid, we interpolate it at location $\mathbf{r}_I = (x_I, y_I)$ by constructing a quadratic interpolant polynomial around $\mathbf{r}_I$. The procedure fetches the cell C containing the point of interest $\mathbf{r}_I$ and all its second-degree neighbors (*i.e.*, neighbor cells of neighbor cells). n relevant data samples $(\mathbf{x}_i, q_i)$, $i \in \{0, 1, \dots, n-1\}$ are accumulated in this manner by parsing this set of neighbor cells (or their appropriate faces, if interpolating face-sampled fields). The interpolation error e_i on the i^{th} sample then reads

$$e_i = \beta_0 + \beta_x (x_i - x_I) + \beta_y (y_i - y_I) + \beta_{xx} (x_i - x_I)^2 + \beta_{xy} (x_i - x_I) (y_i - y_I) + \beta_{yy} (y_i - y_I)^2 - q_i,$$

where β_0 , β_x , β_y , β_{xx} , β_{yy} , β_{xy} are polynomial coefficients to be determined. Using matrix formalism, one has

$$\mathbf{e} = \underline{\underline{\mathbf{X}}} \boldsymbol{\beta} - \mathbf{q}$$

where $\mathbf{e} = [e_0 \ e_1 \ \dots \ e_{n-1}]^T$, $\mathbf{q} = [q_0 \ q_1 \ \dots \ q_{n-1}]^T$, $\boldsymbol{\beta} = [\beta_0 \ \beta_x \ \beta_y \ \beta_{xx} \ \beta_{xy} \ \beta_{yy}]^T$ and

$$\underline{\underline{\mathbf{X}}} = \begin{pmatrix} 1 & (x_0 - x_I) & (y_0 - y_I) & (x_0 - x_I)^2 & (x_0 - x_I)(y_0 - y_I) & (y_0 - y_I)^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & (x_{n-1} - x_I) & (y_{n-1} - y_I) & (x_{n-1} - x_I)^2 & (x_{n-1} - x_I)(y_{n-1} - y_I) & (y_{n-1} - y_I)^2 \end{pmatrix}.$$

The weighted least-square interpolation method intends to find the interpolation coefficients $\boldsymbol{\beta}$ that minimize $\|\underline{\underline{\mathbf{W}}}\mathbf{e}\|_2$ where the weight matrix

$$\underline{\underline{\mathbf{W}}} = \text{diag} \left([\|\mathbf{x}_0 - \mathbf{x}_I\|^{-1} \ \|\mathbf{x}_1 - \mathbf{x}_I\|^{-1} \ \dots \ \|\mathbf{x}_{n-1} - \mathbf{x}_I\|^{-1}] \right)$$

attaches weight factors to interpolation errors: interpolation errors are given a weight inversely proportional to their distance to $\mathbf{x}_I$. The solution of this minimization problem is

$$\hat{\boldsymbol{\beta}}_{WLS} = (\underline{\underline{\mathbf{X}}}^T \underline{\underline{\mathbf{W}}}^2 \underline{\underline{\mathbf{X}}})^{-1} \underline{\underline{\mathbf{X}}}^T \underline{\underline{\mathbf{W}}}^2 \mathbf{q}$$

and the first component of $\hat{\boldsymbol{\beta}}_{WLS}$, i.e., $\hat{\beta}_{WLS,0}$, is a second-order accurate interpolation of q at the desired point $\mathbf{r}_I$.

For each interpolation, a 6×6 symmetric positive definite system must be inverted in two dimensions, which we do using a Cholesky decomposition. While this approach parallelizes easily, it is relatively costly and is used sparingly. For example, velocity field values are interpolated from the faces to the nodes once at the end of each time step, so that the semi-Lagrangian backtrace calculations and their corresponding interpolations can be performed using node-sampled values. By doing so, we perform only one weighted least-squares interpolation per grid point, rather than multiple (as would be required for the backtrace calculations).

1.4 Convergence test

For the given system of PDEs, there is no known analytical solution. Therefore, we investigate the convergence behavior of the solver by considering a numerical verification test for which we select analytical solutions for T_l , T_s , $\mathbf{v}$, P , and $\mathbf{V}_\Gamma$, and augment the system of equations with the appropriate synthetic forcing terms to produce such solutions. We note that for the verification test, all non-dimensional groups and material parameters are set equal to 1. Convergence behavior is then evaluated for these fields based on time-averaged L_∞ norm of the spatial error for each field. To test the convergence of the level set function ϕ , we select the interfacial velocity field such that it deforms the interface sufficiently, and then switch the sign of the velocity field at $t = t_{\text{final}}/2$, so that ϕ should recover its initial state at $t = t_{\text{final}}$ (as is shown in Fig. 1.13). The error in ϕ is thus evaluated by the difference between ϕ_{initial} and ϕ_{final} .

We consider the domain $\Omega = [-\pi, \pi] \times [-\pi, \pi]$, which is initially split into the subdomains Ω_l , Ω_s by the level-set function ϕ such that

$$\left\{ \begin{array}{l} \phi = R - \sqrt{x^2 + y^2}, \\ R = \frac{\pi}{2}, \\ \Omega_l = (x, y) \in \Omega | \phi < 0, \\ \Omega_s = (x, y) \in \Omega | \phi > 0, \\ \Gamma = (x, y) \in \Omega | \phi = 0, \end{array} \right.$$

We select the following solutions for fluid velocity and pressure

$$v_x = \cos t \sin x \cos y,$$

$$v_y = -\cos t \cos x \sin y,$$

$$P = 0,$$

and temperature fields

$$T_l = \sin x \sin y (x + \cos x \cos y \cos t),$$

$$T_s = \cos x \cos y (\cos t \sin x \sin y - 1),$$

which produce an interfacial velocity of

$$\mathbf{V}_\Gamma \cdot \mathbf{n} = (V_{\Gamma,x} n_x + V_{\Gamma,y} n_y),$$

where $V_{\Gamma,x}$ and $V_{\Gamma,y}$ are given by

$$V_{\Gamma,x} = \cos y \sin x - \sin x \sin y - x \cos x \sin y,$$

$$V_{\Gamma,y} = \cos x \sin y - x \cos y \sin x.$$

Simulations are run for $t_0 = 0$ to $t_f = \pi/3$ at 6 different grid level configurations. A test for grid level 5/7 takes 94 seconds on a Dell 7810 tower with 6 3.60 GHz processes. Convergence results are shown in Fig. 1.14 and Table 1.1.

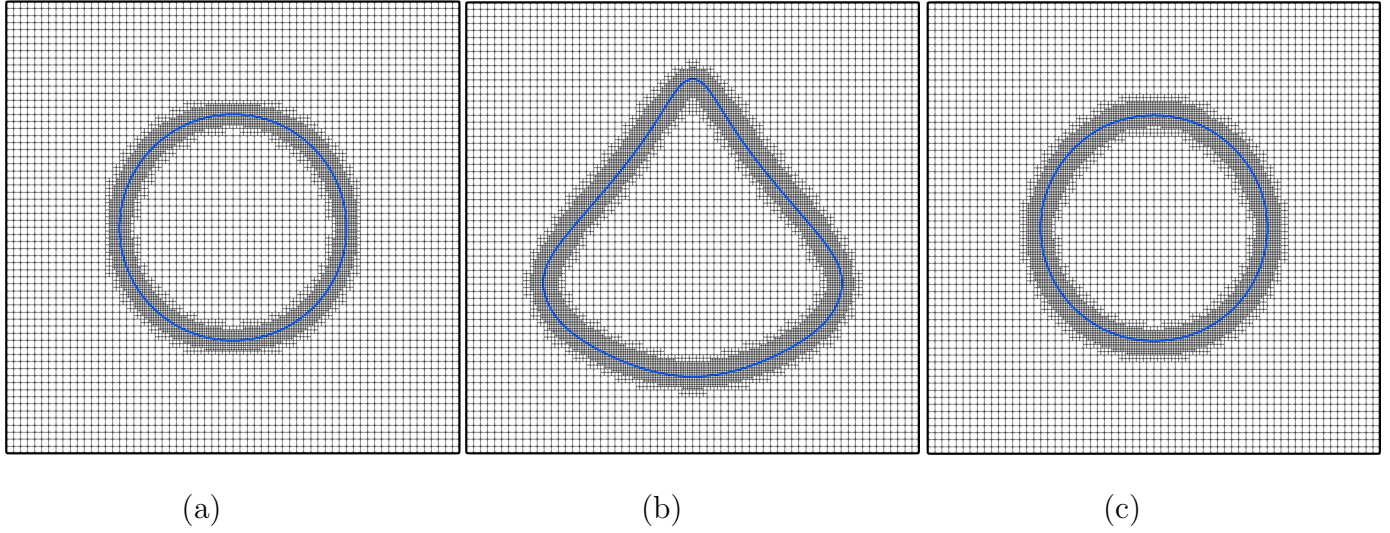


Figure 1.13: Time evolution of ϕ under verification test, (a) at initial state, (b) at half-way time, (c) at final time, for 5/7 grid.

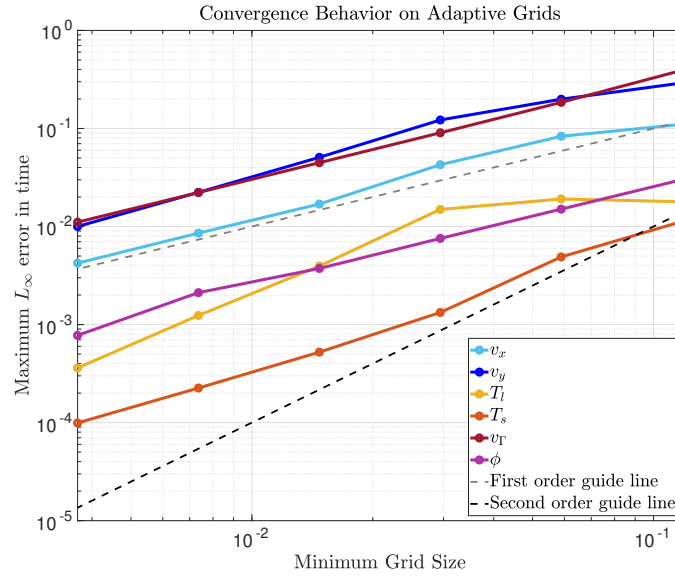


Figure 1.14: Convergence results for numerical verification test.

Convergence rates between first and second order are achieved for v_x , v_y , T_l , and T_s , which is reasonable based on the combination of numerical methods utilized. As the

computation of $\mathbf{V}_\Gamma$ depends on the gradient of the scalar fields T_l , T_s , we expect $\mathbf{V}_\Gamma \cdot \mathbf{n}$ and therefore the level-set function ϕ defining the interface to converge at approximately first order, which they do.

$l_{\min}/l_{\max}$	N	v_x	v_y	T_l	T_s	$\mathbf{V}_\Gamma \cdot \mathbf{n}$	ϕ
3/5	59	1.14×10^{-1}	3.59×10^{-1}	1.72×10^{-2}	1.03×10^{-2}	3.92×10^{-1}	2.61×10^{-2}
4/6	96	9.05×10^{-2}	2.54×10^{-1}	1.90×10^{-2}	4.07×10^{-3}	1.89×10^{-1}	1.05×10^{-2}
5/7	125	4.55×10^{-2}	1.24×10^{-1}	1.48×10^{-2}	8.83×10^{-4}	8.97×10^{-2}	5.01×10^{-3}
6/8	181	1.70×10^{-2}	5.17×10^{-2}	3.92×10^{-3}	1.95×10^{-4}	4.48×10^{-2}	2.25×10^{-3}
7/9	313	8.56×10^{-3}	2.23×10^{-2}	1.23×10^{-3}	5.46×10^{-5}	2.22×10^{-2}	1.19×10^{-3}
8/10	572	4.23×10^{-3}	1.00×10^{-2}	3.59×10^{-4}	1.31×10^{-5}	1.11×10^{-2}	7.74×10^{-4}
Convergence rate	-	1.01	1.07	1.19	1.97	1.03	1.03

Table 1.1: Convergence results for verification test. Displayed is maximum L_∞ error in time for each field at various grid levels, and corresponding convergence rates. N is the average number of nodes per spatial direction.

1.5 Ice growth on a cooled cylinder in crossflow – Quantitative benchmark

Our solver is validated using the configuration of ice growth on a cooled cylinder in a crossflow, as studied experimentally by Okada *et al.*(81) and Cheng *et al.*(82) by comparing the time evolution of interface shape with experimental results. Additionally, we compare the local heat transfer behavior at the forward stagnation point with an empirical correlation developed by Perkins and Leppert (83). The physical system for the problem is shown below in Fig. 1.15. We consider the problem divided into three subdomains: (i) the liquid water Ω_l , (ii) the solid ice Ω_s , and (iii) the cooled cylinder Ω_{cyl}

for which T_{cyl} is assumed constant. The physical properties of ice and water are chosen to match those of (81) to yield the ratios of physical quantities and non-dimensional numbers as given by Table 1.2. The characteristic temperature difference for this problem is given by $\Delta T = T_{\infty} - T_{\text{cyl}}$, where T_{∞} is the freestream fluid temperature.

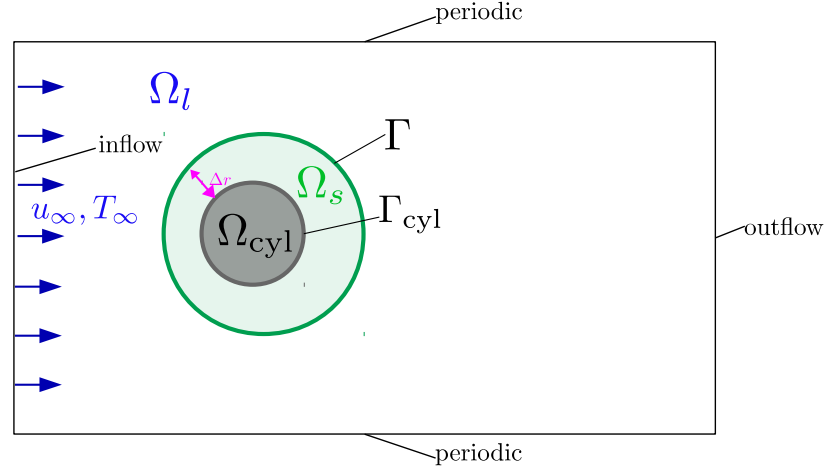


Figure 1.15: Illustration of problem setup for ice growth on cooled cylinder in a crossflow (not drawn to scale)

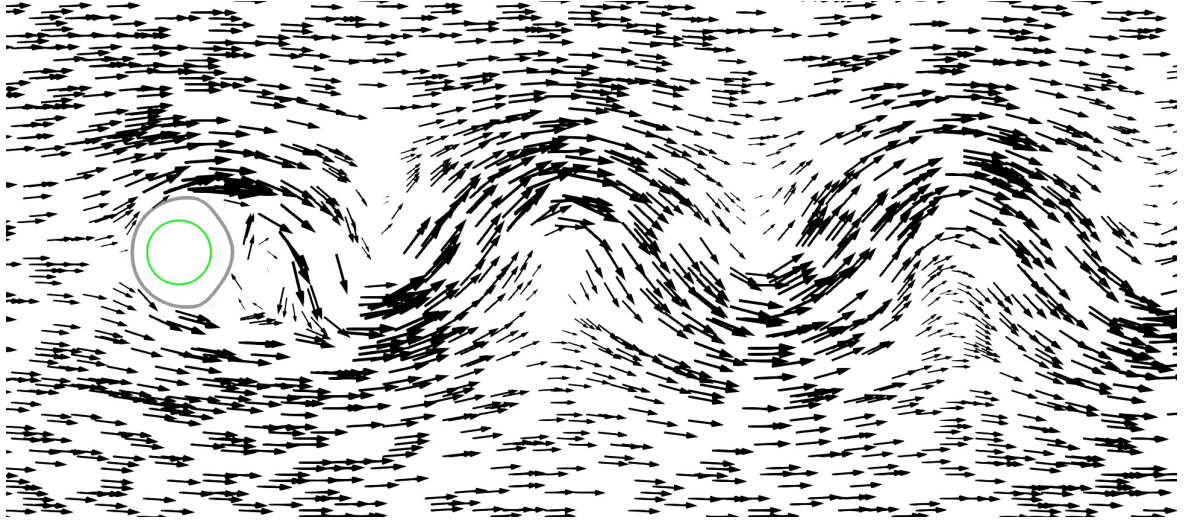
Prandtl number, Pr	13
Ratio of densities, ρ_l/ρ_s	1.09
Ratio of thermal diffusivities, α_l/α_s	0.11
Ratio of surface tension and characteristic length, σ/d	1.2×10^{-8}
Ratio of thermal conductivities, k_l/k_s	0.25
Freestream fluid temperature, T_∞	$2.5^\circ C$
Cylinder diameter, d	35 mm

Table 1.2: Parameters used in the ice growth on cooled cylinder problem, chosen to match conditions from (81).

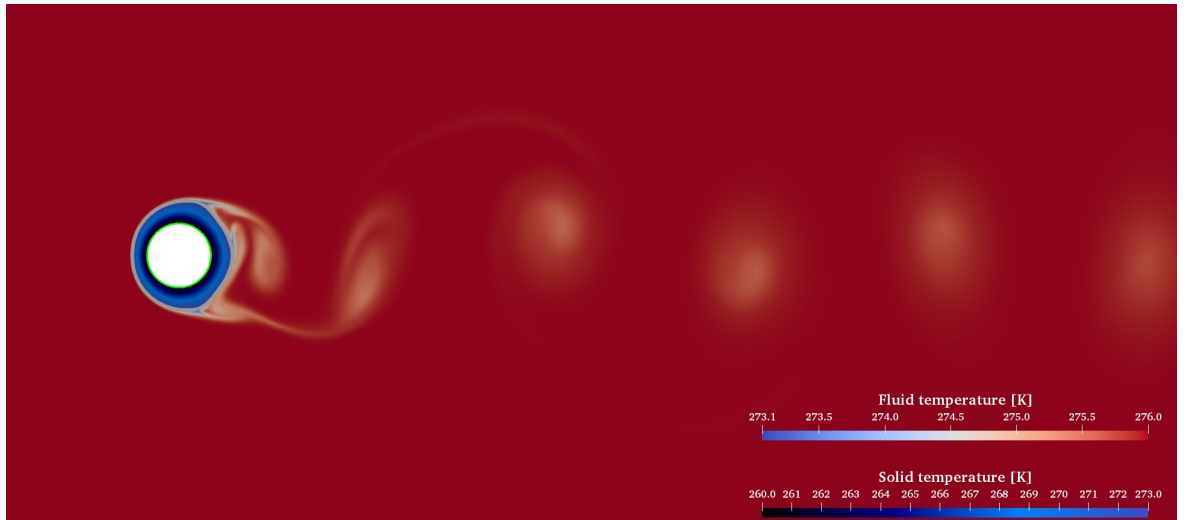
For ice shape comparison, the simulations are run for the cases (i) $Re = 201$, $St = 0.069$ ($T_{cyl} = -2.5^\circ C$) and (ii) $Re = 506$, $St = 0.081$ ($T_{cyl} = -7.5^\circ C$) in order to match the experimental conditions given by Okada *et al.*(81). In the regime we are considering, the momentum and thermal boundary layers δ_m and δ_T are such that $\delta_m \sim \delta_T$. Thus, the maximum level of refinement is chosen such that there are a minimum of 10 – 15 grid cells through the boundary layer thickness for the $Re = 201$ case, and 6 – 12 for the $Re = 506$ case. Since $\delta_T \sim \delta_m \sim (r_{cyl} + r_{ice})/\sqrt{Re_d}$, the boundary layer thickness grows as the thickness of the ice increases, and the thinnest boundary layer we encounter is at the beginning of the simulations. In order to provide an initial ice domain for the solver, the radius of the ice layer is initialized as a cylinder with $r_{ice} = 1.10 r_{cyl}$. The initial size is chosen to be as small as possible for the grid resolution used, whilst still resolving the ice layer by a minimum of 6 grid cells in order to ensure that the initial ice domain is well-defined. This corresponds to an initial ice thickness of ~ 1 mm.

Examples of velocity vectors and temperature fields for a snapshot in time are dis-

played in Fig. 1.16(a) and Fig. 1.16(b), respectively. We recapture the von-Karman vortex street in the flow field, and pockets of cooler fluid temperature can be seen in the wake behind the ice. The ice-water interface evolves to a non-uniform shape, due to the complex flow profile and resulting heat transport around the ice. The details of this shape evolution will be further discussed in Sec. 1.6.



(a)



(b)

Figure 1.16: (a) Velocity vectors and (b) temperature fields for the simulation of ice growth on cooled cylinder at $Re = 506$ and $St = 0.081$, with the cylinder interface plotted in green and the ice interface plotted in gray.

1.5.1 Ice Shape

The ice interface at different times is plotted against the results of (81) in Figs. 1.17(a)-(d). Note that in the plotted results, the angle over the cylinder θ is taken to be 0° at the front of the cylinder and 180° at the back. The general shape that evolves is consistent with the characteristic shape observed in (81) – with a relatively circular front shape, a sharp transition point around $\theta \approx 120^\circ$, and a more blunted back shape. As can be seen in the contour plots shown in Figs. 1.17(a)-(d), we find good qualitative agreement of overall interface shape between simulation and experimental results for both cases at varying times. When comparing interface evolution quantitatively, we note that in (81), the interface between the ice and the water was measured using a vernier caliper. An estimate of the experimental uncertainty of such measurements could be around 0.5 - 1 mm, corresponding to an error of about 3-5 %. Additionally, the initial conditions and boundary conditions (*i.e.* constant temperature in the cooled cylinder) are more challenging to control experimentally than in numerical simulations. For example, (81) notes a T_{cyl} fluctuation of up to $\approx \pm 2.5^\circ C$. Within these experimental uncertainties, Figs. 1.17(a)-(d) demonstrate that the numerical approach reproduces quantitatively the time evolution of the ice shape. Because the ice shape is governed by the flow field around the interface, this result also validates accurate calculation of the flow dynamics.

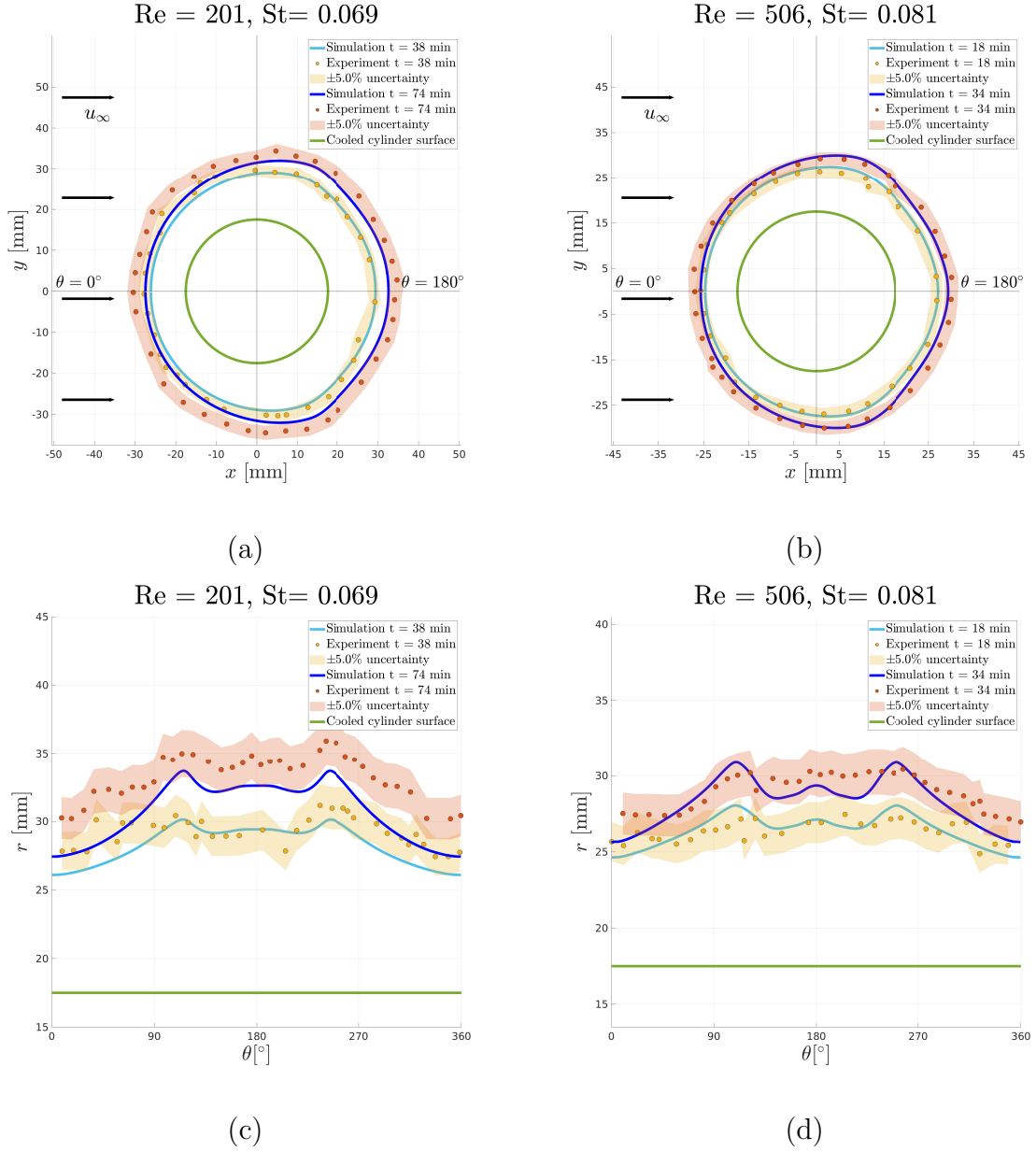


Figure 1.17: Ice interface contour comparison between simulations and experiments of (81) in (a) - (b) $x - y$ coordinates and (c)-(d) in $r - \theta$ coordinates. The green line represents the cooled cylinder surface, the blue lines represent simulation results, the orange and red dots represent experimental results, and the orange and red shaded areas represent the estimated experimental uncertainties of $\pm 5\%$. The coloring corresponds to different times, as denoted in the figure legends.

1.5.2 Local heat transfer at the forward stagnation point

In addition to ice shape evolution, we also compare local heat transfer behavior produced by the solver with experimental results. Since the ice shape at the front is cylindrical, we can compare the local heat transfer at the forward stagnation point with local heat transfer for a circular cylinder under forced convection, as is done in Cheng *et al.*(82). (author?) (83) provide the empirical correlation for the Nusselt number at the stagnation point as given by

$$Nu_{D_s} = 1.08 Re_{D_s}^{0.5} Pr^{0.36}, \quad (1.22)$$

where Re_{D_s} is the Reynolds number at the stagnation point given by $Re_{D_s} = \rho_l D_s u_\infty / \mu_l$. Results are obtained from the simulations of ice growth on a cooled cylinder in cross-flow which are described later in Sec. 1.6 and compared with the empirical correlation described in equation (1.22). We calculate the local Nusselt number at the stagnation point as $Nu_{D_s} = h_s D_s / k_l$, where D_s is the diameter of the ice at the forward stagnation point, and h_s is the local heat transfer coefficient at the forward stagnation point defined as $h_s = (k_l \partial T_l / \partial n)|_{\theta=0^\circ} / (T_\Gamma - T_\infty)$. Simulation results are plotted with the empirical correlation in Fig. 1.18, where the Nu_{D_s} and Re_{D_s} obtained from simulations are those taken at the final time ($t = 40$ minutes) of each simulation. As demonstrated in Fig. 1.18, we find very good agreement between the simulation results and the empirical correlation for the heat transfer.

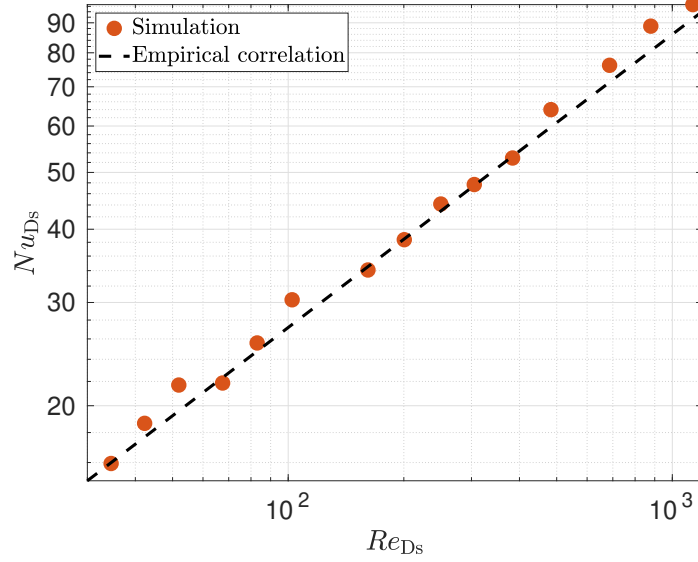


Figure 1.18: Simulation results for the Nusselt number at the forwards stagnation point plotted with the empirical correlation given by equation (1.22) (83).

1.6 Shape dynamics and heat transfer for varying Re and St numbers

Now that the numerical method has been quantitatively validated, we discuss in more detail the shape dynamics and heat transfer results for a large range of Re and St numbers to investigate their effects on the resulting interface morphologies and heat transfer, and how these might be related. We use the same problem configuration as described in Sec. 1.5, and perform simulations for Reynolds numbers $Re = [20, 40, 100, 200, 500]$ and Stefan numbers $St = [0.07, 0.13, 0.26]$. The boundary layer is resolved by 15 – 20 grid points for $Re = 20, 40, 100$ and 10 – 12 grid points for $Re = 200, 500$. It is also worth noting that for $Re = 100, 200, 500$ cases in which vortex shedding emerges, we introduce a slight perturbation in the initial condition of the velocity field as is done by Laroussi *et*

al.(84) in order to reduce the amount of simulation time required to arrive at the steady state. This takes the form $u_0 = U_\infty(1 + 0.25 \sin(2\pi\beta y/H))$, where β is 0.25, and H is the height of the computational domain. In these cases, the interfacial velocity is constrained to zero for a prescribed startup time in order to allow the flow to develop, and we define the start time as the time when the interface is first allowed to move.

1.6.1 Ice shape

We consider the time evolution of the ice-water interface, and in particular the role that Re and St play in the emerging shape. We define a *shape factor* of the ice as the ice contour data normalized by the effective radius at the given time, where the effective radius is defined as $r_{\text{eff}} = \sqrt{A_{\text{ice}}/\pi}$, and A_{ice} is the cross-sectional area of the ice and cylinder. Fig. 1.19 demonstrates such an evolution in time with dimensional contour plots on the left hand side, and shape factor plots on the right hand side.

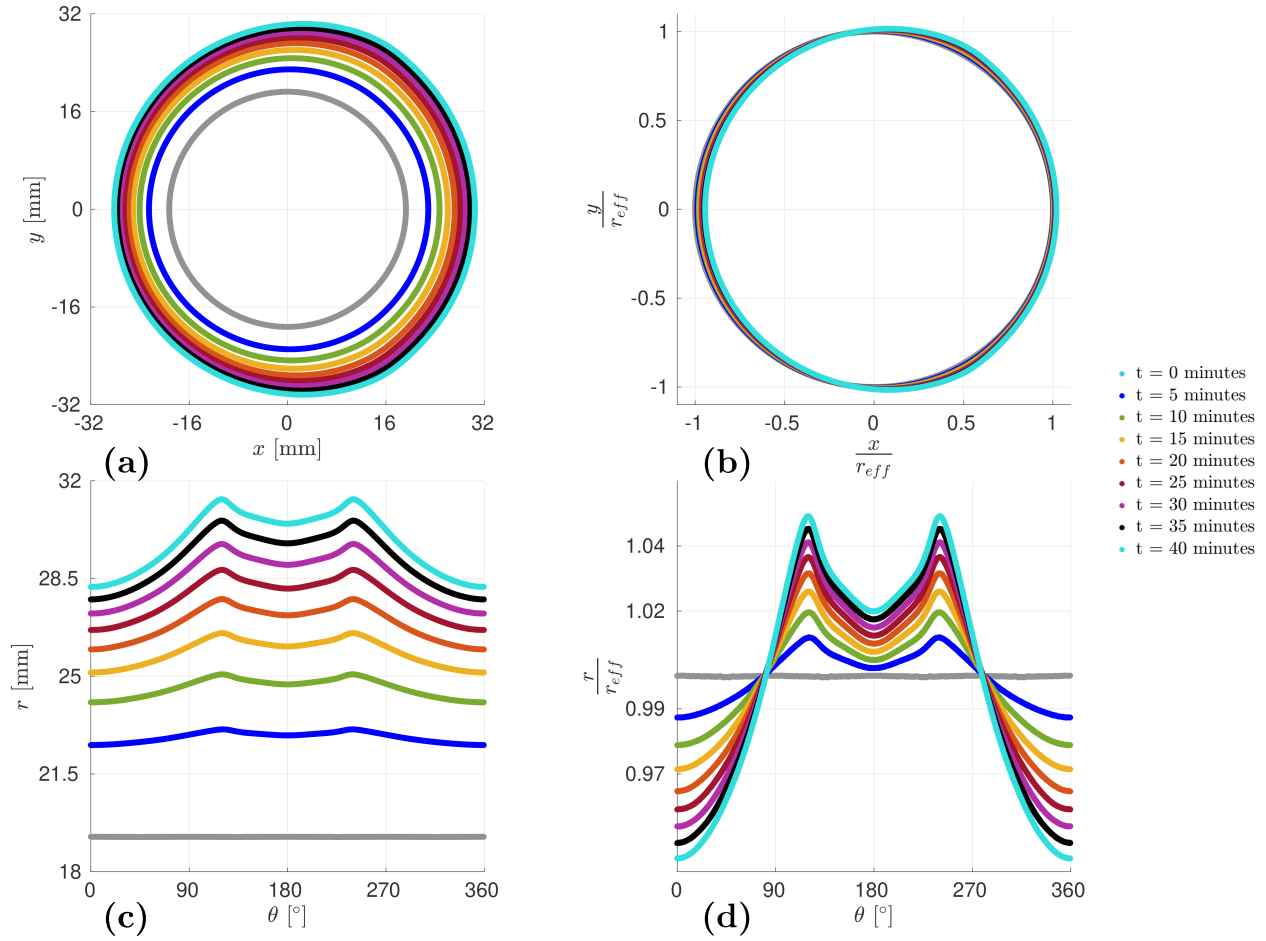


Figure 1.19: Example of ice evolution for $Re = 100$, $St = 0.07$. The interface position at each time is plotted as (a) contour data in xy coordinates, (b) shape factor in xy coordinates, (c) contour data in polar coordinates, (d) shape factor in polar coordinates.

The influence of Re and St on the evolution of the ice interface and the shape factor is reported in Figs. 1.20 - 1.21.

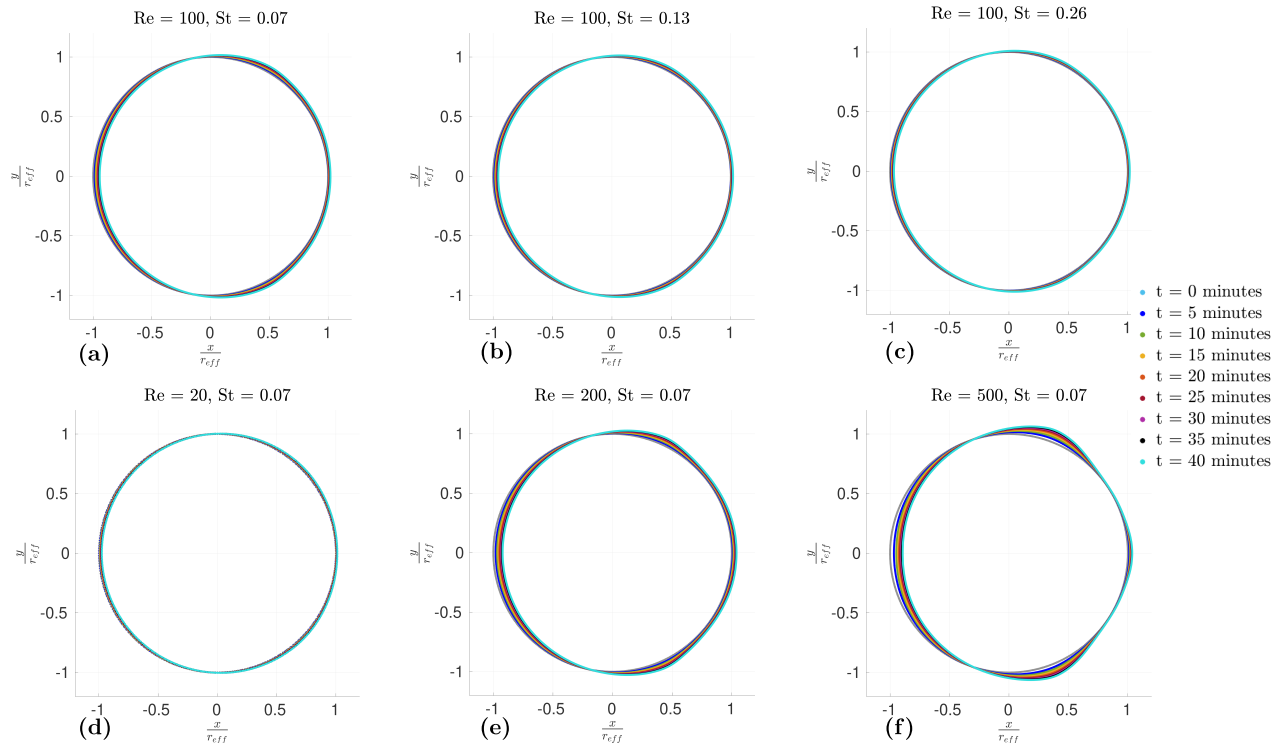


Figure 1.20: Time evolution of the ice shape factor for varying Re and St , with x/r_{eff} on the x -axis and y/r_{eff} on the y -axis. (a)-(c) Influence of increasing St with constant Re . (d)-(f) Influence of increasing Re with constant St .

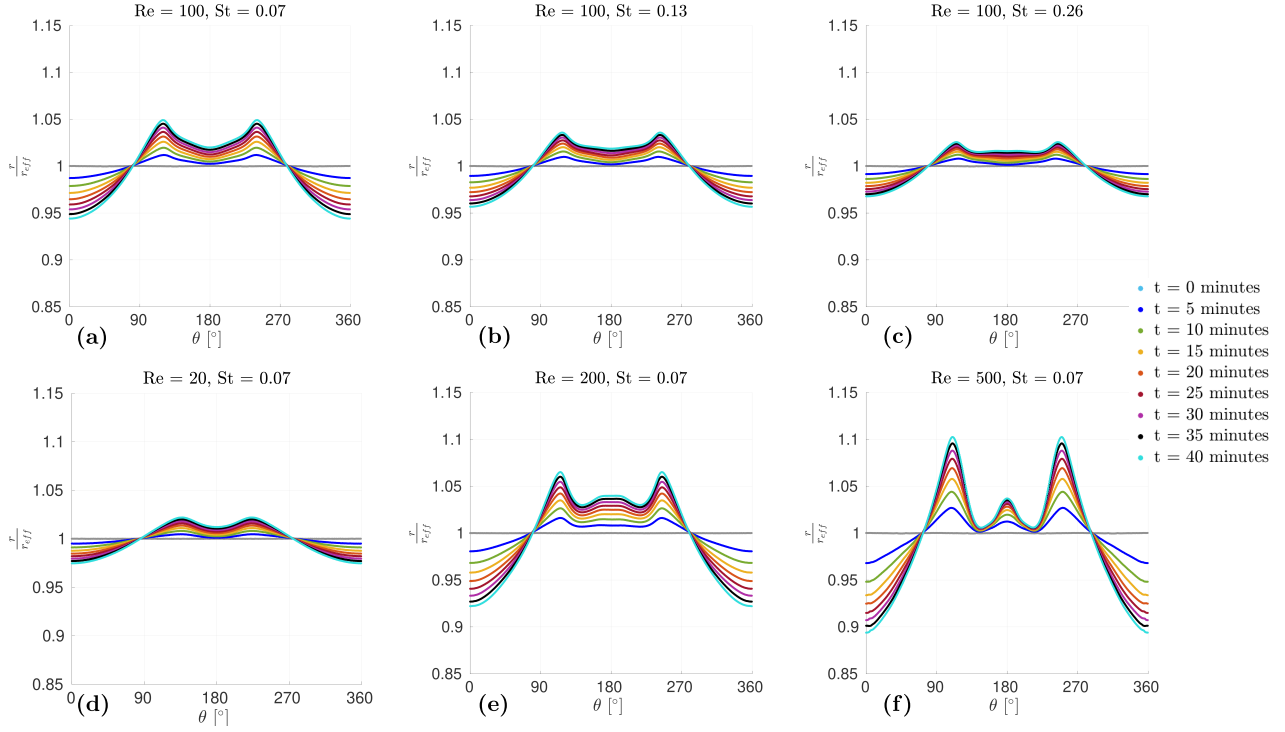


Figure 1.21: Time evolution of the ice shape factor for varying Re and St , with θ on the x -axis and r/r_{eff} on the y -axis. (a)-(c) Influence of increasing St with constant Re . (d)-(f) Influence of increasing Re with constant St .

The balance of diffusive and convective transport of heat is seen to play a significant role in the evolution of the ice shape factor and control the rear symmetry of the ice shape, as well as influence the evolution of ice area, as depicted in Fig. 1.22. Though not displayed above, the cases of no fluid flow ($Re = 0$) maintain a purely cylindrical shape at all times. In general, a lower value of the ratio St/Re leads to a higher shape factor – an ice shape with regions of greater nonuniform curvature, and vice versa. This is likely related to the relative strength of recirculation effects in the wake of the object compared with the Stefan number driven diffusion, as well as the size and role of the

boundary layer, which we discuss in more detail in the following section.

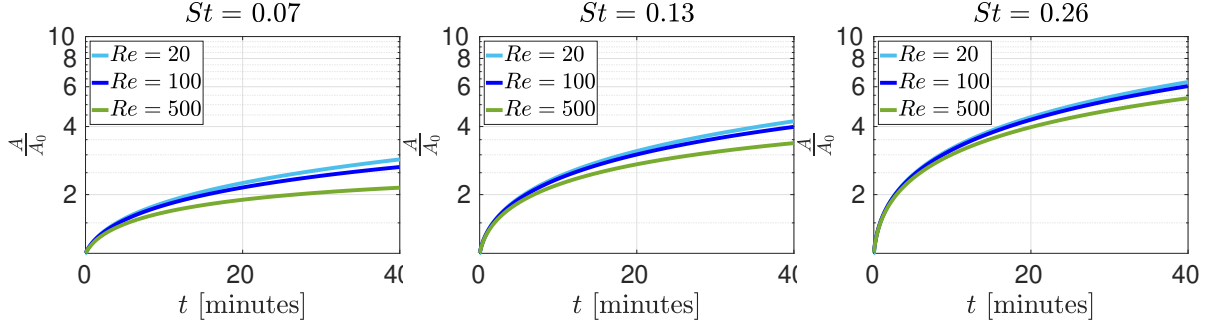


Figure 1.22: Cross-sectional area evolution of ice over time for varying Re and St , where A_0 is the initial area.

1.6.2 Local heat transfer around the ice and its role in interface morphology

The evolution of the ice shape has a unique relationship with heat transfer that can be difficult to predict; this is because the heat transfer along the surface develops to be highly dependent on the shape of the interface at any given time (8; 85; 7). Accurate numerical simulation provides an advantage in studying the coupling between ice growth, heat transfer, and fluid flow, as heat transfer behavior at the interface is readily available from simulation data. Taking a closer look at the heat transfer behavior at the ice-water interface can give us more insight into the roles that advection and diffusion play in determining the emerging interface morphology, especially in regimes where these effects are of comparable influence. We recall that the interfacial velocity is governed by a jump in temperature flux across the interface, *i.e.* $V_{\Gamma} \sim (k_s \partial T_s / \partial n - k_l \partial T_l / \partial n)|_{\Gamma}$. In the case we consider, it is reasonable to assume that the magnitude of the solid heat flux $k_s \partial T_s / \partial n$ is governed mainly by the diffusion of heat in the ice due to the cooled cylinder, captured by the St number. Larger St number implies larger temperature differences across the

ice, and therefore larger solid temperature gradients. Meanwhile, the magnitude of the fluid heat flux $k_l \partial T_l / \partial n$ is governed mainly by the advection of heat due to flow transport around the ice's surface. This flow transport is characterized by the Re number, which will govern flow transport properties such as boundary layer thickness and recirculation effects that ultimately determine the fluid temperature profile and thus fluid temperature gradient near the interface. It is also important to note not only the average effects of diffusion in the solid and advection in the fluid, but the local effects that vary as a function of location along the interface, which may yield helpful explanations in understanding the shape factor profile of the ice-water interface with varying θ around the interface. For example, we report in Figs. 1.20 - 1.21 the variation in shape factor for different trends in Re and St . In general, increasing St tends to result in a more uniformly cylindrical shape of the ice, while increasing Re results in more locations of sharp change in shape along the interface. Additionally, for an increasing Re , ice shape factor minima and maxima varied in both size and azimuthal location along the interface. These trends can be better understood by examining the heat fluxes for both solid and fluid along the interface, and how they vary with Re and St .

First, we examine the effect of Re by considering the magnitude of the local fluid heat flux $k_l \partial T_l / \partial n$, ice radius and temperature contours in the fluid around the interface for two different Reynolds numbers ($Re = 20$ and $Re = 200$) and the same Stefan number ($St = 0.07$), as illustrated in Figs. 1.23-1.24. The role that Re plays is best explained by two factors – the effect of the boundary layer (both thickness and separation point), and the effect of recirculation zones in the wake of the cylinder.

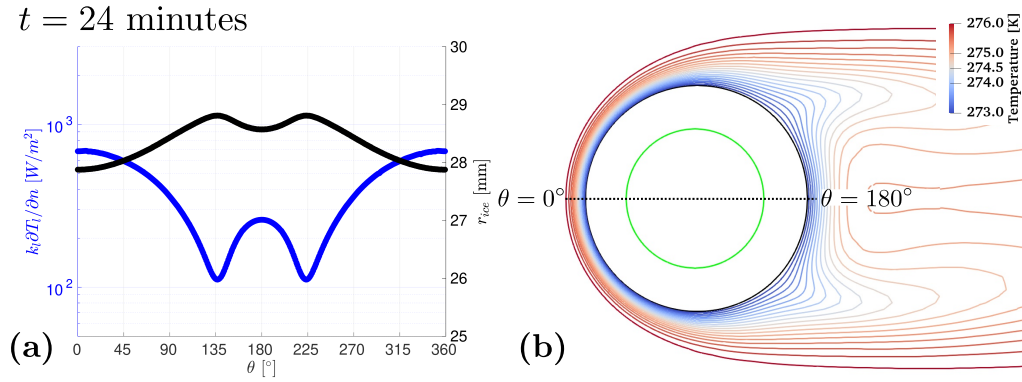


Figure 1.23: For $Re = 20$, $St = 0.07$, (a) local fluid heat flux at the interface and ice radius and (b) fluid temperature contours at $t = 24$ minutes. In this case, the fluid heat flux distribution is symmetrical in θ as correlated with the geometry of the ice shape and maintains the same shape in time – there is no notable time variation in the shape of the distribution within the timescales considered.

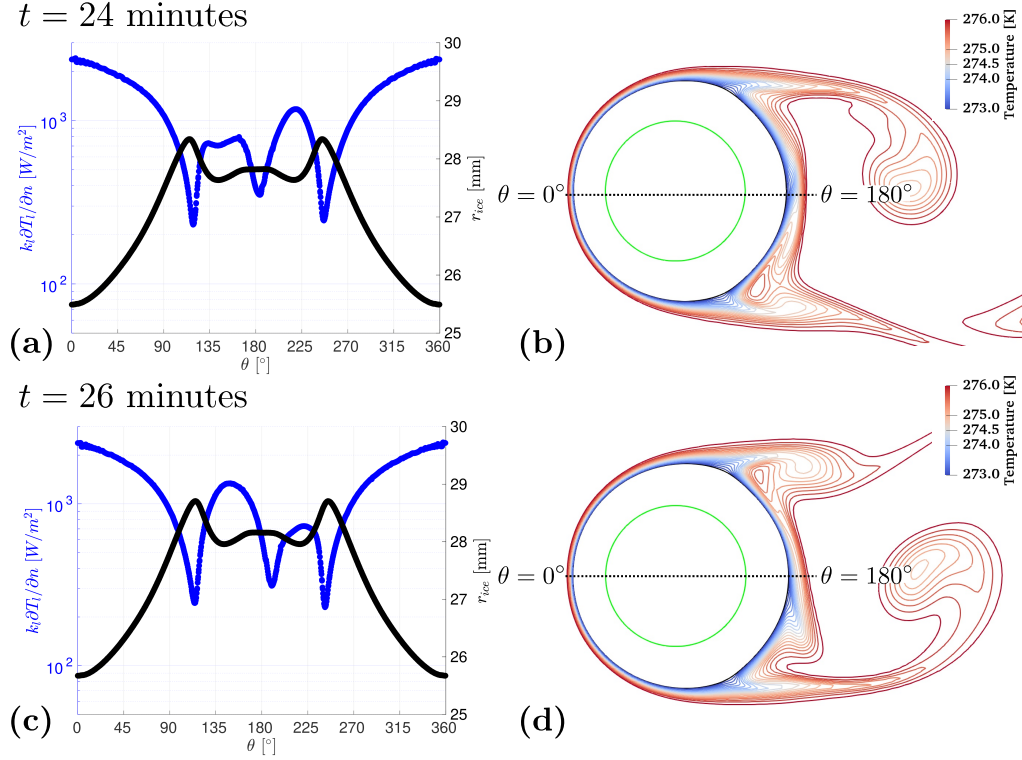


Figure 1.24: For $Re = 200$, $St = 0.07$, local fluid heat at the interface and ice radius at (a) $t = 24$ minutes and (c) $t = 26$ minutes, and fluid temperature contours at (b) $t = 24$ minutes and (d) $t = 26$ minutes. In this case the fluid heat flux distribution is asymmetrical in θ as correlated with pockets of warmer fluid recirculated by vortices, and this distribution varies in time depending on the location of the vortices.

The role of the boundary layer separation can be understood by looking at the fluid heat flux behavior around the interface, and how it correlates with the boundary layer separation and the evolution of the interface shape. In Figs. 1.23-1.24, we see that for both stable flows (*i.e.* no periodic vortex shedding) shown in Fig. 1.23 and unstable flows shown in Fig. 1.24, one can note pockets of cool fluid around the interface at $\theta \approx 135^\circ, 225^\circ$ for $Re = 20$ and $\theta \approx 115^\circ, 245^\circ$ for $Re = 200$ on the temperature contour

plots, which correspond to local minima in heat transfer rate and local maxima in the ice radius. These points correspond to the locations of boundary layer separation, where the flow detaches from the ice surface and thus leaves a quiescent region of flow directly behind the separation point, resulting in regions of cool fluid. Because the fluid temperature is already at or close to the interface temperature, at these locations there is minimal heat transfer and the fluid solidifies more quickly, thus creating a larger ice radius. At these locations, a self-reinforcing behavior begins to emerge between boundary layer separation and fluid heat flux. The boundary layer separation causes a local minima in fluid heat flux and corresponding sharp change in interface shape, which in turn reinforces boundary layer separation at the location. This relationship between the heat flux and boundary layer separation was observed experimentally by Cheng *et al.*(82) for freezing/melting. Huang *et al.*(5) experimentally studied a body dissolving in crossflow (a concentration-driven problem) and noted an analogous phenomena, providing similar reasoning for role of the concentration gradient and boundary layer separation point in the emerging shape.

Additionally, we note that increasing Re leads to a general increase in fluid heat flux at the interface. This effect is also likely related to the boundary layer. Recall that in this regime, the thermal boundary layer δ_T is of the same order as the momentum boundary layer δ_m ($\delta_T \sim \delta_m$). Because $\delta_m \sim 1/\sqrt{Re}$, increasing Re results in a thinner δ_T , thus producing larger fluid temperature gradients near the interface, as the fluid temperature must transition from T_I at the interface to the free-stream value T_∞ in a much narrower region.

Meanwhile, the role of recirculation effects can be examined by noting both similarities and differences between the $Re = 20$ and $Re = 200$ cases. For both cases, warmer pockets of fluid are transported via recirculation effects to the region spanned by $\theta \approx 140^\circ - 220^\circ$, resulting in local maxima in heat transfer. The ice solidifies much more slowly in this region due to the warmer fluid temperatures, giving way to local minima in

ice radius. However, these recirculation effects become more pronounced in cases where vortex shedding begins to arise, as can be noted by some interesting differences between the $Re = 20$ case compared with the $Re = 200$ case. For example, when the flow in the wake of the cylinder is stable in the $Re = 20$ case (*i.e.* no periodic shedding of vortices), the local heat transfer is correlated with the geometry of the ice, but keeps the same shape as a function of time (Fig. 1.23). In contrast, for the $Re = 200$ case shown in Fig. 1.24, one can note the asymmetries associated with which side of the cylinder the vortex shedding has transferred more heat to at a given time. At the $t = 24$ minutes, there is a discernible pocket of warm temperature around $\theta \approx 220^\circ$ on the temperature contour plot (Fig. 1.24a) with corresponding greater heat transfer compared with the heat transfer at $\theta \approx 140^\circ$ (Fig. 1.24b). A similar phenomena can be noticed at $t = 26$ minutes, with the pocket of warm fluid on the opposite side now (Fig. 1.24c,d). Because the vortex shedding is oscillatory in nature and takes place at a much faster time scale than the interface evolution, this asymmetrical effect likely evens out over time to produce a symmetrical shape in ice contour. The effects of recirculation are more pronounced at higher Reynolds numbers in the considered flow regimes, which facilitate higher rates of heat transport to the back of the ice, leading to larger temperature gradients and resulting heat fluxes.

The shape dynamics of a body that is melting/dissolving in flow has been studied theoretically by Huang *et al.*(5) and Moore (13), utilizing the general driving insight that a dissolving body with a uniform flux (and therefore interface velocity) along the surface will preserve its shape while continuing to shrink in size. They employ conformal mapping approaches in conjunction with free-streamline theory (FST) (86; 87; 88) to find analytical solutions for the terminal frontal shape, which is found to be cylindrical. The analytical solutions do not apply past the point of boundary layer separation, but a flattened shape at the back has been observed experimentally by (5). The work of

Huang *et al.*(5) and analytical solutions derived occur at a much higher Reynolds regime than we consider, and consider the case of a body solely dissolving (rather than our case which features elements of both growth and shrinkage), so strict quantitative agreement of shape may not be expected. Nonetheless, the qualitative similarity of the emerging shape in our results and that acknowledged by (5; 13) is evident.

Keeping in mind the effect of recirculation and boundary layer separation, we can now take a closer look at the influence of the Stefan number St on the magnitude of the solid heat flux and the shape of the ice. Because the cooled cylinder is held at a constant temperature (no variation with θ), one would expect that varying the St will produce uniform changes in the solid heat flux $k_s \partial T_s / \partial n$ across the ice's surface. This is highlighted in Fig. 1.25, which reports the time evolution of the fluid heat flux $k_l \partial T_l / \partial n$ and the solid heat flux $k_s \partial T_s / \partial n$ at the ice-water interface for $Re = 20, 200$ and $St = 0.07, 0.26$. The solid heat flux at the interface is fairly uniform for all cases, aside from some slight nonuniformities which likely develop as the interface develops more nonuniformities in shape due to fluid flow effects discussed previously. Additionally, increasing St results in a uniform increase in solid heat flux at the interface. This is in contrast to the effect of increasing Re , which results in not only an increase in average fluid heat flux at the interface, but also in nonuniform changes to the fluid heat flux distribution (in θ) as Re is changed.

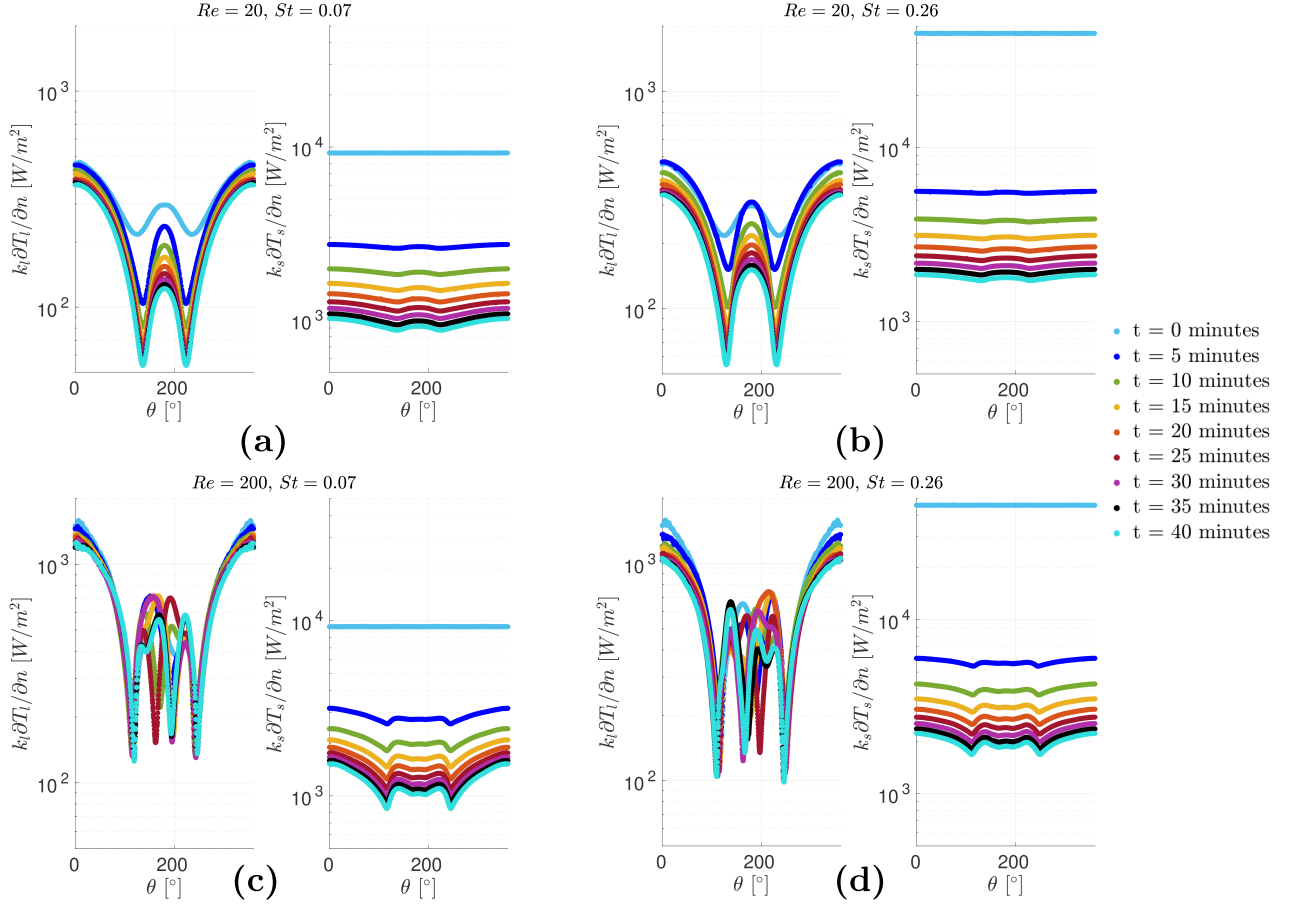


Figure 1.25: The effect of varying Re and St on the magnitudes of the fluid heat flux $k_l \partial T_l / \partial n$ and the solid heat flux $k_s \partial T_s / \partial n$ at the ice-water interface, as shown for $Re = 20, 200$ and $St = 0.07, 0.26$. The left panel of each plot shows the time evolution of the fluid heat flux $k_l \partial T_l / \partial n$ in time, while the right panel of each plot shows the time evolution of the solid heat flux $k_s \partial T_s / \partial n$ in time. Plots viewed from left to right demonstrate the effect of increasing St for constant Re . Plots viewed from top to bottom demonstrate the effect of increasing Re for constant St . Colors represent different times.

Therefore, the relationship between St and Re also plays a role in the emerging shape as it influences the local evolution of V_Γ . This can be thought of in terms of uniform solid conductive effects competing with nonuniform fluid convective effects. Regimes with higher St numbers lead to a uniformly higher contribution from the $k_s \partial T_s / \partial n$ term and thus more uniform cylindrical shapes. In contrast, regimes with relatively higher Re result in higher and more nonuniform contributions from the $k_l \partial T_l / \partial n$ term, yielding interface shapes with more locations of nonuniformity and higher degrees of nonuniformity (*i.e.* higher shape factor). This is of course limited to the cases where the cooled cylinder temperature T_{cyl} is uniform in θ , and it would be interesting to examine the competition of these effects for a case where T_{cyl} is nonuniform. For example, one might be able to exert control over the heat fluxes and emerging shape by accounting for the discussed roles of St and Re and applying carefully selected temperature distributions to T_{cyl} .

1.6.3 Average heat transfer around the ice-water interface

During the formation of ice around the cooled cylinder, a particular quantity of interest is the Nusselt number Nu_d , which describes the ratio of convective to conductive heat transfer in the fluid. The local Nusselt number is defined as $Nu_d = hk_l/d$, where d is the diameter of the cylinder and h is the local heat transfer coefficient defined as $h = (k_l \partial T_l / \partial n) / (T_\Gamma - T_\infty)$. Accurate calculation of the Nusselt number necessitates measurement of the local fluid temperature gradients at the interface, which remains an experimental challenge. Previous experimental evaluations of the Nusselt number for this problem have relied on making a steady state assumption that if the interfacial velocity is close to zero ($V_\Gamma \sim 0$), $k_l \partial T_l / \partial n = k_s \partial T_s / \partial n$, and therefore the solid heat flux $k_s \partial T_s / \partial n$, which is much easier to measure, can be used to calculate the Nusselt number instead. However, Fig. 1.26 highlights that this assumption may not be the most reliable one,

as the difference between solid and fluid heat fluxes at the interface can still be quite large for very small interfacial velocities and the steady state may not be a condition that is easily reached because of the long time scales, or checked because very small V_Γ are still not near the steady state. This is a case where the numerical tool can provide new insight, as the fluid heat fluxes at the interface are readily available from simulation data, and we need not rely on any assumptions to compute Nu_d .

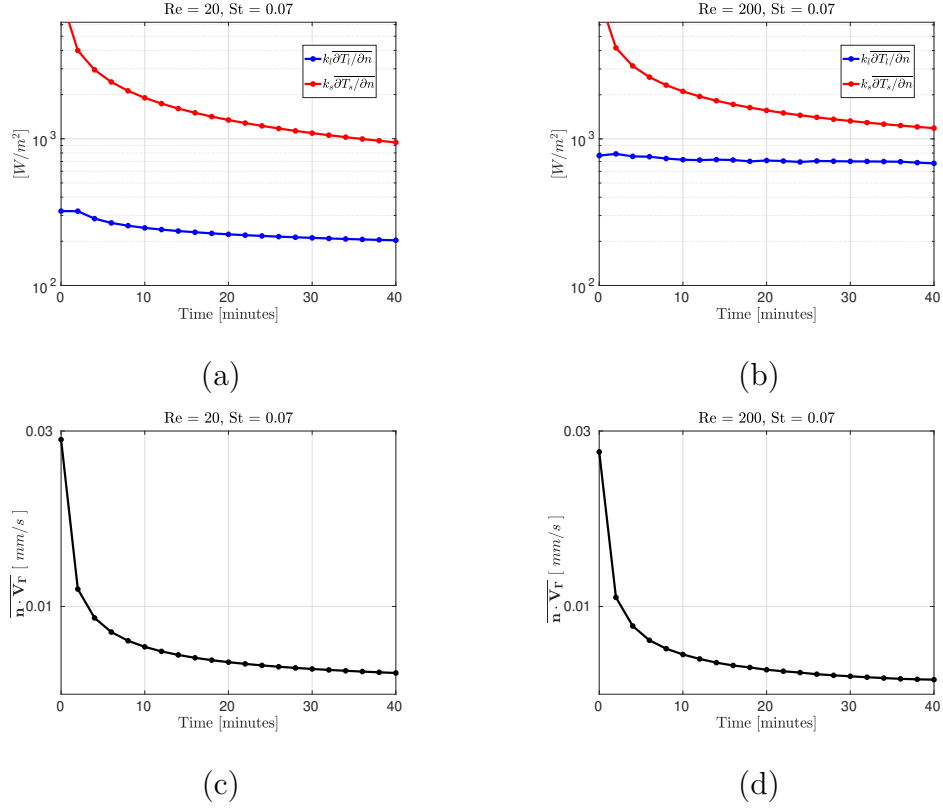


Figure 1.26: Time evolution of heat fluxes at the interface and interfacial velocity for $Re = 20$ and $Re = 200$ with $St = 0.07$. (a)-(b) Solid (red) and fluid (blue) heat fluxes at the interface. (c)-(d) Average interfacial velocity. Although the interfacial velocities approach very near to zero, especially relative to flow velocities $u_\infty \sim 1 - 10$ mm/s, the difference in fluid and solid heat fluxes can remain significant, and it may require much longer timescales in order to reach a true steady state, corresponding to a vanishing interfacial velocity.

Cheng *et al.*(82) proposed an empirical correlation to describe the average Nusselt number at the ice's surface, $\overline{Nu_d}$, for a range of Re and cooling temperature ratios (which can be considered an analog to the Stefan number), where $\overline{Nu_d}$ was found to increase with both Re and St . We recover a similar trend as (82) in terms of Re_d – the average

Nusselt number $\overline{Nu_d}$ scales with $\sim Re_d^{0.5}$. However, in terms of St (or cooling temperature ratio), we obtain an opposite trend. We see that as ΔT and thus St increases, the average Nusselt decreases, while (82) reports the opposite. This discrepancy may be explained by the use of the steady state assumption by (82) in computing Nusselt number at the ice interface as discussed previously, and by noting again the trends observed for local fluid heat flux in Fig. 1.26. As seen in Fig. 1.26, increasing St (and correspondingly, ΔT) does indeed result in an increase in the *solid* heat flux at the interface; however, the same trend is not apparent for the fluid heat flux. If a steady state assumption was made and the solid heat flux was used to compute the Nusselt number instead of the fluid heat flux, it follows that one might arrive at such a trend as reported in (82). Access to the fluid heat fluxes via simulation data allows us to directly capture the trends in fluid heat flux, and therefore provide a more accurate relation for changes in $\overline{Nu_d}$ with respect to St . However, one must then consider why an increase in St results in a decrease, albeit slight, in the fluid heat flux. One possible explanation might be reached by considering how St affects the thickness of the ice, and in turn, the resulting flow dynamics. For example, we have noted that a larger value of St results in a larger magnitude of the solid heat flux $k_s \partial T_s / \partial n$, which in turn leads to a larger interfacial velocity V_I and results in a larger ice thickness for the same given time compared with a lower St value. If the thickness of the ice is larger, then the effective radius of the body r_{ice} seen by the flow, corresponding to the sum of the radius of the cylinder r_{cyl} and the ice thickness, will be larger for a given time. Recalling that the thermal boundary layer is of the same order as the momentum boundary layer ($\delta_T \sim \delta_m$) and the momentum boundary layer scales with the radius of the body as $\delta_m \sim (r_{ice}) / \sqrt{Re_d}$, one may note that this increase in ice thickness will result in a larger boundary layer. A thicker thermal boundary layer will also imply a thicker region over which the fluid temperature must go from the interfacial value T_I to the freestream temperature T_∞ , thus leading to smaller gradients in fluid temperature.

This effect results in a lower value of fluid heat flux $k_i \partial T_i / \partial n$, and consequently a lower local Nusselt number Nu_d , in all the regions where the boundary layer remains attached (*i.e.* $\theta \approx 0 - 135, 225 - 360$), as is illustrated in Fig. 1.27, which in turn leads to a lower values of average Nusselt number for increasing values of St .

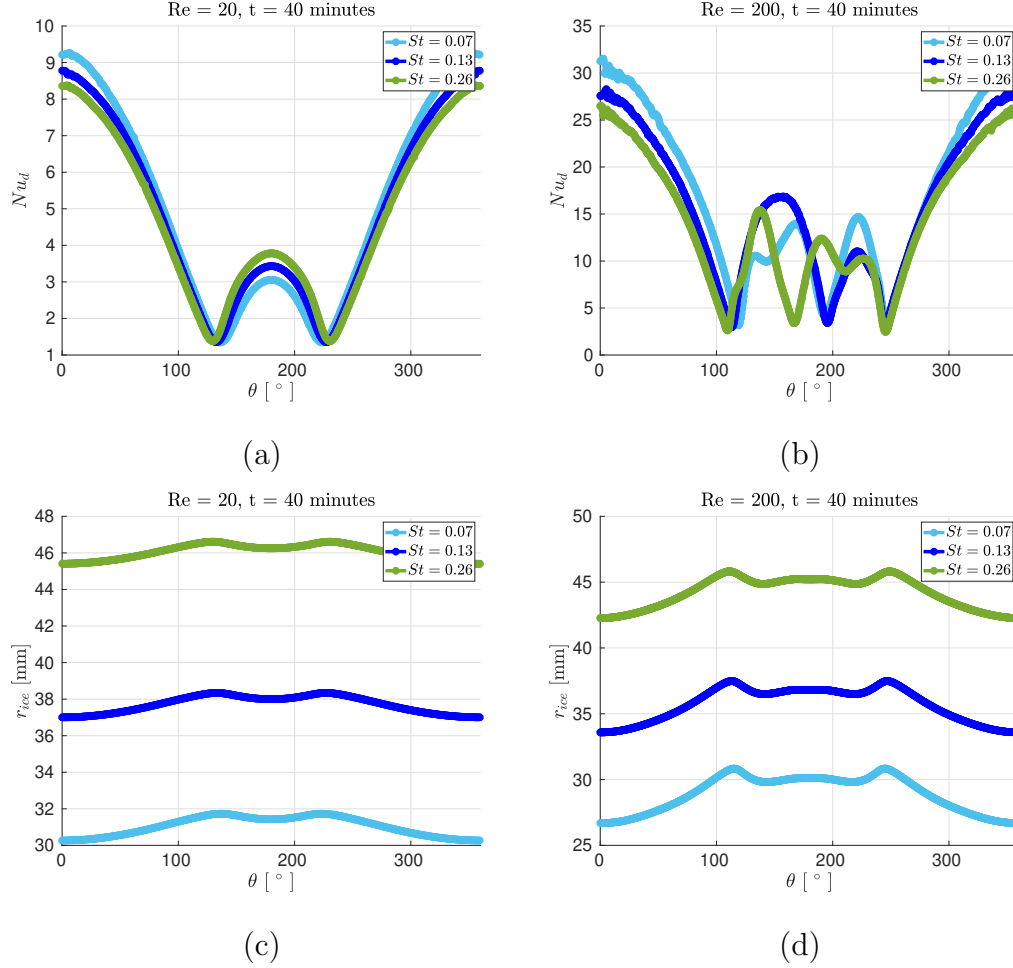


Figure 1.27: Local Nu_d distributions at $t = 40$ minutes for varying St number for (a) $Re = 20$ and (b) $Re = 200$, and r_{ice} distributions at $t = 40$ minutes for varying St number for (c) $Re = 20$ and (d) $Re = 200$.

Therefore, our numerical results allow us to present a new correlation using the average $\overline{Nu_d}$ at the final time of the simulations, $t = 40$ minutes. We report that for all the

cases except $Re = 500$, the average Nusselt number Nu_d at the final simulation time is not changing by more than $\approx 1\%$ over a span of 2 minutes, and for the $Re = 500$ cases it is not changing by more than $\approx 5\%$ over a span of 2 minutes. Our proposed correlation is given by

$$\overline{Nu_d} = 0.59 Re^{0.57} St^{-0.16}, \quad (1.23)$$

where the average Nusselt number over the ice's surface is computed as $\overline{Nu_d} = \frac{1}{2\pi} \int_0^{2\pi} Nu_d d\theta$. The simulation data is plotted along with the proposed correlation in Fig. 1.28.

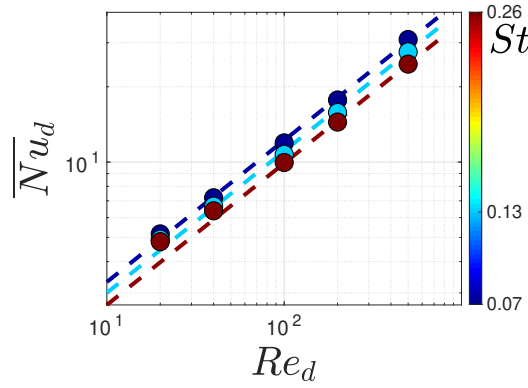


Figure 1.28: Simulation data and proposed correlation given by equation (1.23) are plotted for varying Re and St . The circles represent simulation data, and the dotted lines represent the proposed correlation at different values of St . The colors correspond to different values of St .

1.7 Conclusion

In summary, in this chapter we present a sharp numerical method for the solution of the Stefan problem coupled with an incompressible fluid flow, which utilizes finite-difference and finite-volume discretizations on adaptive quadtree grids, level-set methods for sharp interface capturing, and a pressure-free projection method for solution of the

incompressible Navier Stokes equations. The numerical approach provides an advantage in accuracy and efficiency by combining three main elements – (a) use of adaptive grids for efficient computation of a problem which is multiscale in nature, (b) accurate computation of gradients at the interface which govern the interfacial velocity, and (c) sharp interface representation which allows for accurate application of interfacial boundary conditions. The method is first verified with convergence tests using a synthetic numerical solution, and then validated for ice growth on a cylinder in cross flow for which we find good quantitative agreement between the simulation and experimental results. Then, we use the numerical tool to investigate the role of the Reynolds and the Stefan numbers on the evolution of interface morphologies, flow dynamics, and heat transfer near the interface. We recapture the qualitative phenomena of a self-reinforcing relationship between boundary layer separation and interface shape reported by previous experiments in both freezing and dissolution. The effects of Re and St are explored by discussing their uniform and nonuniform effects on the resulting shape and heat transfer, which have interesting implications about the possibility to control the shape of the interface. For example, in this case the effect of the St was likely largely uniform due to the uniformity of the cooled cylinder temperature; however, it might be interesting to explore how a nonuniform temperature distribution applied to the cylinder could modify the way that the heat transport via solid diffusion and fluid advection interact. Additionally, we highlight the ways in which simulation data can be leveraged to provide more insight into the interfacial temperature gradients, and offer a new scaling relation for average Nusselt number at the interface $\overline{Nu_d}$ as it relates with St and Re .

The numerical approach and methods described in this chapter will serve as a foundation for the work regarding precipitation/dissolution in porous media (Chapter 2) and solidification of multicomponent alloys (Chapter 3) in the forthcoming chapters.

Chapter 2

Adaptation to the concentration driven problem governed by one scalar transport field in the context of mineral precipitation/dissolution

2.1 Introduction

Due to a strong similarity in governing physics, a natural problem of interest to extend the numerical method developed and discussed in the previous chapter is that of precipitation and dissolution, particularly in a porous media, where the concentration(s) that govern the interfacial surface reaction are transported by a fluid flow. Problems of this nature arise in a fairly diverse array of applications, such as catalysis (89), biofilm precipitation (90), and natural landscape formation (5). One key application where such a phenomena takes place is the mineral precipitation and dissolution present in applications for subsurface flows such as carbon sequestration (91) and fracture formation

and/or sealing (92).

Traditionally, a common strategy for modeling the flow and transport in a porous media has been to take a macroscopic (or Darcy-law) approach, wherein the specific geometry of the porous media is not resolved. Rather, the porous media geometry is represented by a continuum scale porosity field, with a corresponding intrinsic permeability K that describes the ability of the porous media to transmit the fluid. The fluid velocity is then assumed to relate to the pressure drop experienced by the media via the Darcy law $\mathbf{v} = -\frac{K}{\mu_l} \nabla P$. For further details on the Darcy scale modeling of this phenomena, we refer the interested reader to (93). In recent years however, there has been an increased focus on the importance of modeling this phenomena at the pore-scale – i.e. utilizing modeling approaches that resolve the geometry of the porous media, where every point in the domain is represented as either in the liquid or the solid phase. Advantages of pore-scale modeling are related to concepts discussed previously in Chapter 1 (recall the variation fluid heat flux and thus interface shape explored in Sec.1.6.1); the transport of concentration and thus the surface reactions may vary along the surface of the interface, and therefore lead to unique emerging interface morphologies that are better represented when capturing the physics at the interface directly. As a result, variations in the interface evolution at the pore scale can have impacts on the bulk behavior of the flow and transport in the porous media. Many studies have shown the importance of capturing the evolving permeability and reactivity of the media in order to accurately model the macro-scale behavior of the porous media (1; 94; 95; 96). For further information on relevant applications, the interested reader is referred to the thorough reviews by Seigneur *et al.*(94) and Ladd and Szymczak(93), which are excellent sources of information on this topic.

Pore-scale simulation efforts have employed a variety of numerical methods. For interface representation, such endeavors have included level-set (97; 98; 99), voxel (100),

and phase-field (101; 102) approaches among others. For the computation of the transport and Navier-Stokes equations, more traditional CFD approaches have consisted of XXXX. Other approaches like particle (103; 104), Lattice-Boltzmann (105), and micro-continuum Darcy-Brinkman-Stokes (106) methods have also been employed. An excellent summary of various approaches employed and related references can be found in Table 1 of Molins *et al.*(1), or in the review by Ladd and Szymczak (93).

As the numerical methods employed in this Thesis are related to sharp representation of physics directly at the interface, this Chapter is focused on extending the techniques described in Chapter 1 in order to simulate precipitation/dissolution of porous media at the pore-scale. As discussed in Chapter 1, the method has key advantages in (a) the sharp representation of the solid-liquid interface and ensuing accurate calculation of gradients at the interface, which drive the geometry of the problem, and (b) the utilization of adaptive quadtree grids, which allow for resolution of fine-scale physics whilst avoiding computational cost associated with uniform grids. The key challenge in extending these techniques to the problem of mineral precipitation/dissolution lies in resolving the large disparity in time scales between the flow and the interface evolution in a way that is computationally feasible. To address this, we present a quasi-steady approach, which is described in detail in Sec.2.3.2 and evaluated further in Sec.2.5.

This chapter is organized as follows. First, we present the governing equations for the pore-scale model we use in Sec. 2.2, and their dimensionless forms in Sec. 2.2.1. Next, we review the numerical method in Sec. 2.3, paying special attention the quasi-steady approach presented in Sec. 2.3.2. The method is validated against a known benchmark problem in Sec. 2.4. In Sec. 2.5, we present a test developed to evaluate performance of the quasi-steady approach in different parameter regimes found in relevant applications. Lastly, in Sec. 2.6 we summarize the described numerical approach and discuss future work and applications of the developed numerical tool.

2.2 Physical model

As is done in Chapter 1, we represent the domain Ω as a liquid subdomain Ω_l and solid subdomain Ω_s , separated by an interface Γ (See Fig. 1.1). The fluid flow is again modeled by the incompressible Navier-Stokes equations, as given by

$$\begin{aligned}\nabla \cdot \mathbf{v} &= 0, \\ \rho_l \left(\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} \right) &= \mu_l \nabla^2 \mathbf{v} - \nabla P + \mathbf{f}.\end{aligned}$$

The transport of the concentration in the fluid, C , is given by the advection-diffusion equation

$$\frac{\partial C}{\partial t} + \mathbf{v} \cdot \nabla C = D_l \nabla^2 C,$$

where D_l is the species diffusivity in the fluid. The mineral precipitation/dissolution reaction occurs at the solid-liquid interface Γ , and can be written as a Robin boundary condition on the concentration C (1) as

$$-D_l \nabla C \cdot \mathbf{n} = \xi r,$$

where ξ is the stoichiometric coefficient for the considered species in the reaction, and $\mathbf{n}$ is the unit outward normal from the fluid domain. The surface reaction rate, r , can take on different expressions depending on the type of reaction. In this work, we assume a linear rate law of the form

$$r(C) = kC,$$

where k is the reaction rate constant for the given reaction considered, typically given in

m/s. The interface evolution can then be described by

$$\mathbf{V}_\Gamma = -rV_m\mathbf{n},$$

where V_m is the molar volume of the solid. Therefore, we can also write the interface velocity in terms of the concentration gradient at the interface, given by

$$\mathbf{V}_\Gamma = \left(\frac{D_l V_m}{\xi} \right) (\nabla C \cdot \mathbf{n}) \mathbf{n}$$

For other presentations of governing equations for this type of phenomena, the interested reader is referred to (1; 107; 93; 94).

2.2.1 Dimensionless form

As such phenomena can occur at a wide variety of different length scales, it is useful for us to formulate and solve the system of equations in a non-dimensionalized form. In selecting the non-dimensionalization, it is worth noting that we may be interested in investigating problems with a pressure drop boundary condition (rather than an imposed velocity), and as a result the characteristic fluid velocity and therefore Reynolds number of the problem may initially be unknown. As a result, we choose to non-dimensionalize the system by the species diffusivity D_l rather than the more standard selection of fluid velocity, an approach similar to that taken in (108). The dimensionless quantities in the problem (expressed with a “hat” notation) are chosen as

$$\mathbf{r}' = \frac{\mathbf{r}}{l}, \quad t' = \frac{D_l t}{l^2}, \quad \mathbf{v}' = \frac{\mathbf{v}}{D_l/l},$$

$$P' = \frac{Pl^2}{\rho_l D_l^2}, \quad C' = \frac{C}{C_{\text{ref}}},$$

where C_{ref} is a reference concentration (typically taken to be the equilibrium concentration of the species C_{sol}), and l is a selected characteristic length scale of the problem, typically selected to be the average size of one grain in the porous media. Applying these non-dimensionalizations, the system of equations becomes (dropping the “hat” notation)

$$\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} = Sc \nabla^2 \mathbf{v} - \nabla P \text{ in } \Omega_l, \quad (2.1)$$

$$\frac{\partial C}{\partial t} + \mathbf{v} \cdot \nabla C = \nabla^2 C \text{ in } \Omega_l, \quad (2.2)$$

$$\mathbf{n} \nabla C + Da C = 0 \text{ on } \Gamma, \quad (2.3)$$

$$\mathbf{V}_\Gamma = \gamma (\mathbf{n} \cdot \nabla C) \mathbf{n} \text{ on } \Gamma. \quad (2.4)$$

Here $Sc = \nu/D_l$ is the Schmidt number which relates momentum diffusivity to species diffusivity, $Da = kl/D_l$ is the Damkohler number which expresses surface reaction rate relative to the species diffusivity, and $\gamma = V_m C_{\text{ref}}$, which expresses the reference concentration relative to the molar volume of the solid.

2.3 Numerical methods

2.3.1 General numerical approach

In general, the numerical methods utilized are the same as those presented in Chap-

ter 1, with the fluid concentration acting as an analog to the fluid temperature modeled in Chapter 1. There are only a few distinctions. Firstly, we assume that the concentration of species in the solid is constant, and therefore only compute solutions for a fluid concentration field. Next, the boundary condition on the fluid concentration at the interface described in (2.3) is a Robin condition, in contrast with the Dirichlet interfacial boundary condition on temperature described in (1.7). To discretize this Robin boundary condition, we employ the 2nd order approach of Bochkov *et al.*(109). Lastly, we make a slight adjustment to the algorithm in order to take a “quasi-steady” approach, which is described in detail in Sec.2.3.2. The adjusted algorithm for the solution process in this particular context is illustrated in Fig. 2.2.

2.3.2 The quasi-steady approach

A key challenge in computing solutions to the presented problem is the large difference in the time scales governing the fluid flow versus those governing the interface evolution, as noted in (93). If we define the simulation timesteps required to solve for the flow and the interface evolution respectively as

$$\Delta t_{\text{flow}} = \text{CFL}_{\text{flow}} \times \frac{\min(\Delta x)}{\max(\boldsymbol{v})}, \quad (2.5)$$

$$\Delta t_{\Gamma} = \text{CFL}_{\Gamma} \times \frac{\min(\Delta x)}{\max(\boldsymbol{V}_{\Gamma})}, \quad (2.6)$$

there may be a ratio in timesteps $\frac{\Delta t_{\text{flow}}}{\Delta t_{\Gamma}}$ ranging from 10^{-10} to 10^1 , depending on the flow speed and surface reaction rate, among other things. As a result, we seek to use a strategy that accounts for this wide range in scales and retains a reasonable level of computational cost. Since the problems applications mainly considered in this work (i.e. mineral dissolution/precipitation) occur mainly in the regime $Re \ll 1$, we assume

that the main contribution to the time variation of the flow field is due to changes in problem geometry; i.e. the flow field is assumed to be steady-state so long as there are not significant changes in geometry. With this assumption in mind, a “quasi-steady” time-step denoted Δt_{QS} is defined under which the flow is assumed steady. As such, under this “quasi-steady” timestep, we only solve the interface evolution, as opposed to the entire physics of the problem. The quasi-steady time-step is given by

$$\Delta t_{\text{QS}} = \text{CFL}_{\text{QS}} \times \frac{\min(\Delta x)}{\max(\mathbf{V}_{\Gamma})}, \quad (2.7)$$

where CFL_{QS} is user-selected, and represents the fraction of a grid cell that the interface is allowed to travel during this “quasi-steady” time step. This concept is depicted in Fig. 2.1.

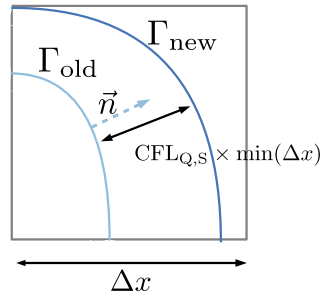


Figure 2.1: Illustration of the concept for how Δt_{QS} is selected, and what role CFL_{QS} plays in this choice. CFL_{QS} represents the fraction of the smallest grid cell size that interface is allowed to travel in the normal direction during the quasi-steady step under Δt_{QS} .

The algorithm for solving the system for a given time t_n therefore becomes as follows:

1. Solve the concentration transport (2.2) with interfacial boundary condition (2.3)

2. Compute $\mathbf{V}_\Gamma$ as in (2.4)
3. Solve the Navier-Stokes equations (2.1)
4. Advect the level-set function ϕ under $\mathbf{V}_\Gamma$ with time step of Δt_{flow} and advance the time by Δt_{flow}
5. if(Quasi-steady approach activated)
 - Advect the level-set function ϕ under $\mathbf{V}_\Gamma$ with time step of Δt_{QS} and advance the time by Δt_{QS}

This process is also illustrated in Fig. 2.2.

It is then a matter of assessing appropriate values of CFL_{QS} to use for different ratios of $\frac{\Delta t_{\text{flow}}}{\Delta t_\Gamma}$ that produces results convergent with what we might expect in a “baseline” (no quasi-steady approximation) case. To develop guidance in selecting this parameter, a series of tests were conducted on a 20 grain porous media geometry. The details of this can be found in Sec. 2.5. Additionally, the quasi-steady approach was evaluated against a known benchmark for different values of CFL_{QS} , and is described in Sec. 2.4.

A note on the reinitialization: It is well known that the reinitialization step used with the level-set method to restore signed distance property is associated with the possibility of mass loss (110). In this particular case, the interface velocity is such a small value that frequent reinitialization may produce noticeable changes to the interface motion, even when a quasi-steady approach is employed. For our purposes, a very basic workaround is to simply not reinitialize every single timestep. This does not produce negative effects on the signed distance property of the level-set function for the same reason that we encounter this issue in the first place – because the interface does not move very much in a given time step. The simulations presented in this work typically use a reinitialization

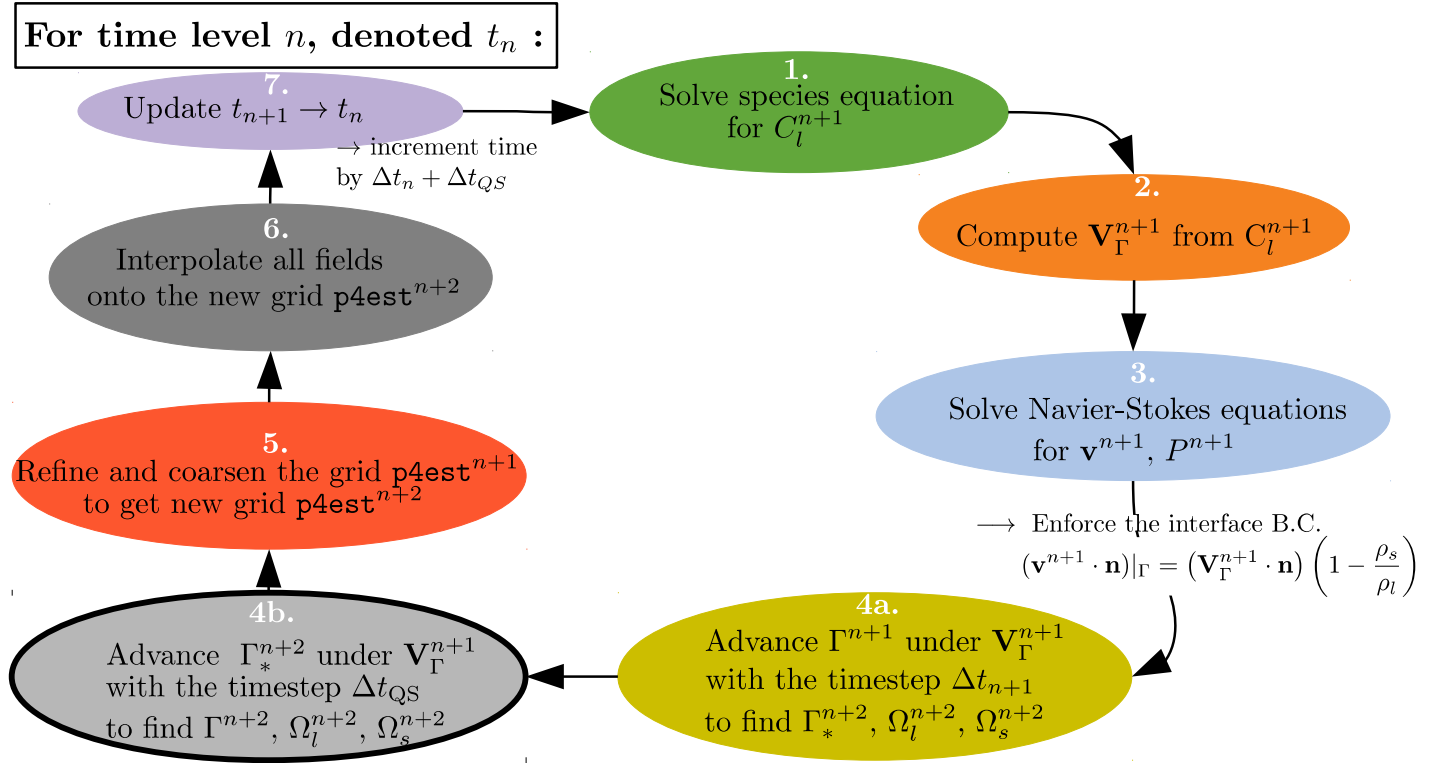
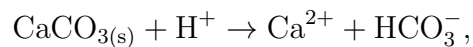


Figure 2.2: Illustration of algorithm used to solve the problem, including the “quasi-steady” approach. The key adjustments to the algorithm previously presented in Fig. 1.2 are highlighted in the addition of step 4b and the change to the time increment in step 7.

rate of once every 50-100 time iterations, which works well to produce results that are reasonable and match the known benchmark described in Sec. 2.4.

2.4 Validation test – comparison with Molins et al.(1)

To assess the validity of the approach in the context of precipitation/dissolution, we compare our simulations with the benchmark problem proposed by Molins *et al.*(1) which describes the dissolution of a single calcite grain in a channel. The dissolution reaction is given by



and the reaction is modeled as a first-order dependence on the concentration of hydrogen ions (H^+). As a result, simulating the transport of the concentration of hydrogen ions is sufficient to model the dissolution reaction. The problem setup and parameters are given below in Fig. 2.3 and Table. 2.1. For further details on this benchmark problem, the interested reader is referred to Molins *et al.*(1).

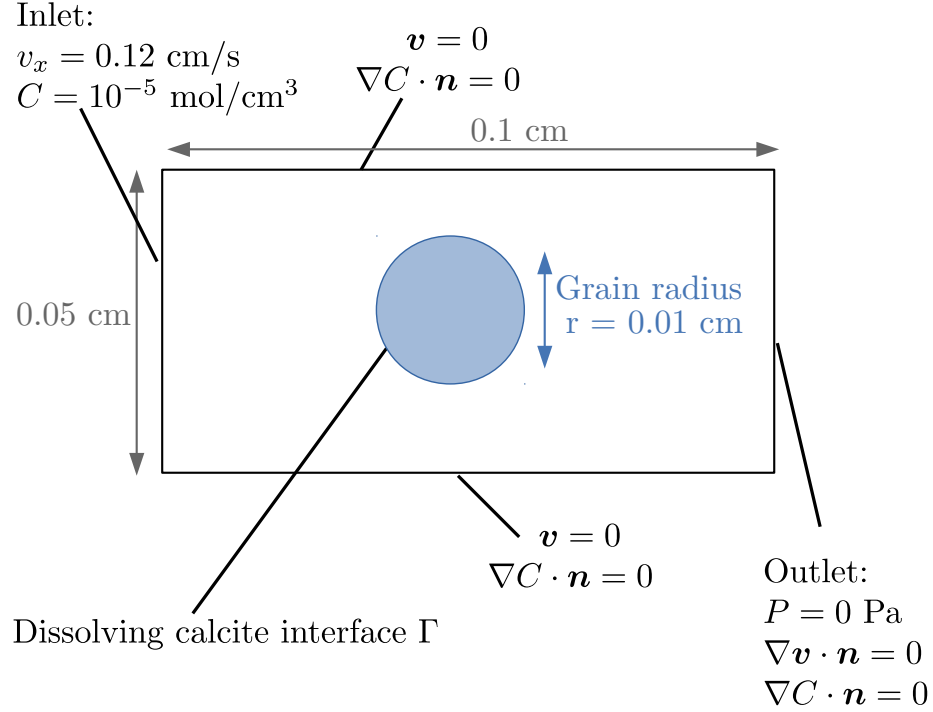


Figure 2.3: Illustration of problem setup for benchmark problem from Molins *et al.*(1).

Parameter	Symbol	Value
Grain radius	r	0.01 cm
Diffusion coefficient	D	cm^2/s
Surface reaction rate constant	k	$1.122 \times 10^7 \text{ cm/s}$
Fluid density	ρ_l	1 g/cm ³
Kinematic viscosity	μ_l	$10 \times 10^{-2} \text{ cm}^2/\text{s}$

Table 2.1: Parameters for the benchmark problem.

In order to compare with the results reported in (1), we calculate the average outlet concentration c_{out} , average reaction rate R , and grain area A as defined in (1). The

effluent concentration and average reaction rate are given by

$$c_{out} = \frac{\int_{\delta S} c \mathbf{v} \cdot d\mathbf{s}}{Q}, \quad (2.8)$$

$$R = \frac{Q(c_{out} - c_{in})}{\xi A}, \quad (2.9)$$

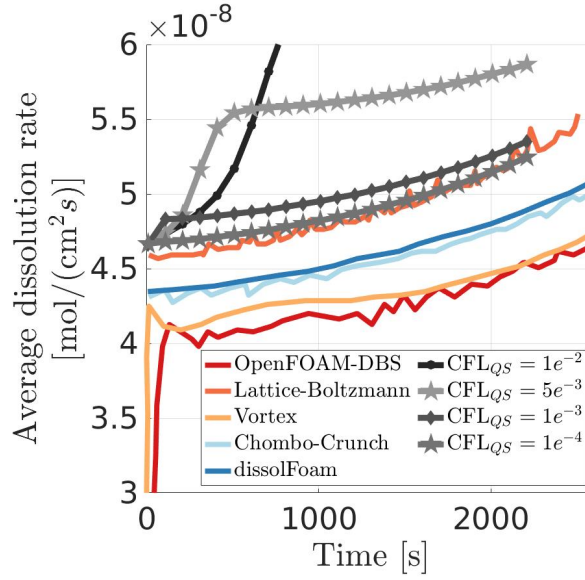
where Q is the outlet flow rate calculated by integrating over the outlet area as

$$Q = \int_{\delta S} \mathbf{v} \cdot d\mathbf{s},$$

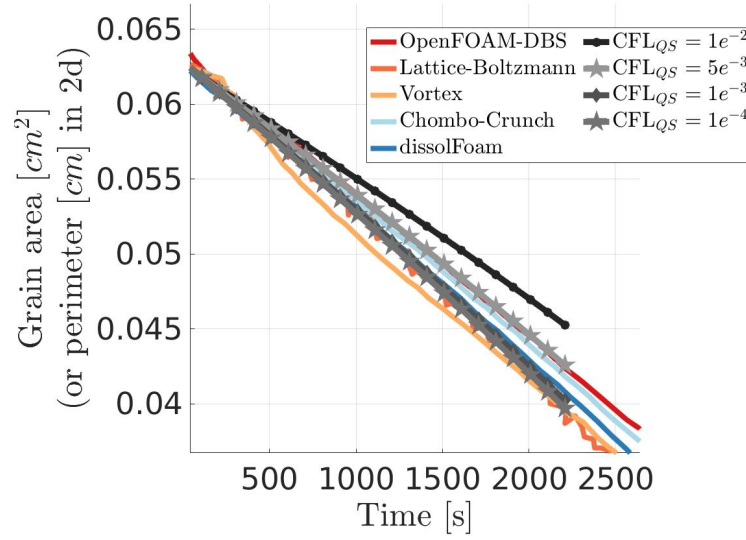
and A is the grain area, taken to be the grain perimeter for the 2d problem.

It is also important to note that the benchmark problem presents two parts for 2D code evaluation – a part I which is steady state wherein the interface is not allowed to evolve (lasting ~ 5 seconds), and a part II moving boundary case lasting $\sim 2000 - 3000$ seconds. In our case the part I case was utilized to establish an appropriate grid size at which to conduct tests for varying values of CFL_{QS} . A choice of $\Delta x_{min} \approx 0.01r$ was found to be sufficient, resulting in approximately 40,000 grid points. The part II test is used to assess the effect of CFL_{QS} choice on the simulation result, which is the focus of the work presented in this section.

We compare against the results reported in (1) for several different selected values of CFL_{QS} . As can be seen in Fig. 2.4, the numerical approach performs well on the benchmark problem for CFL_{QS} values approximately 1×10^{-3} and lower, capturing very similar behavior in evolution of R and grain area compared with the other codes assessed in Molins *et al.*(1). Whilst still capturing appropriate problem behavior, the “quasi-steady” approach is able to significantly reduce the required simulation time, with the $CFL_{QS} = 1 \times 10^{-3}$ simulation completing in roughly 1% of the computational time of our traditional approach.



(a)



(b)

Figure 2.4: Comparison of the numerical approach for varying values of CFL_{QS} with benchmark codes assessed in (1). It can be seen that for the given parameter regime, the approach performs accurately for values of CFL_{QS} $\leq 1 \times 10^{-3}$.

2.5 Assessing appropriate values of CFL_{QS}

2.5.1 Problem Setup

Because there can be such a wide range of values for the ratio between time steps $\Delta t_{\text{flow}}/\Delta t_{\Gamma}$, a test was developed to gain an understanding of what values of CFL_{QS} give converging results for different ranges of $\Delta t_{\text{flow}}/\Delta t_{\Gamma}$ specifically in the context of a porous media. As we are interested particularly in studying problems of precipitation and clogging, we choose to model a simple precipitation reaction of the general form



where $A_{(s)}$ is the solid mineral and $A_{(aq)}$ is the aqueous species, similarly to that done in (111).

The problem setup for these tests is selected to be 20 grain porous media, with a domain size of 2 mm × 2 mm and an average grain radius R of 0.1 mm. The characteristic length scale is selected to be the size of an average grain radius. A pressure drop of 0.1 Pa is prescribed across the domain from left to right, which results in a maximum Reynolds number Re of ~ 0.08 , where Reynolds number is defined as $Re = \rho_l R \max(\boldsymbol{v})/\mu_l$ and R is the average radius of a grain. The inlet concentration is selected to be supersaturated as 4 times the equilibrium concentration in order to simulate a precipitation scenario, and prescribed as a Dirichlet boundary condition. The top and bottom walls are set to be periodic. An illustration of the problem setup and boundary conditions is depicted in Fig. 2.5.

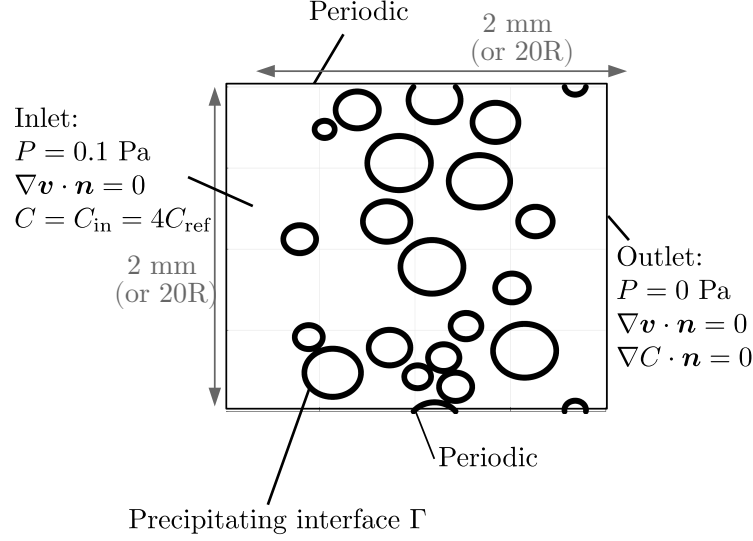


Figure 2.5: Problem setup and boundary conditions used for the 20 grain precipitating porous media test used to assess appropriate values of CFL_{QS} .

Three different regimes of $\frac{\Delta t_{\text{flow}}}{\Delta t_{\Gamma}}$ were evaluated. In order to produce these regimes, the values of γ and Da were varied to produce differing values of the evolving interface velocity. The regimes are referred to as “high”, “middle”, and “low”, in regard to the interface velocity for each regime. The values selected for γ and Da and the resulting ratios in $\Delta t_{\Gamma}/\Delta t_{\text{flow}}$ that were produced are shown in Table 2.2. It is worth noting that the values selected for γ were chosen based on approximate values of this parameter for varying minerals (93), with $\gamma \sim 1 \times 10^{-3}$ matching values for gypsum (107), $\gamma \sim 1 \times 10^{-7}$ approximating values for calcite (93), and $\gamma \sim 1 \times 10^{-5}$ representing some intermediate case. The simulation times chosen for each test are given in Table 2.2, and are selected to show some level of time evolution of quantities of interest for each given regime.

Regime	γ	Da	$\Delta t_\Gamma / \Delta t_{\text{flow}}$	Test simulation time
“High”	1×10^{-3}	200	7×10^3	≈ 3000 seconds (≈ 45 minutes)
“Middle”	1×10^{-5}	2	2×10^6	≈ 3000 minutes (≈ 45 hours)
“Low”	1×10^{-7}	0.2	1×10^9	≈ 600 hours

Table 2.2: Parameters and $\Delta t_\Gamma / \Delta t_{\text{flow}}$ ratio values for each regime selected for the tests to determine appropriate values of CFL_{QS} .

For the different regimes, values of CFL_{QS} ranging from 1×10^{-3} to 1×10^{-7} were tested. Convergence of the solution was tested for flow rate Q , porosity, outlet concentration c_{out} , average interface velocity, and mean tortuosity evolution in time. To assess convergence of the solution, simulations were compared with either a standard approach (when possible), or with an approach using the smallest CFL_{QS} that was computationally reasonable to try. It is also worth noting that the effect of grid resolution for the simulations was also considered, and it was found that a minimum grid size of $0.01R$ produced convergent results with respect to grid refinement. Therefore, all tests were performed with the same grid resolution, with $\min(\Delta x) = 0.01R$.

The calculations of assessed quantities are as follows. The outlet flow rate and average outlet concentration are computed as described in Sec. 2.4 and given by (2.4) and (2.8) respectively. The porosity is given by

$$\text{Porosity} = \frac{A_{\text{domain}} - A_{\text{grains}}}{A_{\text{domain}}}. \quad (2.11)$$

To compute tortuosity, the approach used in (14) is taken. First, we consider a given streamline that starts on the inlet $x = x_0$ at the point (x_0, y_0) and passes through a parallel outlet $x = x + d$, where d is the length of the domain. Then, we define y_0 as the location along the inlet, and the points (x_n, y_n) as the set of points in the streamline

originating at y_0 . We can then calculate the arclength $\lambda(y_0)$ as

$$\lambda(y_0) = \sum_{n=1}^{N-1} \sqrt{(x_{n+1} - x_n)^2 + (y_{n+1} - y_n)^2}. \quad (2.12)$$

The local tortuosity, $\tau(y_0)$ is then defined as

$$\tau(y_0) = \lambda(y_0)/d. \quad (2.13)$$

To define the mean local tortuosity, we simply take the average of $\tau(y_0)$ over the inlet. The streamline coordinates used for calculation of tortuosity were computed using ParaView's streamline capability (112).

2.5.2 Results

Before assessing the different values of CFL_{QS} for each test regime, simulations of 30 second duration using the standard numerical approach were conducted for each of the experiments at varying grid levels to decide on an appropriate grid resolution (similar to the part I test of the Molins *et al.*(1) benchmark discussed in Sec.2.4). The method achieved reasonable convergence of all the quantities considered for a selection of $\Delta x_{\min} \approx 0.04R$ and $\Delta x_{\max} \approx 0.3R$, resulting in approximately 100,000 grid points.

Results for the time evolution for flow rate, outlet concentration, porosity, mean local tortuosity, and average interface velocity are given in Figs. 2.6, 2.7, and 2.8 for the high, middle, and low regime respectively. All quantities are plotted in dimensionless units except for flow rate, which is shown in mm/s . For the high regime, it is computationally feasible to compare the results for varying CFL_{QS} with the standard approach, because the time scale of the problem is still short enough to be computationally feasible. However, for the middle and low regimes, the interface velocity moves so much slower than

the flow that the number of time steps required to solve the problem with the standard method becomes unreasonable. As a result, for the middle and low regimes, the lowest value of CFL_{QS} used is treated as a baseline to compare with.

In the case of the high regime, relatively good convergence of all values is found for $CFL_{QS} \leq 1 \times 10^{-3}$ compared with the standard approach. This corresponds with a maximum allowable choice of Δt_{QS} that is roughly 10 times Δt_{flow} ; essentially we solve 10 interface advection steps for every one step of transport and Navier-Stokes. In contrast, for the middle regime, a choice of $CFL_{QS} \leq 1 \times 10^{-4}$ appears to be sufficient, although this case may benefit from a further investigation. The values of Q , c_{out} , mean local tortuosity, and average interface velocity all appear relatively in agreement for $CFL_{QS} = 1 \times 10^{-4}$ and $CFL_{QS} = 1 \times 10^{-5}$; however, there appears to still be some variation in porosity evolution. Still, the trends are relatively close, and CFL_{QS} may prove good enough when considering the difference in computational cost between the two selections (a reduction in computational time by a factor of 10). This corresponds with a choice of Δt_{QS} that is roughly 100 times Δt_{flow} – namely, we may solve only one transport + Navier Stokes step for every 100 interface advection steps. Lastly, for the low regime case, reasonably converged results are obtained with $CFL_{QS} \leq 1 \times 10^{-5}$, corresponding with a choice of Δt_{QS} that is 10,000 times Δt_{flow} . There is some variation still seen in c_{out} and the average v_{Γ} between $CFL_{QS} = 1 \times 10^{-5}$ and $CFL_{QS} = 1 \times 10^{-6}$, but the variations are still quite small in scale. This test may also benefit from additional examination at a longer time scale, so that the quantities of interest have more time to further evolve, but it serves as a good preliminary estimate.

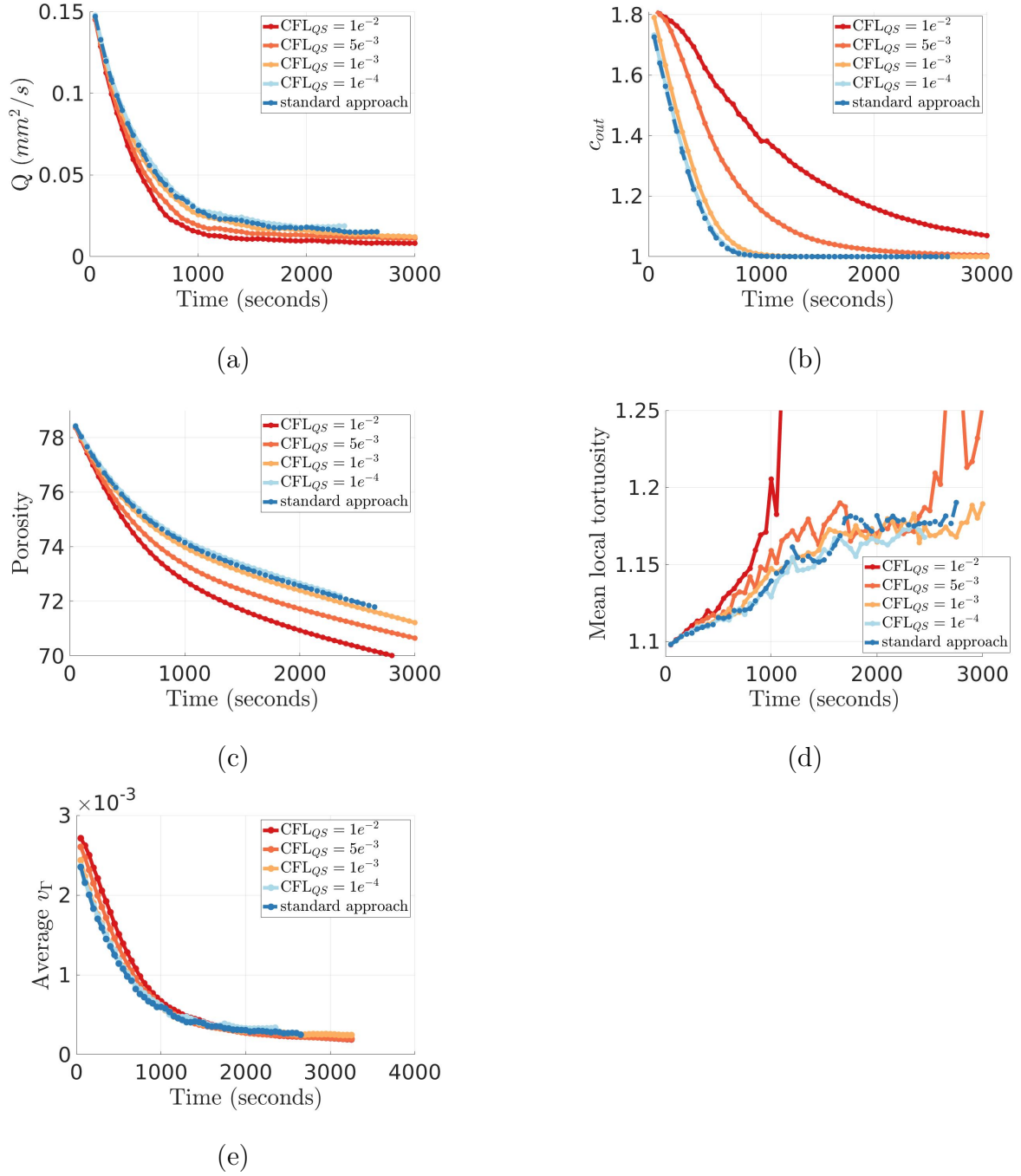


Figure 2.6: Simulation results for the high regime, with $\Delta t_{\Gamma}/\Delta t_{\text{flow}} = 7 \times 10^3$. In this case, a selection of $CFL_{QS} \leq 1 \times 10^{-3}$ is able to reproduce results from the standard numerical approach.

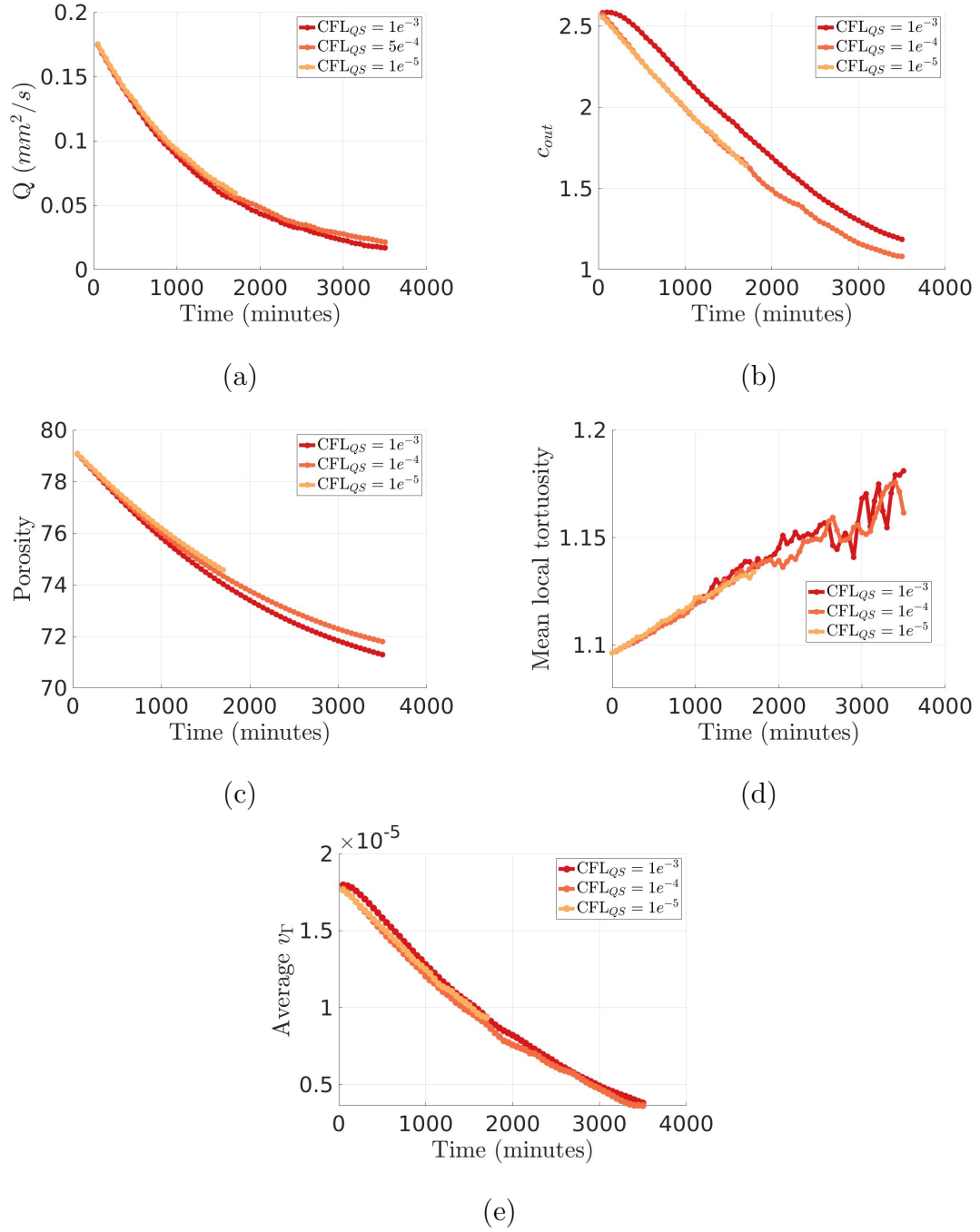


Figure 2.7: Simulation results for the middle regime, with $\Delta t_{\Gamma}/\Delta t_{\text{flow}} = 2 \times 10^6$. In this case, a selection of $\text{CFL}_{QS} \leq 1 \times 10^{-4}$ is able to give reasonable results, however further evaluation may be beneficial.

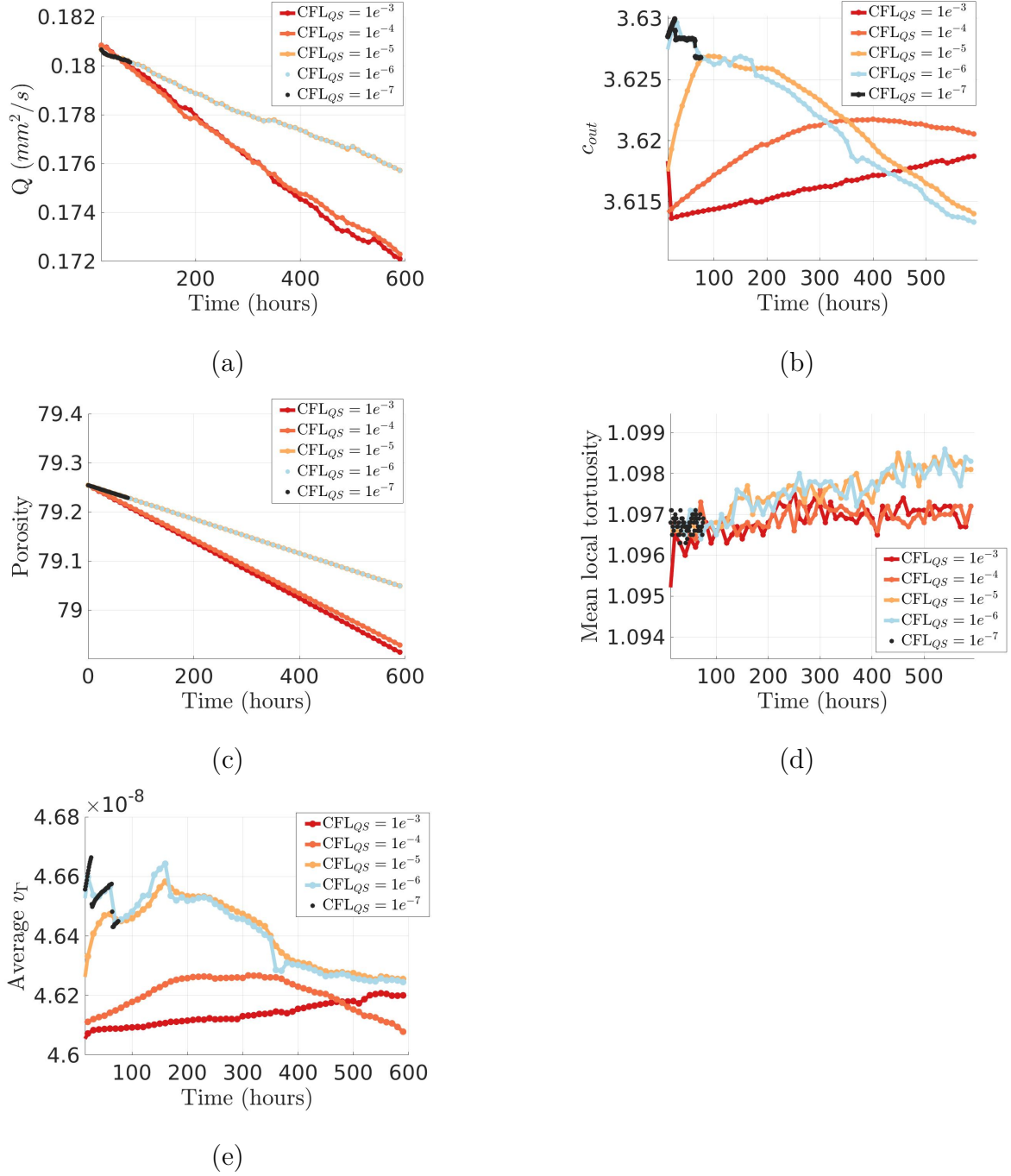


Figure 2.8: Simulation results for the low regime, with $\Delta t_{\Gamma}/\Delta t_{\text{flow}} = 1 \times 10^9$. In this case, a selection of $\text{CFL}_{\text{QS}} \leq 1 \times 10^{-5}$ is able to provide results that converge with smaller selections of CFL_{QS} .

The results of all tests performed are summarized in Table 2.3, which additionally includes parameters and results from the benchmark test in Sec.2.4. These results may serve as guidelines for selecting CFL_{QS} when applying the quasi-steady approach, depending on the problem regime being studied.

Regime	Re	γ	Da	$\Delta t_{\Gamma}/\Delta t_{flow}$	$CFL_{QS} \leq \dots$	$\Delta t_{QS}/\Delta t_{flow} \leq \dots$
“High”	0.08	1×10^{-3}	200	7×10^3	1×10^{-3}	1×10^1
Molins benchmark (Sec. 2.4)	0.3	3.67×10^{-4}	178	1×10^5	1×10^{-3}	1×10^2
“Middle”	0.08	1×10^{-5}	2	2×10^6	1×10^{-4}	2×10^2
“Low”	0.08	1×10^{-7}	0.2	1×10^9	1×10^{-6}	1×10^4

Table 2.3: Guidelines for selection of CFL_{QS} values when using the quasi-steady approach.

In the future, it may be beneficial to additionally examine any effect that flow velocity (and/or resulting Reynolds number Re) has on appropriate selection of CFL_{QS} .

2.6 Conclusion

In summary, we have extended the numerical framework presented in Chapter 1 to the simulation of concentration-driven precipitation/dissolution problems in porous media. While the numerical methods utilized can provide useful gains in pore-scale resolution of such problems, the key challenge in this endeavor is addressing the wide disparity in time scales driving the fluid flow and interface velocity, which can render the problem too computationally expensive to feasibly tackle. To address this, we have presented a quasi-steady approach which augments the algorithm from Chapter 1 to handle this gap in time scales while maintaining computational practicality. The quasi-steady approach is then validated against a known benchmark problem in Sec.2.4, and assessed for reasonable convergence under a variety of parameter regimes in Sec.2.5. To build upon the guidelines

established in Sec.2.5, it may be worthwhile to further investigate the effect of average Reynolds number in the media on appropriate selection of CFL_{QS} .

Future work may include application of the numerical tool developed to investigate precipitation/dissolution problems of interest in porous media. In particular, the method is convenient for studying problems with an imposed pressure drop, which may be a more realistic boundary condition in the context of such applications. Additionally, the method is able to handle situations of both shrinkage (dissolution) and growth (precipitation), and could be used to investigate problems of clogging. For example, Soullaine *et al.*(113) conducted numerical experiments over a parameter sweep of Damkohler (Da) and Peclet (Pe) numbers to characterize different patterns of dissolution in porous media. The method presented here is well situated to perform analogous investigations in the precipitation (clogging) case. Similar studies have been conducted by Kang *et al.*(105) (using Lattice Boltzmann methods), and Tartakovsky *et al.*(104) (using a particle tracking approach), and it would be interesting to compare the results of these with the sharp-interface approach presented herein. Drawing connections between trends in outlet measurable quantities, like flow rate and outlet concentration, and the parameter regime of the precipitation/dissolution reaction taking place, could be of use and interest to experimental researchers studying such phenomena. Additionally, the presented method may be extended to study problems involving biofilm growth, where some of the governing equations used are relatively analogous.

Chapter 3

Extension to multi-component alloy solidification – temperature and concentration driven solidification with multiple scalar transport fields

3.1 Introduction

Another natural extension of the numerical approach developed in Chapter 1 is to tackle Stefan-type problems with governing physics of a similar type, but increased complexity. One key example, which is the focus of this Chapter, is the solidification of multi-component alloys. In contrast to the systems covered in Chapters 1 and 2, multi-component alloy solidification is governed by the transport of multiple scalar transport fields in the fluid domain; i.e. it is dictated by the transport of temperature and multiple concentration fields, as opposed to solely temperature (as covered in Chapter 1) or one concentration field (as covered in Chapter 2).

The accurate simulation of such problems is a field of great importance, as varying solidification regimes can influence the microstructure and thus the resulting material properties of these alloys. For example, in modern turbine engines, nickel-based superalloys (114; 115) play a crucial role as high-temperature structural materials. To ensure optimal performance of superalloy components, the casting microstructure has a significant influence. Therefore, gaining a comprehensive understanding of microstructure formation during solidification is of utmost importance (116; 117; 118; 119).

One key factor affecting microstructure is thermosolutal convection, which arises from density inversion along the direction of gravity during solidification(116). This convection process has the ability to modify various microstructure features, including dendrite morphology, dendrite spacing (117), and microsegregation. Of particular concern is the formation of freckle defects during directional solidification of superalloys(118). These defects are directly linked to the convective flow of the melt within the mushy zone and can have significant implications for the integrity of the material. In this context, investigating the thermosolutal convection and its impact on microstructure formation during solidification is crucial to enhance the manufacturing processes and performance of nickel-based superalloy components in high-temperature applications, particularly in modern turbine engines.

In recent times, significant progress has been made in exploring the growth of dendrites during alloy solidification, utilizing both experimental and numerical methods. Early investigations involved transparent aqueous and organic systems to study dendrite growth. Advancements in X-ray and synchrotron X-ray techniques have further enabled in situ observations of dendrite growth during solidification, as demonstrated in lightweight Ga–In (120) and Al–Cu (121) alloys. These studies have led to the discovery and analysis of phenomena such as channel plume flow, dendrite remelting, and overgrowth. Despite these advancements, direct observation of dendrite growth and melt

flow in superalloy solidification remains a challenge due to the limited attenuation contrast between the solid and melt phases. Consequently, current techniques have not yet allowed for a direct, clear visualization of these processes in superalloy systems.

The growth of a crystal during solidification is primarily governed by heat and mass transfer which is affected by crystalline anisotropy, surface tension, molecular kinetics, and undercooling. At the atomic level, the solute concentration during solidification should be continuous, rendering boundary conditions unnecessary. However, when dealing with macroscopic computations, the approach which most closely resembles the governing physics is to consider the concentration as discontinuous at the solidification front. Consequently, the liquid-solid interface is treated as a sharp boundary, requiring preservation of jump or boundary conditions at that point. Naturally, developing a physically accurate sharp interface model for simulating dendritic growth poses a challenge for numerical simulations.

Traditional numerical methods overcome this challenge of sharp boundary conditions by employing a variety of diffusive interface methods in numerical simulations. These strategies typically involve employing artificial smearing techniques near the interface to calculate the concentration and temperature fields. These approaches include phase-field (122; 123; 124; 125; 126; 127; 128) , enthalpy (129; 130; 131; 132; 133), front-tracking (134; 135; 136), level-set (137) and cellular automata methods(138).

On the other hand, sharp interface methods do not employ any kind of smearing at the interface and preserve the jump conditions at the interface as derived from the governing PDEs. On account of the difficulty of implementing the sharp interface methods, only a handful papers have been published. Y. Yang and co-authors (34) used an immersed interface approach to simulate dendritic growth in a sharp manner. Theillard and co-authors (49) developed a sharp interface approach for diffusion-driven solidification of binary alloys on adaptive quad trees. This method was extended by Bochkov

et al. (139) to simulate diffusion-driven multi-component alloy solidification by using a modified Newton iterative method instead of fixed iterative approach to solve the system of nonlinear coupled PDEs. Meanwhile the work of Bayat et al. (2), as covered in Chapter 1 of this thesis, introduced a numerical framework for coupling the solidification of a pure substance with the convective effects dictated by the presence of an incompressible flow. Knowing the importance of thermosolutal convection in determining microstructure in ternary alloy solidification, we extend the work of Bochkov et al. (139) using the approach described in (2) (Chapter 1) to couple the solidification process with an incompressible flow and its resulting transport of temperature and species, thus achieving simulation of multi-component alloy solidification driven by both diffusive and convective effects.

In this Chapter, we discuss the work of combining the numerical framework described in Chapter 1 with the methods developed by Bochkov *et al.*(139) to accurately simulate the solidification of multicomponent alloys with convective effects. Firstly in Sec. 3.2, we describe the governing equations and the initial and boundary conditions used to model problems of this nature. Next, we discuss the numerical methods utilized and the general algorithm for the proposed numerical framework in Sec. 3.3. In Sec.3.4, we demonstrate the performance of the method via a numerical convergence test. In Sec.3.4.2, we perform proof-of-concept simulations for the solidification of a ternary alloy in the cases of both diffusion-only and convection-diffusion transport of concentration and temperature, and draw comparisons between the two. Lastly in Sec.3.5, we summarize our findings and discuss potential future applications.

3.2 Problem Description

3.2.1 Governing equations

Solidification in a multi component mixture is often modelled as a moving boundary problem. As is done analogously in the previous two chapters, we consider a physical domain Ω , which is separated into liquid Ω_l and solid Ω_s subdomains and define separate partial differential equations on each subdomain. We apply the interface boundary conditions at the evolving interface Γ resolved in a sharp manner and wall boundary conditions on the domain boundary $\delta\Omega$.

We model the solidification process of an alloy containing $N + 1$ different elements (a solvent that constitutes majority of the alloy and N solutes) in a macroscopic manner without resolving atomic-level specifics. In the fluid, the temperature T_l and the dissolved solute concentrations of N elements (C_{lJ} , $J \in [1, N]$) satisfy the advection-diffusion equations

$$\partial_t T_l + \mathbf{v} \cdot \nabla T_l = \alpha_l \nabla^2 T_l + g_l, \quad (3.1)$$

$$\partial_t C_{lJ} + \mathbf{v} \cdot \nabla C_{lJ} = D_{lJ} \nabla^2 C_{lJ} + h_{lJ}, \quad (3.2)$$

where α_l is the thermal diffusivity of the J^{th} solute, D_{lJ} is the concentration diffusivity of the J^{th} solute and g_l and h_{lJ} are the forcing terms. The fluid velocity $\mathbf{v}$ and pressure P satisfy the incompressible Navier-Stokes equations, with a Boussinesq approximation taken (56) to capture the natural convection effects that may result from concentration

and/or temperature induced differences in local density. These are as given by

$$\rho_l (\partial_t \mathbf{v} + \mathbf{v} \cdot \nabla \mathbf{v}) = \mu_l \nabla^2 \mathbf{v} - \nabla P + \rho_l \left[1 - \beta_T (T_l - T_{l,\infty}) - \sum_J^N \beta_{C,J} (C_{l,J} - C_{l,J,\infty}) \right] + \rho_l \mathbf{f}_l, \quad (3.3)$$

$$\nabla \cdot \mathbf{v} = 0, \quad (3.4)$$

where μ_l is the liquid viscosity, β_T and $\beta_{C,J}$ are thermal and solutal expansion coefficients respectively, and $T_{l,\infty}$ and $C_{l,J,\infty}$ are free-stream temperatures and coefficients respectively. The term $\mathbf{f}_l$ represents any other external body force on the fluid, and is typically selected to be $\mathbf{f}_l = 0$.

In the solid, the temperature T_s and the dissolved solute concentrations of N elements ($C_{s,J}$, $J \in [1, N]$) satisfy the diffusion equations,

$$\partial_t T_s = \alpha_s \nabla^2 T_s + g_s, \quad (3.5)$$

$$\partial_t C_{s,J} = D_{s,J} \nabla^2 C_{s,J} + h_{s,J}, \quad (3.6)$$

where α_s is the thermal diffusivity of the J^{th} solute, $D_{s,J}$ is the concentration diffusivity of the J^{th} solute and $g_s, h_{s,J}$ are the forcing terms. We assume that the alloy parameters. The diffusion of the solutes in the solid phase is several orders of magnitude slower than in the liquid phase for typical metal alloys and so we neglect the transport of the species in the solid. Therefore, the species concentration only need to be solved for in the liquid domain.

3.2.2 Initial and Boundary Conditions

At initial time $t = t_0$, the state of the system is defined as

$$\Gamma(t_0) = \Gamma_0, \quad (3.7)$$

$$T_\nu(t_0, \mathbf{r}) = T_{0\nu}(\mathbf{r}), \quad \mathbf{r} \in \Omega_\nu(t_0), \quad \nu = s, l, \quad (3.8)$$

$$C_{\nu J}(t_0, \mathbf{r}) = C_{0\nu J}(\mathbf{r}), \quad \mathbf{r} \in \Omega_\nu(t_0), \quad \nu = s, l, \quad J \in [1, N], \quad (3.9)$$

where $\Gamma_0, T_{0\nu}(\mathbf{r}), C_{0\nu J}(\mathbf{r})$ describe the initial solid-liquid interface, temperature field and concentration fields. The temperature and concentration fields must satisfy the following conditions described below. We assume the phase transition occurs at the thermodynamic equilibrium and so temperature is continuous across the solidification front:

$$[T] = 0, \quad \mathbf{r} \in \Gamma(t), \quad (3.10)$$

and satisfies the Gibbs-Thomson relation:

$$T_l = T_{liq}(C_{l1}, C_{l2}, \dots, C_{lN}) + \epsilon_v(\mathbf{n})\mathbf{v}_\Gamma \cdot \mathbf{n} + \epsilon_c(\mathbf{n})\kappa, \quad \mathbf{r} \in \Gamma(t), \quad (3.11)$$

where $T_{liq}(C_{l1}, C_{l2}, \dots, C_{lN})$ describes the liquidus surface of the alloy, $\epsilon_v(\mathbf{n})$ and $\epsilon_c(\mathbf{n})$ define the curvature and kinetic undercoolings, and κ is the front's mean curvature. T_{liq} is assumed to be a linear function of solutal concentrations

$$T_{liq}(C_{l1}, C_{l2}, \dots, C_{lN}) = T_m + m_{l1}C_{l1} + m_{l2}C_{l2} + \dots + m_{lN}C_{lN}, \quad (3.12)$$

where T_m is the melting temperature of the pure solvent and $m_{l1}, m_{l2}, \dots, m_{lN}$ are the liquidus slopes corresponding to each of the solutes. The current work is not restricted

to such a case, the liquidus slopes may no longer be constants but functions of the local composition. The undercooling coefficients ϵ_v and ϵ_c can depend on the normal vector $\mathbf{n}$ to the solidification front pointing outward from the solid domain, accounting in a way for specific crystalline structures of alloys. An example of this, is a two-dimensional four-fold crystalline structure described by

$$\epsilon_v(\mathbf{n}) = \epsilon_v(1 - 15\epsilon \cos(4\cos^{-1}(\mathbf{n} \cdot \mathbf{n}_0))), \quad (3.13)$$

$$\epsilon_c(\mathbf{n}) = \epsilon_c(1 - 15\epsilon \cos(4\cos^{-1}(\mathbf{n} \cdot \mathbf{n}_0))), \quad (3.14)$$

where ϵ_v and ϵ_c are curvature and kinetic undercooling magnitudes, ϵ is the degree of anisotropy and $\mathbf{n}_0$ is the preferred crystal growth direction. The thermal balance at the interface leads to the following Stefan condition:

$$\mathbf{n} \cdot \mathbf{v}_\Gamma = \frac{(k_l \nabla T_l - k_s \nabla T_s) \cdot \mathbf{n}}{\rho_s L}, \quad (3.15)$$

where k_s and k_l are the thermal conductivities of the solid and liquid respectively and L is the latent heat of fusion of the alloy.

Chemical equilibrium governs the compositions of liquid and solid phases at the solidification front, which is described by parameters called partition coefficients k_J , $J \in [1, N]$, which represent the ratios of component concentrations in solid and liquid phases, that is:

$$C_{s,J} = k_J C_{l,J}, \quad \mathbf{r} \in \Gamma(t), \quad J \in [1, N]. \quad (3.16)$$

The conservation of species at the solidification front lead to the solute-rejection equations

$$D_{IJ}\partial_{\mathbf{n}_l}C_{IJ} - C_{IJ}(1 - k_J)\mathbf{v}_\Gamma \cdot \mathbf{n} = 0, \quad (3.17)$$

Lastly, the conservation of mass at the interface yields the same expression as developed in Chapter 1, namely

$$\mathbf{n} \cdot \mathbf{v}|_\Gamma = \mathbf{v}_\Gamma \cdot \mathbf{n} \left(1 - \frac{\rho_s}{\rho_l}\right),$$

where the quantity $(1 - \frac{\rho_s}{\rho_l})$ is known as the solidification shrinkage factor (57).

The type of boundary conditions (Dirichlet, Neumann or Robin) on the domain boundary $\delta\Omega$ depends on the physical setup. The total heat flux is specified and the boundary is impermeable to solutes:

$$\alpha_s \partial_{\mathbf{n}_\nu} T_\nu = g_{T_\nu}, \quad \mathbf{r} \in \Omega_\nu \cap \delta\Omega, \quad \nu = s, l, \quad (3.18)$$

$$D_{IJ}\partial_{\mathbf{n}_l}C_{IJ} = 0, \quad \mathbf{r} \in \Omega_l \cap \delta\Omega, \quad J \in [1, N], \quad (3.19)$$

where $g_{T_\nu}(t, \mathbf{r}), \nu = s, l$, are prescribed heat fluxes for the solid and liquid phases. We have observed that changing boundary conditions have minimal impact on the performance of this computational method.

3.3 Numerical approach

The key thrust of the numerical approach is to essentially combine the framework to couple traditional Stefan-type problems with fluid flow introduced in Chapter 1 (and in Bayat *et al.*(2)) with the technique for solving diffusion-driven multi-component alloy solidification introduced by Bochkov *et al.*(139). Both works developed methods using

sharp-interface methods on adaptive quadtree grids, using the same general suite of numerical methods. Thus, the main novelty in this work is the successful combination of both strategies to accurately simulate the multi-component solidification process *coupled* with the solution of the Navier-Stokes in order to capture the advective transport of temperature and concentration fields which can significantly influence the solidification regimes present.

The general approach we introduce to solve the problem is provided first in Sec.3.3.1. Then in Sec.3.3.2 we review the methods used to solve the boundary value problems (BVPs) for temperature and concentration transport. Lastly in Sec.3.3.3, we briefly summarize the Newton-type iteration strategy developed by Bochkov *et al.*(139) to address the nonlinearly coupled boundary conditions of this system, which is utilized in this work.

3.3.1 General algorithm to solve the model

The initial conditions for temperatures in the solid and liquid domain, concentration fields in the solid and liquid domain and velocity fields are given at time t_{-1} and t_0 . We solve the problem in time as follows

1. Solve the advection-diffusion equations for the liquid temperature and the concentration fields in the liquid domain to obtain T_l^{n+1} and $C_{l,j}^{n+1}$. For the solid domain, we solve the diffusion equation for T_s^{n+1} . Advective terms are treated using the Semi-Lagrangian formulation as described in Chapter 1, Sec. 1.3.5. The diffusion terms are treated implicitly and the discretizations used are the node-based ones described in Chapter 1 Sec.1.3.4, also mentioned in Sec. 3.3.2. The nonlinearly coupled system of PDEs is solved using an approximate Newton iterative method developed by Bochkov *et al.*(139) as summarized briefly in Sec. 3.3.3.

2. Compute the interface velocity $\mathbf{v}_\Gamma^{n+1}$ using (3.17).
3. Solve the Navier Stokes equation using the pressure-free projection method to evaluate $\mathbf{v}^{n+1}$ and P^{n+1} using the methods detailed in Chapter 1 Sec. 1.3.7, enforcing the interfacial boundary condition given by (3.2.2).
4. Advance the interface Γ^{n+1} to obtain Γ^{n+2} by advecting the level-set function under $\mathbf{v}_\Gamma^{n+1}$ with Δt_{n+1} .
5. Refine and coarsen the current grid, $\mathbf{grid}^{n+1}$ according to the desired criterial i.e. a uniform band around the interface Γ^{n+2} , or certain ranges of vorticity to finely resolve the vortex structures.
6. All fields are moved to the new grid using the quadratic non-oscillatory interpolation technique described in (2) and Chapter 1 1.3.9.
7. Advance to the next time step.

3.3.2 Solving the BVPs for temperature and concentration

To solve the boundary value problems for temperature and concentrations given by (3.1), (3.2), the methods employed are largely similar to those discussed in Chapter 1. The key differences are mainly in the application of boundary conditions – which include Robin-type and jump conditions at the interface. Once again, the advective terms are treated with the Semi-Lagrangian method described in Chapter 1 Sec. 1.3.5. The time discretization for the solid temperature is treated using the 2nd order BDF scheme as described in (139). With these discretizations applied, we are left with the Poisson-type equations of the general form (assuming a generic scalar field q)

$$(a - \eta \nabla^2) q^{n+1} = f.$$

In this general form, the original equations can be recovered as

1. For $q = T_l$:

- $\eta \rightarrow \alpha_l$, the thermal diffusivity of the liquid.
- $a \rightarrow \frac{\alpha}{\Delta t_n}$, and $f \rightarrow T_d^n \left(\frac{\alpha}{\Delta t_n} - \frac{\beta}{\Delta t_{n-1}} \right) + T_d^{n-1} \left(\frac{\beta}{\Delta t_{n-1}} \right)$, where α and β are the coefficients for the Semi-Lagrangian discretization

2. For $q = C_{l,j}$, $j = 1, \dots, J$:

- $\eta \rightarrow D_{l,j}$, the species diffusivity of component j .
- $a \rightarrow \frac{\alpha}{\Delta t_n}$, and $f \rightarrow C_{j,d}^n \left(\frac{\alpha}{\Delta t_n} - \frac{\beta}{\Delta t_{n-1}} \right) + C_{j,d}^{n-1} \left(\frac{\beta}{\Delta t_{n-1}} \right)$, where α and β are the coefficients for the Semi-Lagrangian discretization

3. For $q = T_s$:

- $\eta \rightarrow \alpha_s$, the thermal diffusivity of the solid.
- $a \rightarrow \frac{a_0}{\Delta t_n}$, and $f \rightarrow \frac{1}{\Delta t_n} \sum_{i \geq 1} a_i T_s^{n-i}$, where a_0, a_i are BDF coefficients.

To solve these Poisson-type equations for the node-sampled fields, we employ the approach of (64; 67), which was shown to produce 2nd order accuracy for the solution and its gradients on adaptive quadtree grids. The Dirichlet boundary condition at the interface applied for the C_0 field is done so using the finite-difference method described in (? 140; 72). To impose the Robin interfacial boundary condition necessary for $C_1, \dots, C_J$, the finite-volume approach of (109) is taken. Lastly, to impose the jump conditions required for the temperature continuity and Stefan conditions, we utilize the finite-volume method developed in (141). It is also worth noting that after solving each BVP, it's solution is extended across the interface to allow for calculation of the interface velocity and provide valid values for all fields in each domain for the next timestep. The extrapolation

technique used is the PDE-based approach in (79) and described in Chapter 1 Sec. 1.3.8. For further details on the process of solving these BVPs, the interested reader is referred to (139).

3.3.3 Solving the nonlinearly coupled system of equations

To solve the system of PDEs governing the transport of temperature and concentrations, which features nonlinearly coupled boundary conditions at the interface, Bochkov *et al.*(139) introduced an approximate Newton iteration method, which is used analogously in this work. For a detailed description of this strategy as well as its accompanying derivations, the interested reader is referred to Bochkov *et al.*(139) – specifically to Section 3.2.2 and Appendices A and B. However, the method is briefly summarized below for the general understanding of the reader.

In essence, the method developed by Bochkov *et al.*(139) applies variational calculus to relate changes made in the interfacial boundary condition on the first concentration field (denoted $\Delta C_{l,1}^*$) with a change in the error in satisfying the Gibbs-Thomson relation (denoted ϵ). As a result, a Newton-type iteration can be applied by solving the system of PDEs with an initial guess on the boundary condition of $C_{l,1}$ (denoted $C_{l,1}^*$, computing the error in satisfying the Gibbs-Thomson relation, and then adjusting the boundary condition $C_{l,1}^*$ in such a way to achieve convergence in the Gibbs-Thomson error. A functional derivative relating the change in the boundary condition of C_1 with the Gibbs-Thomson error is defined using Lagrangian multipliers. As a result, computing the appropriate changes to $C_{l,1}^*$ needed to perform the approximate Newton iteration requires the solution of an adjoint system of PDEs, which gives the Lagrangian multipliers needed to calculate the functional derivative. Essentially, for each PDE to solve for a given field in our original system of equations, the adjoint system contains an analogous

PDE to solve for the field's corresponding Lagrange multiplier. In example, the original system solves for the variables T_l , T_s , $\mathbf{v}_\Gamma$, and $C_{l,J}$ with $J = 1, \dots, N$; meanwhile, the adjoint system solves for the corresponding Lagrange multipliers Λ_{T_l} , Λ_{T_s} , $\Lambda_{\mathbf{v}_\Gamma}$, and $\Lambda_{C_{l,J}}$ with $J = 1, \dots, N$.

The general procedure for solving the system of equations using this approach for one time-step is given as

1. Provide an initial guess for the Dirichlet boundary condition on the first concentration field, $C_{l,1}^{*,0}$
2. Set `iter` = 0
3. Solve (3.2) for $C_{l,1}$ with approximate Dirichlet boundary condition $C_{l,1}^*$
4. Compute the front velocity $\mathbf{v}_\Gamma$ based on the solute-rejection equation (3.17) using the solution for $C_{l,1}$
5. Solve (3.2) for $C_{l,J}$, for $J = 2, \dots, N$, enforcing the solute-rejection equation (3.17) as a Robin boundary condition, using the value of $\mathbf{v}_\Gamma$ calculated in Step 4.
6. Solve (3.1) and (3.5) for T_l and T_s respectively, enforcing the Stefan condition given in (3.15) as a jump condition in temperature at the interface
7. Compute the error in the Gibbs-Thomson condition for this iteration, ϵ^{iter} .
8. if ($|\epsilon| > \text{tolerance}$ and `iter` < `itermax`)
 - (a) Solve the adjoint system of PDEs in an analogous manner to the process in Steps 1 - 6 to compute the Lagrange multipliers
 - (b) Compute the updated guess for the boundary condition on the first concentration field, $C_{l,1}^{*,\text{iter}+1}$.

(c) `iter = iter + 1`

(d) Go to Step 3.

The important detail to note regarding this strategy is regarding the derivation of the system of adjoint PDEs (presented in Appendix A of Bochkov *et al.*(139)), and the effect on this derivation caused by the incorporation of advective transport terms. Let us recall that the numerical method utilized to handle the discretization of the advective terms is the Semi-Lagrangian approach, detailed in Chapter 1 Sec.1.3.5. A key advantage of employing the Semi-Lagrangian schemes is that due to the use of departure points in the discretization, all terms related to the advective transport are added to the right hand side of the linear system. As a result, the linear system of the advection-diffusion equation is no different than a purely diffusion case, and the advection-diffusion transport equations take on the same time-discretized form as the diffusion-only equations. Additionally, the work in Bochkov *et al.*(139) derives the system of adjoint PDEs used in this approximate Newton iteration method by applying the Lagrangian multiplier approach to the *time-discretized* system of PDEs. As a result, all the derivations presented in Appendix A of Bochkov *et al.*(139) are equivalently true for the numerical method utilized in the present work.

3.4 Results

3.4.1 Convergence test

For the presented system of PDEs there is no known analytical solution, similarly to the case in Chapter 1. As a result, in order to evaluate the convergence behavior of the numerical approach, we select a synthetic solution for each of the fields and augment the system of PDEs appropriately so that the forcing terms on the right-hand side of the

equations and interfacial boundary conditions produce the desired solutions. For this test, we solve for three concentration fields – thus resembling the solution process for a quaternary alloy system.

We select the fields

$$T_l(x, y, t) = \sin(x) \sin(y) (x + \cos(x - \pi/2)) \cos(y) \cos(t),$$

$$T_s(x, y, t) = \sin(x) \cos(y),$$

$$C_0(x, y, t) = \cos(t) (x \cos(y) \sin(x) + y \sin(y) \cos(x)) + 5,$$

$$C_1(x, y, t) = \cos(t) (x \cos(y) \sin(x) - y \sin(y) \cos(x)) + 5,$$

$$C_2(x, y, t) = \cos(t) (y + \cos(5x)) + 5,$$

$$v_x(x, y, t) = \cos(t) \cos(y) \sin(x),$$

$$v_y(x, y, t) = -\cos(t) \cos(x) \sin(y),$$

$$P(x, y, t) = 0,$$

$$\mathbf{v}_\Gamma(x, y, t) = 1.0.$$

The parameter values chosen are $\rho_l = \mu_l = \rho_s = \alpha_l = \alpha_s = L = 1$. The species diffusivity coefficients are selected as $D_{l,1} = 0.5$, $D_{l,2} = 0.7$, and $D_{l,3} = 1$. The liquidus surface expression is assumed to be linearized, with liquidus slope values as $m_{l,1} = m_{l,2} = m_{l,3} = 0.33$. The Boussinesq approximation coefficients β_T , β_{C1} , β_{C2} , β_{C3} were tested at values of both 1 and 100 with nearly identical results, and the results using 100 are presented here. The free-stream temperature and concentration values in the Boussinesq approximation expression are all selected to be equal to zero. The initial level-set function given by

$$\phi = R - \sqrt{x^2 + y^2},$$

where $R = \pi/2$, $\Omega_l = [(x, y) \in \Omega | \phi < 0]$, $\Omega_s = [(x, y) \in \Omega | \phi > 0]$, and $\Gamma = [(x, y) \in \Omega | \phi = 0]$. The domain size is chosen to be $[-\pi, \pi] \times [-\pi, \pi]$.

The test is run for 100 time iterations, with a constant time step of $\Delta t = 1.0 \times 10^{-4}$ used for all grid resolutions. The errors are taken to be the maximum L_∞ error across all time levels, and 5 different grid resolutions are tested.

The methods all converge as expected, producing between 1st and 2nd order convergence for all fields. The convergence behavior is shown in Fig. 3.1, and the maximum L^∞ errors in time and average convergence rates for each field can be found in Table ??.

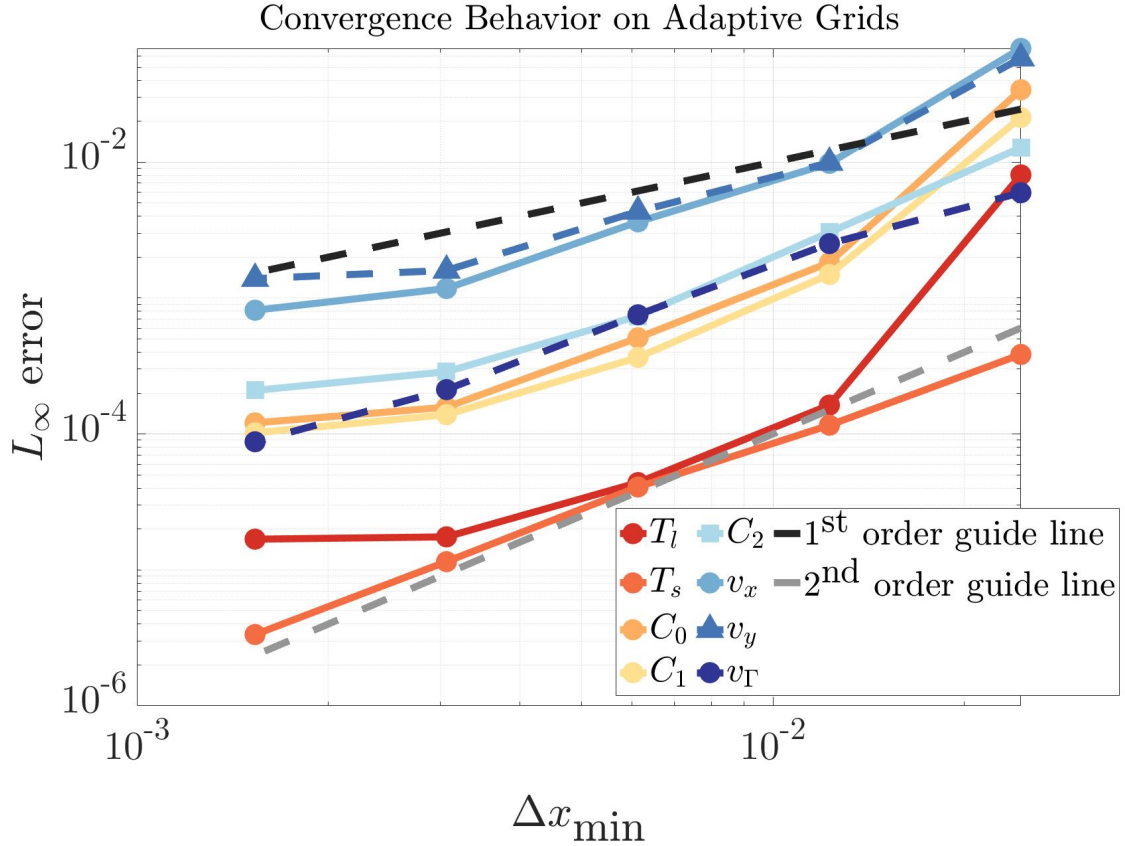


Figure 3.1: Convergence behavior on adaptive grids. All grids produce between 1st and 2nd order convergence for all fields.

$l_{\min}/l_{\max}$	5/7	6/8	7/9	8/10	9/11	Convergence rate
$\min(\Delta x)$	2.45×10^{-2}	1.23×10^{-2}	6.14×10^{-3}	3.07×10^{-3}	1.53×10^{-3}	-
T_l	8.05×10^{-3}	1.63×10^{-4}	4.39×10^{-5}	1.75×10^{-5}	1.68×10^{-5}	2.10
T_s	3.85×10^{-4}	1.16×10^{-4}	4.08×10^{-5}	1.15×10^{-5}	3.35×10^{-6}	1.70
C_0	3.41×10^{-2}	1.82×10^{-3}	5.10×10^{-4}	1.58×10^{-4}	1.21×10^{-4}	1.98
C_1	2.13×10^{-2}	1.49×10^{-3}	3.66×10^{-4}	1.39×10^{-4}	1.02×10^{-4}	1.88
C_2	1.28×10^{-2}	3.05×10^{-3}	7.37×10^{-4}	2.87×10^{-4}	2.09×10^{-4}	1.53
v_x	6.86×10^{-2}	9.79×10^{-3}	3.64×10^{-3}	1.18×10^{-3}	8.14×10^{-4}	1.59
v_y	5.81×10^{-2}	9.92×10^{-3}	4.34×10^{-3}	1.60×10^{-3}	1.39×10^{-3}	1.34
v_Γ	5.96×10^{-3}	2.52×10^{-3}	7.53×10^{-4}	2.12×10^{-4}	8.77×10^{-5}	1.57

Table 3.1: Maximum L^∞ errors in time and average convergence rates for each field.

3.4.2 Simulations of ternary alloy directional solidification – comparing diffusion-only vs. convection-diffusion cases

To illustrate the potential capabilities of the developed numerical method, we now perform simulations of a ternary Ni - 5.8 wt% Al - 15.2% wt Ta alloy for a diffusion-only case and a convection-diffusion case in the presence of a shear flow. The domain is 0.32 cm (3200 μ m) wide by 0.16 cm (1600 μ m) tall, and is initialized with a planar solidification front that contains 5 seeds of 125 μ m radius. A thermal gradient of $G_T = 100$ K/cm with a cooling velocity of $V = 0.001$ cm/s is applied across the domain from top to bottom by enforcing appropriate corresponding Neumann boundary conditions in temperature, given as

$$\begin{aligned}\partial_n T_l &= G_T \text{ at } y = H \\ \lambda_s \partial_n T_s &= -G_T [\lambda_l + V (L + \rho_l c_{p,l} H)] \text{ at } y = 0,\end{aligned}$$

where L is the latent heat of fusion, G_T is the thermal gradient, V is the cooling velocity, and H is the height of the domain. At the top wall, a homogeneous Neumann condition is imposed for the concentration fields. In the convection-diffusion case, a shear flow is induced by imposing a bulk velocity in the x-direction of $v_x = 1.0$ cm/s and a zero

110

pattern of the interface shape. This is presumably due in part to the periodic boundary conditions in the x direction, which provide this problem symmetry for diffusive transport of the temperature and concentrations. In the convection-diffusion case however, the effects of the shear flow traveling from left to right across the domain are visible on the resulting shape of the dendrites. In this scenario, for each seed, the larger dendrite is preferentially the one in the upstream direction; i.e. the left dendrite for each seed evolves to be the larger one. This is likely related to the flow transport of temperature and concentration, creating higher temperature and concentration gradients on the left sides of the seeds and dendrite formations. Visualizations of the flow velocity vectors are provided in Fig. 3.4. Assumably for similar reasons, another key difference between the resulting interface shape between the two cases is the size of the secondary dendritic arms. In the convection-diffusion case, we see that the secondary arms are also preferentially larger on the upstream side of the larger main dendrite for each seed compared with the downstream side. Fig.3.4(b) illustrates the velocity vectors against concentration contours for component 1 to highlight the different regions of concentration and how they might be transported by the flow in the inter-dendritic spaces via recirculation effects.

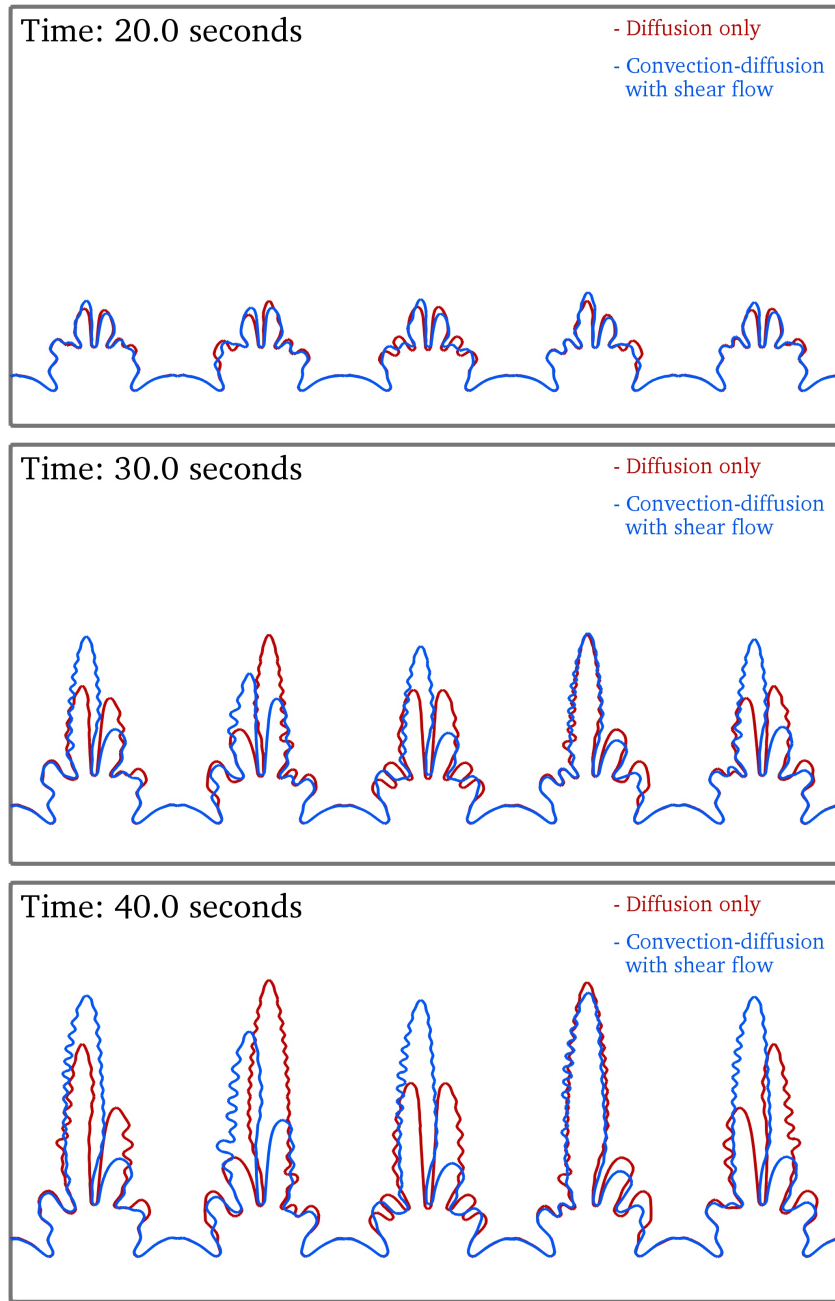


Figure 3.3: Interface contours throughout the simulation for the cases of diffusion-only (red) and convection-diffusion with shear flow (blue). In the case of convective effects (with flow coming from the left) it can be seen that the larger dendrite arms for each seed form preferentially on the upstream side. Additionally, the secondary arms forming off the main dendrite are larger in the upstream direction.

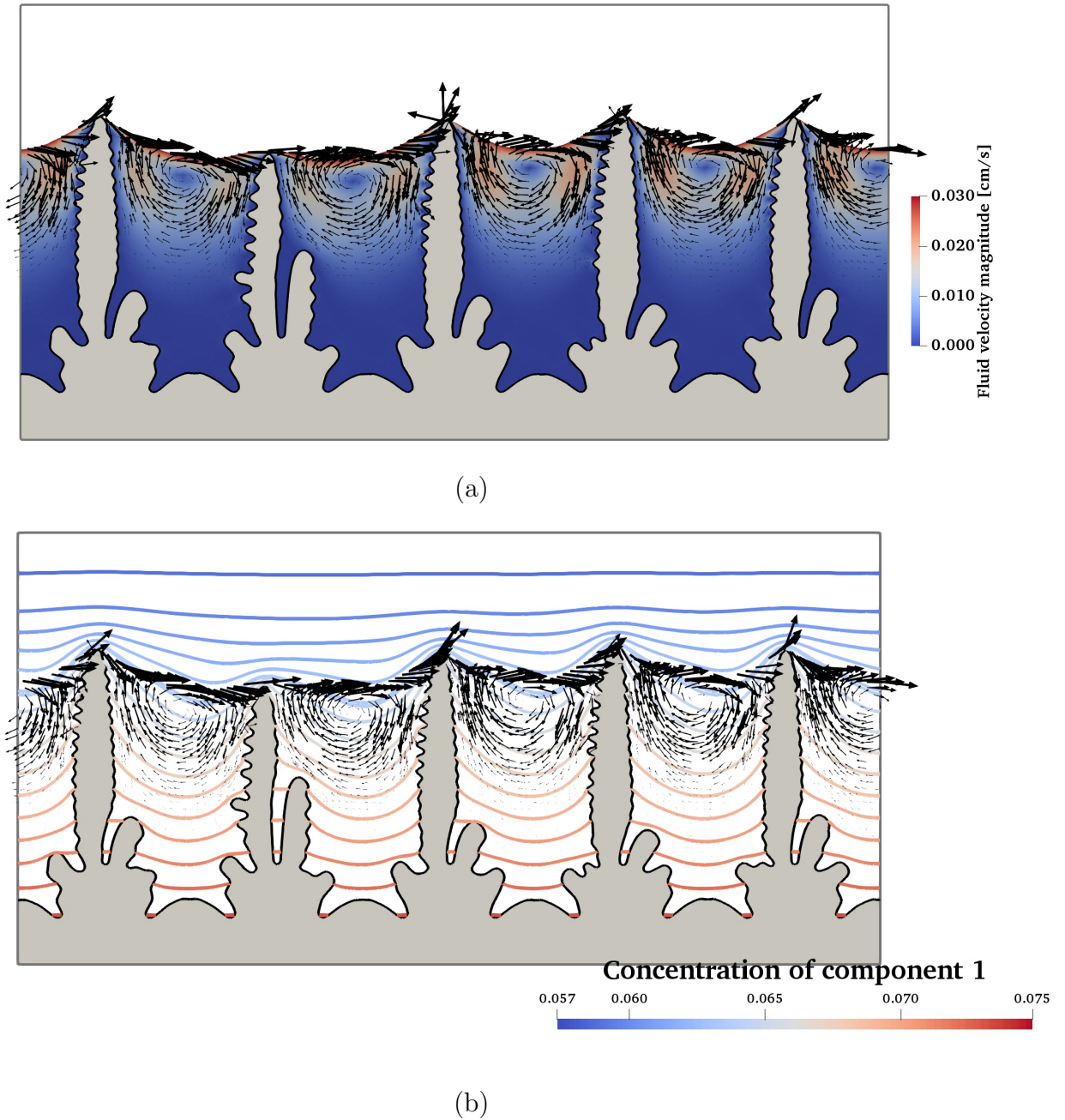


Figure 3.4: (a) Fluid velocity magnitude and velocity vectors at $t = 40$ seconds for the convection-diffusion case. The values for the colormap and vector glyphs are thresholded for fluid velocity magnitude values less than 0.030 cm/s, so that recirculation effects in between dendrites can be better visualized. (b) Fluid velocity vectors (thresholded as in above) and concentration contours for component 1.

Concentration isocontour profiles for components 1 and 2 are additionally illustrated for both the diffusion-only and convection-diffusion cases in Figs.3.5, 3.6, 3.7, and 3.8. In particular, Fig.3.5 and 3.6 demonstrate the slight differences in concentration profile resulting from the shear flow, as the convection-diffusion contours are skewed more to the right as a result of flow transport. Additionally the solid concentration contours in the resulting solidified material differ as well between diffusion-only vs. convection-diffusion cases, if for no other reason than the fact that the final interface morphology is different, which will affect those concentration distributions.

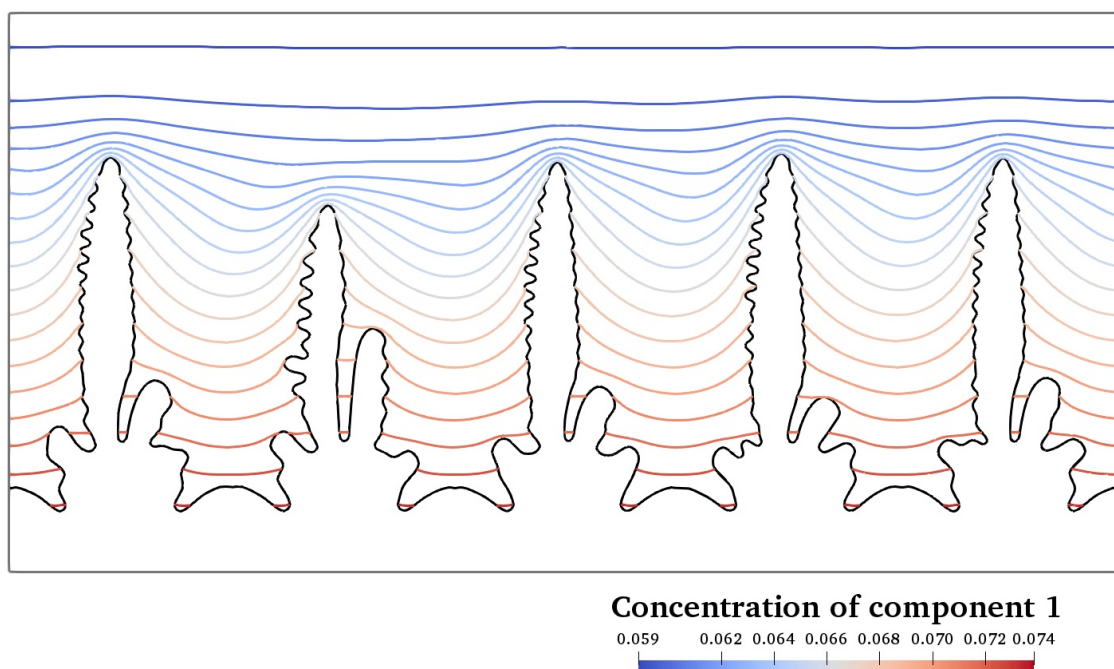
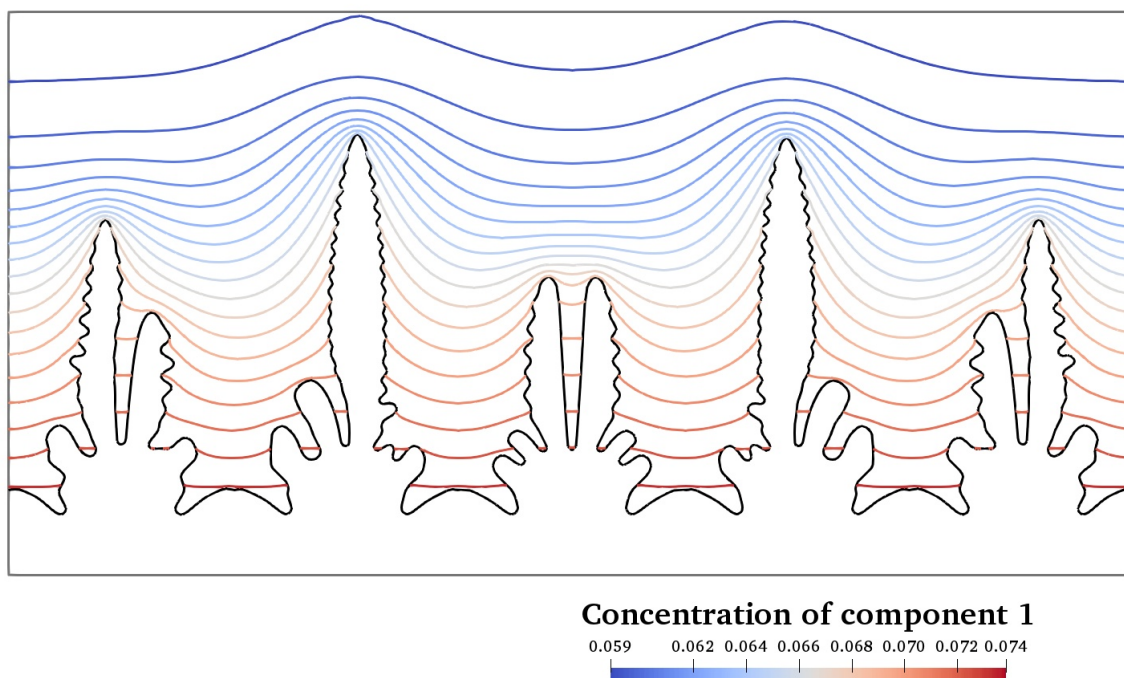


Figure 3.5: Contours for the concentration of component 1 in the liquid for (top) the diffusion-only case and (bottom) the convection-diffusion case with a shear flow.

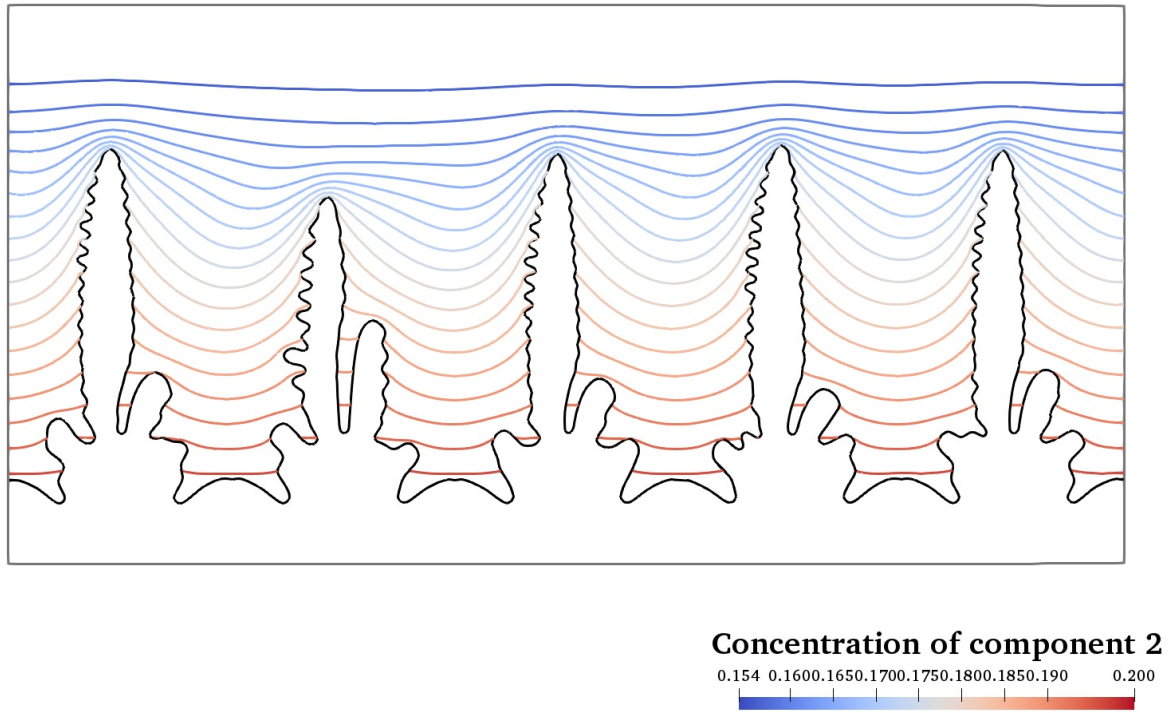
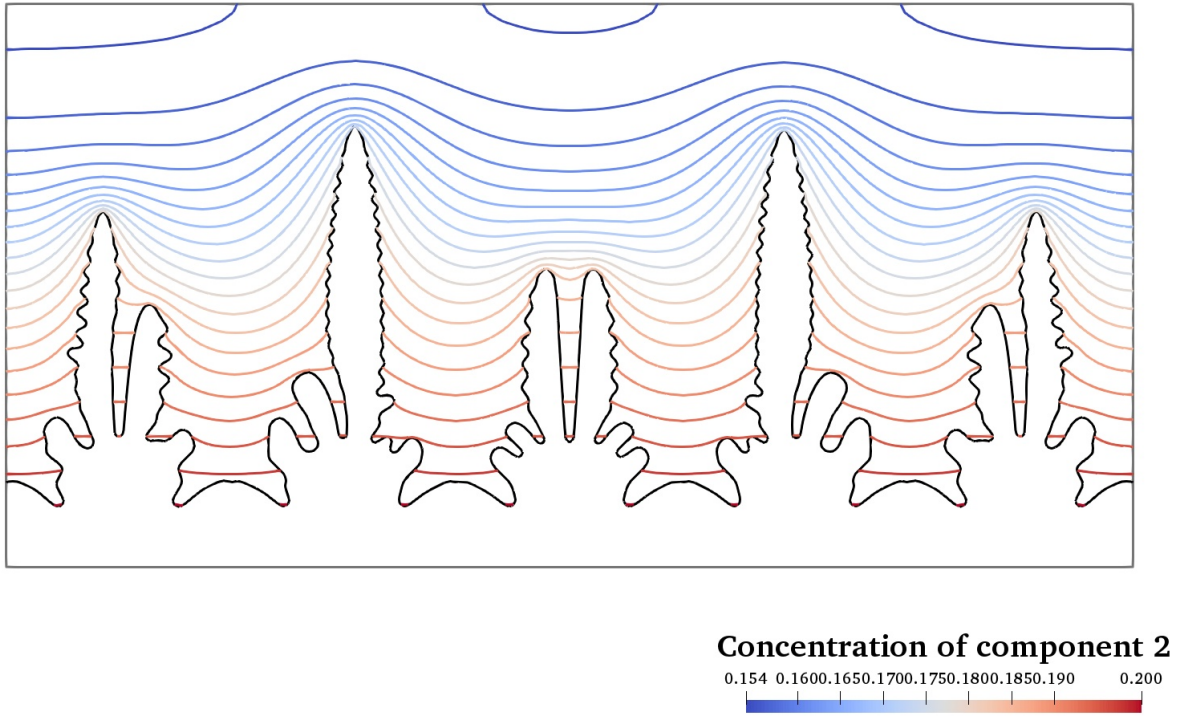


Figure 3.6: Contours for the concentration of component 2 in the liquid for (top) the diffusion-only case and (bottom) the convection-diffusion case with a shear flow.

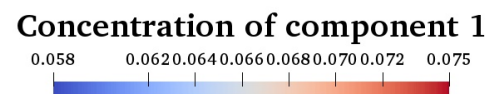
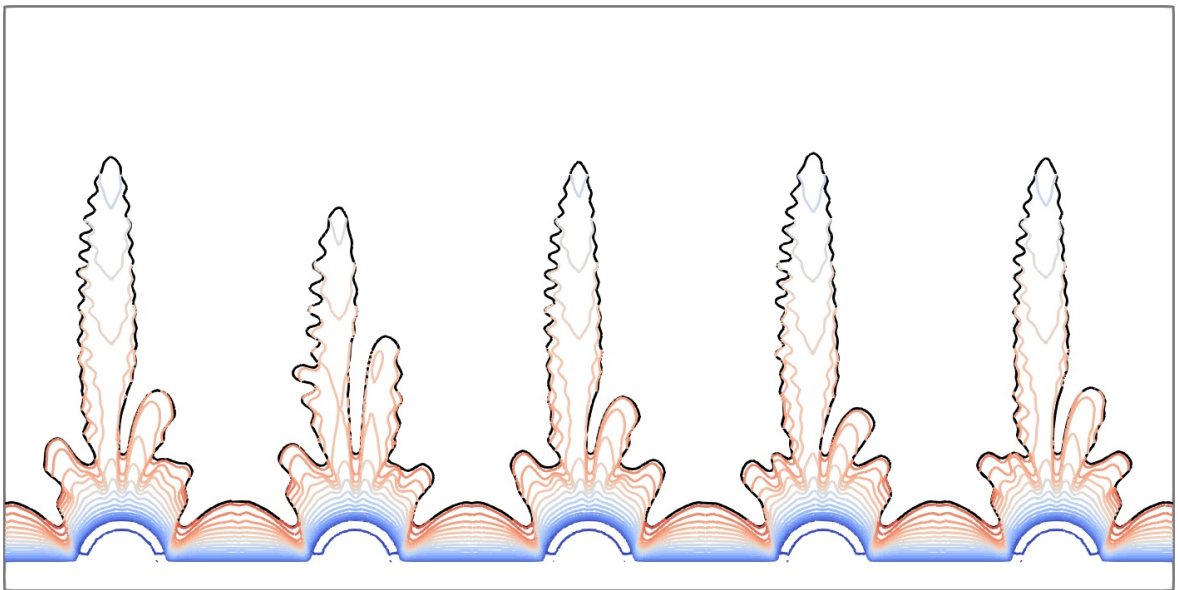
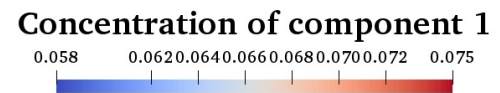
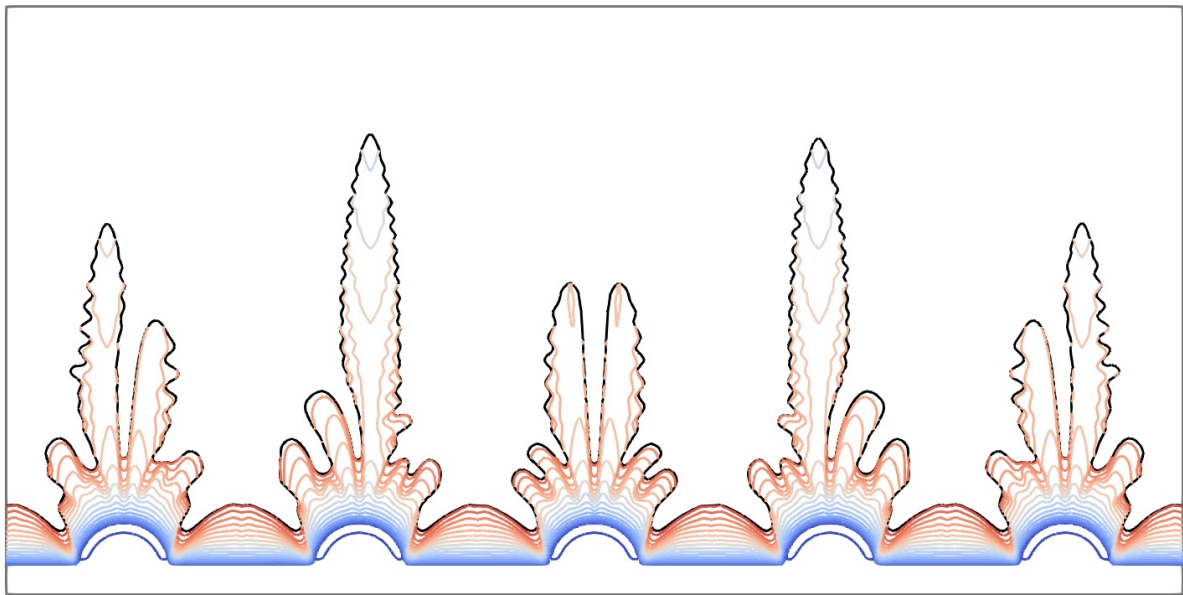


Figure 3.7: Contours for the concentration of component 1 in the solid for (top) the diffusion-only case and (bottom) the convection-diffusion case with a shear flow.

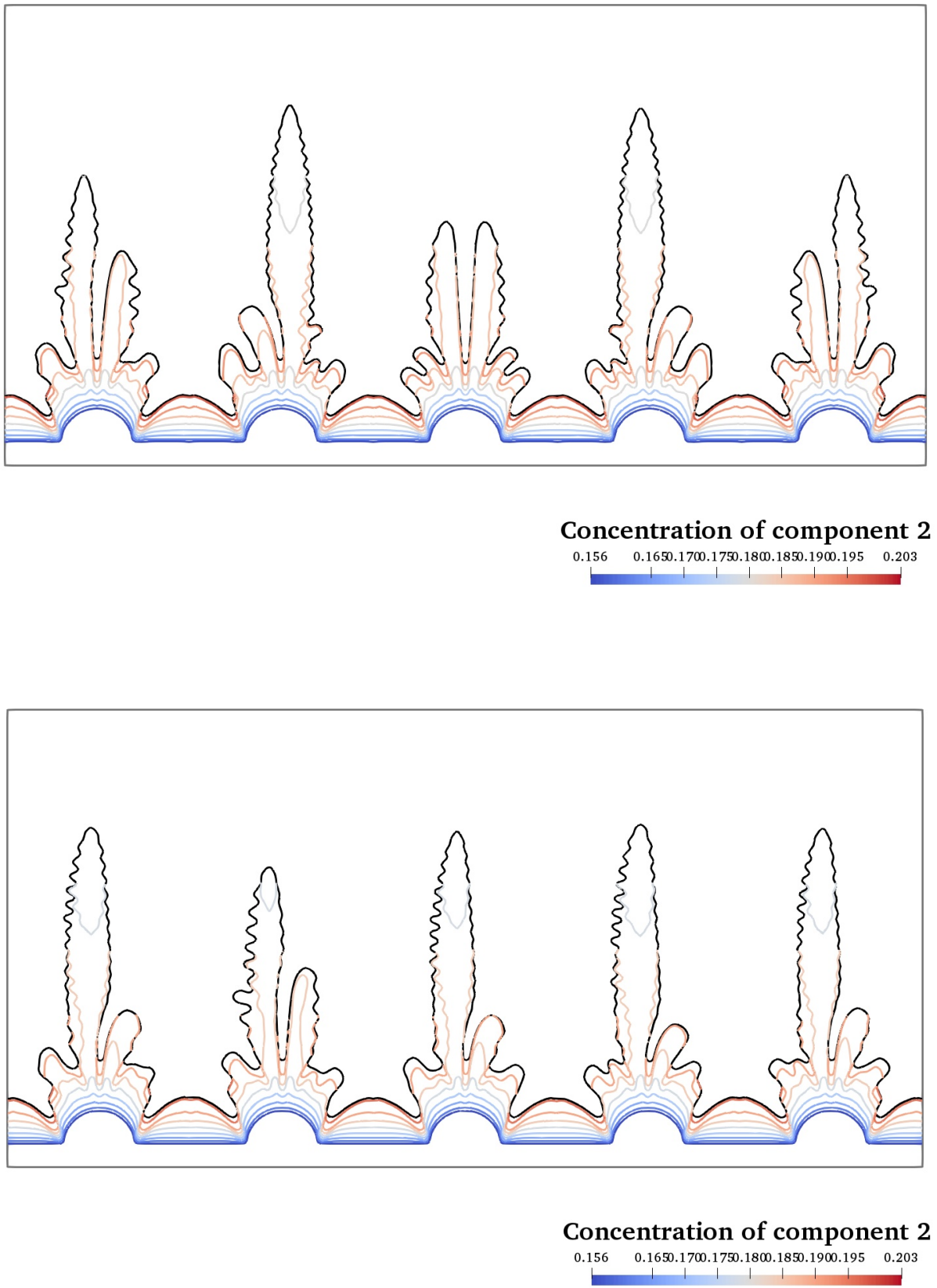


Figure 3.8: Contours for the concentration of component 2 in the solid for (top) the diffusion-only case and (bottom) the convection-diffusion case with a shear flow.

While this set of simulations is mainly a small illustration of the developed method, it illustrates the added value of capturing the convective transport of concentrations and temperatures for alloy solidification applications. For example, the simulations can be leveraged to extract relevant data, such as concentration distributions in the melt and the final solidified material, that may be more difficult to discern experimentally. Future work can include employment of the method to problems of higher complexity (i.e. starting from a planar solidification front rather than from seed regimes), or investigation of parameter regimes relevant for various applications.

3.5 Conclusion

In summary, we have developed a numerical method for the simulation of multi-component alloy solidification including the consideration of convective effects. This work is achieved by combining the numerical framework developed by Bayat *et al.*(2) and described in Chapter 1 for solving Stefan-type problems with convective effects with the Newton-type iteration strategy introduced by Bochkov *et al.*(139) for the solution of diffusion-driven multi-component alloy solidification. The method is evaluated with a numerical convergence test, and performs with between 1st and 2nd order convergence for all fields. Lastly, in Sec. 3.4.2, we utilize the method for proof-of-concept simulations of directional solidification of a ternary alloy in both the diffusion-only and convection-diffusion cases. Future work might consist of applying the developed numerical tool to study relevant multi-component alloy solidification problems of interest. This could include studying the effects of both forced and natural convection on resulting solidification regimes. For example, it would be interesting to investigate the impact of convective effects on the transition from cellular to dendritic solidification regimes, and when and how the transition takes place. Another interesting avenue of study could be natural

convection regimes that arise in solidification contexts, such as that explored in (143).

Bibliography

- [1] S. Molins, C. Soulaine, N. I. Prasianakis, A. Abbasi, P. Poncet, A. J. Ladd, V. Starchenko, S. Roman, D. Trebotich, H. A. Tchelepi, *et. al.*, *Simulation of mineral dissolution at the pore scale with evolving fluid-solid interfaces: Review of approaches and benchmark problem set*, *Computational Geosciences* **25** (2021), no. 4 1285–1318.
- [2] E. Bayat, R. Egan, D. Bochkov, A. Sauret, and F. Gibou, *A sharp numerical method for the simulation of stefan problems with convective effects*, *Journal of Computational Physics* **471** (2022) 111627.
- [3] L. Ristroph, M. N. Moore, S. Childress, M. J. Shelley, and J. Zhang, *Sculpting of an erodible body by flowing water*, *Proceedings of the National Academy of Sciences* **109** (2012), no. 48 19606–19609.
- [4] L. Ristroph, *Sculpting with flow*, *Journal of Fluid Mechanics* **838** (2018) 1–4.
- [5] J. Mac Huang, M. N. J. Moore, and L. Ristroph, *Shape dynamics and scaling laws for a body dissolving in fluid flow*, *Journal of Fluid Mechanics* **765** (2015).
- [6] J. M. Huang, M. J. Shelley, and D. B. Stein, *A stable and accurate scheme for solving the stefan problem coupled with natural convection using the immersed boundary smooth extension method*, *Journal of Computational Physics* **432** (2021) 110162.
- [7] Z. Wang, E. Calzavarini, C. Sun, and F. Toschi, *How the growth of ice depends on the fluid dynamics underneath.*, *Proceedings of the National Academy of Sciences of the United States of America* **118** (2021), no. 10.
- [8] M. Epstein and F. B. Cheung, *Complex freezing-melting interfaces in fluid flow*, *Annual Review of Fluid Mechanics* **15** (1983), no. 1 293–319.
- [9] T. Hirata, K. Nagasaka, and M. Ishikawa, *Crystal ice formation of solution and its removal phenomena at cooled horizontal solid surface. part i : ice removal phenomena*, *International Journal of Heat and Mass Transfer* **43** (2000), no. 3 333–339.

- [10] B. Melissari and S. A. Argyropoulos, *The identification of transition convective regimes in liquid metals using a computational approach*, *Progress in Computational Fluid Dynamics, an International Journal* **4** (2004), no. 2 69–77.
- [11] A. Kumar and S. Roy, *Heat transfer characteristics during melting of a metal spherical particle in its own liquid*, *International journal of thermal sciences* **49** (2010), no. 2 397–408.
- [12] F. Gallaire and P.-T. Brun, *Fluid dynamic instabilities: theory and application to pattern forming in complex media*, *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* **375** (2017), no. 2093 20160155.
- [13] M. N. J. Moore, *Riemann-hilbert problems for the shapes formed by bodies dissolving, melting, and eroding in fluid flows*, *Communications on Pure and Applied Mathematics* **70** (2017), no. 9 1810–1831, [<https://onlinelibrary.wiley.com/doi/pdf/10.1002/cpa.21689>].
- [14] S.-H. Chiu, M. N. J. Moore, and B. Quaife, *Viscous transport in eroding porous media*, *Journal of Fluid Mechanics* **893** (2020).
- [15] U. Telgmann, H. Horn, and E. Morgenroth, *Influence of growth history on sloughing and erosion from biofilms*, *Water Research* **38** (2004), no. 17 3671–3684.
- [16] J. Langer, *Instability and pattern formation in crystal growth*, *Rev. of Mod. Phys.* **52** (1980) 1–28.
- [17] A. Karma and W.-J. Rappel, *Quantitative phase-field modeling of dendritic growth in two and three dimensions*, *Phys. Rev. E* **57** (1997) 4323–4349.
- [18] B. Nestler, D. Danilov, and P. Galenko, *Crystal growth of pure substances: Phase-field simulations in comparison with analytical and experimental results*, *J. Comput. Phys.* **207** (2005) 221–239.
- [19] A. Karma and W.-J. Rappel, *Phase-field modeling method for computationally efficient modeling of solidification with arbitrary interface kinetics*, *Phys. Rev. E* **53** (1996).
- [20] K. Elder, M. Grant, N. Provatas, and J. Kosterlitz, *Sharp interface limits of phase-field models*, *SIAM J. Appl. Math* **64** (2001) 021604.
- [21] W. J. Boettinger, J. A. Warren, C. Beckermann, , and A. Karma, *Phase-field simulations of solidification*, *Ann. Rev. Mater. Res.* **32** (2002) 163–194.
- [22] D. Benson, *Computational methods in Lagrangian and Eulerian hydrocodes*, *Comput. Meth. in Appl. Mech. and Eng.* **99** (1992) 235–394.

- [23] D. Benson, *Volume of fluid interface reconstruction methods for multimaterial problems*, *Applied Mechanics Reviews* **52** (2002) 151–165.
- [24] R. DeBar, *Fundamentals of the KRAKEN code*, tech. rep., Lawrence Livermore National Laboratory (UCID- 17366), 1974.
- [25] W. Noh and P. Woodward, *SLIC (simple line interface calculation)*, in *5th International Conference on Numerical Methods in Fluid Dynamics*, pp. 330–340, 1976.
- [26] D. Youngs, *An interface tracking method for a 3D Eulerian hydrodynamics code*, tech. rep., AWRE (44/92/35), 1984.
- [27] N. Al-Rawahi, *Numerical Simulation of Dendritic Solidification with Convection: Two-Dimensional Geometry*, *Journal of Computational Physics* **180** (Aug., 2002) 471–496.
- [28] T. Myers and S. L. Mitchell, *Application of the combined integral method to stefan problems*, *Applied Mathematical Modelling* **35** (2011), no. 9 4281–4294.
- [29] G. Beckett, J. A. Mackenzie, and M. L. Robertson, *A moving mesh finite element method for the two-dimensional stefan problems*, *Journal of Computational Physics* **168** (2001), no. 2 500–518.
- [30] E. Javierre, C. Vuik, F. J. Vermolen, and S. van der Zwaag, *A comparison of numerical models for one-dimensional stefan problems*, *Journal of Computational and Applied Mathematics* **192** (2006), no. 2 445–459.
- [31] A. I. Fedoseyev and J. D. Alexander, *An inverse finite element method for pure and binary solidification problems*, *Journal of Computational Physics* **130** (1997), no. 2 243 – 255.
- [32] M. L. Bars and M. G. Worster, *Solidification of a binary alloy: Finite-element, single-domain simulation and new benchmark solutions*, *Journal of Computational Physics* **216** (2006), no. 1 247 – 263.
- [33] S. Chen, B. Merriman, S. Osher, and P. Smereka, *A simple level set method for solving stefan problems*, *Journal of Computational Physics* **135** (1997), no. 1 8–29.
- [34] Y. Yang and H. Udaykumar, *Sharp interface cartesian grid method iii: solidification of pure materials and binary solutions*, *Journal of Computational Physics* **210** (2005), no. 1 55–74.
- [35] F. Gibou and R. Fedkiw, *A fourth order accurate discretization for the Laplace and heat equations on arbitrary domains, with applications to the Stefan problem*, *J. Comput. Phys.* **202** (2005), no. 2 577 – 601.

- [36] F. Gibou, R. Fedkiw, R. Caflisch, and S. Osher, *A level set approach for the numerical simulation of dendritic growth*, *J. Sci. Comput.* **19** (2003) 183–199.
- [37] H. Chen, C. Min, and F. Gibou, *A numerical scheme for the Stefan problem on adaptive Cartesian grids with supralinear convergence rate*, *J. Comput. Phys.* **228** (2009), no. 16 5803–5818.
- [38] N. Zabaras, B. Ganapathysubramanian, and L. Tan, *Modelling dendritic solidification with melt convection using the extended finite element method*, *Journal of Computational Physics* **218** (Oct., 2006) 200–227.
- [39] H. Udaykumar, S. Marella, and S. Krishnan, *Sharp-interface simulation of dendritic growth with convection: benchmarks*, *International journal of heat and mass transfer* **46** (2003), no. 14 2615–2627.
- [40] T. V. Vu, A. V. Truong, N. T. Hoang, and D. K. Tran, *Numerical investigations of solidification around a circular cylinder under forced convection*, *Journal of Mechanical Science and Technology* **30** (2016), no. 11 5019–5028.
- [41] T. V. Vu and J. C. Wells, *Numerical simulations of solidification around two tandemly-arranged circular cylinders under forced convection*, *International Journal of Multiphase Flow* **89** (2017) 331–344.
- [42] T. V. Vu, *Fully resolved simulations of drop solidification under forced convection*, *International Journal of Heat and Mass Transfer* **122** (2018) 252–263.
- [43] F. Gibou, C. Min, and R. Fedkiw, *High resolution sharp computational methods for elliptic and parabolic problems in complex geometries*, *J. Sci. Comput.* **54** (2013) 369–413.
- [44] A. Helgadóttir and F. Gibou, *A Poisson–Boltzmann solver on irregular domains with Neumann or Robin boundary conditions on non-graded adaptive grid*, *Journal of Computational Physics* **230** (May, 2011) 3830–3848.
- [45] C. Min and F. Gibou, *A second order accurate level set method on non-graded adaptive Cartesian grids*, *J. Comput. Phys.* **225** (2007) 300–321.
- [46] M. Mirzadeh, M. Theillard, and F. Gibou, *A Second-Order Discretization of the Nonlinear Poisson–Boltzmann Equation over Irregular Geometries using Non-Graded Adaptive Cartesian Grids*, *Journal of Computational Physics* **230** (Dec., 2010) 2125–2140.
- [47] M. Mirzadeh and F. Gibou, *A conservative discretization of the Poisson–Nernst–Planck equations on adaptive cartesian grids*, *J. Comput. Phys.* **274** (2014) 633–653.

- [48] F. Gibou, R. P. Fedkiw, L.-T. Cheng, and M. Kang, *A second-order-accurate symmetric discretization of the poisson equation on irregular domains*, *Journal of Computational Physics* **176** (2002), no. 1 205 – 227.
- [49] M. Theillard, F. Gibou, and T. Pollock, *A sharp computational method for the simulation of the solidification of binary alloys*, *J. Sci. Comput.* (2014).
- [50] H. Chen, C. Min, and F. Gibou, *A supra-convergent finite difference scheme for the poisson and heat equations on irregular domains and non-graded adaptive cartesian grids*, *J. Sci. Comput.* **31** (2007) 19–60.
- [51] M. Mirzadeh, M. Theillard, A. Helgadottir, D. Boy, and F. Gibou, *An adaptive, finite difference solver for the nonlinear Poisson–Boltzmann equation with applications to biomolecular computations*, *Communications in Computational Physics* **13** (2012), no. 1 150–173.
- [52] J. Papac, F. Gibou, and C. Ratsch, *Efficient symmetric discretization for the Poisson, heat and Stefan-type problems with Robin boundary conditions*, *Journal of Computational Physics* **229** (Feb., 2010) 875–889.
- [53] E. Brun, A. Guittet, and F. Gibou, *A local level-set method using a hash table data structure*, *J. Comp. Phys.* **231** (2012) 2528–2536.
- [54] A. Guittet, M. Theillard, and F. Gibou, *A stable projection method for the incompressible navier–stokes equations on arbitrary geometries and adaptive quad/octrees*, *Journal of Computational Physics* **292** (2015) 215–238.
- [55] R. Egan, A. Guittet, F. Temprano-Coletto, T. Isaac, F. J. Peaudecerf, J. R. Landel, P. Luzzatto-Fegiz, C. Burstedde, and F. Gibou, *Direct numerical simulation of incompressible flows on parallel octree grids*, *Journal of Computational Physics* **428** (2021) 110084.
- [56] W. M. Deen, *Analysis of transport phenomena*, .
- [57] J. A. Dantzig and M. Rappaz, *Solidification: -Revised & Expanded*. EPFL press, 2016.
- [58] S. Osher and J. A. Sethian, *Fronts propagating with curvature dependent speed: algorithms based on Hamilton-Jacobi formulations*, *J. Comput. Phys.* **79** (1988), no. 1 12–49.
- [59] F. Gibou, R. Fedkiw, and S. Osher, *A review of level-set methods and some recent applications*, *Journal of Computational Physics* **353** (2018), no. Supplement C 82 – 109.

- [60] M. Mirzadeh, A. Guittet, C. Burstedde, and F. Gibou, *Parallel level-set methods on adaptive tree-based grids*, *Journal of Computational Physics* **322** (2016) 345–364.
- [61] C. Burstedde, L. C. Wilcox, and O. Ghattas, *p4est: Scalable algorithms for parallel adaptive mesh refinement on forests of octrees*, *SIAM Journal on Scientific Computing* **33** (2011), no. 3 1103–1133.
- [62] M. Sussman, P. Smereka, and S. Osher, *A level set approach for computing solutions to incompressible two-phase flow*, *J. Comput. Phys.* **114** (1994) 146–159.
- [63] G. Russo and P. Smereka, *A remark on computing distance functions*, *Journal of computational physics* **163** (2000), no. 1 51–67.
- [64] C. Min, F. Gibou, and H. D. Ceniceros, *A supra-convergent finite difference scheme for the variable coefficient poisson equation on non-graded grids*, *Journal of Computational Physics* **218** (2006), no. 1 123–140.
- [65] J. Strain, *A fast modular semi-lagrangian method for moving interfaces*, *Journal of Computational Physics* **161** (2000), no. 2 512–536.
- [66] J. Strain, *Tree methods for moving interfaces*, *J. Comput. Phys* (1999).
- [67] C. Min and F. Gibou, *A second order accurate level set method on non-graded adaptive cartesian grids*, *Journal of Computational Physics* **225** (2007), no. 1 300–321.
- [68] C. Min and F. Gibou, *A second order accurate projection method for the incompressible navier–stokes equations on non-graded adaptive grids*, *Journal of Computational Physics* **219** (2006), no. 2 912–929.
- [69] D. Xiu and G. E. Karniadakis, *A semi-lagrangian high-order method for navier–stokes equations*, *Journal of computational physics* **172** (2001), no. 2 658–684.
- [70] H. Chen, C. Min, and F. Gibou, *A supra-convergent finite difference scheme for the poisson and heat equations on irregular domains and non-graded adaptive cartesian grids*, *Journal of Scientific Computing* **31** (2007), no. 1-2 19–60.
- [71] C. Min, F. Gibou, and H. D. Ceniceros, *A supra-convergent finite difference scheme for the variable coefficient poisson equation on non-graded grids*, *J. Comput. Phys.* **218** (2006), no. 1 123–140.
- [72] G. H. Shortley and R. Weller, *The numerical solution of laplace’s equation*, *Journal of Applied Physics* **9** (1938), no. 5 334–348.

- [73] G. Yoon and C. Min, *Convergence analysis of the standard central finite difference method for poisson equation*, *Journal of Scientific Computing* **67** (2016), no. 2 602–617.
- [74] A. J. Chorin, *Numerical solution of the navier-stokes equations*, *Mathematics of computation* **22** (1968), no. 104 745–762.
- [75] D. Brown, R. Cortez, and M. Minion, *Accurate projection methods for the incompressible navier–stokes equations*, *Journal of Computational Physics* **168** (04, 2001) 464–499.
- [76] J. Kim and P. Moin, *Application of a fractional-step method to incompressible navier-stokes equation*, .
- [77] F. Losasso, R. Fedkiw, and S. Osher, *Spatially Adaptive Techniques for Level Set Methods and Incompressible Flow*, *Computers and Fluids* **35** (2006) 995–1010.
- [78] F. Losasso, F. Gibou, and R. Fedkiw, *Simulating water and smoke with an octree data structure*, *ACM Trans. Graph. (SIGGRAPH Proc.)* (2004) 457–462.
- [79] D. Bochkov and F. Gibou, *Multidimensional extrapolation over interfaces with kinks and regions of high curvatures*, *arXiv preprint arXiv:1912.09559* (2019).
- [80] T. D. Aslam, *A partial differential equation approach to multidimensional extrapolation*, *Journal of Computational Physics* **193** (2004), no. 1 349–355.
- [81] M. Okada, K. Katayama, K. Terasaki, M. Akimoto, and K. Mabune, *Freezing around a cooled pipe in crossflow*, *Bulletin of JSME* **21** (1978), no. 160 1514–1520.
- [82] K. Cheng, H. Inaba, and R. Gilpin, *An experimental investigation of ice formation around an isothermally cooled cylinder in crossflow*, .
- [83] H. Perkins Jr and G. Leppert, *Local heat-transfer coefficients on a uniformly heated cylinder*, *International journal of heat and mass transfer* **7** (1964), no. 2 143–158.
- [84] M. Laroussi, M. Djebbi, and M. Moussa, *Triggering vortex shedding for flow past circular cylinder by acting on initial conditions: A numerical study*, *Computers & Fluids* **101** (2014) 194–207.
- [85] Y. Hao and Y.-X. Tao, *Heat transfer characteristics of melting ice spheres under forced and mixed convection*, *J. Heat Transfer* **124** (2002), no. 5 891–903.
- [86] J. Hureau, E. Brunon, and P. Legallais, *Ideal free streamline flow over a curved obstacle*, *Journal of computational and applied mathematics* **72** (1996), no. 1 193–214.

- [87] S. Alben, M. Shelley, and J. Zhang, *How flexibility induces streamlining in a two-dimensional flow*, *Physics of Fluids* **16** (2004), no. 5 1694–1713.
- [88] M. N. Moore, L. Ristroph, S. Childress, J. Zhang, and M. J. Shelley, *Self-similar evolution of a body eroding in a fluid flow*, *Physics of Fluids* **25** (2013), no. 11 116602.
- [89] L. Chen, Q. Kang, and W. Tao, *Pore-scale study of reactive transport processes in catalyst layer agglomerates of proton exchange membrane fuel cells*, *Electrochimica acta* **306** (2019) 454–465.
- [90] Y. Tang, A. J. Valocchi, C. J. Werth, and H. Liu, *An improved pore-scale biofilm model and comparison with a microfluidic flow cell experiment*, *Water Resources Research* **49** (2013), no. 12 8370–8382.
- [91] Q. Kang, P. C. Lichtner, H. S. Viswanathan, and A. I. Abdel-Fattah, *Pore scale modeling of reactive transport involved in geologic co₂ sequestration*, *Transport in porous media* **82** (2010), no. 1 197–213.
- [92] T. A. Jones and R. L. Detwiler, *Mineral precipitation in fractures: Using the level-set method to quantify the role of mineral heterogeneity on transport properties*, *Water Resources Research* **55** (2019), no. 5 4186–4206.
- [93] A. J. Ladd and P. Szymczak, *Reactive flows in porous media: challenges in theoretical and numerical methods*, *Annual review of chemical and biomolecular engineering* **12** (2021) 543–571.
- [94] N. Seigneur, K. U. Mayer, and C. I. Steefel, *Reactive transport in evolving porous media*, *Reviews in Mineralogy and Geochemistry* **85** (2019), no. 1 197–238.
- [95] C. Noiriel, C. I. Steefel, L. Yang, and J. Ajo-Franklin, *Upscaling calcium carbonate precipitation rates from pore to continuum scale*, *Chemical Geology* **318** (2012) 60–74.
- [96] L. E. Beckingham, C. I. Steefel, A. M. Swift, M. Voltolini, L. Yang, L. M. Anovitz, J. M. Sheets, D. R. Cole, T. J. Kneafsey, E. H. Mitnick, *et. al.*, *Evaluation of accessible mineral surface areas for improved prediction of mineral reaction rates in porous media*, *Geochimica et Cosmochimica Acta* **205** (2017) 31–49.
- [97] X. Li, H. Huang, and P. Meakin, *Level set simulation of coupled advection-diffusion and pore structure evolution due to mineral precipitation in porous media*, *Water Resources Research* **44** (2008), no. 12.
- [98] X. Li, H. Huang, and P. Meakin, *A three-dimensional level set simulation of coupled reactive transport and precipitation/dissolution*, *International Journal of Heat and Mass Transfer* **53** (2010), no. 13-14 2908–2923.

- [99] M. Vu and P. Adler, *Application of level-set method for deposition in three-dimensional reconstructed porous media*, *Physical Review E* **89** (2014), no. 5 053301.
- [100] Q. Kang, D. Zhang, and S. Chen, *Simulation of dissolution and precipitation in porous media*, *Journal of Geophysical Research: Solid Earth* **108** (2003), no. B10.
- [101] Z. Xu and P. Meakin, *Phase-field modeling of solute precipitation and dissolution*, *The Journal of chemical physics* **129** (2008), no. 1.
- [102] Z. Xu and P. Meakin, *Phase-field modeling of two-dimensional solute precipitation/dissolution: Solid fingers and diffusion-limited precipitation*, *The Journal of chemical physics* **134** (2011), no. 4.
- [103] A. M. Tartakovsky, P. Meakin, T. D. Scheibe, and B. D. Wood, *A smoothed particle hydrodynamics model for reactive transport and mineral precipitation in porous and fractured porous media*, *Water resources research* **43** (2007), no. 5.
- [104] A. M. Tartakovsky, P. Meakin, T. D. Scheibe, and R. M. E. West, *Simulations of reactive transport and precipitation with smoothed particle hydrodynamics*, *Journal of Computational Physics* **222** (2007), no. 2 654–672.
- [105] Q. Kang, I. N. Tsimpanogiannis, D. Zhang, and P. C. Lichtner, *Numerical modeling of pore-scale phenomena during co2 sequestration in oceanic sediments*, *Fuel processing technology* **86** (2005), no. 14-15 1647–1665.
- [106] C. Soulaire and H. A. Tchelepi, *Micro-continuum approach for pore-scale simulation of subsurface processes*, *Transport in porous media* **113** (2016) 431–456.
- [107] F. Dutka, V. Starchenko, F. Osselin, S. Magni, P. Szymczak, and A. J. Ladd, *Time-dependent shapes of a dissolving mineral grain: Comparisons of simulations with microfluidic experiments*, *Chemical Geology* **540** (2020) 119459.
- [108] N. Zabaras and D. Samanta, *A stabilized volume-averaging finite element method for flow in porous media and binary alloy solidification processes*, *International Journal for Numerical Methods in Engineering* **60** (2004), no. 6 1103–1138.
- [109] D. Bochkov and F. Gibou, *Solving poisson-type equations with robin boundary conditions on piecewise smooth interfaces*, *Journal of Computational Physics* **376** (2019) 1156–1198.
- [110] L. Zhao, X. Bai, T. Li, and J. Williams, *Improved conservative level set method*, *International Journal for Numerical Methods in Fluids* **75** (2014), no. 8 575–590.
- [111] H. Fazeli, M. Masoudi, R. A. Patel, P. Aagaard, and H. Hellevang, *Pore-scale modeling of nucleation and growth in porous media*, *ACS Earth and Space Chemistry* **4** (2020), no. 2 249–260.

- [112] J. Ahrens, B. Geveci, C. Law, C. Hansen, and C. Johnson, *Paraview: An end-user tool for large-data visualization*, *The visualization handbook* **717** (2005) 50038–1.
- [113] C. Soullaine, S. Roman, A. Kovscek, and H. A. Tchelepi, *Mineral dissolution and wormholing from a pore-scale perspective*, *Journal of Fluid Mechanics* **827** (2017) 457–483.
- [114] G. Liu, L. Liu, Z.-H. Han, G.-J. Zhang, and J. Zhang, *Solidification behavior of re-and ru-containing ni-based single-crystal superalloys with thermal and metallographic analysis*, *Rare Metals* **36** (2017) 792–798.
- [115] X.-D. Yue, J.-R. Li, and X.-G. Wang, *The microstructure of a single crystal superalloy after different aging heat treatments*, *Rare Metals* **37** (2018) 210–216.
- [116] A. Bogno, H. Nguyen-Thi, A. Buffet, G. Reinhart, B. Billia, N. Mangelinck-Noël, N. Bergeon, J. Baruchel, and T. Schenk, *Analysis by synchrotron x-ray radiography of convection effects on the dynamic evolution of the solid–liquid interface and on solute distribution during the initial transient of solidification*, *Acta Materialia* **59** (2011), no. 11 4356–4365.
- [117] J. E. Spinelli, D. M. Rosa, I. L. Ferreira, and A. Garcia, *Influence of melt convection on dendritic spacings of downward unsteady-state directionally solidified al–cu alloys*, *Materials Science and Engineering: A* **383** (2004), no. 2 271–282.
- [118] T. Pollock and W. Murphy, *The breakdown of single-crystal solidification in high refractory nickel-base alloys*, *Metallurgical and Materials Transactions A* **27** (1996) 1081–1094.
- [119] W. Zhang and L. Liu, *Solidification microstructure of directionally solidified superalloy under high temperature gradient*, *Rare Metals* **31** (2012) 541–546.
- [120] N. Shevchenko, S. Boden, G. Gerbeth, and S. Eckert, *Chimney formation in solidifying ga-25wt pct in alloys under the influence of thermosolutal melt convection*, *Metallurgical and Materials Transactions A* **44** (2013) 3797–3808.
- [121] A. Clarke, D. Tourret, Y. Song, S. Imhoff, P. Gibbs, J. Gibbs, K. Fezzaa, and A. Karma, *Microstructure selection in thin-sample directional solidification of an al-cu alloy: In situ x-ray imaging and phase-field simulations*, *Acta Materialia* **129** (2017) 203–216.
- [122] E. W. Hester, L.-A. Couston, B. Favier, K. J. Burns, and G. M. Vasil, *Improved phase-field models of melting and dissolution in multi-component flows*, *Proceedings of the Royal Society A* **476** (2020), no. 2242 20200508.

- [123] J. Ramirez and C. Beckermann, *Examination of binary alloy free dendritic growth theories with a phase-field model*, *Acta Materialia* **53** (2005), no. 6 1721–1736.
- [124] W. J. Boettinger and J. A. Warren, *Simulation of the cell to plane front transition during directional solidification at high velocity*, *Journal of crystal growth* **200** (1999), no. 3-4 583–591.
- [125] A. Karma, *Phase-field formulation for quantitative modeling of alloy solidification*, *Physical review letters* **87** (2001), no. 11 115701.
- [126] J. Ramirez, C. Beckermann, A. Karma, and H.-J. Diepers, *Phase-field modeling of binary alloy solidification with coupled heat and solute diffusion*, *Physical Review E* **69** (2004), no. 5 051607.
- [127] S. G. Kim, *A phase-field model with antitrapping current for multicomponent alloys with arbitrary thermodynamic properties*, *Acta Materialia* **55** (2007), no. 13 4391–4399.
- [128] M. Ohno and K. Matsuura, *Quantitative phase-field modeling for dilute alloy solidification involving diffusion in the solid*, *Physical Review E* **79** (2009), no. 3 031603.
- [129] V. R. Voller and C. Swaminathan, *Eral source-based method for solidification phase change*, *Numerical Heat Transfer, Part B Fundamentals* **19** (1991), no. 2 175–189.
- [130] R. Almgren, *Variational algorithms and pattern formation in dendritic solidification*, *Journal of computational physics* **106** (1993), no. 2 337–354.
- [131] V. R. Voller, *An enthalpy method for modeling dendritic growth in a binary alloy*, *International Journal of Heat and Mass Transfer* **51** (2008), no. 3-4 823–834.
- [132] A. Jakhar, A. Bhattacharya, P. Rath, and S. K. Mahapatra, *Effect of thermal anisotropy on binary alloy dendrite growth*, *International Journal of Heat and Mass Transfer* **127** (2018) 1114–1127.
- [133] S. Karagadde, A. Bhattacharya, G. Tomar, and P. Dutta, *A coupled vof-ibm-enthalpy approach for modeling motion and growth of equiaxed dendrites in a solidifying melt*, *Journal of Computational Physics* **231** (2012), no. 10 3987–4000.
- [134] D. Juric and G. Tryggvason, *A front-tracking method for dendritic solidification*, *Journal of computational physics* **123** (1996), no. 1 127–148.
- [135] N. Al-Rawahi and G. Tryggvason, *Numerical simulation of dendritic solidification with convection: Three-dimensional flow*, *Journal of Computational Physics* **194** (2004), no. 2 677–696.

- [136] N. Al-Rawahi and G. Tryggvason, *Numerical simulation of dendritic solidification with convection: two-dimensional geometry*, *Journal of Computational Physics* **180** (2002), no. 2 471–496.
- [137] L. Tan and N. Zabaras, *A level set simulation of dendritic solidification with combined features of front-tracking and fixed-domain methods*, *Journal of Computational Physics* **211** (2006), no. 1 36–63.
- [138] T. Dobravec, B. Mavrič, and B. Šarler, *A cellular automaton–finite volume method for the simulation of dendritic and eutectic growth in binary alloys using an adaptive mesh refinement*, *Journal of computational Physics* **349** (2017) 351–375.
- [139] D. Bochkov, T. Pollock, and F. Gibou, *Sharp-interface simulations of multicomponent alloy solidification*, *arXiv preprint arXiv:2112.08650* (2021).
- [140] F. Gibou, R. P. Fedkiw, L.-T. Cheng, and M. Kang, *A second-order-accurate symmetric discretization of the poisson equation on irregular domains*, *Journal of Computational Physics* **176** (2002), no. 1 205–227.
- [141] D. Bochkov and F. Gibou, *Solving elliptic interface problems with jump conditions on cartesian grids*, *Journal of Computational Physics* **407** (2020) 109269.
- [142] J. Cui, B. Li, Z. Liu, F. Qi, B. Zhang, and J. Zhang, *Numerical investigation of segregation evolution during the vacuum arc remelting process of ni-based superalloy ingots*, *Metals* **11** (2021), no. 12 2046.
- [143] V. Kumar, A. Srivastava, and S. Karagadde, *Role of microstructure and composition on natural convection during ternary alloy solidification*, *Journal of Fluid Mechanics* **913** (2021) A41.