LARGE SCALE SIMULATION OF MICROSTRUCTURAL EVOLUTION IN ALLOY SOLIDIFICATION

A Thesis

Presented to

The Faculty of the Department of Mechanical Engineering

California State University, Los Angeles

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

in

Mechanical Engineering

By

Nicholas Anthony Cusato

May 2023

© 2023

Nicholas Anthony Cusato

ALL RIGHTS RESERVED

The thesis of Nicholas Anthony Cusato is approved.

Mohsen Eshraghi, Committee Chair

Sergio Felicelli

Mario Medina

David Raymond, Department Chair

California State University, Los Angeles

May 2023

ABSTRACT

Large-Scale Simulation of Microstructural Evolution in Alloy Solidification

By

Nicholas Anthony Cusato

During the past two decades, researchers have shown interest in large-scale simulations to analyze alloy solidification. These simulations provide the basis for understanding important industrial applications using various numerical schemes to solve for two main phenomena: transport and morphology evolution. Transport phenomena that consist of heat transfer, diffusion, and fluid flow whereas the evolution of morphology needs models to capture the solid/liquid interface during solidification. Regardless of the technique, understanding complex, real-world phenomena requires large-scale simulations that can be achieved using parallelization techniques, such as MPI and CUDA. This thesis outlines the advances in both mathematical models and computational hardware/software that enable large-scale simulations of solidification microstructure. The secondary purpose of this thesis is to present a solution for measuring permeability, as well as performing simulations of large-scale. The lattice Boltzmann method was utilized to simulate the fluid flow in order to measure the permeability of the structure. Phase field method was used to capture the solidification morphology while finite difference method was used to solve for solute diffusion and the phase field equations. The simulations provide greater insight into higher-order branching and the parameters required to scale performance. In addition, methods were developed for importing 3D experimental or simulation morphologies into the code for permeability calculations.

ACKNOWLEDGMENTS

I would like to thank Dr. Eshraghi for his encouragement and guidance during my research at Cal State LA. I would like to thank Dr. Nabavizadeh for the tutorship, dedicating his time teaching me the practice of large-scale simulations. I would like to thank my wife Dana for all the late nights and early mornings supporting my work. I would like to thank the National Aeronautics and Space Administration (NASA) who partially funded this work through Grant Number NNX16AT75G. I would like to thank the Centers of Research Excellence in Science and Technology (CREST) for their fellowship and support who has funded this work by both NSF CREST-CEaS, using NSF award No. HRD-1547723, and NSF CREST-CATSUS, using NSF award No. HRD-2112554.

TABLE OF CONTENTS

Abstract iv

Acknowledgments vi

List of Tables ix

List of Figures x

Chapter

1. Introduction 1

Motivation 5

Background 6

Methods for Simulating the Morphology 9

Cellular Automation 14

Phase-field 16

Direct-Interface-Tracking 19

Level-set 20

Dendritic Needle Network 22

Features and Applications 26

Dendrite Interactions and Competitive Growth 31

Columnar to Equiaxed Transition 31

Solute Transport and Segregation 35

Natural and Forced Convection 42

Permeability 44

Additive Manufacturing Processes 48

2. Methodology 53

Finite Element/ Difference/ Volume....54
Lattice Boltzmann....59
Computational Approaches....66
3. Results and Discussion....69
Simulating Dendritic Growth....69
Results....69
Discussion....79
Optimizing Computations....80
Results....80
Discussion....81
Measuring Permeability....82
Results....82
Discussion....83
4. Conclusion....89
Future Developments....91

References....94

Appendices

A. Code A 1....134
B. Code A 2....138

LIST OF TABLES

Table

1. Simulation Parameters Comparison.....67
2. Al-3wt%Cu material properties 69
3. Large Domain Dataset Comparison of COMSOL and LBM 84
4. Small Domain Dataset Comparison of COMSOL and LBM 86

LIST OF FIGURES

Figure

1. 3D model of dendrite interaction showing tertiary arm growth for unfavorably oriented dendrite 7
2. (1) Comparison of pseudo-binary model (a) and MC model (b); (2) 3D Simulation of 99 nucleation seeds; (3) 2D evaluation of solute concentration 8
3. Speedups of single CPU core serial-programming vs multi-GPU parallel-programming efficiency 9
4. Large scale domain of dendrite growth with enlarged composition analysis 11
5. 3D columnar dendritic microstructure 12
6. Hybrid-YZ (Top) and Hybrid-Y (Bottom) protocols for parallelization..... 13
7. Directional solidification of Fe0.6wt.%C alloy in a domain consisting of 400x100x1000 cells and a grid size of 5 μm 16
8. Columnar dendrite growth where inclination angle is indicated by color at the following time steps in order: 5.4 s, 10.7 s, 53.6 s, and 107.1 s 18
9. Evolution of directional solidification with thermal gradient after (a) 2 s, (b) 3.2 s, (c) and (d) 12 s, (a-c) shows grain boundaries and (d) solute map 20
10. Solute Concentration of Ta without (left) and with convection (middle) and 3D crystal growth with front tracking and required undercooling for nucleation 7k (right) 22
11. Comparison of simulation to polished microgravity sample 24
12. Comparison of model simulation types 25

13. Dendrite Interactions comparing different angles of UO dendrites of 5˚ and 15˚ at time (a) 1.3s and (b) 187.5s 29

14. Voronoi grain from Meso-scale simulation 30

15. Evolution of (a) equiaxed (b) columnar (c) mixed dendrites and (d) solidification map for Inconel 718 alloy 32

16. Simulations of CET evolution comparing 2D (a) and (c) and 3D (b) and (d) during the unidirectional solidification of alloy 718-5 wt. % Nb 33

17. Undercooling on simulated grain structures range of mean nucleation undercooling (ΔT_N) (a) 1 K; (b) 2 K; (c) 5 K; and (d) 10 K. (V= $5.5 \cdot 10^{-4}$m/s, G= 3000 - 100tK/m 34

18. CET simulation and initial circular germ regime transition to four branches grain....35

19. Tertiary arm dendrites used to predict spacing in 2D 36

20. Dendrite morphologies and solute concentration distributions for 3D (a) and (b) and 2D (c) and (d), comparing different forms of gravitational acceleration 37

21. Chimney formation and solute concentration field during directional solidification of Pb-10 wt% Sn alloy 39

22. Freckle formation for Ga-25 wt% In binary alloy (a) to (c) Simulation, (d) to (f) experimental result, for G=250 K/m in the vertical direction at different times ..41

23. (a) ATP Efficiency comparison; (b) evolution of solute concentration and flow velocities simulated by LB method; (c) parallelization efficiency with scaling GPUs 44

24. (a) Surface-based representation for Al–20 wt.% Cu for Nucleation Density, (N_V=90mm^3) (b) 3D printed dendritic network 46

25. Domain of 1.152 × 1.152 × 0.768 mm^3 with 51 dendrites, extracting red part illustrating arbitrary direction flow 47

26. (a) Comparison of experimental and simulated microstructure by liquid solute concentration in the interfacial cells (b) Simulated melt track showing temperature gradient (top) and cooling rate (bottom) compared with the experimental solidified melting pool region 49

27. (a) 3D simulation orientation map (b) Longitudinal cross-section (c) Evolution of grain structure during a single-track L -PBF AM of 316L- SS at section located at 8 μm below the substrate surface; GB: Grain Boundary 50

28. Simulation of grain structure for multi-layer AM builds with a unidirectional raster pattern. (a) Midsection EBSD; (b) 3D grain structure; (c): 3D pole figure of these simulations.....51

29. Grain size simulation with respect to different localized melt strategies comparing simulation (top) and experimental pole figures (bottom).....52

30. Comparison of identical 2D, FV morphologies of Pb-5 wt% Sn alloy simulations using different time steps: (a) 0.3 ms and (b) 0.8 ms 56

31. The three common FD schemes: Explicit (left), Implicit (middle), and Crank-Nicolson (right).....58

32. 2D simulation of flow melt around solidified dendrites, where colors represent solute concentration and contours represent temperature.....60

33. 1D lattice schemes: D1Q2 (left), D1Q3 (middle), and D1Q5 (right) 63

34. 2D lattice schemes: D2Q4 (left) D2Q5 (middle), and D2Q9 (right).....64

35. 3-D lattice schemes: D3Q15 (left) and D3Q19 (right).....65

36. Simulation of smaller scale showing secondary dendrite arms forming tertiary arm growth.....70

37. Case 1: Visualization of (a) Solute Concentration profile (b) Phi at the 131,172,413th iteration, approximately 16.4 seconds. 72

38. Case 1: 0.2 × 0.2 × 8 mm^3 Visualization of Phi at 262,344,827 iterations (32.8 seconds) (a) Full size (b) Close-up 73

39. Case 1: 0.2 × 0.2 × 8 mm^3 Visualization of Phi at 524,689,654 iterations (79.25 seconds) (a) Full size at upper level of Phi (b) Close-up at lower level of Phi….74

40. Case 2: 181,455,172 iterations (22.7 seconds) comparing (a) phi and (b) full scale C and (c) close-up of concentrations of solute..... 75

41. Case 2: 292,951,724 iterations (36.6 seconds) simulation output of Phi measuring the upper limit and exhibiting tertiary 76

42. Case 2: 415,379,310 iterations (51.9 seconds) of dendrite growth comparing (a) phi, (b) full-size and (c) close-up of solute 77

43. Case 2: 550,924,137 iterations (68.9 seconds) simulation of dendrite growth comparing (a), (b) full-size and (c) close-up of solute 78

44. Case 3: Solid formation (Phi) from seed at 109,310,344 iterations (13.6 seconds).....79

45. Comparing the solute concentration for each of the cases (a) Case 1, (b) Case 2, (c) Case 3 at approximately 16 seconds 80

46. NVIDIA Visual Profiler output for visualizing simulation process 81

47. Permeability visualization of a simple spherical dendrite 82

48. Visualizing permeability through a porous media in the x-direction.....83

49. Permeability Comparison for a Large Domain (L=300 μm) 85

50. Permeability Comparison of COMSOL and LBM for a Small Domain Dataset……...87

51. Dendrite split into segments (top segment isolated) 88

52. “Stl” cat conversion to points for LB flow 89

53. RNN model to predict microstructural evolution in latent space for PF modeling…. 92

LIST OF TERMS

Cellular Automaton (CA)

Phase-field (PF)

Direct-Interface-Tracking (DIT)

Level-Set (LS)

Dendritic Needle Networks (DNN)

Finite Element (FE)

Finite Difference (FD)

Finite Volume (FV)

Lattice-Boltzmann (LB)

Two-dimensional (2D)

Three-dimensional (3D)

Message Passing Interface (MPI)

Compute Unified Device Architecture (CUDA)

Adaptive mesh refinement (AMR)

Columnar-to-equiaxed transition (CET)

Columnar-to-Equiaxed Transition in the Solidification Processing project (CETSOL)

Flux intensity factor (FIF)

Molecular dynamics (MD)

Multicomponent (MC)

Metal droplets deposition method (MDDM)

Secondary dendritic arm spacing (SDAS)

Bhatnagar-Gross-Krook (BGK)

Single Program, Multiple Data Streams (SPMD)

Hierarchical Data Format 5 (HDF5)

Multiphysics Object-Oriented Simulation Environment (MOOSE)

Portable, Extensible Toolkit for Scientific Computation (PETSc)

Mesoscale Microstructural Simulation Project (MMSP)

Structured Adaptive Mesh Refinement Application Infrastructure (SAMRAI)

Additive Manufacturing (AM)

Laser engineered net shaping (LENS)

Electron Beam Selective Melting (EBSM)

Laser powder bed fusion (L-PBF)

Directed energy deposition (DED)

Electron backscattered diffraction (EBSD)

Primary dendrite arm spacing (PDAS)

Gäumann, Trivedi and Kurz (GTK model)

GFLOPS (giga floating-point operations per second)

Favorably oriented (FO)

Unfavorably oriented (UO)

Fourier Transformation (FT)

Voronoi Tessellation (VT)

Selective laser sintering (SLS)

Fused filament fabrication (FFF)

Kozeny-Caman (KC)

Poirier-Heinrich (PH)

Vacuum arc remelting (VAR)

Active Parameter Tracking (APT)

back-propagation neural network (BPNN)

principal component analysis (PCA)

isometric feature mapping (Isomap)

Uniform Manifold Approximation and Projection (UMAP)

gated recurrent unit (GRU)

recurrent neural network (RNN)

long short-term memory (LSTM)

visualization toolkit (vtk)

stereolithography (stl)

Initial Graphics Exchange Specification (IGES)

Solidworks part (sldprt)

TECPLOT (tec)

pickle (pkl)

CHAPTER 1

Introduction

The research in simulation of dendritic growth during alloy solidification used to be limited to small-scale, single dendrites that do not fully capture all of the physics behind solidification. The computation power and cost required for more superb analysis have been the main hindrance for performing more complex, large-scale simulations. This thesis outlines the advances in both mathematical models and computational hardware/software that enable large-scale simulations of solidification microstructure, including features and applications of the simulations. The secondary purpose of this thesis is to present a solution for measuring permeability, as well as performing simulations of large-scale. Discussing optimization and comparison of the techniques to provide a comprehensive understanding of simulating microstructural evolution of alloys.

Large-scale simulations of dendritic growth can greatly benefit manufacturing and materials engineers and scientists offering a detailed understanding of microstructural evolution during solidification processes. Reducing the trial and error of the manufacturing process, the simulations save time, giving greater understanding to the microstructure and defect formation. Many advances in in-situ X-ray observations of solidification have identified areas that can benefit from large-scale realistic modeling of dendrites Lee and Hunt [1] first examined slabs of Al-Cu alloys for hydrogen pores in situ during directional solidification (DS). They discovered that as the growth speed of dendrites increases, along with hydrogen content, the pores' shape evolves into a "worm-hole-like" shape. Defects that form can be caused by numerous conditions, such as gas bubbles, segregation, tears, among other factors which can be difficult to predict, due to

the rheological behavior of the mushy zone [2], [3]. This cycle of casting and taking x-rays of the samples are both costly and time consuming. Therefore, a lot of research has been conducted to understand the physics behind these defects [4]–[11] and reproducing them using computer simulations. Initial simulations produced two-dimensional (2D) axisymmetric, branchless features that lacked details needed to understand the issues that develop during manufacturing processes. There are three-dimensional (3D) aspects that are not captured by 2D simulations. However, 3D simulations require extensive computing power and algorithms to scale. Recent advances in large-scale simulation of solidification microstructure have produced some astonishing new discoveries [12]. This has been made possible thanks to development in parallelization and supercomputing. State-of-the-art hardware and algorithm techniques have enabled researchers to produce computations that used to be seen as impossible. The ability to see high resolution in the formation of microstructures has never been more accessible. The size, spacing, and morphology of dendritic arms show a significant effect on the solidified materials' properties [1].

Solidification modeling is a complex multiphysics problem, including fluid flow, heat, and solute diffusion aspects. There is also a need to solve equations for phase change or track the interface to simulate the morphology. The morphology of the dendrites can be captured using Cellular Automaton (CA), Phase-field (PF), Direct-Interface-Tracking (DIT), Level-Set (LS), or Dendritic Needle Networks (DNN) methods [21]. In terms of reproducing physics, PF is one of the best methods to simulate dendrite growth accurately, comparing to front-tracking, boundary integral methods, and CA. However, depending on the aspect under consideration, each method has its own

strengths and shortcomings. The majority of large-scale simulations of dendrite growth use PF and CA methods. The transport phenomena, which consists of either heat transfer, diffusion, and fluid flow, can be solved using Finite Element (FE), Finite Difference (FD), Finite Volume (FV), or Lattice-Boltzmann (LB) methods. Scalability is accomplished using computational approaches that can take advantage of the increasing number of processing units. The selected numerical methods need to be scalable, making it easier to implement different parallelization techniques, using threads, Message Passing Interface (MPI) [13]–[18], and Compute Unified Device Architecture (CUDA) [19]–[22] among other methods. Reducing computational time is still a challenge for large-scale simulation. Therefore, many strategies have been developed over the years to address computational time. Some of the researchers have focused on utilizing parallel computing algorithms to speed up the simulation. Others have tried to optimize the meshing method and reduce the overall computational time by using the adaptive mesh refinement (AMR) strategy [55] or applying different grid size and time step methods [24], [25].

From computational power point of view, technology is increasing towards Moore's law and the exascale era is already upon us [29]–[31][6]. Supercomputing has provided a small deviation from expectations, now known as the post-Moore era [29], [30]. Now the strategy for increasing the ability to model dendritic solidification is finding limitations in mathematical models and algorithms. This limitation is partly associated with interpolation of the quantities around the interface of the dendrite where sharp gradients are present. By implementing an adaptive grid refinement model, the code's capability will be extended to more accurate prediction of the interfacial region. In the AMR algorithm, finer grids are used for sharp gradients regions, such as solid-liquid interface,

while the mesh coarsens in the rest of the computational domain. As a result, most computation time is spent solving the region occupied by the mushy zone. This method has been used by many researchers separately or in combination with other parallelization methods to reduce the computational time [12]–[14]. Recently, parallel-GPU AMR has shown great promise to combine the effectiveness of the AMR scheme with the power of GPU computing [31]–[33]. This allows for high-speed simulations of purely diffusive dendrite growth.

The outstanding large-scale simulations have inspired many researchers to perform analysis on realistic grain growth. As an example, Miyoshi et al. [34] conducted a phase-field simulation study of ideal grain growth in an ultra-large-scale domain. Using a supercomputer, TSUBAME 2.5 at the Tokyo Institute of Technology, which consists of 1408 nodes, the team was able to investigate the number of sample grains that are necessary for repeatable results for grain size distributions. The largest domain consisted of 2560^3 grid points, which consisted of 3,125,00 initial grains. After 100,000 timesteps, 0.4% of the initial grains remained. At this time and scale, it was the largest simulation performed by a factor of 10. This was an effective performance for observing and quantifying the steady-state growth behaviors. Another impressive simulation of large-scale was performed by Sakane et al. [38][11] using a quantitative PF model to simulate a dilute binary alloy's (Al-3 wt%Cu) grain growth under the influence of forced convection. A directional solidification condition was produced in a system with dimensions of 3.072x3.078x3.072 mm^3. The computation of 1024^3 mesh points, during 60,000 steps, was completed using 128 GPUs within four hours. Their PF model could

provide consistent results independent of the interface thickness. An LB model was coupled for simulating the melt flow.

The literature review here aims to highlight important research on large-scale simulation of dendritic growth and other microscale solidification features. We will discuss numerical methods as well as computational approaches used for producing large-scale simulations of solidification microstructure. The actual domain size of these simulations depends on the physics considered and the features that are being simulated. The structure of the literature review will focus on the increasing complexity of features. First, simulating the morphology is discussed in comparison of the methods. Secondly, many case studies will be discussed in the following order: dendritic interactions and competitive growth, columnar to equiaxed transition, solute transport and segregation, natural and forced convection, permeability, and lastly, applications for additive manufacturing (AM). The details for each subject are outlined with respect to both the mathematical/computational models as well as the physical nature of the simulations.

Background

There are many unique use-cases for large-scale simulations of solidification microstructure. The original focus of initial works was to provide a large macroscale domain for the direct numerical simulation of dendrites. Stochastic-based methods of modeling [36], [37] were used to produce larger simulation domains. They had to produce special dynamic allocation techniques to minimize the computation costs for the large number of cells [38]. The trade-off between 2D and 3D is important for distinct use cases. A dendrite growing in a 3D domain versus a 2D domain in the presence of convection will experience less effect of flow on its growth and morphology. This is

because of the obvious reason that the moving melt can flow around the dendrite arms in 3D. We studied the comparison of natural convection and forced convection on a single dendrite evolution in 2D and 3D using LB [39]. 3D dendrites grow faster than 2D, while secondary arms are also more likely to form. Jegatheesan et al. [40] studied the effects of distributing nanoparticles in a buoyancy driven convection solidification, using an enthalpy-based method. When considering methods for convection, reduction in volume from solidification (shrinkage driven flow), was not considered as it was studied only in 2D. However, adding a third dimension showed improved nanoparticle transport, due to the enhanced diffusion.

In a 3D space, competitive growth can be studied in detail, which cannot be expressed in 2D. When a dendrite enters the space of another, dendritic branches can block the growth based on orientation. Sakane et al. [41] studied the dendrite interactions in directional solidification of an Al-Cu binary alloy bicrystal using 512 GPUs with 1024^3 meshes. Figure 1 shows this interaction at 600,000 steps, where a tertiary arm is interacting with a secondary arm. A PF-LB model was utilized with MPI to demonstrate that 3D phase field simulations can be performed within reasonable computation time (12 hours).

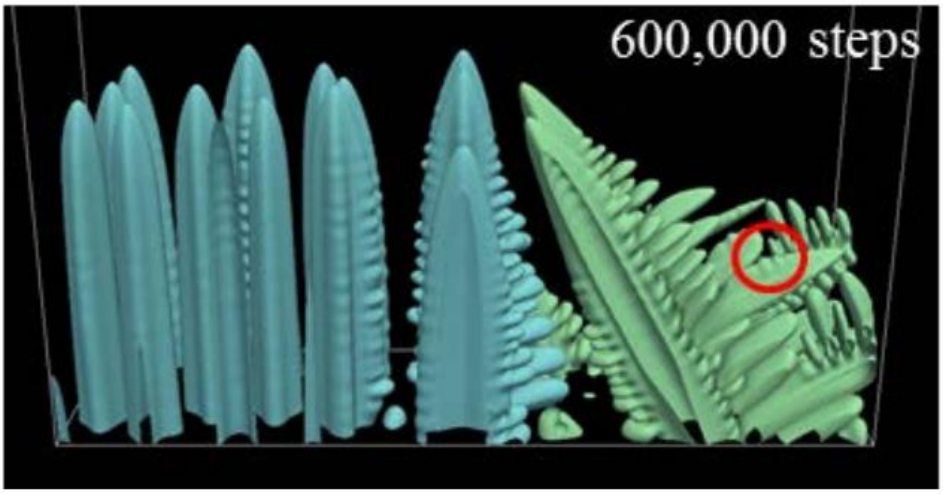


Figure 1: 3D model of dendrite interaction showing tertiary arm growth for unfavorably oriented dendrite [41].

Another helpful comparison of 2D and 3D simulations of dendrite growth was accomplished by Sun et al. [42], comparing the effectiveness of the multi-component (MC) PF model. In this case, a 2D simulation of Ti6Al4V was performed using Al and V solute simultaneously, which was the first of its kind. The results showed that the driving force and growth are magnified artificially in the pseudo-binary in comparison to the MC (ternary) PF model, violating the sharp interface (Figure 2). As a result of the benchmark comparisons, the MC PF model was scaled into a 3D model, using a 512^3 nodal box with a capillary length of 0.840 μm. Using 4 Tesla K80 GPUs for a total of nearly 20k cores, the resulting simulation showed that the inhibited growth of the shortest dendrite by diluted Al (V) was merged by its surrounding dendrites. This phenomenon has been studied experimentally in additive manufacturing for incremental arm spacing and coarsening of primary dendrites. Special computational approaches are required to capture this detail.

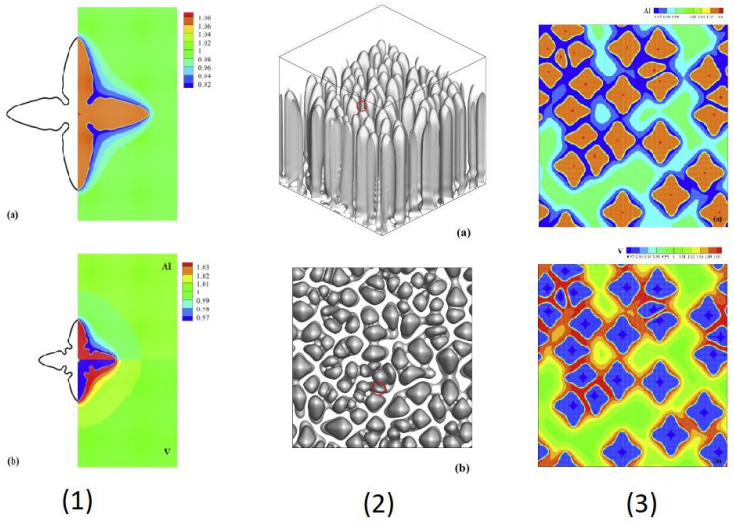


Figure 2: (1) Comparison of pseudo-binary model (a) and MC model (b); (2) 3D Simulation of 99 nucleation seeds; (3) 2D evaluation of solute concentration [42].

In recent years, general-purpose computing on CPUs and GPUs with MPI and CUDA has been employed to speed up large-scale simulations of solidification microstructure [43]–[49]. When MPI is used for parallelization, the computational domain is decomposed into finite subdomains in all directions. The information exchange between subdomains is carried through halo regions or ghost nodes. Based on the location of the subdomain, ghost nodes are either physical boundary conditions or contain neighbor subdomain boundary information. The ghost nodes are updated at each temporal iteration using MPI and the intercommunication between the subdomains [50]. Programming with CUDA, the CPU is known as the “host”, while the GPU is called the “device.” A program written by CUDA programming model consists of a host program or function that runs on a CPU and kernel function which is responsible for solving the governing equations in parallel on the GPU. The host program divides the computational domain into thousands of thread blocks. Each thread block consists of a multiple of 32 threads.

The kernel function is executed by total number of threads in parallel in device. It was shown that the computation time for a GPU with massive computation capacity and bandwidth, is two orders of magnitude faster than compared with a serial CPU core [51].

Under certain conditions, CPUs are effective for simulations in comparison to GPUs. Sun et al. [42] characterized the accelerated performance quantitatively, based on total node points run on a single Intel Xeon E5-2699 v4 core and 4 NVIDIA GPUs. Figure 3 shows that at a lower number of nodes, the GPU efficiency is low. At 64^3 nodes, the speedup ratio is only 8.83. The data communication between the devices is time consuming, while the calculation source of the GPU is largely unoccupied. Furthermore, GPU efficiency increases as the number of nodes increases. With greater development, acceleration in GPUs can be doubled by using shared memory [81], which is located at and shared by the same block.

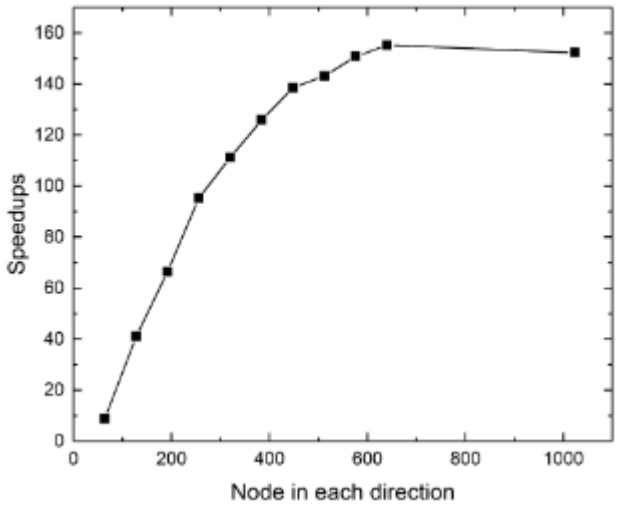


Figure 3: Speedups of single CPU core serial-programming vs multi-GPU parallel-programming efficiency [42].

The evolution of technology has given rise to diversity in applications. Computer processing for large-scale simulations was improved due to the number of processors.

Supercomputers were the source of this amount of computation required, while CPUs were soon replaced by GPUs. This allowed for even faster calculations. Eventually, the bottlenecks for the speed were determined by the programming. Innovations in the GPU technology, such as NVLink, which allows for direct communication between GPUs, were assisted with toolkits, such as NVIDIA's CUDA, to program the GPUs directly. The TSUBAME3.0 (the most current of the series of supercomputers based in Tokyo Institute of Technology) employs this technology allowing four 20GB/s data link per GPU for a total of 47.2 PFlop performance in half-precision [53]. MPI has shown effective use throughout trials. The combination of techniques and optimizations has proven to show significant success in GPU-rich supercomputers.

et al. [54] produced a large-scale simulation for 2D dendritic growth, which was accomplished through the MPI parallelization. Parallel programming utilizing this type of communication between the distributed-memory systems is the standard for large-scale simulations. The programming is approached using a Single Program, Multiple Data Streams (SPMD) [55]. SPMD uses each processor executing the same program on different data for MPI processes. Using a notion of rank to distinguish processes, the point-to-point communication is the fundamental primitive for sending and receiving. The scalability enabled this type of research to utilize Oak Ridge National Laboratory's Cray XT5 system (Kraken supercomputer) [56], [57]. The dendrites were grown to a reasonable size in an "incubation region", then storing the result for restarting using the scalability of 41,472 of the total 112,000 cores of the Kraken supercomputer. Figure 4 is the final snapshot of the dendrite incubation domain, where the magnified portion shows the flow of alloy melt between solidifying dendrites.

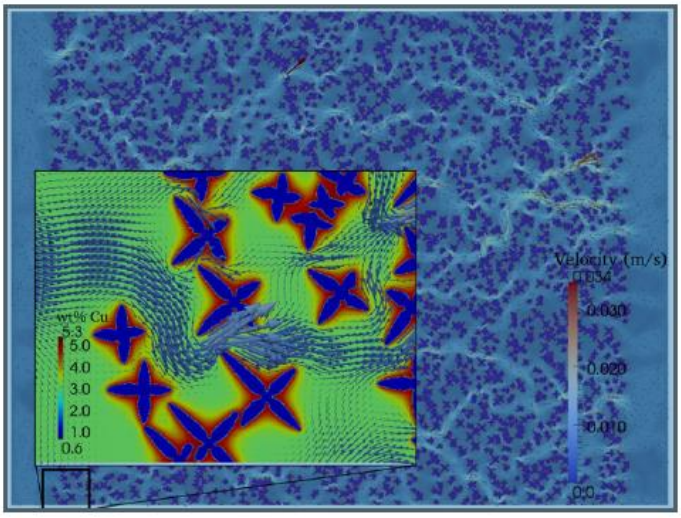


Figure 4: Large scale domain of dendrite growth with enlarged composition analysis [54].

The parallelization required to have this amount of detail for such a large domain was only possible with a technique of spatial domain decomposition. This popular method of splitting the spatial domain into equally-sized domains is specific to the number of cores used for the computation. The benefit of CA-LB model is that only the subdomain boundary values need to be exchanged between the subdomains [54]. Implementing this in binary Hierarchical Data Format 5 (HDF5), yielded high efficiency that resulted in a 50% reduction of memory and computational time required, enabling high scalability.

Continuing this research, Eshraghi et al. [58] utilized the CA-LB model to simulate 3D dendrite growth in a macroscale domain of approximately 36 billion grid points ($1mm^3$). The scale-up performance (strong scaling), where the number of processors was increased with fixed domain size, was compared to the speed-up performance of a fixed processor load by scaling the domain size (weak scaling). Using the combination, the entire domain was filled with dendrites, as shown in Figure 5. The competition between

the dendrites shows that the ones with orientations other than 90 degrees are blocked by the perpendicular dendrites.

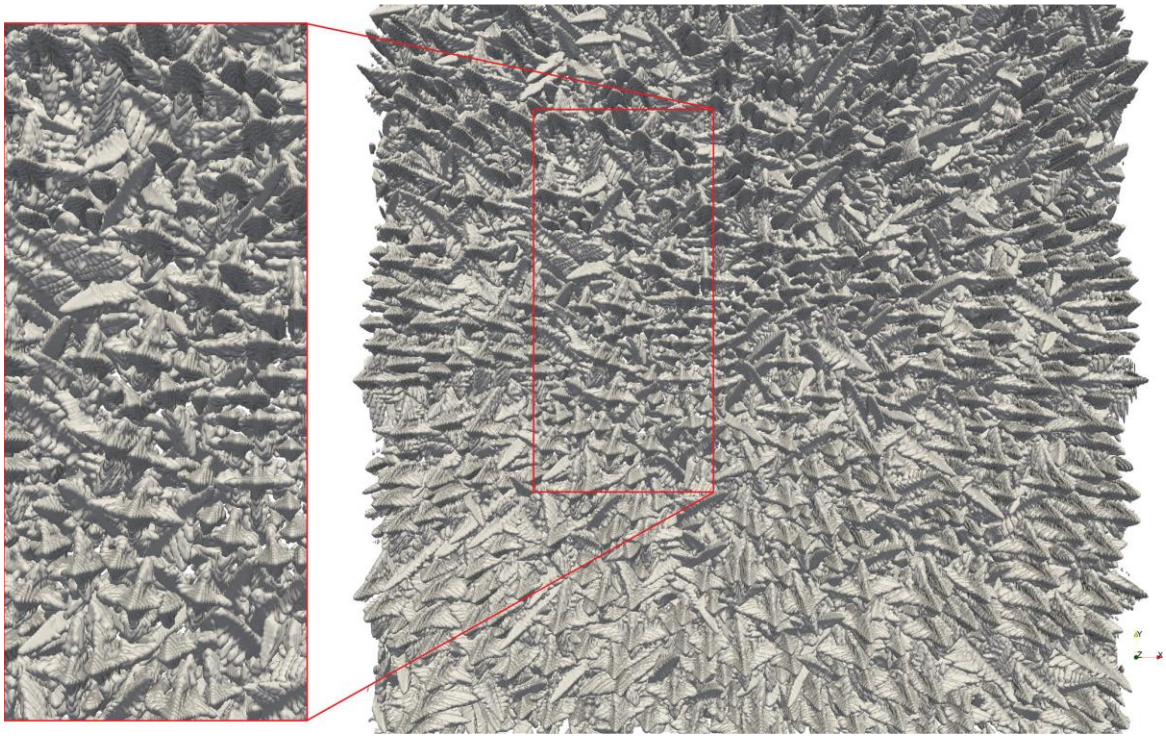


Figure 5: 3D columnar dendritic microstructure [58].

Shimokawambe et al. [59] were able to perform the first-ever peta-scale PF simulation of dendrite growth. The process was implemented locally on a single GPU using CUDA; then using 4000 GPUs, MPI was implemented so host CPUs were used as a bridge for the data exchange. They used overlapping techniques to utilize both GPUs and CPUs to optimize the scalability, which were defined as Hybrid-YZ and Hybrid-Y. First, Hybrid-YZ exploits data independency for array elements by dividing each subdomain in five regions, which are computed separately. The CPU cores were used to compute four *y* and *z* boundary regions, while GPU was used to compute inside the region to produce halo regions. Hybrid-Y instead assigns only boundary regions in the y

orientation of the boundary to CPUs. The communication between the different protocols is illustrated in Figure 6, which depicts the phases of domain designations.

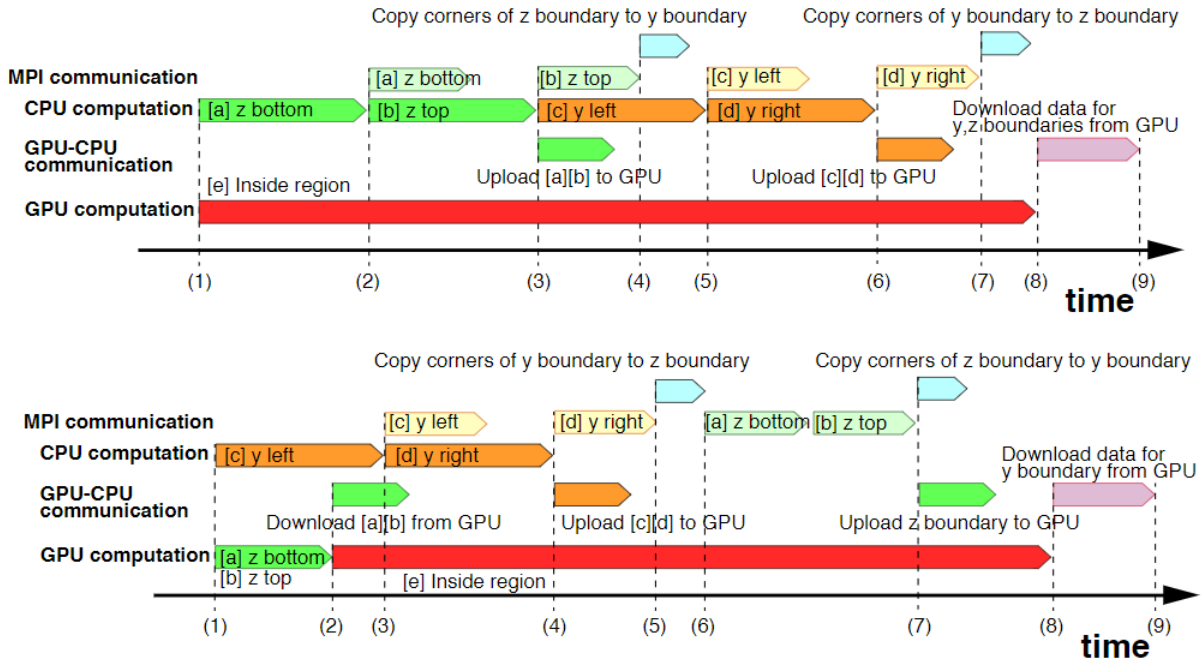


Figure 6: Hybrid-YZ (Top) and Hybrid-Y (Bottom) protocols for parallelization [59].

Hybrid techniques for computational simulations have been the focus of research since modern high-performance computing (HPC) systems have been introduced to simulations. Yamanaka et al. [52] proved that GPU-accelerated PF simulation was faster than its CPU counterpart by two orders of magnitude. Heterogeneous computing has resulted in the development of many frameworks [60], [61]. These many frameworks are developed to enhance the computation and portability of these HPC techniques. "Multiphysics Object-Oriented Simulation Environment" (MOOSE) [62], FEniCS [63], "Portable, Extensible Toolkit for Scientific Computation" (PETSc) [64], Mesoscale Microstructural Simulation Project (MMSP) [65], and Structured Adaptive Mesh Refinement Application Infrastructure (SAMRAI) [66] are a few resources that have

been built upon the models discussed before. High performance interconnection networks [67] have optimized research regarding many of the features below.

Methods for Simulating the Morphology

The process of capturing the interface morphology in simulations can be accomplished with CA, PF, DIT, or LS methods. Each method has its unique advantages and deficiencies. CA, which is relatively simple, stands alone from the others because of its local structure and will be discussed first. PF is probably the most powerful and will be analyzed with respect to some of the most notable case studies. Third, DIT will be discussed to feature the differences using a couple research examples. Lastly, LS method will be reviewed, which is similar to DIT but requires the knowledge about the temperature gradients at the interface. By simulating the morphology of dendrites, complexity in geometry can be incorporated for large-scale domain for manufacturing purposes.

Cellular Automaton

CA is ideal for large-scale modeling simulations, as its structure is considered local with respect to the objective domain. The ability to approach the morphology naturally, in a localized manner, allows for easy scalability on many processors. In the 1940s, the CA method was developed by John von Neumann using simple microscopic laws to model complex, macroscopic phenomena physically [68]. With a successful application in metallurgy, the CA method can accurately simulate processes, such as grain growth, cracking, diffusion, or mechanical deformation. By superimposing a grid of cells on the area of interest, these physical processes' evolution can be reproduced. State

indices and variables are attributed to cells that evolve based on transition rules. Cagigas-Muñiz et al. [69] performed an efficient study of CA transition rules, such as Game of Life, a Forest Fire model, a cyclic CA, and the WireWorld CA, which outperform the standard GPU implementations. This is effective for scalability as interactions of cells can be defined in three dimensions. Advantages over PF include being able to develop a spatial resolution by the order of magnitude of the smallest microstructure feature size [70]. CA is highly parallelizable, making it two orders magnitude faster than alternative PF methods [71].

One of the largest strictly CA methodologies was recently performed by Zhang et al. [72]. The study focused on simulation of Fe-C alloy during isothermal and directional solidification, while using the CA model to simulate the columnar-to-equiaxed transition (CET). Using the CA model, in combination with GPUs and MPI, the maximal speed-up ratio was measured to be 153.19. They were able to study the effects of increasing the cooling rate, which promotes the occurrence of equiaxed dendrites ahead of the solidification front. The CA code was able to process 768^3 grids within 27.42 min for 8000-time steps. They were able to solve the problem of data race in the CA model by introducing an additional field variable with a modification to the capture rule. CA is regularly coupled with solute transport methods, for its functional, elemental nature that allows it to scale [36], [37], [73]. The CA algorithm accounts for heterogeneous nucleation, the preferential growth directions, and growth kinetics of the dendrites [38].

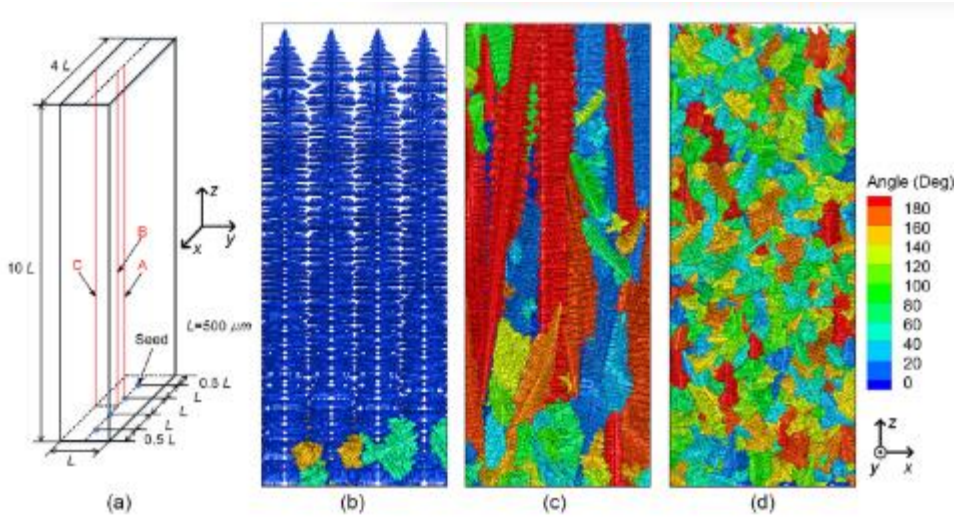


Figure 7: Directional solidification of Fe0.6wt.%C alloy in a domain consisting of 400x100x1000 cells and a grid size of 5 μm [72].

Phase-field

The PF method is a powerful and versatile tool to model microstructural dynamics [74], [75]. Many phenomena can be represented using a simple set of differential equations [76]. Using continuous fields to describe the interfaces, discontinuities of properties and boundary conditions are represented by a variation of one or more auxiliary fields. These phase fields across a diffused interface are solved by integrating partial differential equations for the whole domain. Free-boundary problems with arbitrary complex interfaces are notoriously challenging [77].

Kobayashi [78] first brought the PF modeling of dendritic growth to the public's attention in 1993, showing a simple simulation for one component melt growth. He showed its ability solve the free-boundary problem. PF solves the time evolution equation of the PF variable, Ø. This was introduced to express the phase state (solid or liquid) of the material, in which the sharp interface is replaced by a diffuse interface. With a finite thickness, the PF variable has a smooth, but steep change. The versatility of the PF

method allows for the simulation of the interface migration without tracking of the interface [79]. By simply solving an equation, PF can solve for solid concentration, curvature of interface energy, and capture the interface based on computationally tracking. Kobayashi had produced movies of grain growth simulation in 3D prior to 1990 [80]! This has inspired researchers to reproduce this method for larger scale domains.

PF is very computationally taxing, where the meshes for the interface require high density. AMR has been implemented in large scale under certain conditions [81], such as 3D modeling [82]–[84] (but mostly 2D conditions due to limitations), as computational efficiency decreases as the volume fraction of the interface increases [85]. Therefore, parallel computational schemes are necessary for very large-scale simulations using a quantitative PF model. Shibuta et al. [86] utilized a supercomputer and PF to simulate the competitive growth of dendrite assemblages. They discussed convergence behavior that enables the use of a large interface thickness. At the time (2011), it was the largest reported simulation of dendrite growth for a domain of 3.072 mm^3 for a total time of 100 s, using 768 GPUs on the supercomputer TSUBAME2.0 (TSUBAME 2.5's predecessor). It is computationally oppositional to obtain a realistic looking microstructure and an extremely thin interface [87]. PF parameters are derived at a thin interface limit [88], which is competitively being pushed further.

Two years later, Takaki et al. [89] used the TSUBAME2.0 for a very-large-scale 3D PF simulation of directional solidification of 3.072 x 3.078 x 3.072 mm^3, which was equivalent to 4096 x 4104 x 4096 meshes. This research produced interesting results (Figure 8), where unfavorably oriented dendrites and highly complicated interactions could survive. As a result, controlling microstructures in terms of crystallographic

structures has been considered for more complex structures. This was important as it expanded on both the domain size and complexity that was previously discussed by Shibuta et al [86].

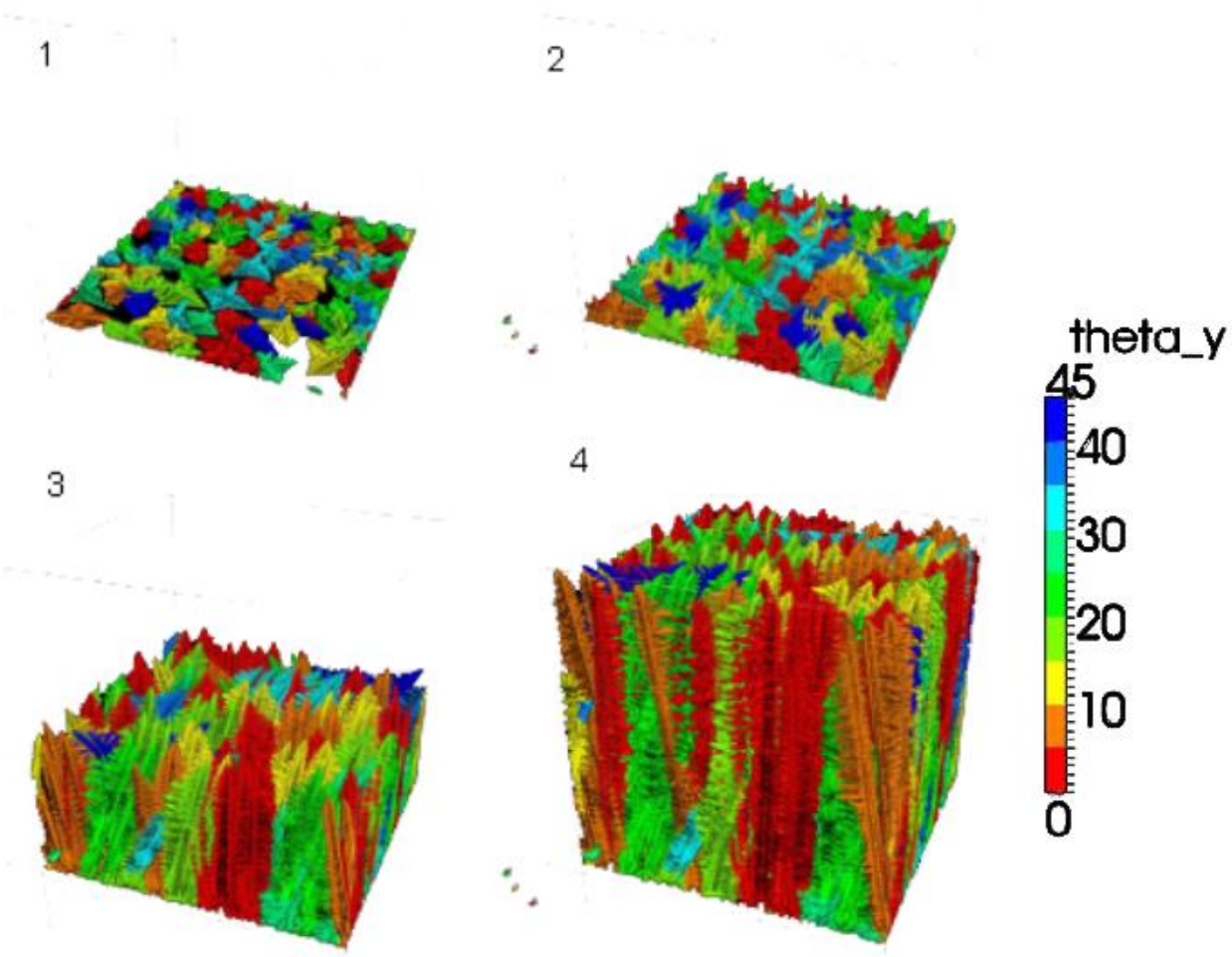


Figure 8: Columnar dendrite growth where inclination angle is indicated by color at the following time steps in order: 5.4 s, 10.7 s, 53.6 s, and 107.1 s [79].

In order to simulate large-scale microstructures using PF models, the PF interface is required to be scaled up much larger than the physical interface to remove several artifacts that step from the thickness of the interface [90]. Using interpolation functions for average diffusivities and grand potentials of the bulk phases, the interface stretching is eliminated. The interface also faces (unrealistic) movement from the relaxation that can

be countered by asymptotic analysis. Lastly, artificial solute trapping can be mitigated by anti-trapping formulations [91]. These conditions need to be met in order to scale to higher resolutions while maintaining accuracy using PF methods. PF's advantage over the other methods lies within the field variables, which eliminate the need to identify and track the interface.

Direct-Interface-Tracking

DIT, a front tracking method, is a successful methodology for solving the energy equation, as well as momentum and mass conservation equations. This requires treating the interface as incompressible. Since it requires the calculation of the temperature gradients at the interface, in combination with the normal velocity and curvature of the interface, DIT is less powerful in comparison to the other techniques mentioned [92]. N the mesh size is refined. It has been observed that the mesh dependency is influenced By the methods of curvature and solid fraction calculations [93]. The simulated columnar dendrite evolution of an Al-4 wt.% Cu alloy was calculated in a domain of 600 x 1200 meshes on a single CPU in about 11 hours of computational time. Figure 9 shows

Zhu et al. [94] used a quantitative virtual front tracking model, which is characterized by its mesh independency, for simulating a 2-D dendrite growth in the low Pèclet number regime. This means that the results converge to a finite value whe the evolution of the dendrites after 12 s, comparing the grain boundaries and solute map. The equilibrium composition was compared in good agreement to the LGK model.

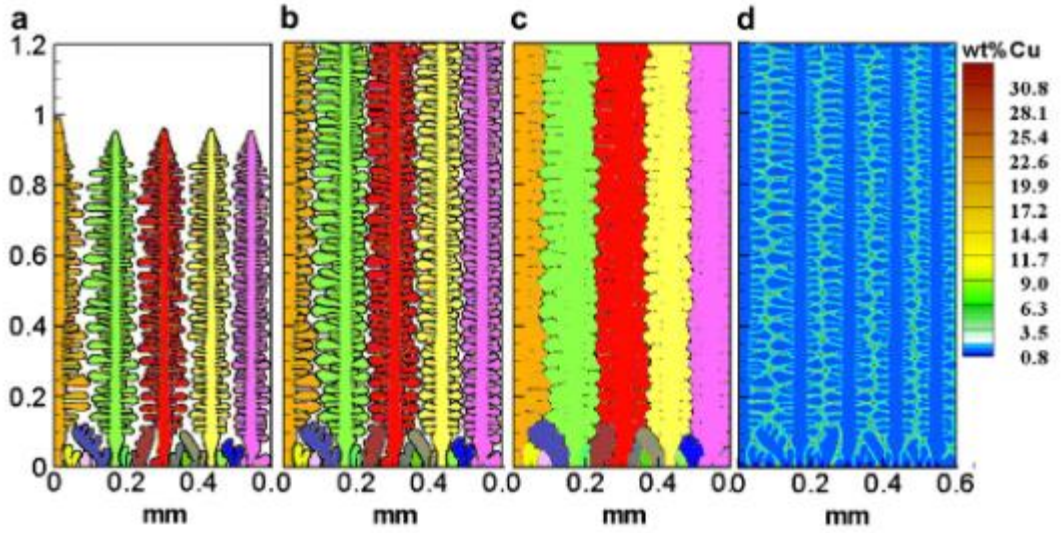


Figure 9: Evolution of directional solidification with thermal gradient after (a) 2 s, (b) 3.2 s, (c) and (d) 12 s, (a-c) shows grain boundaries and (d) solute map [94].

This process is considered with direct numerical simulations of flows with phase change. Tryggvason et al. [95] expanded on a multiphase flow model, where different material properties in each phase were added at the phase boundaries. These techniques have been widely used for sharp front solidification processes. Juric et al. [96] used this method to observe complex dendritic structures, such as liquid trapping, tip-splitting, side branching, and coarsening. This efficiently handles discontinuities in material properties between liquid and solid phases. DIT produces a method for simulating the morphology that can be further defined based on tracking criteria. Front Tracking explicitly provides the location of the interface at all times.

Level-set

Tracking the interface can be accomplished either explicitly or implicitly. Explicit tracking, such as front tracking, requires special care for topological changes (i.e. Merging or breaking) [97]. Overcoming this is difficult in 3D; however, implicit

representations, such as LS or PF, can handle these topological changes in a straightforward manner. They represent the front as a level set of continuous functions. LS simulations are simple to an advantage, especially when computing the curvature. LS is similar to PF [98], as it also interchanges the interface with a field variable; however, it requires the knowledge of the direction in which the solid front is advancing (along with its velocity and calculation of the normal vector from the interface) [99].

Tan et al. [100] utilized a level set simulation that combines the features of front-tracking and fixed-domain. A domain decomposition of 8 domains was performed using the Cornell CTC supercomputer [101] for 5 hours, while the 3D simulation of a single dendrite required 12 hours with 16 nodes (each of the nodes consisted of two 2048 MHz CPUs). They were able to prove the method provides accurate tracking of the interfaces, computation of heat/mass/momentum transport avoiding boundary conditions, adaptive meshing, and capability of multiple solid phases. The research involved a reasonable mesoscale value for a mesh width of 12.2 μm for the diffused interface for a full mesh of 1024 x 2048. In another study, Tan et al. [102] modeled multiple dendrite interaction with undercooling in the front using a LS method in a 3D domain. A solidification speed of 3000 μm/s and thermal gradient of 1400 K/cm was required for nucleation and is shown in the simulation below (Figure 10).

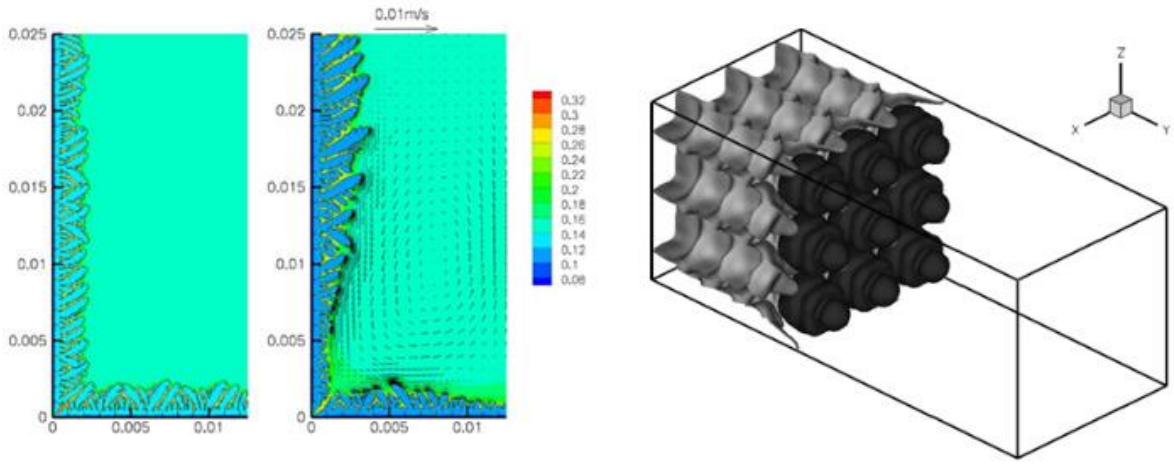


Figure 10: Solute Concentration of Ta without (left) and with convection (middle) [9] and 3D crystal growth with front tracking and required undercooling for nucleation 7k (right) [102].

LS method is a helpful alternative to PF, to avoid the asymptotic analysis needed for PF method [103], [104]. It is an extension of front-tracking methods, where energy conservation issues derive from discretization errors. This is based on the direct application of temperature boundary conditions and the computation of heat fluxes, which was first addressed in Stefan problems [77], where the interface velocity is calculated from interpolation of the heat flux nodes near the interface. The LS method is useful for a direct calculation of the growth and shape of the solidification of dendrites without the need to apply the boundary condition explicitly at the freezing interface. Using a fixed FE grid, LS avoids the need for moving or adaptive griding, while providing an explicit and accurate tracking of the interface front [105].

Dendritic Needle Network

DNN is another novel method that is featured with respect to simulating the complex structure of dendrites. A mesoscale simulation resolution is needed for this, as each branch of the dendritic grain is considered a thin needle crystal, modeled as a network. Phase-field and CA simulations are used to predict the dynamics of the

individual branches for solidification or complex dendritic networks at a scale much larger than the diffusion length. Tourret and Karma [106] created a 2D multiscale model for DNN for both isothermal and directional solidification that is validated through the comparison with analytic solutions. This DNN approach allows for investigation into the dynamics and stability of spatially extended dendritic arrays. The optimization of this model proves to be four times faster than normal phase-field simulations. A 3D DNN simulation was scaled from 2D by defining the flux intensity factor (FIF), which is the strength of the concentration gradient at the tip of the needle. This is comparable to calculating the stress factor of a crack's tip, which is calculated using a contour integral and modeled as a parabolic tip. The main discrepancy for a 3D analysis is caused by the Laplace's equation having no solution for a line terminating at a single point, which was accounted for by considering the solute flux intensity for a needle of finite thickness. The results were compared with microscopy of a sample from NASA's microgravity solidification experiments, which showed a similar characteristic scale of the array spacing. Tourret and Karma [107] elaborated on this 3D DNN model using the same parameters with isothermal and directional solidification. Al-7wt%Si alloy that was modeled in the previous microgravity experiment was analyzed with more scrutiny. They utilized a new 2D formulation for thick branches with paraboloidal tips. The comparison of the 3D simulations to samples from the Columnar-to-Equiaxed Transition (CET) in the Solidification Processing project (CETSOL) [108] is shown in Figure 11. The simulation results of DNN, featured in white, follow a similar primary dendritic spacing to the markers of the samples featured in purple. This research has provided a basis for

predicting quantitatively fully 3D microstructure spacing of individual branches, which determine the mechanical strength of the structure.

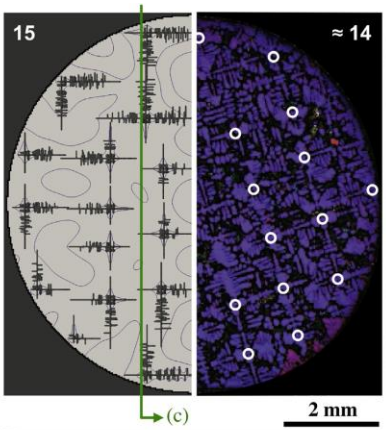


Figure 11: Comparison of simulation to polished microgravity sample [107].

Tourret et al. [58] expanded on the DNN research by focusing on isothermal growth of an equiaxed grain in a supersaturated liquid in 3D. Using PF, needle-based, and envelope-based approaches, benchmarks for an undercooled isothermal equiaxed growth were used to compare steady-state growth predictions. The theoretical Ivantsov solution provided the Péclet number that formed the basis of comparison, scaled with respect to the tip radius and velocity. The comparison of the model types is shown in Figure 12. While the lack of truncation radius bounding was apparent in the shapes, the coarseness of the dendrites had caused the velocity to decrease due to the effect of the boundary conditions.

Nonetheless, this illustrates the effectiveness of DNN and Grain Envelope Models (GEM) at reproducing PF results with reasonable accuracy. This comparison of models features an operational compromise between resolution and computational efficiency. While accuracy is important for large scale simulations, multiscale models enable predictions of primary dendritic spacings, similar to industrial casting processes [109].

This is a large step towards creating realistic models; however, the computation required for DNN is still dependent on compute intensive models of PF or CA. Therefore, much research has been deployed to scale these models.

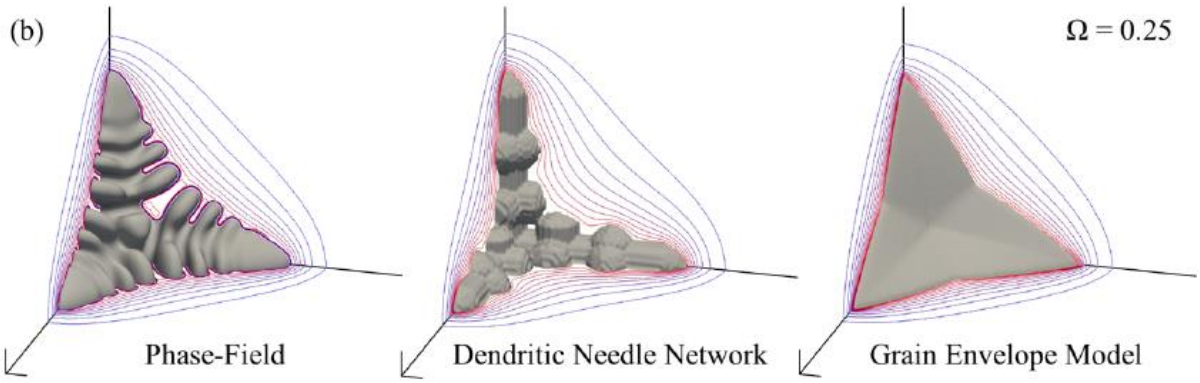


Figure 12: Comparison of model simulation types [110].

In context with PF, DNN solves a broad range of phenomenon, where strong assumptions are made for the meso-scale models. Normally, these models don't account for the detailed evolution of the solid/ liquid interface. DNN's representation of the dendritic structure as an array of needles, allows for both steady-state and transient growth regimes to be described, while accounting for the diffusive interactions between them. This bridges the PF approach with coarse grained stochastic models. Multiscale DNN allows for convective effects to be more effectively studied. Most recently, Isensee and Tourret [111] compared the oscillatory growth behavior to X-ray in-situ imaging, identifying the fundamental mechanisms in which the gravity-induced buoyancy alters the dynamics of the crystal growth. DNN is a helpful approach for scaling simulations of morphology; however, in order to predict realistic microstructural evolution, incorporating fluid flow is crucial.

Features and Applications

The applications of large-scale simulations are as diverse as the features that are observed. The following sections provide a general understanding of the top areas of focus for simulating dendritic growth evolution, which include the follow: dendrite interactions and competitive growth, columnar to equiaxed transition, solute transport and segregation, natural and forced convection, permeability, and additive manufacturing processes. Competitive growth and dendrite interactions provide insights into how orientation and spacing can influence evolution. Columnar to equiaxed transition focuses on how the transition can be predicted using processing parameters. Solute transport and segregation highlight interesting phenomena such as freckle formation and solute plumes. Natural and forced convection explain growth phenomena from different flow dynamics. Permeability provides insight toward both experimental and inspiration for the research shown in the second half of this manuscript. AM Processes give context to manufacturing processing parameters used by simulation techniques. Together they provide a comprehensive understanding for the basis of this research.

Dendrite Interactions and Competitive Growth

The dendritic growth is dependent on many factors that can generate many different phenomena. These effects need to be understood in detail to be able to predict the changes in the microstructure. Traditionally, Walton and Chalmers competitive growth model [123] is utilized for the selection of growing multiple dendrites. Nickle-based superalloys have shown some astonishing growth phenomenon [124]–[126], which cannot be simulated using the conventional model. First, this section will highlight the

traditional model, then will expand on the phenomenon and other effects associated with competitive growth.

In 2017, Yang et al. [127] used PF simulation to create a 3D dendrite growth in a nickel-based superalloy. The results were generated using a single NVIDIA GTX1080 GPU for a total of 774.292 GFLOPS (giga floating-point operations per second). From dendrite arm spacing to inclination angles, the dendrite morphology evolution during this superalloy's casting can be understood to optimize the mechanical properties. The simulation was accomplished using a simplified approach by assuming the alloy as a pseudo-binary alloy, which was first popularized by Raghavan in 2012 [128]. PF has the capacity to simulate both isothermal and non-isothermal dendrite growth, which makes it versatile for applications, such as asynchronous concurrent GPU computations. This allowed Yang et al. to show the interactions between the dendrites in an equiaxed multi-dendrite domain, which caused the dendrite arms to grow with a deviation from their initial crystal orientations [129]. As they fill the entire domain, the dendrites coarsen and coalesce causing the melting of the secondary arms.

The comparison was accomplished using directional grain growth, allowing for the height to be competitive. This illustrated that the growth is dependent on both the inclination angle and the positions of the adjacent dendrites. This simulation shows how the favorably oriented (FO) dendrites outperformed the unfavorably oriented (UO) dendrites. This simulation was performed using an asynchronous concurrent algorithm to show that 774.29 GFLOPS were obtained in 512^3 computational grids on a single NVIDIA GTX1080 GPU. The growth of the dendrites obtained a height of 6.42 mm.

However, in contrast to this study, the following study illustrates conditions where the opposite is true (UO outgrows FO).

Takaki et al. [130] published a study focused on simulating the competitive growth with converging grain boundaries in a large domain of 3.072×6.144 mm^2. The directional solidification of 3D dendrites was simulated to compare the effect of unfavorably oriented (UO) grain inclination angles. Using the Walton-Chalmers model [123] Takaki modeled a contrasting phenomenon. Where at the grain boundaries, the UO dendrites overgrow the favorably oriented (FO) dendrites! This unusual overgrowth is more common when the thickness of sample and the UO grain inclination angle are small. The secondary arms have higher growth at boundaries. Tertiary arm growth is enlarged at the convergence with the domain boundary.

Takaki used this competitive growth model for 3D analysis, where a collision of FO and UO grains occurred in the middle of the domain to form a straight GB. The FO and UO grains shared different properties, where the arrangements of the dendrites become gradually ordered [131]. The FO grains form a hexagonal arrangement, while the UO grain migrates in a lateral direction. This occurred as the UO dendrites penetrated deeper into the FO grains with respect to the reduction of angle of orientation. This interaction, termed “space-to-face interaction” [131]. Figure 13 shows the convergence of the differently oriented grains with respect to different degrees of orientation. This perspective from above the columnar dendrites shows that the UO dendrites with a lesser degree of orientation have a greater convergence, which is expected from previous competitive growth patterns.

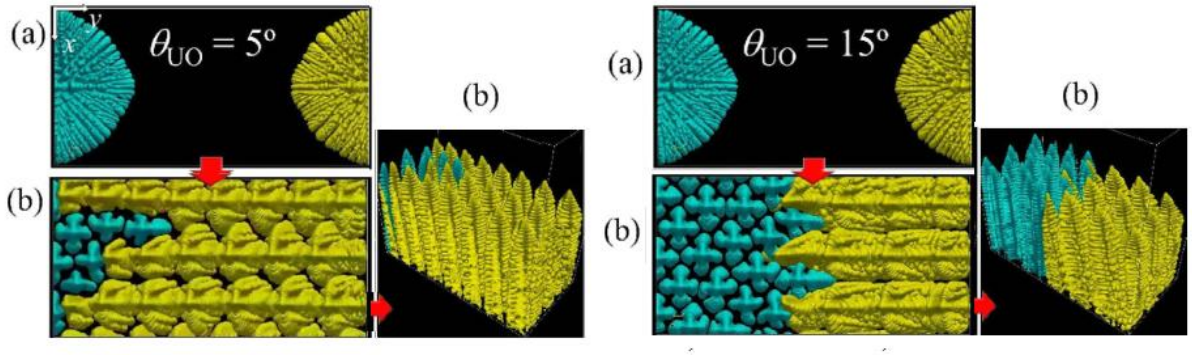


Figure 13: Dendrite Interactions comparing different angles of UO dendrites of 5˚ and 15˚ at time (a) 1.3s and (b) 187.5s [130].

In large-scale grain growth, the Walton-Chalmers model is largely accepted as the general competitive-growth model for grain selection. In this model, favorably oriented (FO) grains block unfavorably oriented (UO) grains. In contrast, when analyzing primary arm spacing, Fourier Transformation (FT), Voronoi decomposition, and minimum spacing tree have been employed to evaluate the arm arrays [38]. This has been the basis for new models to be developed for more accuracy. Voronoi Tessellation (VT) has been applied to approximate the morphology of equiaxed dendritic grains. The dendrites are formed from a mesoscale domain, where Voronoi grains are formed in a triangulation technique exemplified below (Figure 14) forming a polyhedral structure. These assumption models allow for complex model domains to be generated. For example, Feng et al [132] simulated a 3D semi-solid microstructure using 1000 grains, where VT was used to approximate the final grain morphology to compare the volumetric inflow caused by shrinkage. Permeability will be discussed in more detail in the next section.

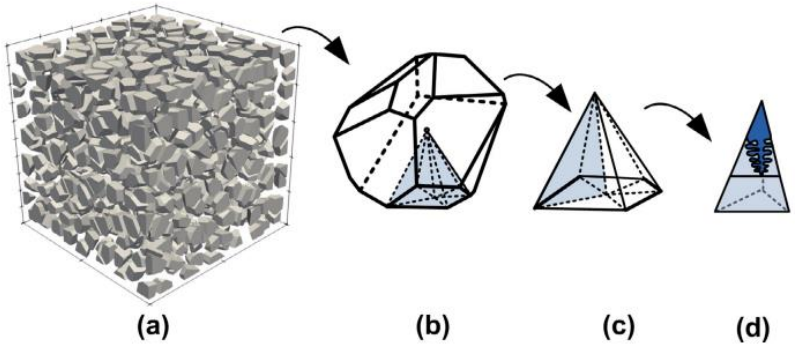


Figure 14: Voronoi grain from Meso-scale simulation [132].

In order to evaluate arrays, ordering and spacing of primary arms, especially in 3D directional growth, different methods have been employed historically, such as minimum spacing tree [133], Fourier transformation [134], Voronoi decomposition [135], and many others [136]–[139]. Using a modified Voronoi decomposition technique, where the small sides are removed from the Voronoi cells, the primary arm array was evaluated by Takaki et al [140]. The hexagonal pattern showed typical-hepa defects that decreased with respect to time for tilted columnar dendrites in a computational domain of 1.152 × 1.152 × 0.768 mm^3. These enabled predictions of the primary arm spacing which is crucial for the integrity of the material. This was ultimately possible through a convergence analysis of the tip undercooling of the dendrite/ cell.

The ability to predict dendrite spacing is important for permeability research, as fluid flow can only be researched under certain conditions. Porosity, which is an important type of defect that can form from casting, can be caused mostly by either shrinkage, where the volume changes upon solidification with a restricted feeding of the liquid, or condensation of dissolved gases in the melt upon freezing. This has a strong negative effect on ductility and fatigue life, where internal pores create initiation sites for

cracks and local stress concentrators [141]. As convergence occurs, permeability is another condition that requires much research for consideration of realistic predictions.

Columnar to Equiaxed Transition

In context of design, columnar or equiaxed grains may be desired depending on the expected properties. The transition between columnar and equiaxed has been investigated for many years, where heterogeneous nuclei are used commonly to promote equiaxed grains. An example where equiaxed grains are preferred is for processes of direct-chill casting of aluminum alloys [142]. However, its high angle grain boundaries can reduce creep rapture life. Therefore, it is important to control the grain structures using conditions, such as high thermal gradients and low growth rates. We recently performed a 3D PF simulation of columnar to equiaxed transition (CET) for Inconel 718 alloy in a domain of 0.2 x 0.1 x 0.4 mm^3. A CET solidification map was created to compare growth rates and temperature gradients for the evolution of dendrites in equiaxed, columnar, and mixed regimes as depicted in Figure 15. A model was developed to predict primary dendrite arm spacing (PDAS) of columnar growth in a wide range of temperature gradient, solidification rate, and initial grain sizes. This novel approach is effective for optimizing process parameters for melting and solidification on a preexisting substrate, such as in AM or welding applications.

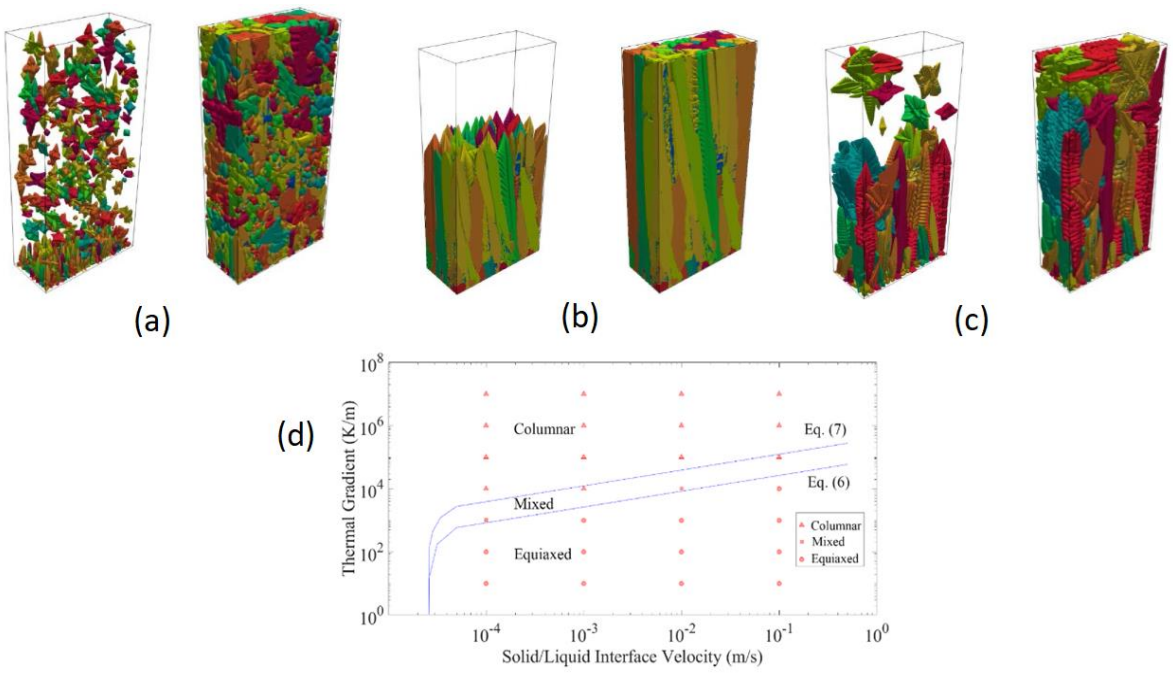


Figure 15: Evolution of (a) equiaxed (b) columnar (c) mixed dendrites and (d) solidification map for Inconel 718 alloy [15].

Studies to determine the mechanisms behind CET have been a popular focus for research. Hunt [144] proposed an analytical model to predict CET based on the potential for equiaxed grains to nucleate ahead of the columnar front's undercooled region. The solid fraction was calculated using the truncated Scheil equation empirically to relate the cooling rate to the tip undercooling. Then, it was modified by Gäumann, Trivedi and Kurz (GTK model) [145] to include non-equilibrium effects of rapid solidification. The Hunt's model shares the same relationship for CET based on key parameters of pulling velocity, thermal gradient, and composition.

The process of predicting the CET has been demonstrated using a stochastic model for alloy 718 [146]. This meso-scale experiment was performed in 2D and compared to 3D computation to illustrate the competition of nucleation and growth for

the equiaxed and columnar morphologies. The visualization shows that the CET begins earlier in 3D than 2D, because the grain growth isn't confined in the third (z) direction. There are also more nucleation sites available in the 3D simulation, where some dendrites are cut from a different plane. Again, in mesoscale, the model predicted a larger segregation pattern in the 2D compared to 3D. In microscale, the 3D columnar dendrites showed finer detail than the 2D observations. Modeling the CET evolution provided an efficient methodology of simulating microstructural features in a mesoscale domain using a stochastic model with relatively low computational resources. The simulations of microstructure cross sections are shown below (Figure 16) comparing 2D and 3D, where the CET begins earlier in the 3D simulation. The domain size was 10×30 mm^2 in 2D and $10 \times 30 \times 2$ mm^3 in 3D.

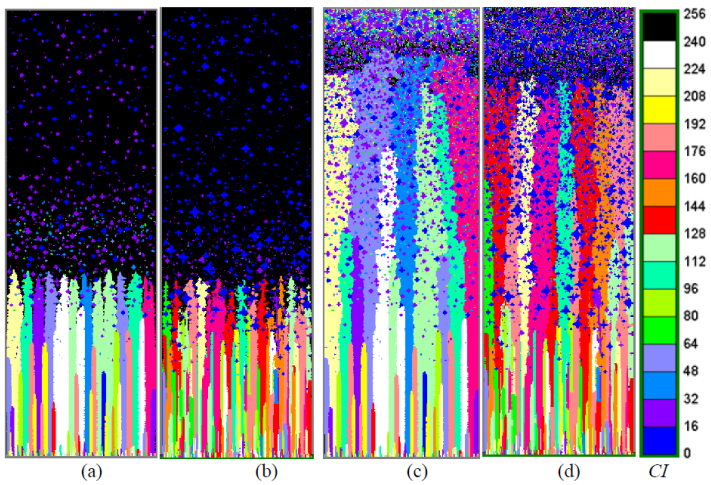


Figure 16: Simulations of CET evolution comparing 2D (a) and (c) and 3D (b) and (d) during the unidirectional solidification of alloy 718-5 wt. % Nb [146].

Dong et al. [147] showed a situation where equiaxed grains were placed ahead of the growth of columnar front during solidification of Ni-based superalloys. This real-life situation results in casting defects, such as stray grains or "tree rings" [148] This was the

precursor of the CET simulations, where it showed the gradual transition, realistic dendrites, and complex solute concentration profiles. The size of the domain was $2.5\times4mm^2$ for a total of 500×800 cells in the domain. In another work, Dong et al. [142] focused on the solute interactions within the CET modeling. It was a unique finding that the solute interactions were strong in the secondary and tertiary arms, while a weak interaction between solute and arms was observed in the columnar tips. Figure 17 illustrates the simulation for inoculating a melt with heterogeneous nuclei while reducing the activation barrier and increasing the density of the nuclei (right to left). The change in temperature ranges from 1K (a) to 10K (d), simulating the change in undercooling in a large-scale domain.

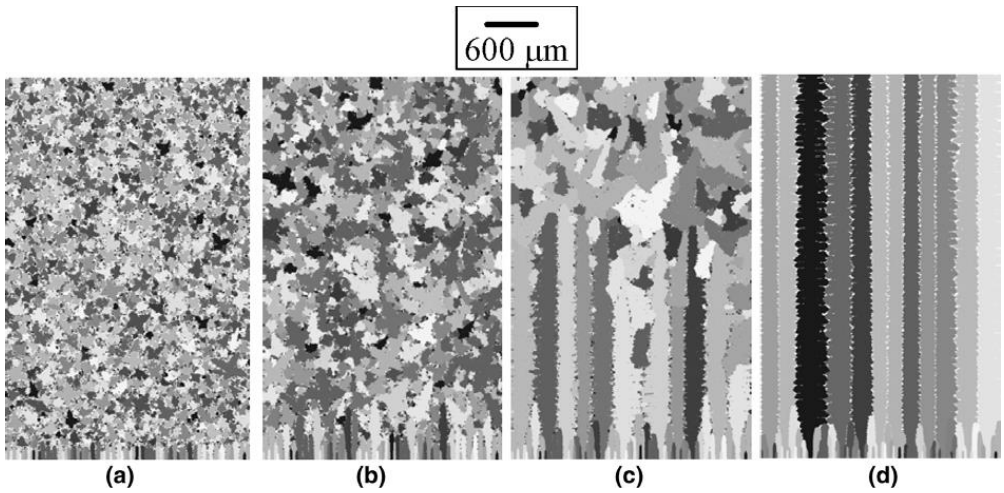


Figure 17: Undercooling on simulated grain structures range of mean nucleation undercooling (ΔT_N) (a) 1 K; (b) 2 K; (c) 5 K; and (d) 10 K. ($V= 5.5\cdot10^{-4}m/s$, $G= 3000 - 100tK/m$ [142].

DNN provides a promise for describing the transient growth dynamics of higher order branches, which normally relies on strong assumptions. Geslin et al. [149] effectively simulated the CET in 2D, using DNN in a large domain of $22.5\times30\ mm^2$ size. Using a sharp-interface model for directional solidification, they were able to observe

complex phenomena, such as: “circular growth” (branchless growth) (Figure 18), abrupt and progressive transitions with different grain structures. Therefore, when casting large samples in a crucible, the columnar front that progresses towards the inside of the sample is accurately simulated, where the slower solidification rates favor the nucleation of equiaxed grains [141].

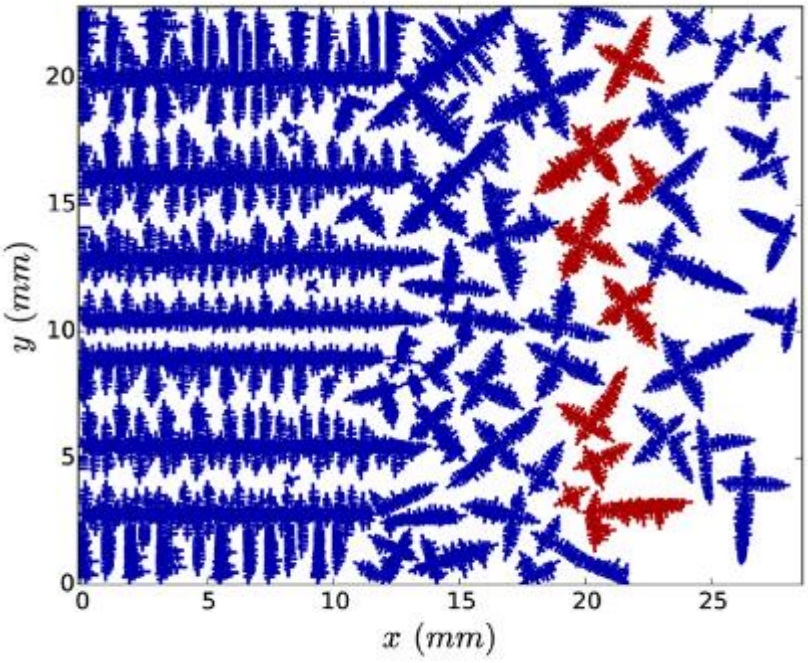


Figure 18: CET simulation and initial circular germ regime transition to four branches grain. [149].

Solute Transport and Segregation

Solute transport is a phenomenon that is very flow-dependent. This can have effects on the development of the dendrites under well-defined thermal conditions. Effects on dendrite spacing and symmetry have been notably studied [169], [170]. Wang et al. [36] were able to utilize a CAFE model to simulate a controlled solute diffusion in the solidification of a binary alloy. Using a 2D model, they simulated a tertiary dendritic arm growth from a secondary dendrite arm, which was then blocked by another

secondary dendrite. This was helpful in predicting the primary dendrite arm spacing. The average primary arm spacing was found to be dependent on the current growing conditions and the way those conditions were reached. Figure 19 shows the measurement technique to predict dendrite spacing. With large-scale simulations that utilize higher resolution, tertiary dendrite grain growths can be visualized.

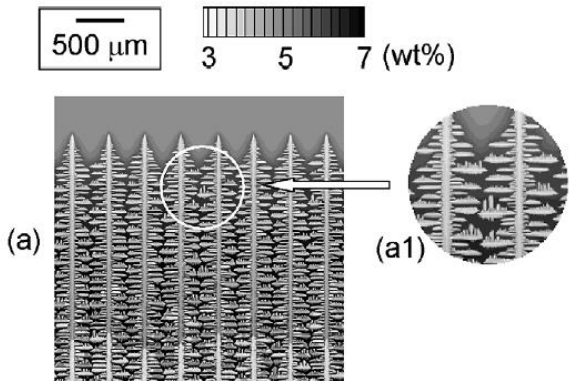


Figure 19: Tertiary arm dendrites used to predict spacing in 2D [36].

These features for prediction models serve as a helpful analysis to develop a real-life understanding of the grain interactions. From interactions of both equiaxed and directional grain growth, the competition models show a clear understanding of the evolutional interactions. Even highlighting the transition between the two phases, comparing the mesoscale interactions on a microscale level requires a manageable distribution of resources. The prediction models for spacing utilize innovative techniques that give a new understanding of competitive grain growth, even within the same primary arm. The features for this analysis have provided the foundation for more progress to be made with more complexity in the dendritic evolution. Solute plumes are a resulting situation that occurred in a directional solidification of Ga-In alloys. This solute-rich liquid that flew up is believed to be a crucial factor that causes, during solidification, a

freckle defect [171]. Takaki et al [172] studied the effects of natural convection in both 2D and 3D simulations. Performing a series of simulations where gravity is changed, they were able to show that as the gravity decreases in the negative region, primary arm spacing increases. Furthermore, the downward flow enhances growth of secondary arms, as unstable dendrite growth is caused by a large upward flow. Figure 20 shows the differences in the dendrite tips in 3D, while 2D shows plume and freckle-like solidification defects forming. Using PF-LB simulations on the TSUMBAME 2.5, they were able to simulate a domain of $0.384 \times 0.384 \times 1.536$ mm^3 with 95 hours of computation time.

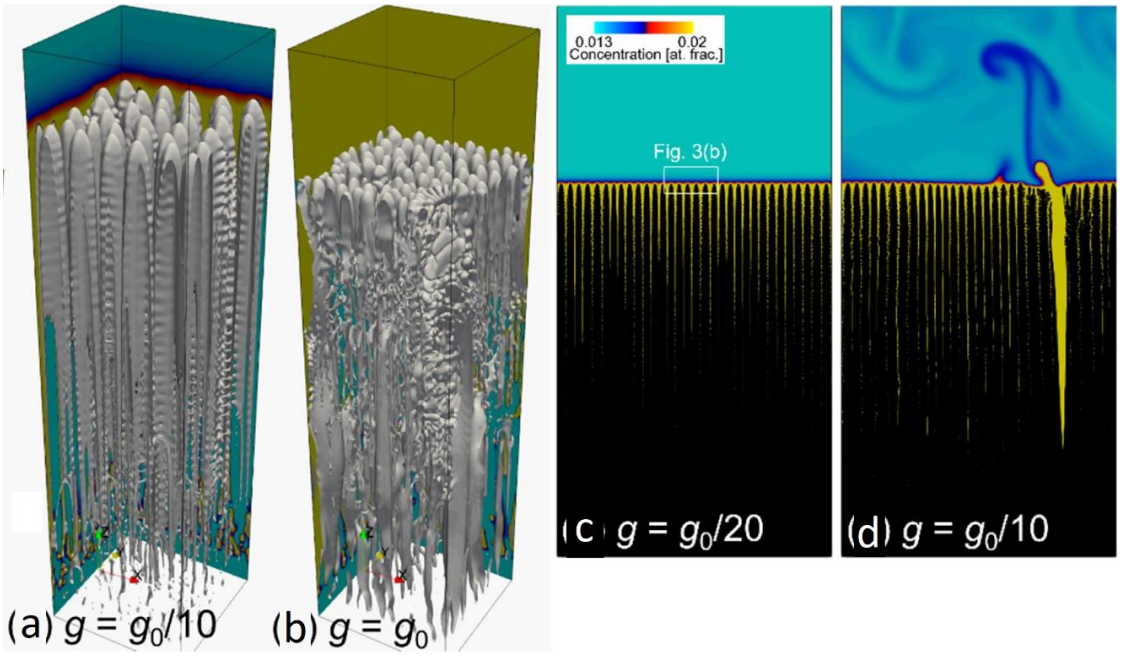


Figure 20: Dendrite morphologies and solute concentration distributions for 3D (a) and (b) and 2D (c) and (d), comparing different forms of gravitational acceleration [172].

The flow influences both solute transport variations and the upstream-downstream dendritic growth. The effects of convection can limit the downstream growth, while the up-stream dendrites are promoted by the convection-induced anisotropy. The solute

plumes are the result of gravity assisting in the direction of the dendrite growth. The presence of convection can change both time- and length-scales, which can result in differences in morphologies from the purely solute transport. Our team studied the buoyancy-induced flow during directional solidification of 3D thin domains, comparing Al-Cu and Pb-Sn alloy systems [173]. Sn is lighter than Pb, where more solute is rejected into the melt and its concentration around the interface increases. Upward buoyant force is caused by the resulting from the decrease of density of the liquid mixture around the interface. This large solute boundary layer that forms with a peak at the center is shown below (Figure 21) and is also referred to as a chimney. This phenomenon has the potential to be stable during the solidification process, which carries solute up, forming a recirculating flow form. This can lead to decreased growth or remelting in this region during later stages of solidification and can form a freckle defect once completely solidified.

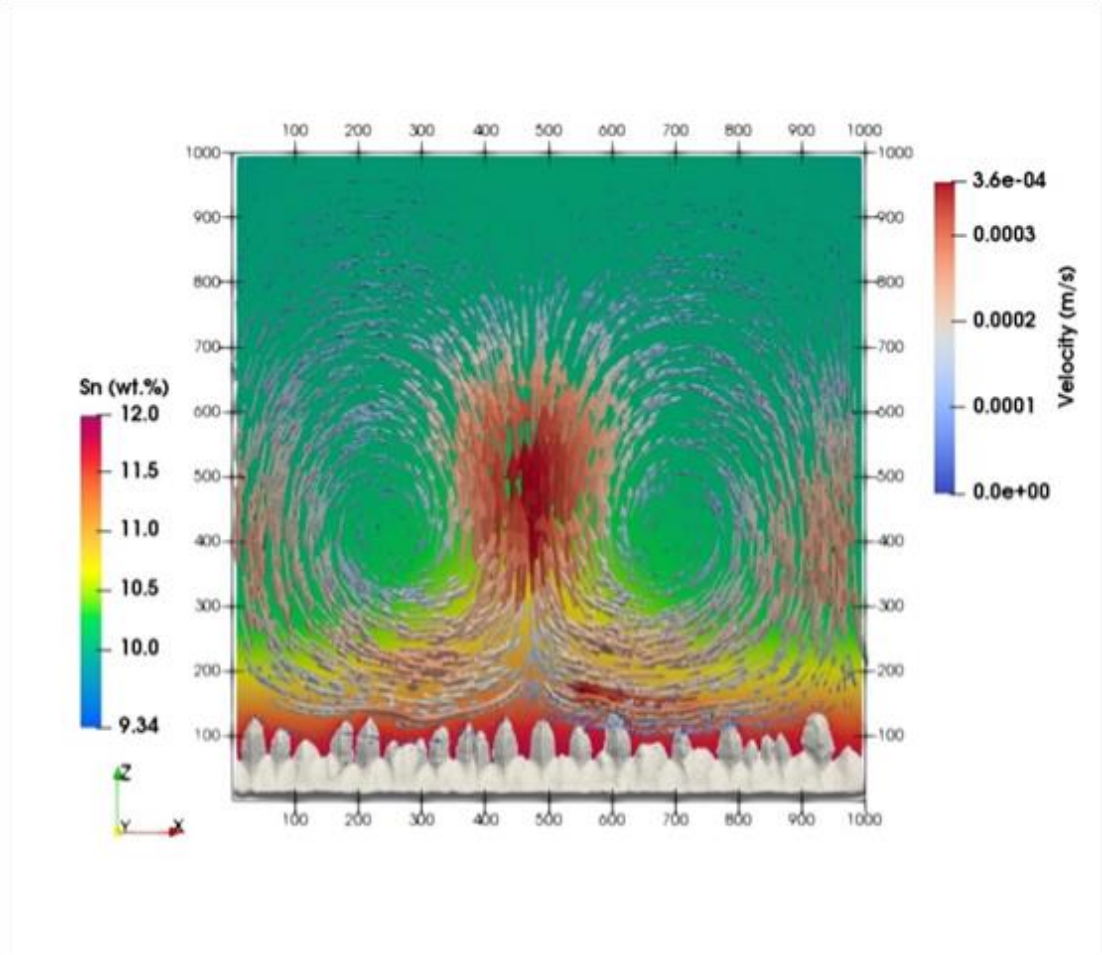


Figure 21: Chimney formation and solute concentration field during directional solidification of Pb-10 wt% Sn alloy [173].

Macrosegregation is the solute composition inhomogeneities at a macroscopic scale of a casting [174]. Heat treatment can remove these imperfections; however, spatial variations of mechanical properties can occur with respect to the nature, amount, and size of these defects. Gross compositional defects, such as the formation of freckle or segregated chimneys, is caused by macrosegregation. This is associated with the following causes: solidification shrinkage [175]–[178], natural and forced convection [179]–[181], grain movement [182], mushy deformation [181], and cavitation bubbles [183]. Fragmentation has been researched as an important phenomenon using large-scale

simulations, as it is responsible for grain refinement [181]–[187]. It is undesirable for certain manufacturing processes, such as single crystal turbine blade casting [188] or AM of Metastable β-Ti alloys [189], [190]. Process parameters can define the formation of freckles that can be caused by remelting, where there is a sudden rise in temperature, deceleration of a growth front, or a change in flow conditions. Simulations are an effective method to study this capillary force driven phase transformation process. [59]

Kao et al. [191] developed a large-scale model to simulate freckle formation for the casting of Ga-25 wt% In. Freckles form due to remelting and fragmentation of dendrite arms by thermosolutal fluid flow, especially for alloys where the partitioned solute is lighter than the bulk fluid. Examples of such phenomena include Ni-based superalloy and Ga-25 wt% In. They utilized the LB method for fluid flow and CA to simulate solidification growth. The coupling between these two main equations is achieved by natural convection force and energy and solute transport equations at each node. They utilized MPI parallel algorithm to accelerate the large-scale simulation [192]. The formation of the freckles in the simulation is shown in Figure 22. The domain for the numerical model was a total of $32 \times 32 \times 0.16$ mm^3, which was equivalent to 164 million cells.

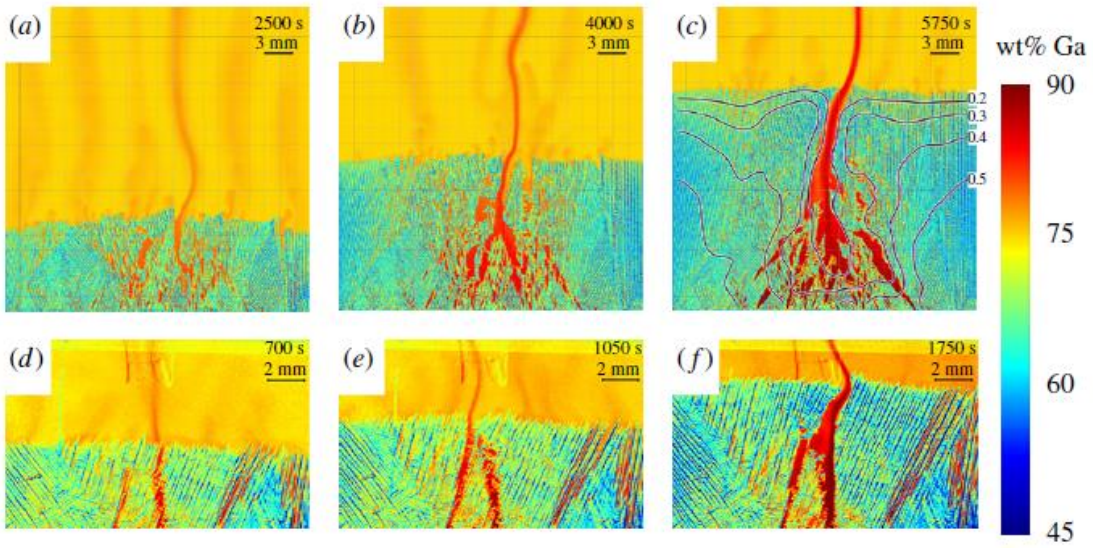


Figure 22: Freckle formation for Ga-25 wt% In binary alloy (a) to (c) Simulation, (d) to (f) experimental result, for G=250 K/m in the vertical direction at different times [191].

Zang et al. [177] studied a comparison of forced and natural convection simulations in both equiaxed and columnar dendrite evolution. Understanding the effects on Al-Cu alloy, using PF-LB in an AMR algorithm, they were able to illustrate some unique phenomena. Dendritic fragmentation, angulation of dendrite arms, and splitting are dendritic growth behaviors that were subjects of interest for the forced convection simulations, where the direction and intensity of the convection had a significant influence. Fragmentation occurs regardless of orientation type for columnar dendrites, but instead dependent on convection type.

Freckle formation, also known as channel segregation, is a unique byproduct of vacuum arc remelting (VAR). At the final stage of the casting process, this is employed to assist in the quality of casting. Solute is ejected upward and solidifies as freckles. Research has been conducted in understanding the segregation defects on the VAR process, simulating the ingot evolution under different arc distribution [193]. However,

much detail can be obtained employing large-scale simulations of the microstructure, where solute transport is a driving factor.

Natural and Forced Convection

Forced convection has been a subject of interest for large-scale simulations, as it has a lot of influence on solute distribution. Research has provided context for dendritic growth and compared the effects with and without convection. Another helpful comparison is between natural convection and forced convection, which is designated by boundary conditions for the walls and velocity conditions. Convection was originally solved using NS solutions [194]. Simulations of dendrite growth with melt flow was first possible in the 90s using PF [195]–[197], while being limited to 2D [198], [199]. Early simulations of 3D were limited to a single dendrite [203], however, it is important to scale these algorithms for complex microstructures. Yuan et al. [201] studied the effects of dimensionality on dendritic growth simulations for convection, using a modified projection method of NS. When 2D flow has a blockage from the primary dendrite arms, the 3D flow has the ability to wrap around the primaries. Therefore, it demonstrated that 3D simulations are necessary to correctly predict unconstrained solidification microstructures. Forced convection has been studied by Jakhar et al. [202] in combination with thermal isotropy, where pressure fields are solved using the SIMPLER algorithm. The model was extended to multi-dendritic simulations with random distributions and orientations in order to study microstructural evolution. Takaki et al. [203] performed a large-scale PF and LB simulation ($0.384 \times 0.384 \times 1.536$ mm^3) studying the effects of natural convection during directional solidification. Comparing 2D and 3D, the effects of gravity are smaller for 3D, while average primary arm spacing increases as gravity

decreases; similar for both. Downward flow enhances growth of secondary arms, while upward flow, larger than a critical value, can produce plumes and freckle-like solidification defects. Using the TSUBAME 2.5 supercomputer, the computation took about 95 hours for 1.5×10^6 computation steps in 3D.

Computational cost has been a limiting factor for research on convection, where LB is most efficient for multi-GPU computation, as the growth is able to be simulated with solid motion, liquid flow, collision and coalescence of multiple solids, and subsequent grain growth. However, much effort has been made to enhance this ability. Sakane et al. [204] created a 2D simulation for a large number of dendrites (350) utilizing PF and LB methods. Figure 23 shows the evolution of solute concentration and flow velocities, where a sedimentation path forms from the dendrites. Assuming inelastic collisions, the coalescence of the grains is observed. The nuclei generated at the top of the domain settle downward, while growing equixially. The ability to scale performance was widely studied in this research, utilizing Active Parameter Tracking (APT) [205], [206]. Tracking the execution time with and without APT was compared to grains with and without motion, proving the efficiency of the model, where APT excels with convection. The parallel efficiency of the model showed that a simulation can be performed in a 64x larger domain with only twice the amount of time compared to a single GPU. The simulation was performed with 2048 × 2048 grid points and 5 grains per one GPU. APT algorithms are employed to simulate coalescence-free grain growth within a reasonable computation time [207].

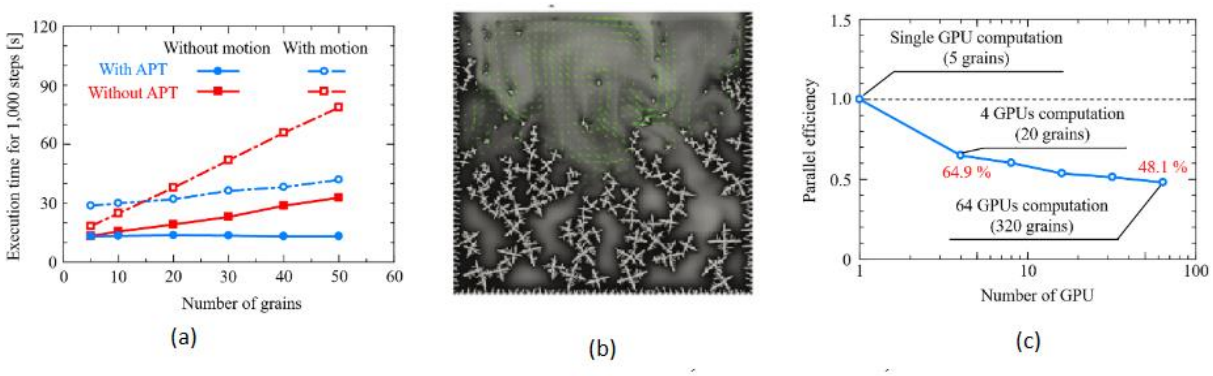


Figure 23: (a) ATP Efficiency comparison; (b) evolution of solute concentration and flow velocities simulated by LB method; (c) parallelization efficiency with scaling GPUs [204].

Zhang et al. [208] combined the PF and LB with a parallel AMR algorithm to for several studies of convection, both natural and forced, in 2D and 3D. They quantified the effect of both convection and undercooling, comparing the length ratio of dendrite arms. The investigation showed that as the undercooling decreases and the effect of convection increases, the length ratio has a peak value. This is due to crystal size decrease comparative to a higher undercooling. Secondly, the effect of gravity with a lateral force of convection was studied, where columnar dendrites grow anti-parallel to gravity. The accumulation of the solute from gravity stunts the growth; however, with the convection, the primary trunks of the dendrites show a constant deflection angle until reaching a critical value. Sun et al. [209] used a 2D LB model to show that asymmetrical dendrites grow faster in an upstream direction, while slower downstream. Studies like this can help to optimize the solidification conditions for manufacturing, by analyzing the convection's effects on dendrite growth.

Permeability

Permeability predictions have produced an effective method to analyze the distinct solidification conditions of the liquid flow through a mushy zone. Anisotropic

porous media uses Darcy's law, which is derived from the Navier-Stokes equation using an averaging procedure [150]. Interfacial stresses occur when the solid fraction is high enough for the solid to form a continuous structure. In a mesoscale domain, the solute distribution is limited by the permeability of the solidification, with consequences for grain refinement [151]. 3D interdendritic flow simulations have been performed using microtomography mappings to measure permeability in Al-Cu [152]. However, the compromise between resolution and sample size is limited to the camera. Therefore, simulations have provided a solution with parallel programming, simulating both the morphology and the fluid flow. The complexity of the transport phenomena has resulted in interesting studies in large-scale, such as, cross-permeability [153], where experiments measuring permeability has a limitation of experimental volume fraction [154].

Permeability research has been characterized both numerically and experimentally. Permeability simulations have been validated in hypoeutectic aluminum alloys by Khajeh et al. [155], where the simulated microstructures for a dendritic network were modeled using CA technique. The Brinkman-Darcy equation was used as the mathematical expression for the permeability, while an X-ray microtomography scan of a solidified Al-20 wt.% Cu alloy was used to generate the computational domain with a solid fraction of 0.73. The model was validated using a large-scale analog of the simulated structures 3D printed using selective laser sintering (SLS), as shown in Figure 24. The results showed that the dimensionless parameter, θ, determines the permeability behavior of the dendritic/eutectic structure. The large values are controlled by the active dendrite network, while the smaller values are controlled by the permeability of both the dendritic and eutectic networks in a dual fashion. 3D Printing has been a helpful tool to

perform experimental measurements from a simulated model. Most recently, Berger et al. [156] used a fused filament fabrication (FFF) technique to scale a PF model from 150 μm^3 to 5 cm^3 of an Al-Si-Mg alloy sample with a fraction solid varying from 0.61 up to 0.91.

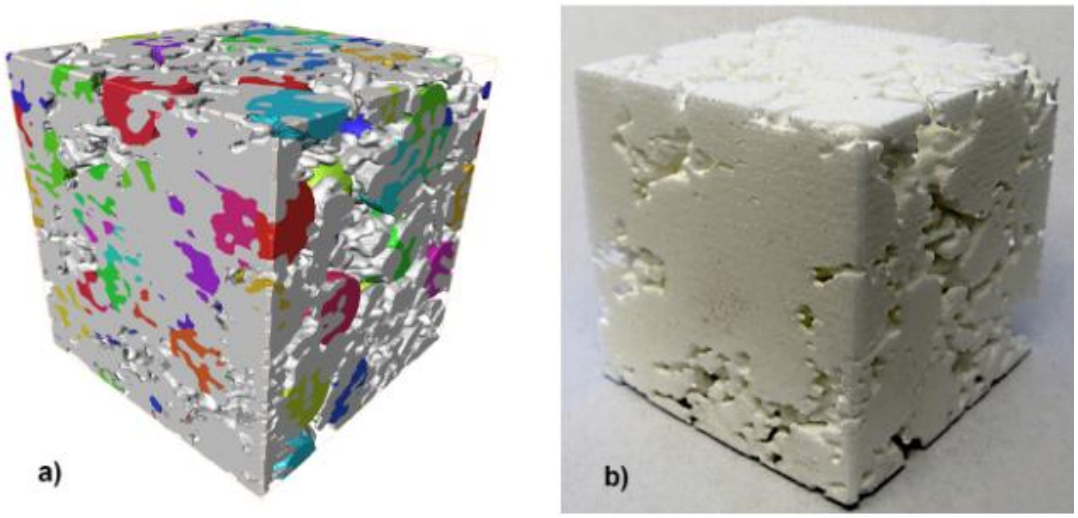


Figure 24: (a) Surface-based representation for Al–20 wt.% Cu for Nucleation Density, (N_v=90mm^3) (b) 3D printed dendritic network [155].

Validation of these models requires more detail to ensure the correct physics is being modeled [190]. Mitsuyama et al. [157] performed an analysis on permeability of a large domain of 1.152 × 1.152 × 0.768 mm^3, using PF, LB, and approximated using the Kozeny-Caman (KC) equation [158], which is used most frequently to express permeability. Alternatively, the Poirier-Heinrich (PH) [159] equation can be used, but with uncertain accuracy due to its derivation from 2D simulations and experimental data [160]. Therefore, much research has gone into validating these models. Such discrepancy has invoked studies based on different types of flow and growth patterns. In parallel flow, a solid fraction can change in dimensions with respect to the liquid that flows through the entire columnar dendrites, as shown in Figure 25. The simulations validated the use of a

KC coefficient of $K_c = 3$ and the permeability tensor for this use case. Thus, the permeabilities in arbitrary directions for columnar solidification structures can be calculated without simulation. The main area of interest regards the specific interface areas and the temporal changes, described by the solid fraction.

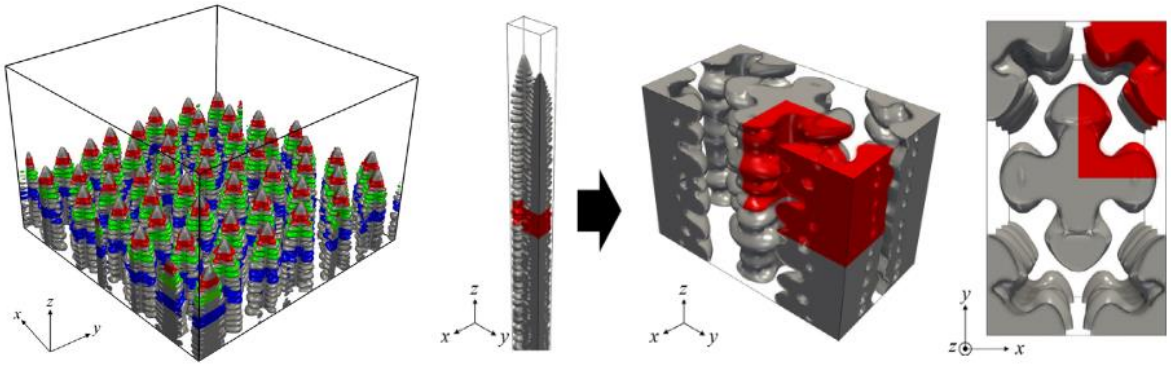


Figure 25: Domain of 1.152 × 1.152 × 0.768 mm^3 with 51 dendrites, extracting red part illustrating arbitrary direction flow [157].

While approximation for equiaxed dendrites is effective, the KC equation is not as certain for directional solidification [161], [162]. Takaki et al. [163] utilized a new permeability prediction method [164]–[168] using a parallel process of GPUs. They performed an analysis on permeability for columnar solidification structures with a periodic regular hexagonal array simulated using PF and LB. They were able to develop smooth variations of the qualities of solidification morphologies, which the permeability was shown to be independent of the array ordering of a consistent primary arms structure. This dimensionless permeability for a specific interface area is attributed to the parallel GPU computing that performed this large-scale simulation. This reiterates the importance of large-scale simulation to study what normally is not possible in real time.

Additive Manufacturing Processes

Additive Manufacturing (AM) is an important application where solidification microstructure and related phenomena can significantly alter the material properties [112], [113]. Porosity, propagation of cracks, or precipitation of second phases can have unique effects on the mechanical properties of the material. Many factors can contribute to this, where this section elaborates on different types of AM methods and studies that produce advances in understanding of these processes. The molten pool is an interesting area of focus, where different dendrite morphologies can be obtained by controlling the thermal gradient and cooling rates. As an example, a FE-CA model was used by Yin et al. [114] to simulate dendritic growth in the molten pool of the laser engineered net shaping (LENS) process, studying laser moving speed, layer thickness, and substrate size. Comparing simulation to experimental results are an effective measure for success. Yu et al.[115] researched a multigrid CA model to simulate these properties from an Electron Beam Selective Melting (EBSM) of a Ni-based superalloy, Inconel 718. Using experimental data, the growth of tertiary dendrite arms was validated. This 48 thermos-fluid model was compared to experimental results of single-track scans, as shown in Figure 26. By studying how primary dendrite arms grow in the melt pool in mesoscale,

this research provided a promising approach for studying shrinkage porosity and propagation of hot cracking.

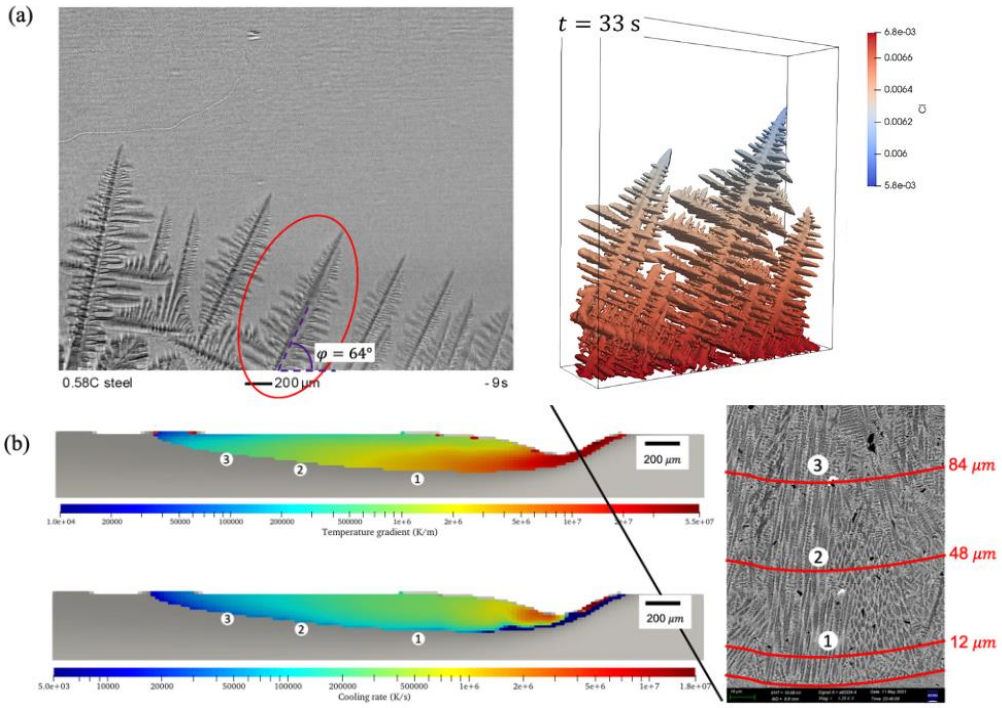


Figure 26: (a) Comparison of experimental and simulated microstructure by liquid solute concentration in the interfacial cells (b) Simulated melt track showing temperature gradient (top) and cooling rate (bottom) compared with the experimental solidified melting pool region [115].

The complexity of AM requires many components to efficiently model the processes. An interesting aspect of AM processes is the potential for location-specific microstructure control. Shi et al. [116] studied the effect of laser beam shaping on morphology, size, and crystallographic texture for laser powder bed fusion (L-PBF) of stainless steel. First, they used a process modeling code, ALE3D, for solving continuity, momentum, and energy equations, which was developed at Lawrence Livermore National Laboratory using a hybrid finite element and finite volume formulation [117]. The output of the ALE3D was the temperature field for all nodes through time. The transient temperature field was then imported to a simplified CA model to capture grain structure.

The ALE3D solver needed a coarser grid compared to the CA model. Therefore, the output temperature field from the ALE3D solver was projected on a finer CA mesh. They utilized the DREAM 3D software [118] and experimental measurement for the initial grain structure required in the CA model. The output of the CA model, shown in Figure 27, was the grain structure and crystallographic texture during L-PBF.

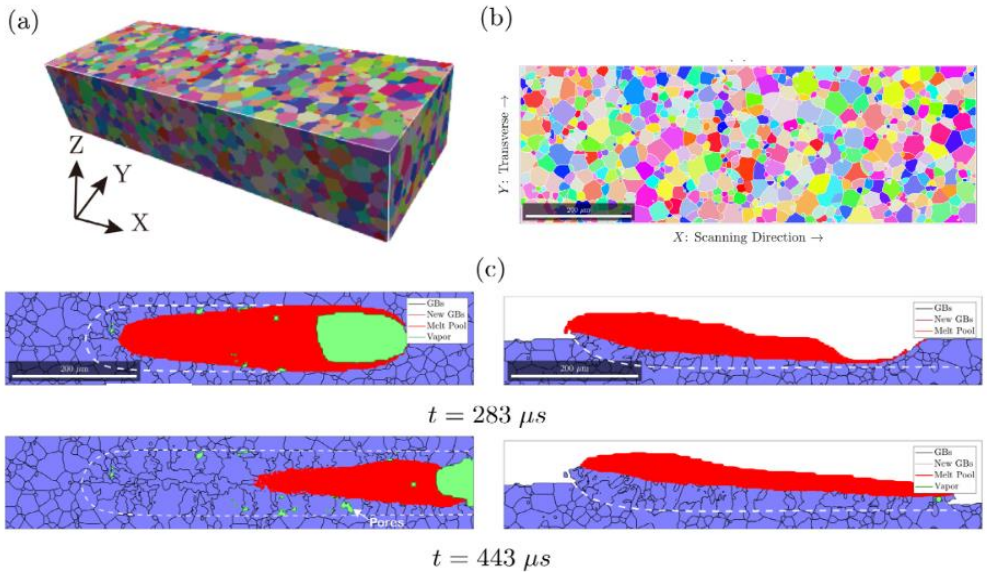


Figure 27: (a) 3D simulation orientation map (b) Longitudinal cross-section (c) Evolution of grain structure during a single-track L -PBF AM of 316L- SS at section located at 8 μm below the substrate surface [119] GB: Grain Boundary.

The properties of AM builds are highly dependent on the process parameters. Lian et al. [120] proposed a 3D CA-FV method to study the process parameters including laser scan speed and laser power to predict the grain structure for the single track directed energy deposition (DED) AM process of the Inconel 718 alloy. They also presented the grain growth for a multiple-layer deposition process with different raster patterns. The comparison of 3D simulation results and electron backscattered diffraction (EBSD) and pole figure experimental results is shown in Figure 28. In their proposed method, the

cellular automaton method enriched with grain nucleation scheme was used to predict columnar, equiaxed and mixed grains, while the FV method was used to solve heat transfer and thermocapillary flow.

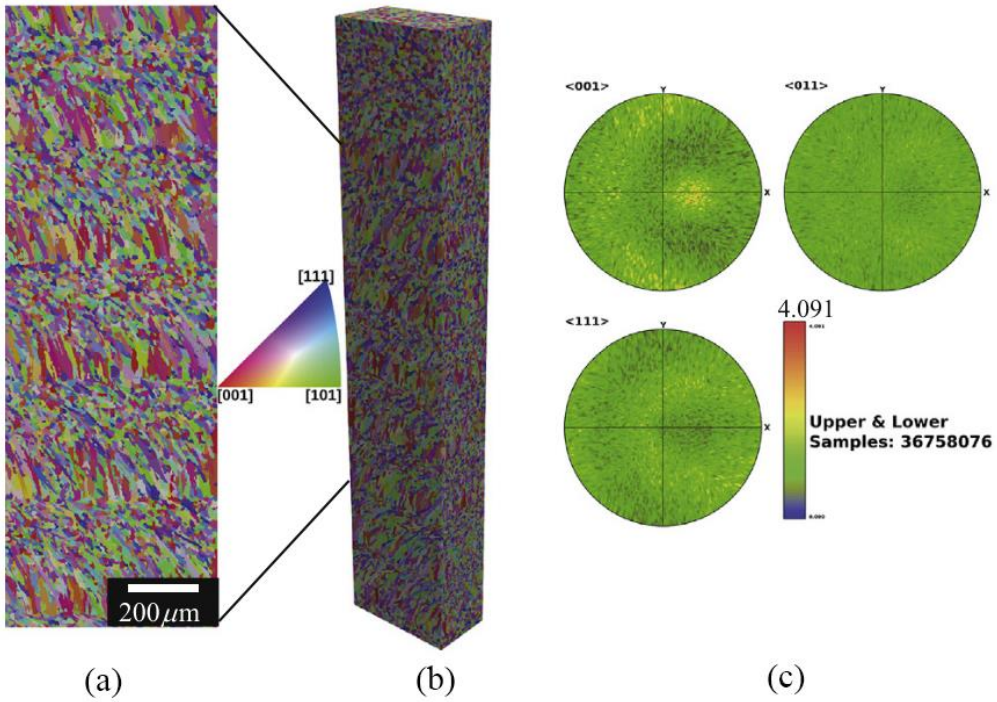


Figure 28: Simulation of grain structure for multi-layer AM builds with a unidirectional raster pattern. (a) Midsection EBSD; (b) 3D grain structure; (c): 3D pole figure of these simulations [120].

Laser powder-bed fusion (L-PBF) is the most popular process for manufacturing functional parts for different applications.[165]–[168]. The simulation of L-PBF requires considering localized phenomena that are highly transient, making the simulation very complex. Marangoni convection in the melt pool, rapid solidification, topological depression of the melt, and thermal cycling are examples of such complex phenomena [116]. Elahi et al. [121] recently presented a computational framework for this type of simulation, using a combination of CALPHAD calculations for alloy properties, macroscale FE thermal simulations, and microscopic PF models for the melt pool

solidification. They were able to calculate a billion grid points on a single cluster node of eight GPUs, providing insight into grain texture selection with detail of dendrites for a realistic multiscale SLM simulation.

The large-scale simulations have enabled innovative melting strategies for AM, such as localized melt-scanning to control the grain size and spacing of the primary dendrite arms. Raghavan et al. [122] produced predictions for grain sizing for a corresponding qualitative texture plot. This process allows for consistent solidification microstructure across the build. The comparison of the simulations is shown below with different types of grains and spacings (Figure 29). By comparing experimental results with simulation, context for the types of microstructures is validated with the types of melt strategies.

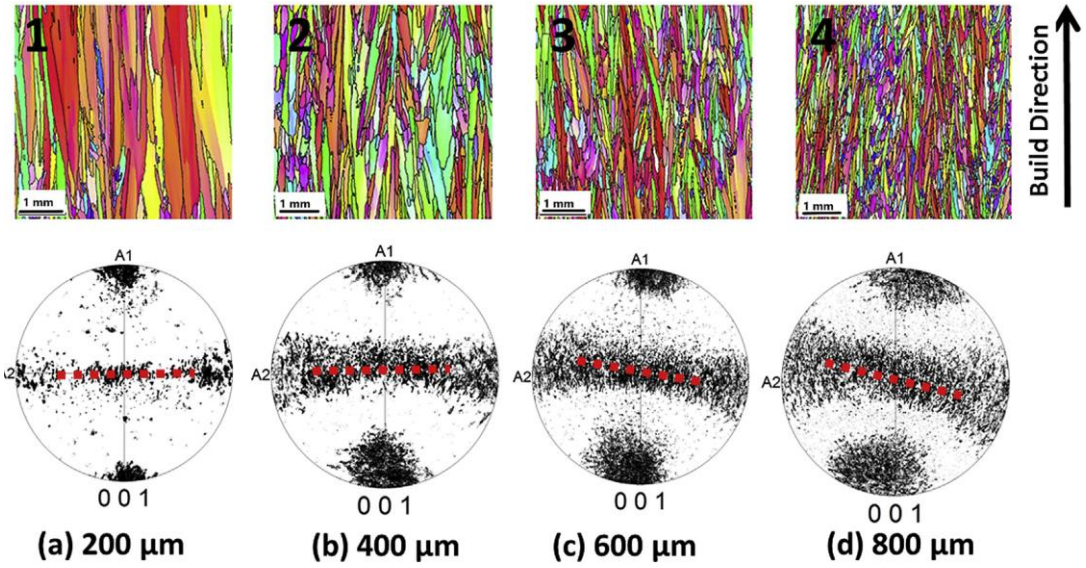


Figure 29: Grain size simulation with respect to different localized melt strategies comparing simulation (top) and experimental pole figures (bottom) [122].

CHAPTER 2

Methodology

The complex physics associated with the solidification process requires solving a number of differential equations for heat transfer, diffusion, fluid flow, and phase transformations. The numerical methods for solving these equations often include finite element (FE), finite difference (FD), finite volume (FV), and lattice Boltzmann (LB). The first three are compared with respect to each other, due to the many similarities shared between them, while LB is featured separately. Solving the transport phenomena can be accomplished in either mesoscale (FE, FD, FV) or microscale using molecular dynamics (MD). Solving a large-scale problem is impossible using this MD method; however, LB bridges the gap between the two by not considering each particle's behavior in isolation, as per MD. LB uses a distribution function to represent behavior properties of a collection of particles.

The conventional method for mesoscale is usually divided into two approaches, continuous or discrete. Using the continuous approach, an infinitesimal control volume and the conservation of energy, mass, and momentum are used to obtain partial differential equations, which is difficult to solve with complex geometry, boundary conditions, and nonlinearity. This problem is solved by discretizing the domain into finite elements, grids, or volumes in a macroscopic scale. Physical properties, such as temperature, pressure, and velocity are represented at nodal points or averaged or assumed linearly/bilinearly across a finite volume/ nodal point [210]. This section covers the methodology used to simulate dendrite growth.

Finite Element/ Difference/ Volume

Many scholars have used FE, FD, FV method for solving the governing partial differential equations. Time and space are structured in a grid, where a continuous field variable is approximated at discrete grid points [141]. When extending into higher volumes, FV methodology enables conformation to an irregularly shaped grid, compared to FD method's cartesian grid. FE method divides a domain into discrete units, which is distinguished by the connectivity between the nodes and suited for irregular structures. The trade-off between resolution and computational speed is dependent on the methodology chosen.

FE method is also used to solve the governing equations for solute transport. Feng et al. [132] used FE to implement a fluid flow model for a mesoscale simulation. Utilizing the Galerkin FE method [211], an elemental matrix was developed and solved using an open access program C++ library (IML++), for solving both systematic and nonsystematic linear systems [212]. These resources have been helpful in developing models for many uses, such as quantitatively predicting the fluid flow behavior induced by solidification shrinkage [132].

FV, on the other hand, is functionally designed to work in a 3D environment. Finite volume works as an integral scheme across an area, similar to FE, in which the chance for error is minimized. Integral schemes are computationally slower than differential schemes (FD), but this is dependent on the boundary conditions. FV is a conservative formulation that allows for a mesh to apply boundary conditions for a flow [108]. Krane et al. [213] coupled the CA method with FV to study the development of artificial anisotropy in growth kinetics. For CA, the domain was divided into finite cells, where the

coupled technique tracks the solid-liquid interface in a simple capture model for various crystallographic orientations. The FV model solves the solute diffusion in 2D, while the CA tracks the growth. Coupling the growth and diffusion is effective as they are interdependent. The diffusion along the composition gradients causes departures from equilibrium in liquid composition near the interface, which drives the growth (and remelting in this case). This process is difficult as many dependent factors are necessary for consistencies, where the time step and grid size depend on both the steady state value and the composition of the alloy. The morphology with respect to time is shown below (Figure 30) featuring a grid size of 1μm. Much effort has been made to simplify this repeated process in mesoscale [214], such as creating a framework for FV [215] for casting processes (OpenCast) or integrating with sharp solid-fluid interfaces with the Eulerian-Lagrangian framework [216]. Navier-Stokes is unable to simply combine thermodynamics with the source term, therefore, FV is a common strategy for integrating these extra physics. LB, however, can naturally inject complex physics of combining thermodynamics with the source term into the model, such as phase changes.

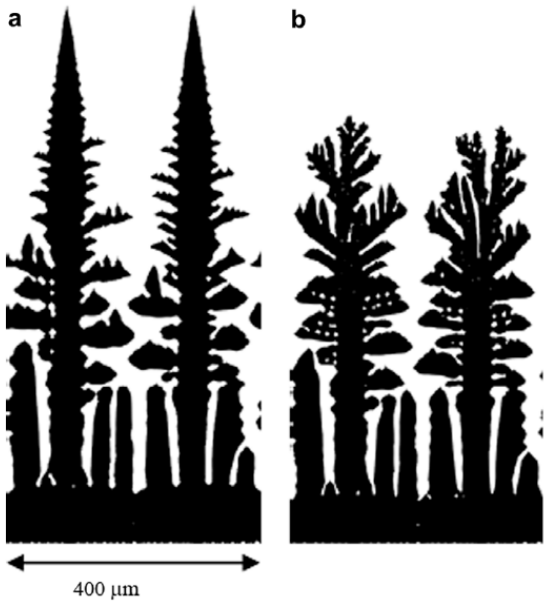


Figure 30: Comparison of identical 2D, FV morphologies of Pb-5 wt% Sn alloy simulations using different time steps: (a) 0.3 ms and (b) 0.8 ms [213].

As an alternative to LB, finite methods have provided useful integrations for modeling fluid flow with respect to meshing. Research has shown it to be robust and computationally efficient, allowing for simplification of tracking the movement around the interface. Integration has been applied as a moving mesh algorithm for quantitative PF equations by Karma et al. [217]. The sharp interface equations define the dendrite evolution in terms of the diffusion equation, which is expressed in a moving frame. Li et al. [218] expanded on the moving mesh framework for 3D multiphase flows. This adaptive grid method is useful for the disparity between the relatively small thickness and global length scale, which requires a locally refined grid inside the interface. This research has inspired others to utilize these methods for greater potential. Wang et al. [219] used these adaptive FE methods to measure the scalability of these domains. Comparing the tip velocities for undercooling cases, complex dendrites were studied for

the potential to simulate more realistic physical problems (with a CPU that required about 60 hours of time for 47,905 nodes). This numerical technique presents a framework for multicomponent (MC) alloys. The previously stated research has its limitations with respect to computation and time. FE has been used historically to model the transient thermal stresses and residual stresses from laser heat treatments [220]. With this context, it requires a higher degree of computation for realistic models. 3D finite element models provide a model for scalability [221]–[223], where other algorithms can enable multiscale complexity.

To solve the FD mathematical model, a 1-D heat conduction equation (1) is obtained and then extended to higher dimensions. The PF equation is solved with FD, where the temperature field is represented as a series of grid points in space.

$$\frac{\partial T}{\partial t} = \alpha \frac{\partial^2 T}{\partial x^2} \qquad 0 \leq x \leq L;\ t > 0 \tag{1}$$

In discrete time steps, we define spatial and temporal increments, where spatial derivatives are constructed using Taylor series expansions about T_i^n at a particular fixed time. Using linear interpolation between values at t_n and t_{n+1}, while separating the terms to opposite sides, producing an equation (2) for the grid Fourier number (Fo_g). There are three common schemes (Figure 31) for choosing the domain of dependence (T_i^{n+1}), which is obtained by choosing different values of ζ: Explicit (ζ = 0), Crank-Nicolson (ζ = ½), and Fully Implicit or Backward Euler (ζ = 1). Each method for selecting nodes further simplifies the equation below.

$$T_i^{n+1} - Fo_g\zeta\,(T_{i-1}^{n+1} - 2T_i^{n+1} + T_{i+1}^{n+1}) = T_i^n + Fo_g(1-\zeta)T_{i-1}^n - 2T_i^n + T_{i+1}^n + \frac{1-2\zeta}{2}\,(\frac{\partial^2 T}{\partial t^2}\,\Delta t + \mathcal{O}(\Delta t^2, \Delta x^2)$$

(2)

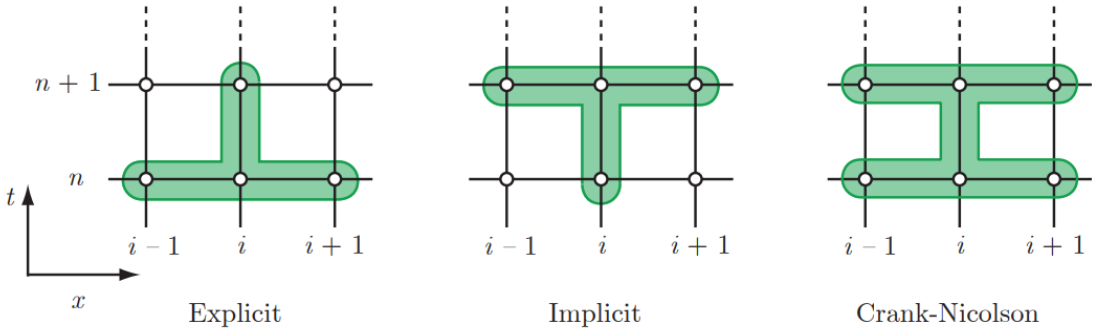


Figure 31: The three common FD schemes: Explicit (left), Implicit (middle), and Crank-Nicolson (right) [141].

Expanding the FD scheme to higher dimensions is straightforward, where the time derivatives are evaluated in the same way as in the one dimension. The FD representation of the Laplacian is new, which can expand using techniques such as the five-point stencil (3).

$$\nabla^2 T = \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = \frac{T_{i-1,j} - 2T_{i,j} + T_{i+1,j}}{\Delta x^2} + \frac{T_{i,j-1} - 2T_{i,j} + T_{i,j+1}}{\Delta x^2} + \mathcal{O}(\Delta x^2)$$

(3)

The algorithm for the simulations consists of a combination of PF and LB methods. The solute transport/diffusion equations are solved using FD method, whereas the fluid flow equation is solved using the LB method. The PF variable, which describes the evolution of the solid/liquid interface, is either represented as φ = 1 for a solid region

or φ = -1 for a liquid. PF can be solved using the method described above, which is used to solve differential equations (4).

$$\tau \frac{\partial \varphi}{\partial t} = \nabla \cdot (W^2 \nabla \varphi) + \frac{\partial}{\partial x}\left[W \frac{\partial W}{\partial \theta} |\nabla \varphi|^2 \right] + \frac{\partial}{\partial y}\left[W \frac{\partial W}{\partial \theta} |\nabla \varphi|^2 \right] + \partial\, \partial z \left[W \frac{\partial W}{\partial \theta} |\nabla \varphi|^2 \right] - \frac{df(\varphi)}{d\varphi} - \lambda \frac{dg(\varphi)}{d\varphi} u \quad (4)$$

Lattice Boltzmann

LB method is a relatively newer approach for solving the solute transport and fluid flow. It is different from conventional modeling techniques, as the system is modeled as a collection of particles moving on the discrete computational space's lattices. It is powerful for simulating both single and multiphase flows in complex geometries, which gives it advantage over the conventional Navier-Stokes solvers [224]–[227]. It is very efficient in describing the fluid flow computationally [228] and coupled with different interface capturing methods. Interesting observations about the movement of dendrites can be made using large-scale LB models, such as translation and rotation [131], [229]–[231].

LB has been an important tool for our research on large-scale simulation of dendritic solidification. Jelinek et al. [232] were able to visualize the flow of Al-3 wt% Cu alloy melt in 2D (Figure 32). The arrows represent the velocity vectors of the melt, while the contours represent temperature, where cooling occurs at the front and back as heat is applied at the sides. By incorporating effects of melt convection, solute diffusion, and heat transfer, LB was able to be scaled using MPI and matched to CA using an identical mesh. LB's simple formulation is split into two steps: collision, which is completely local, and streaming, where MPI is used to transfer the distribution functions.

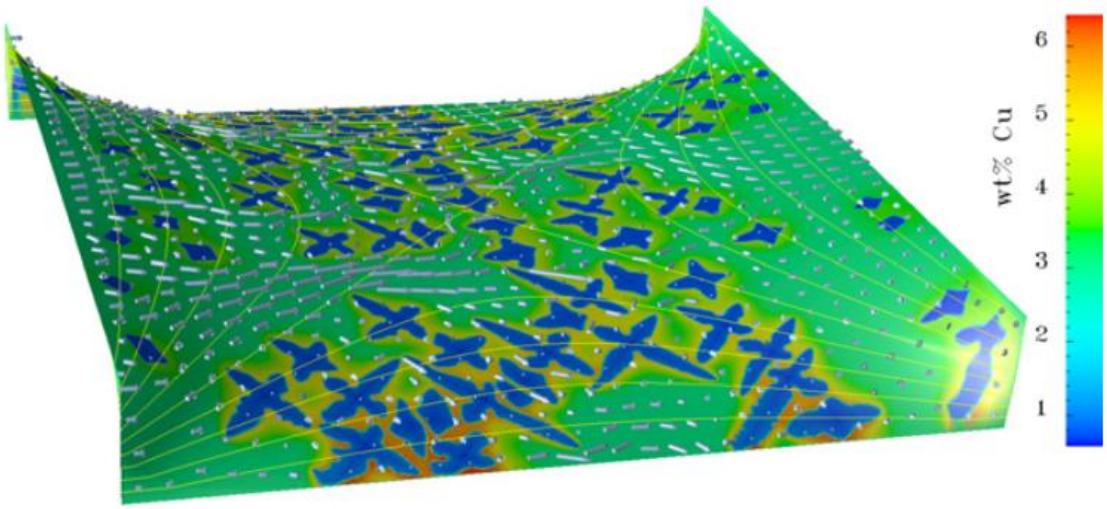


Figure 32: 2D simulation of flow melt around solidified dendrites, where colors represent solute concentration and contours represent temperature. [232].

LB is very suitable for parallel processing, as this method doesn't rely on the assembly of a large global matrix, which makes CA a natural approach for coupling. However, LB has been coupled with PF too [143], [229], where the equations of motion are solved for tracking the translational and rotational motion of the solid phase. Medvedev et al. [233] proposed a mesoscopic scheme to simulate dendritic solidification with both motion and rotation of grains, which laid the framework for larger-scale simulations. This scheme was translated into 3D for multiple solid particles by Subhedar et al. [234], which optimized the diffuse interface-flow simulations. Software such as OPENPHASE [235] utilize PF and LB methods to grow spherical seeds and rotate in a simple sheer flow.

LB has advantages that are clear; computations are local, and easy to handle in terms of complexity. It is efficient for parallelization, while handling accuracy, numerical stability, and constitutive versatility. Therefore, the transport phenomena can be computed in a variety of use cases outside of fluid flow, such as reaction systems, phase

changes, material processing, as well as heat transfer. Typically, simple simulations are performed using LB; however, with greater computation power, LB can be utilized for more than one use-case in a single simulation, such as studying phase change in combination to heat transfer. Ren et al. [236] verified this with experimental temperature profiles. After decoupling, a negative relationship formed between the re-molten volume of the temperature difference, where the influence of dispositive position and relative position of the adjacent component were observed and analyzed for metal droplets deposition method (MDDM) under heat conduction. LB was used to predict the heat transfer and phase change in the multi-layer deposition. Sakane et al. [237] used a domain decomposition method to simulate the free growth of an equiaxed dendrite in a domain of $2\times2\times6$ mm^3. This allows for multiple-GPU parallelization, where the boundaries moved to divide the dendrite evenly.

When the heat transfer equation is solved in large-scale simulation for most metals, since thermal diffusivity is 100–1000 times larger than solute diffusivity, the temporal resolution required for the solute diffusion equation is too fine for energy equation [57]. If all the numerical models for simulating microstructure growth are explicit, then separate spatial scale and temporal scales for each physics can be employed to alleviate the problem and reduce the computational time. In other words, unique grid sizes and time steps can be used for different physics. The grid size and time steps are selected through the Courant–Friedrichs–Lewy condition [238] for each physics. However, the length scales should be fine enough to capture secondary dendritic arm spacing (SDAS) and inter-dendritic flow for fluid flow and solidification growth features. This will result in a much coarser grid for fluid flow that reduces the overall computational cost [291].

This method has been successfully applied by many researchers to reduce the computational time of large-scale simulation [24], [25], [191], [239]. Nabavizadeh et al. [239] studied the accuracy and computational efficiency of a multiples grid and time step scheme for a natural convection benchmark problem. They showed that by appropriate selection of the grid and time step, computational savings up to ten-fold could be obtained compared to when the same time step and grid size is used. The model is also accurate and only loses 9% accuracy for the worst case [239].

LB is based on a probabilistic method of the meso-scale, using an average length scale of particles to discretize an entire system. The Boltzmann equation solves for the transport of these particles, which is explained by a distribution function $f(r, c, t)$. The number of particles is tracked at time (t) at positions between r and $r + dr$, along with the velocities between c and $c + dc$. When setting the number of particles equal before and after an external force is applied, this can result in 2 different equations, depending on if collisions between particles takes place.

$$f\,(r\, +\, cdt, c\, +\, Fdt, t\, +\, dt)drdc\, -\, f\,(r, c, t)drdc\, =\, 0 \tag{5}$$

If collisions take place, the rate of change between the final and initial status of the distribution function is substituted with a collision operator, Ω. Applying Newton's second law to solve for acceleration, which is derived by dividing by dt. Therefore, the Boltzmann transport equation is derived (equation (6)), where the vector *r* can be expressed in the 3-D Cartesian coordinate system.

$$\frac{\partial f}{\partial t}\, +\, c \cdot \frac{\partial f}{\partial r}\, +\frac{F}{m}\, \cdot \frac{\partial f}{\partial c}\, = \Omega \tag{6}$$

The collision term is complicated, which makes the Boltzmann equation difficult to solve. Therefore, BGK (Bhatnagar-Gross-Krook) approximation is used to simplify the collision operator, substituting with new coefficients, ω (collision frequency) and τ (relaxation factor).

$$\Omega = \omega(f^{eq} - f) = \frac{1}{\tau}(f^{eq} - f), \qquad where\ \omega = \frac{1}{\tau} \tag{7}$$

Discretizing along specific directions, called linkages, rewrites the Boltzmann equation. This form is without external forces and is the workhorse for the LB method, replacing the conventional Navier-Stokes in Computational Fluid Dynamics simulations.

$$\frac{\partial f_i}{\partial t} + c_i \nabla f_i = \frac{1}{\tau}.(f_i^{eq} - f_i) \tag{8}$$

Applying this to LB can be accomplished in various lattice arrangements up to 3 dimensions. For 1D lattices, D1Q2, D1Q3, and D1Q5 (Figure 33) are determined based on the orientation of the central node with respect to the neighboring nodes. D1Q2 has no particle residing on the site, which is opposite to D1Q3 (most popular) and D1Q5.

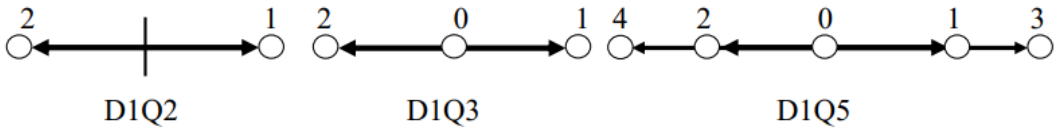


Figure 33: 1D lattice schemes: D1Q2 (left), D1Q3 (middle), and D1Q5 (right) [210].

For a 2D lattice scheme, there are three options: D2Q5, D2Q4, and D2Q9 (Figure 34). D2Q5, which is more stable than D2Q4, has four velocity vectors issuing from the central node but cannot be used to simulate fluid flow. D2Q9 is the most common for

solving fluid flow problems, having nine velocity vectors and a central particle speed equivalent to zero.

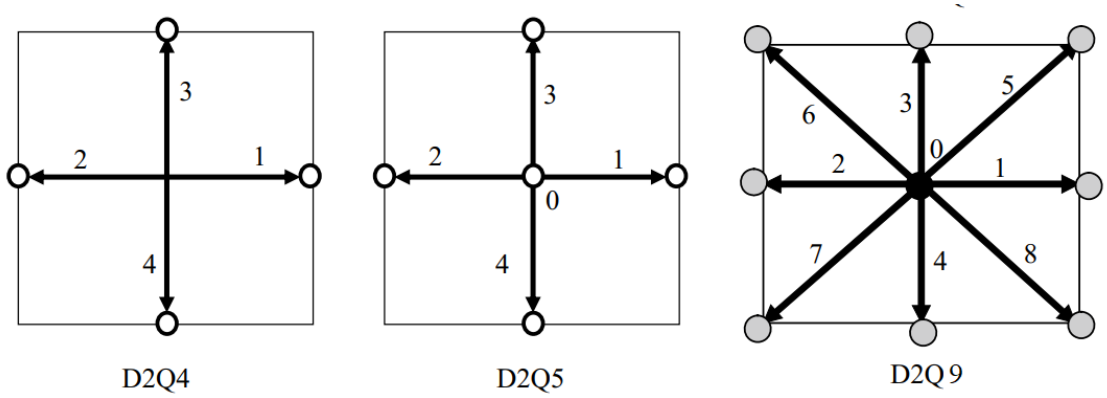


Figure 34: 2D lattice schemes: D2Q4 (left) D2Q5 (middle), and D2Q9 (right) [210].

For 3D problems, two models are more commonly used to simulate the lattice scheme: D3Q15 and D3Q19. Each has a central distribution function with a central vector of zero speed. D3Q15 uses 15 velocity vectors, while D3Q19 has 19 velocity vectors. Less lattice links can be used for energy and species conservation; however, for fluid flow, a higher number of lattice links must be used. Weighting factors are an Important consideration when choosing a lattice scheme. In comparison to FD, velocity and temperature are unknown for each node, but for LB, the distribution functions or Particles depend on the dimension and nature of the problem. For example, the isotropy

of lattices must be conserved in the selection of the weighting factor for each lattice direction.

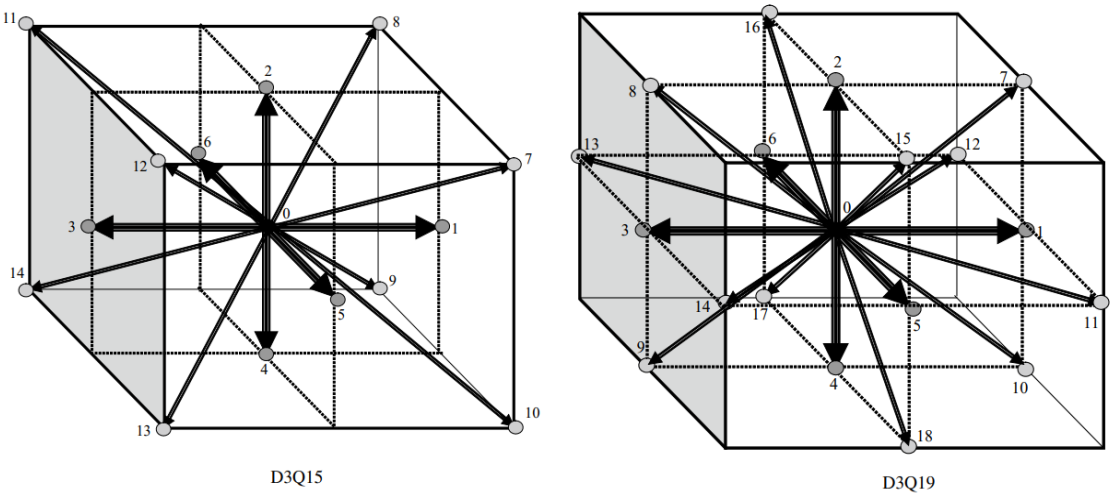


Figure 35: 3-D lattice schemes: D3Q15 (left) and D3Q19 (right) [210].

The PF can be combined with the LB method for solute diffusion. Colliding in three dimensions (D3Q15), this is represented as the following equation (9).

$$e_i = c \times \begin{cases} (0,0,0) & for\ i = 0 \\ (\pm 1,0,0),\ (0,\pm 1,0),(0,0,\pm 1) & for\ i = 1-6 \\ (\pm 1,\pm 1,\pm 1) & for\ i = 7-14 \end{cases} \tag{9}$$

The BGK approximation (10) is used to solve for the distribution function (g_i) and its evolution for the solute concentration at a particular location in space. The lattice speed (c) is calculated by using the lattice spacing (Δx) and time step (Δt), where $c = \Delta x/\Delta t$.

$$g_i(x + e_i\Delta t, t + \Delta t) - g_i(x,t) = -\frac{1}{\tau_c}\left(g_i(x,t) - g_i^{eq}(x,t)\right) \tag{10}$$

The solute transport model's relaxation time (τ_c) and equilibrium distribution function (g_i^{eq}) are used to further approximate the diffusion equation using the diffusion coefficient of liquid (D_l) (11).

$$D_l = \frac{c^2 \Delta t}{6} (2\tau_c - 1) \tag{11}$$

Thus, the equilibrium distribution $g_i^{eq} = w_i C_l$ is solved using the weight coefficients (w_i) associated for a D3Q15 lattice (12). The liquid's solute concentration (C_l) can be recovered using the summation of the distribution function $C_l = \sum_{i=0}^{14} g_i$.

$$w_i = \begin{cases} \frac{16}{72} & for\ i = 0, \\ \frac{8}{72} & for\ i = 1 - 6, \\ \frac{1}{72} & for\ i = 7 - 14 \end{cases} \tag{12}$$

The solute is redistributed between the solid and liquid phases as the dendrite solidifies, where the amount of the redistributed solute is determined using the following equation (13).

$$\Delta C_l = C_l(1 - k_e)\Delta f_s \tag{13}$$

The fraction of solid (f_s) can be solved using the PF variable (ϕ), $f_s = (\varphi + 1)/2$, which can be used to define the microstructural evolution for modeling the solidification. The following approach is computational to enable scaling and efficiency.

Computational Approaches

First, simulations were performed on a smaller scale ($0.2 \times 0.2 \times 0.6$ mm^3), which was used to determine the parameters to scale to a larger domain ($0.2 \times 0.2 \times 8$ mm^3).

This was accomplished by comparing two of the main parameters, Epsilon and Omega (Table 1), which affects the morphology of the dendrite growth. These parameters were selected in means of variation for both high and low values to ascertain which was most valuable in terms of simulating grain growth. A larger value of epsilon will produce a needle-shaped growth, while a smaller value produces a cauliflower-shaped growth evolution. Omega, on the other hand, is more closely related to timestep sizing.

Table 1: Simulation Parameters Comparison

Case Number	Epsilon	Omega
Case 1	0.085	1.0
Case 2	0.04	1.0
Case 3	0.085	1.7

Optimization was a large focus of the research, as parallel programming can specify which GPU can be used during specific processes of the simulation. The process for parallelization is accomplished by specifying threads in kernels using the CUDA programming language. CUDA is traditionally written in C++ but can also be wrapped in Python using PyCUDA. The code consists of parameters, which are declared using Python, while the kernel code is written in C++. The parameters consist of number of particles per lattice row (for x, y, and z directions), mesh size (dxx), Peclet number (time frequency), omega, and epsilon. The process used the nohup command to run the process to completion in the background. The CUDA framework is parallelized, which means when the kernels are deployed into the GPU. This frees the CPU for general computations, such as initiating another kernel execution in another GPU.

Measuring permeability was performed using several techniques, to validate the predictions. First, permeability must be measured using simulated morphology. As part of the process, a dendrite was simulated in evolution. For the first measurement, an equiaxed dendrite was simulated in a simplified form, as a sphere. This sphere was simulated in an evolution by using various sizes form 1.5 - 120 μm. The measurement of permeability was performed using COMSOL and a LB method using two different domain sizes: large (300μm) and small (30μm).

For the second simulation, the large-scale, columnar dendrites that were simulated from the prior experiments were used. The output from the simulations produced a visualization toolkit (vtk) file that contained information about the solute concentration, temperature, and the phi. To validate the predictions, the outputs need to be converted to stereolithography (stl) file format. This was accomplished using Paraview, which allowed triangulation on the contour of the phi. The resulting output was imported into Solidworks, which was used to create a bounding box, which serves as the domain for the fluid flow. This was also imported into COMSOL to measure the permeability using an alternative method for validation purposes.

CHAPTER 3

Results And Discussions

The procedure for measuring permeability of a solidifying alloy required three main stages, which will be discussed in more detail in the following sections. First of which, simulating the dendritic growth was the most time consuming. Secondly, optimizing computations was a requirement to scale. Lastly, measuring permeability was the goal, which required the previous stages.

Simulating Dendritic Growth

Results

The simulations were performed on a GPU server, equipped with an Intel C620 CPU and 3 GPUs: Tesla V100S 32GB. Each GPU has 5120 cores, which clocks at 1245 MHz frequency. The material modeled was Al-3wt%Cu alloy, where the material properties are shown below in Table 2.

Table 2: Al-3wt%Cu material properties

Symbol	Definition	Value
ρ	Density (kgm^{-3})	2475
μ	Viscosity (m^2/s)	0.00014
D_l	Solute diffusion in the liquid ($m^2\ s^{-1}$)	3.0×10^{-9}
m_l	Liquidous slope ($Kwt\%^{-1}$)	-2.6
k	Partition coefficient	0.14
Γ	Gibbs-Thomson coefficient (Km)	2.4×10^{-7}
T_m	Melting Temperature of pure Al (K)	933.47

The first series of simulations was of the smaller scale ($0.2 \times 0.2 \times 0.6$ mm^3), which demonstrated the efficiency of the simulation. However, due to the scale of the simulation, higher-order dendrite arms failed to form, which is shown below (Figure 36).

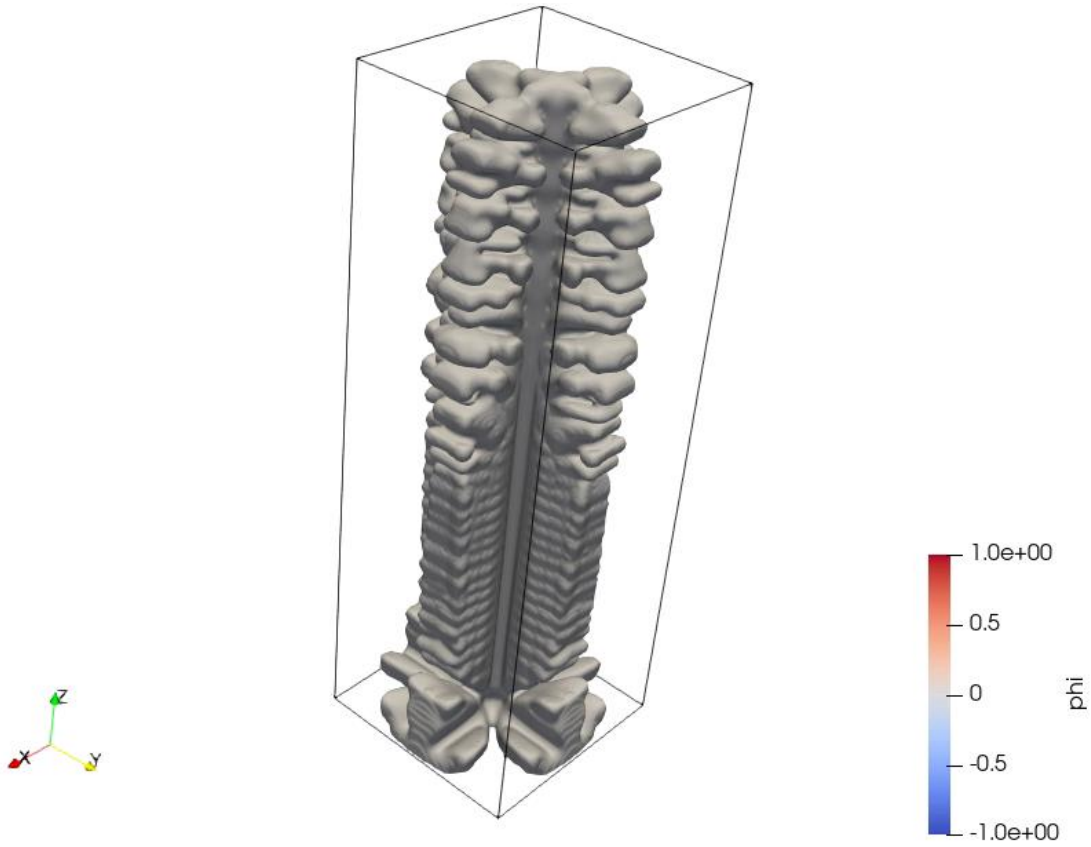


Figure 36: Simulation of smaller scale showing secondary dendrite arms forming tertiary arm growth at approximately 5.96 seconds.

Both small- and large-scale simulations were performed on a single dedicated GPU. The large-scale simulation was scaled up from the small-scale simulation in the z-direction by a magnitude of 13.3x ($0.2 \times 0.2 \times 8$ mm^3). For a total of 634,000,000 time-steps and a time-step size of 0.125 microseconds, which equals 79.25 seconds of physical time, the simulation produced 1585 outputs. The total amount of compute time for

performing the simulation for Case 1 was approximately 487.5 hours. Initially, TECPLOT (tec) files needed to be converted to vst format manually; however, after modifying the code, the outputs were automatically exported in readable format (vst). A second constraint was the pickle (pkl) files, which were used to serialize the file on disk for a quicker runtime, were necessary to start the next iteration, which saved the parameters. This accumulated and consumed valuable space, which required the code to be modified to delete the previous output's predecessor pkl file. Afterwards, the process for completing an iteration required little intervention, except to modify the shell script with the previous output number to direct the code to the correct pkl filename.

The results have been compiled from many hours of simulation time. The simulations produced details that are shown below. Case 1 features images of both solute concentration and phi. The first image (Figure 37) features solidification at 131,172,413 iterations with secondary dendrites visibly formed around the primary arm. The simulation shows holes in the bottom as the solidification has a phi value not within the range, which is an artifact of the visualization. This is due to solid touching the domain walls.

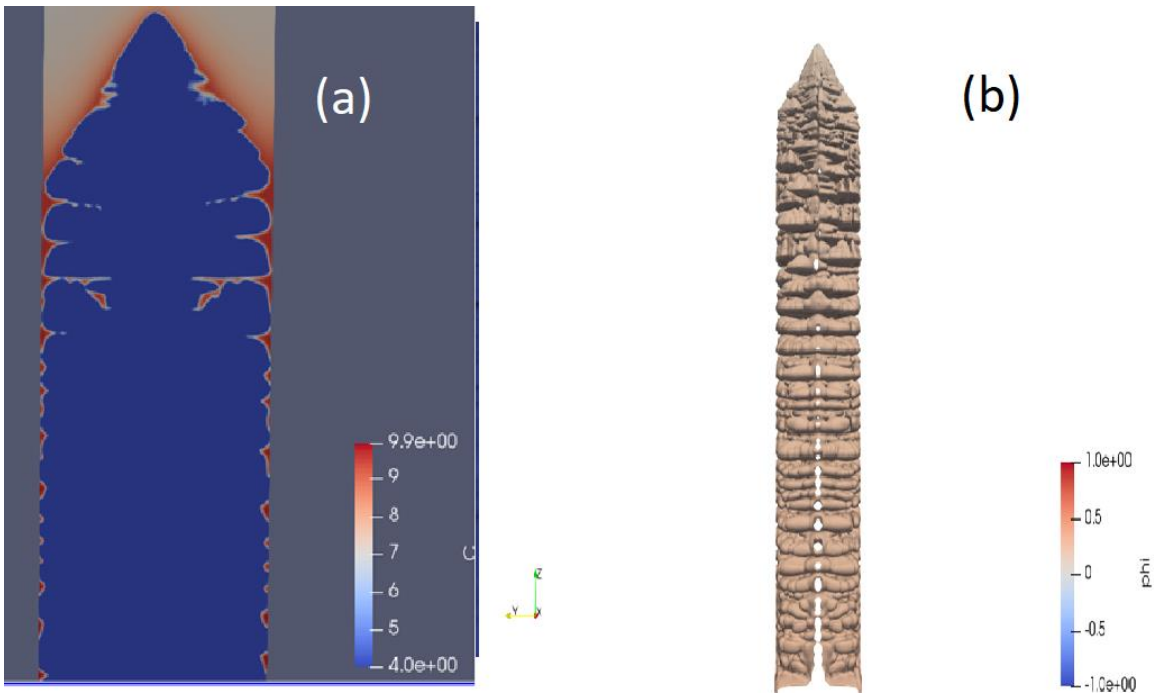


Figure 37: Case 1: Visualization of (a) Solute Concentration profile (b) Phi at the 131,172,413th iteration, approximately 16.4 seconds.

The second figure (Figure 38) compares the full-size image to a close-up of the Case 1 dendrite at the 240th output, which is equivalent to approximately 262,344,827 iterations. The dendrite features a coarse surface with solidified secondary dendrites. The top appears flat rather than pointed as seen in early stages of Case 1. Many of the secondary dendrites that were visible in the earlier stages have solidified into the primary dendrite.

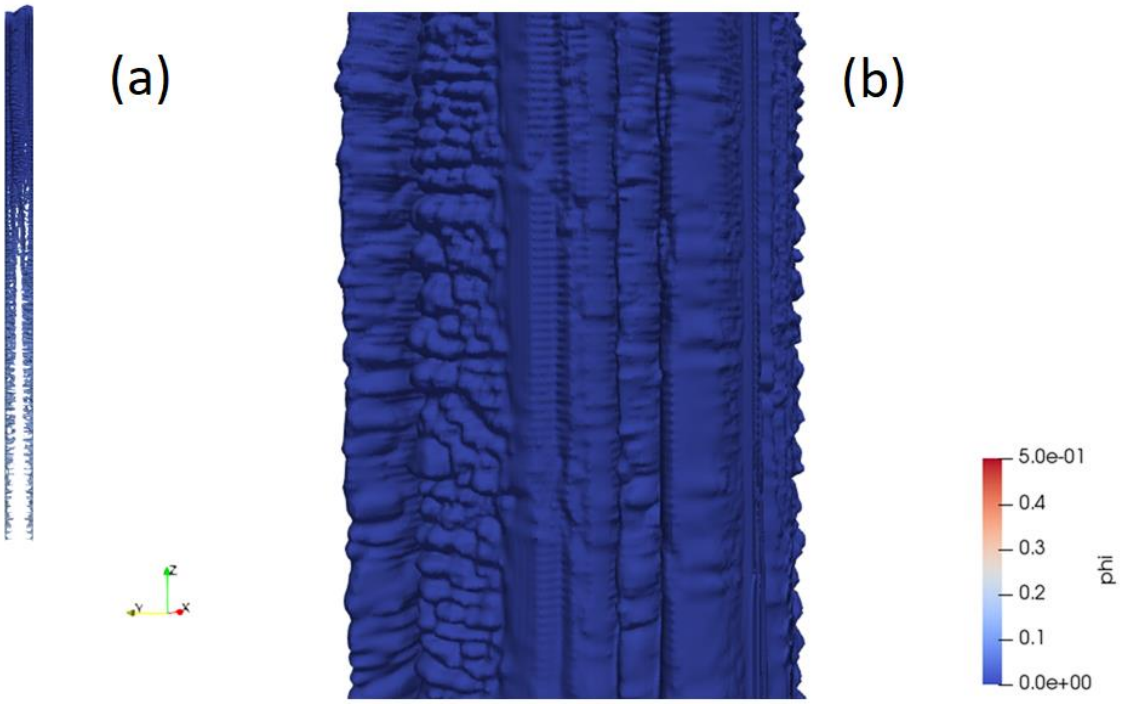


Figure 38: Case 1: 0.2 × 0.2 × 8 mm^3 Visualization of Phi at 262,344,827 iterations (32.8 seconds) (a) Full size (b) Close-up.

The solidification at 634,000,000 time-steps of Case 1 (Figure 39) produced simulation results similar to the first image, but to a larger extent. The domain has nearly doubled in size over 79.25 seconds. When interpreting the results, the levels of phi can be adjusted to change the visual appearance, which is featured in the full-size image. The more solidified portion is highlighted on the right, while the left showcases the mushy zone towards the top.

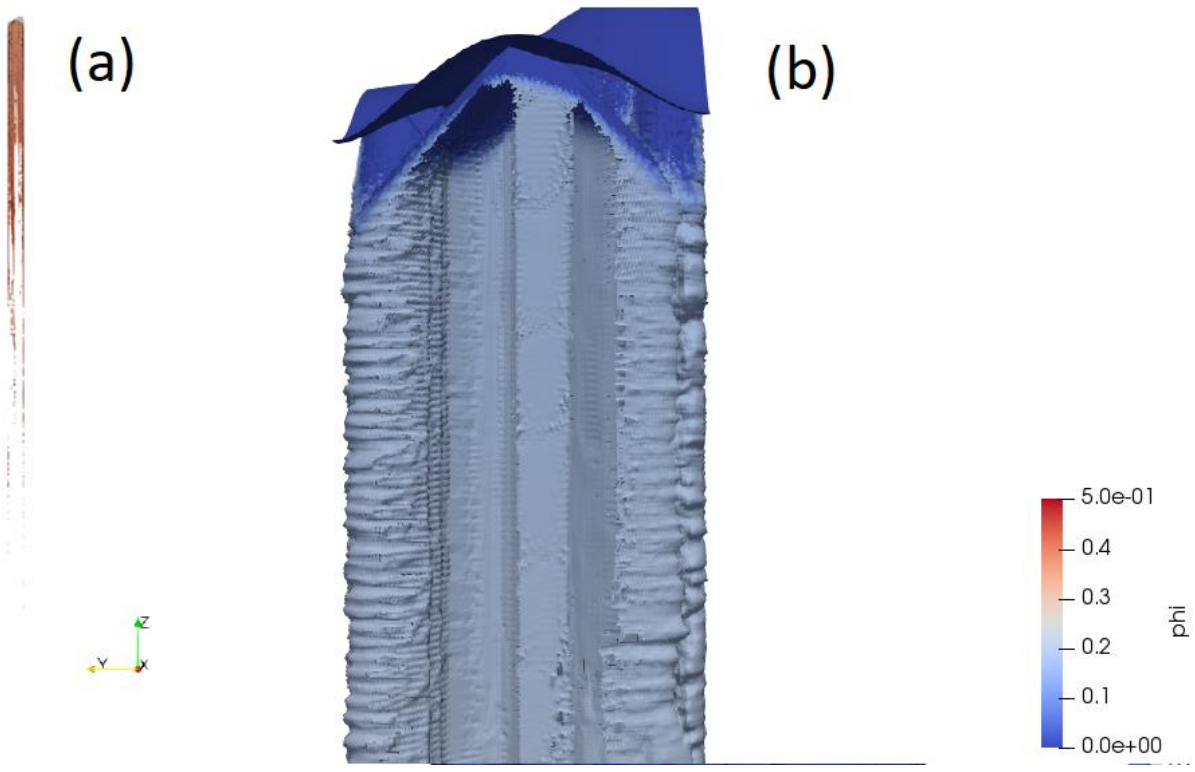


Figure 39: Case 1: 0.2 × 0.2 × 8 mm^3 Visualization of Phi at 524,689,654 iterations (79.25 seconds) (a) Full size at upper level of Phi (b) Close-up at lower level of Phi.

At the 166th output of 181,455,172 iterations, Case 2 presents some interesting results. Case 2 represents an epsilon of 0.04 (smaller than Case 1) and Omega of 1.0 (same as Case 1). The first image below (Figure 40) compares the phi and solute concentrations for both full size and close-up. The right image shows solute concentration, in which solute is in higher concentration down the middle. This is distinctly different from the evolution of the microstructure of the first case.

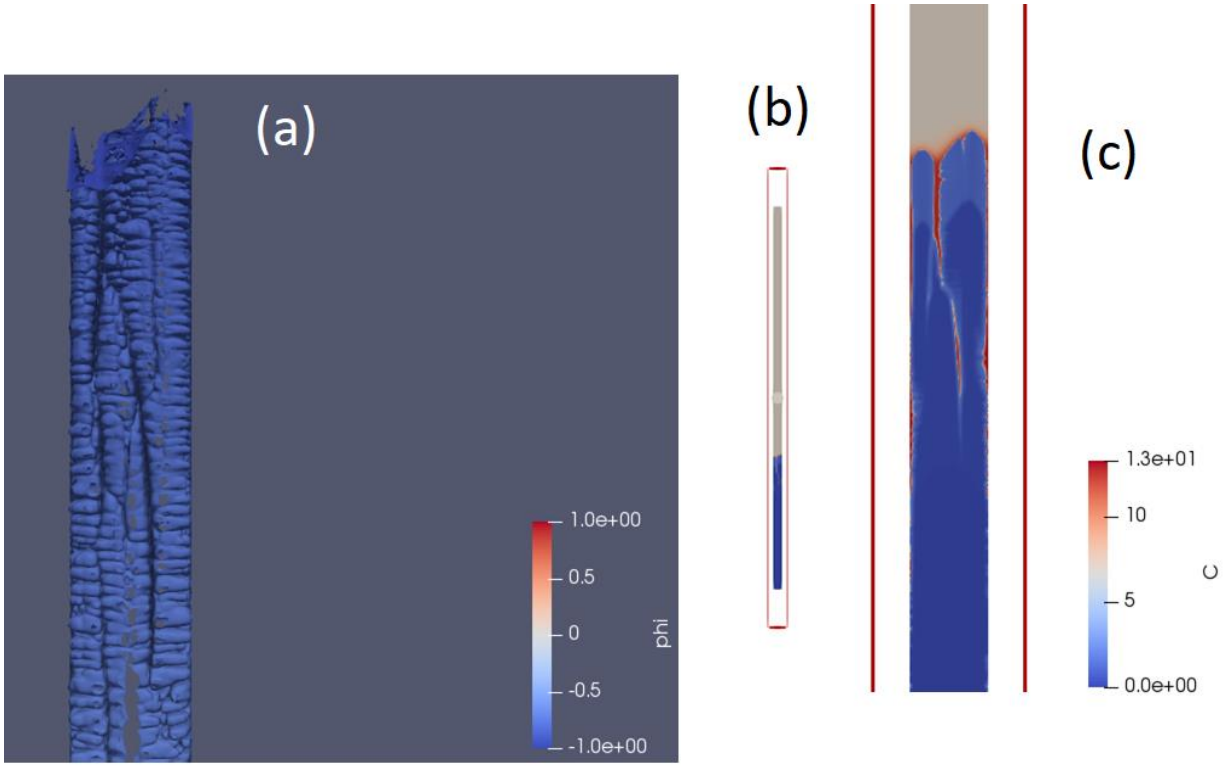


Figure 40: Case 2: 181,455,172 iterations (22.7 seconds) comparing (a) phi and (b) full scale C and (c) close-up of concentrations of solute.

At about 268th output for the second case (292,951,724 iterations and 36.6 seconds), the results showcase the phi visualization, which (Figure 41) shows secondary dendrites with tertiary dendrites protruding around the top. This visualization serves as a good comparison to the previous visualizations, as setting the limits for the phi value can produce uniquely different visualizations as the artifacts of the visualization software.



Figure 41: Case 2: 292,951,724 iterations (36.6 seconds) simulation output of Phi measuring the upper limit and exhibiting tertiary dendrites.

At the 380th output (415,379,310 iterations and 51.9 seconds) of the Case 2 simulation, the phi was compared with the solute concentration (Figure 42). This showed a similar pattern as shown at 22.7 seconds. The color of the phi shows a lighter blue, which is expected as solidification is more progressed. The full-scale image shows the progression of the solidification towards the top of the domain. The close-up shows the solution concentration being high towards the middle.

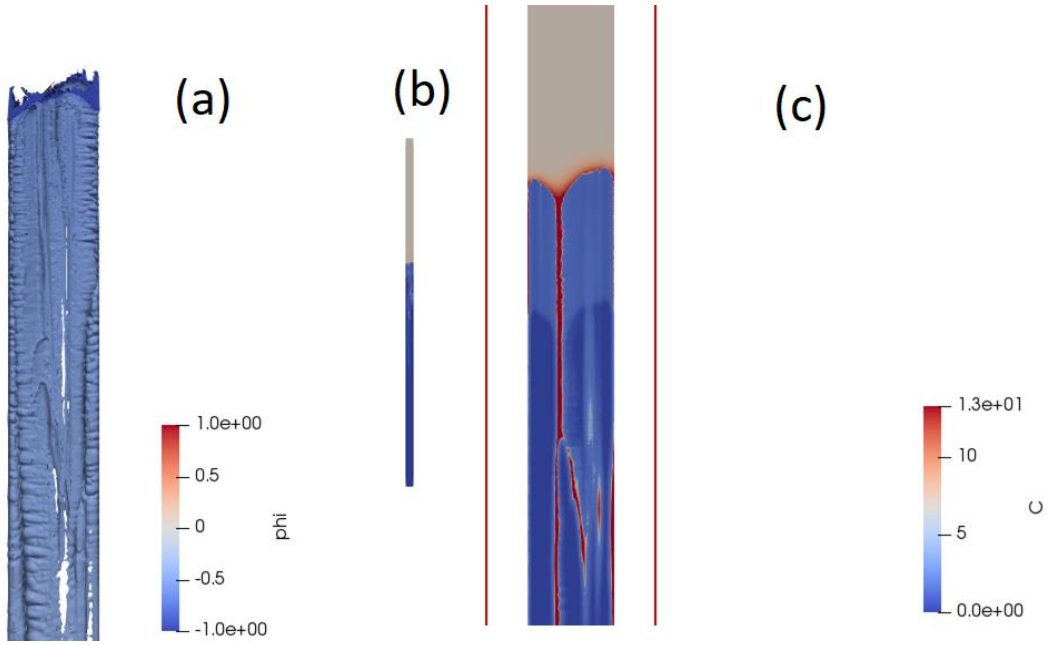


Figure 42: Case 2: 415,379,310 iterations (51.9 seconds) of dendrite growth comparing (a) phi, (b) full-size and (c) close-up of solute concentration.

The final image series of 550,924,137 iterations and 68.9 seconds (Figure 43) for Case 2 shows a similar comparison of phi and solute concentration. First, the phi produces similar solidification microstructure, as shown in the previous image; however, the gaps have fully solidified, showing a smoother exterior. The full-scale image of the solute concentration shows the domain nearly solidified. The close-up shows the gradient more clearly, where the characteristic red is slicing through the middle, representing the mushy zone.

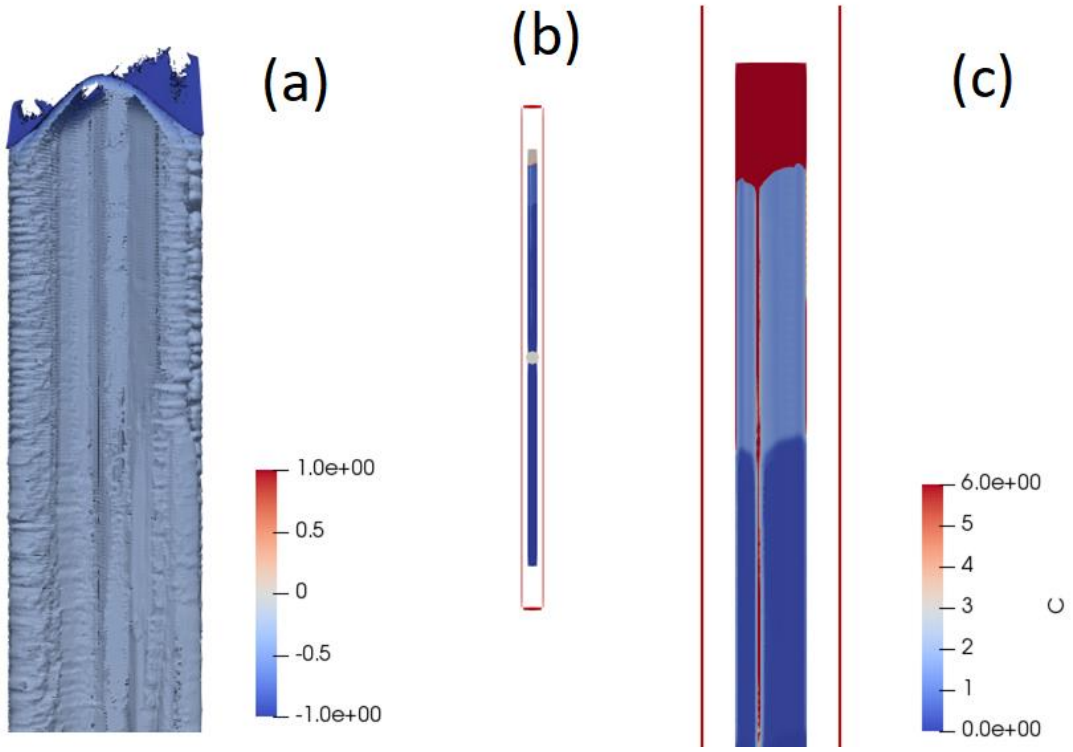


Figure 43: Case 2: 550,924,137 iterations (68.9 seconds) simulation of dendrite growth comparing (a), (b) full-size and (c) close-up of solute concentration.

Lastly, case 3 took an extensively longer amount of time, as the simulation was only commenced for about 100 outputs (109,310,344 iterations and 13.6 seconds). The output was visualized using a phi method. The dendrite is yet to form, while the results of the previous cases produced larger formations nearly 5x the rate, which is due to selection of omega that affects the time step size. The phi simulation results (Figure 44) shows a gray seed forming into a dendrite.

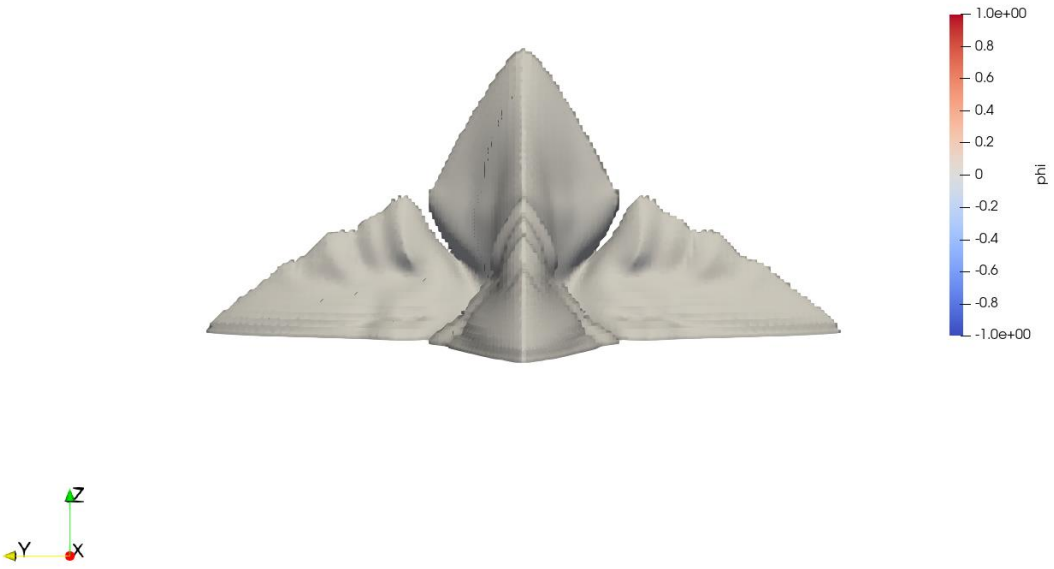


Figure 44: Case 3: Solid formation (Phi) from seed at 109,310,344 iterations (13.6 seconds).

Discussion

As mentioned before, in comparison, the 3 cases were compared to illustrate the differences between the values of omega and epsilon. In this progression, case 3 has only just begun to form, while the two others have shown more progress. The first case shows a sharper front (as expected for a larger epsilon value), while the second case has a flatter front. This comparison illustrates the effectiveness of each of the parameters, which is shown below (Figure 45). The solid front is circled as the main focus of comparison.

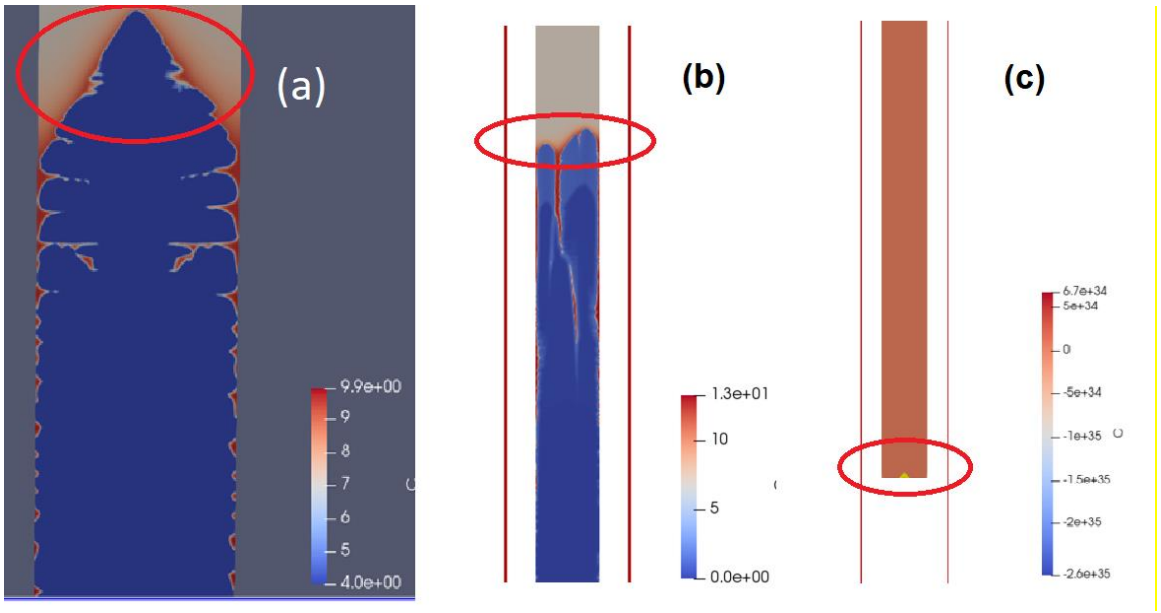


Figure 45: Comparing the solute concentration for each of the cases (a) Case 1, (b) Case 2, (c) Case 3 at approximately 16 seconds.

Optimizing Computations

Results

By optimizing resources, scaling to larger supercomputers is possible. It was attempted to utilize 3 GPU concurrently. However, this requires understanding of how to specify the kernel operations for the modeling of the dendrite morphology and the solute transport. In order to visualize the operations, NVIDIA Visual Profiler (NVVP) was used to understand the current kernel distribution, shown below (Figure 46). The runtime API shows which processes are being run, while the streams show the concurrent kernels. However, only one thread is deployed, which means only one API is running at a time. By specifying additional threads and deploying kernels across GPUs, then scalability is achieved.

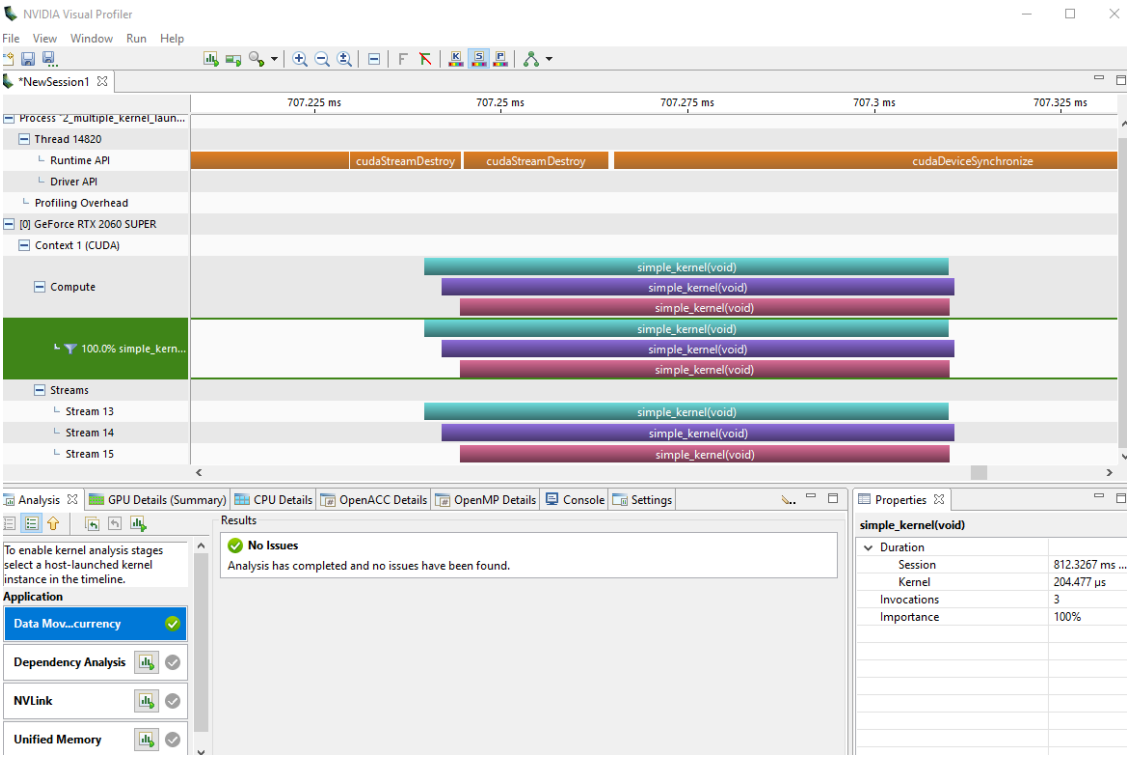


Figure 46: NVIDIA Visual Profiler output for visualizing simulation process.

Discussion

Research into programming each of the GPUs to work in parallel was considered using NVLINK, which is a new technology for high-speed GPU-to-GPU interconnect. This was introduced with the Nvidia RTX GPU series, which removes the blocking that occurs with inter-GPU communications. However, this provides little utility for shared threading for the algorithms. The shared processes would require combined compute power, which is obtained using the MPI programming method. MPI requires understanding of programming using C# and was planned for future optimizations. However, integrating CUDA with MPI is compatible and would be ideal for future development.

Measuring Permeability

Results

Prior to performing the analysis on permeability, in collaboration with a research group in Germany, a csv file was provided for a simulating a LB flow through a microstructural surface. This was modified using Python (Figure 47) and inserted into the LB code, and then, converted into a stl, which was imported into Paraview to visualize the relationship between the flow and the solid. The results were recorded in the image below (Figure 47). The higher density flow (red) shows the flow having higher magnitude towards the center of the object. This is expected, as the object curves away from the center. The blue color of the flow extends around the edges, closer to where the solid is blocking the flow.

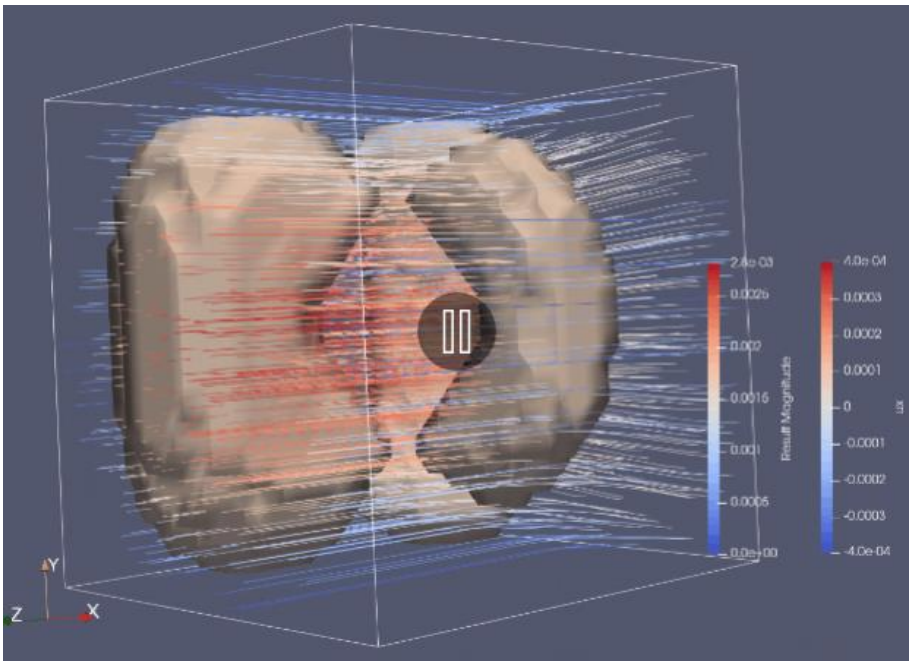


Figure 47: Permeability visualization of a simple spherical dendrite.

A solidified porous medium (240 × 240 × 120 μm^3) was obtained to perform an analysis of the permeability. After completing the simulation, the results were visualized

in Paraview. The results (Figure 48) show a flow (red and blue lines) moving horizontally in the x-direction through pores of a solidification microstructure. The visuals show similar results as expected, as the velocity is higher in places where less blockage is caused by the solid. This was accomplished with the goal to implement a stl conversion for the LB simulation.

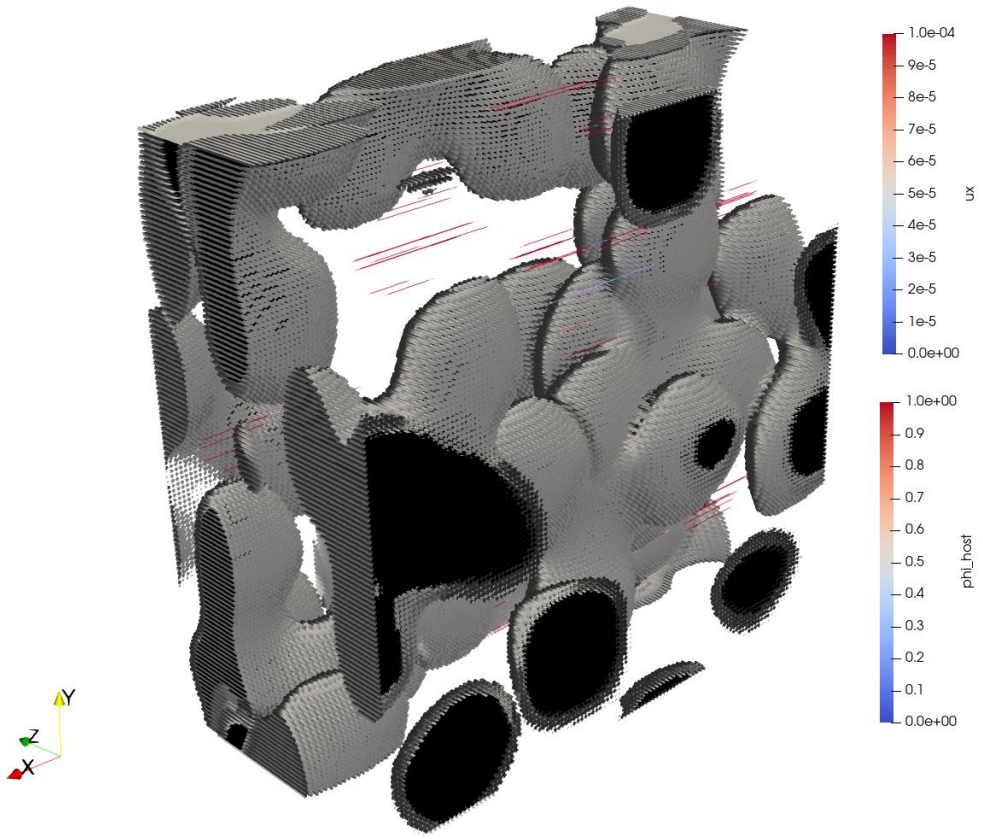


Figure 48: Visualizing permeability through a porous media in the x-direction.

Discussion

Permeability was measured using two methods: COMSOL and a lattice Boltzmann Method (LBM) generated solution. This was used to compare permeability around a sphere. Using a range of sizes for the sphere with consistent mesh size of 1 μm,

the permeability was compared for two different domain sizes: large (300μm) and small (30μm). The first, which is featured below (Table 3), shows the inverse relationship between the permeability and radius. The data used for permeability using LBM was adjusted to match the units in COMSOL. The difference between the two was measured and recorded.

Table 3: Large Domain Dataset Comparison of COMSOL and LBM

Radius (μm)	LBM (m^2)	COMSOL (m^2)	Difference (%)
15	2.99E-09	1.67E-09	44.35
30	2.06E-09	1.36E-09	34.22
60	8.66E-10	7.90E-10	8.82
75	5.09E-10	5.61E-10	-10.16
80	4.18E-10	4.93E-10	-18.06
90	2.69E-10	3.74E-10	-39.28
100	1.60E-10	2.74E-10	-71.11
120	3.92E-11	1.29E-10	-229.12

The results are also shown below in Figure 49. The results compare the radius of the equiaxed dendrite to permeability. As the dendrite grows in a fixed domain size of 300 μm, the permeability is measured in comparison between COMSOL and LB method. The results show a high difference at the largest value of radius, which is deceiving from the chart. As the permeability values decrease, the difference grows substantially as the

chart does not account for incremental changes in scale. However, validating with the table above, the closest approximation is at 60 µm.

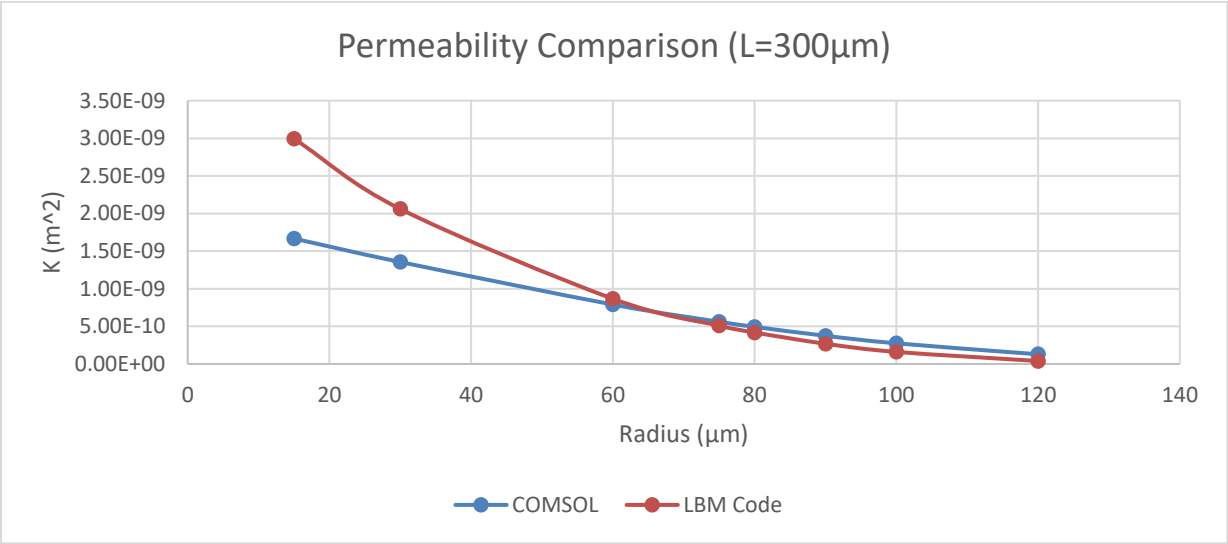


Figure 49: Permeability Comparison for a Large Domain (L=300 µm).

The results from the small domain (30µm) are shown below (Table 4) with similar comparison of COMSOL and LB method as the large domain. The difference is the greatest at the smallest radius, while a larger radius performs best, which is contrary to the smaller domain size comparison.

Table 4: Small Domain Dataset Comparison of COMSOL and LBM

Radius (μm)	LBM (m^2)	COMSOL (m^2)	Difference (%)
1.5	3.95E-11	1.67E-11	136.43
3	2.60E-11	1.36E-11	91.36
6	1.23E-11	7.91E-12	55.73
7.5	8.20E-12	5.62E-12	46.07
8	6.41E-12	4.94E-12	29.76
9	4.36E-12	3.76E-12	16.06
10	2.94E-12	2.75E-12	6.60
12	1.12E-12	1.30E-12	-13.87

The figure below (Figure 50) compares the results of the small domain datasets from the table above. The chart shows a similar exponential decay as the large domain datasets. However, the results converge towards the larger radius. If the domain size is too large for the object, the results show a greater difference in permeability between the two methods.

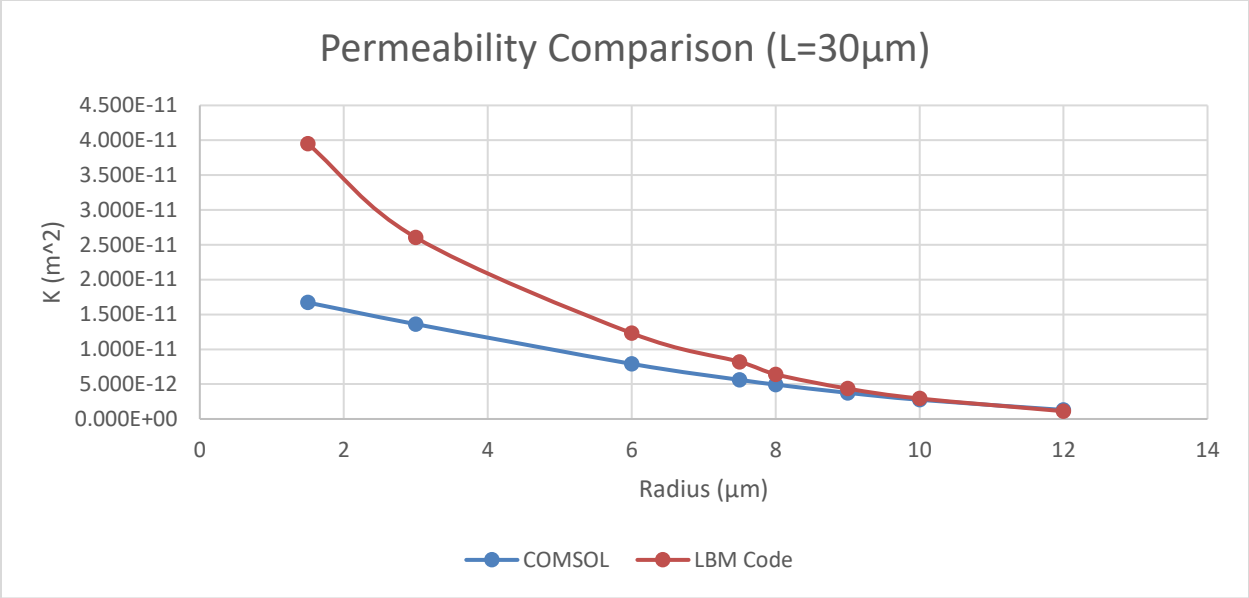


Figure 50: Permeability Comparison of COMSOL and LBM for a Small Domain Dataset.

The process for determining the permeability was a challenging exercise for the procedure we chose to use, which required methods to import a stl format as a Solidworks part (sldprt) format. COMSOL required a bounding box for producing a flow around a solid, therefore, the supported file format for importing the two solids was restricted to Initial Graphics Exchange Specification (IGES) format. This wouldn't be an issue if the conversion of the stl to a sldprt was possible. However, under much effort, the "stl" was only able to be imported as a surface part. This doesn't allow for modifications to the imported part; therefore, the bounding box was unable to perform the cavity, which would conform the negative space from the dendrite into the bounding box. Alternative methods include joining or combining the two bodies and removing the dendrite. The normal parameters were removed from the vtk format when exporting the dendrite from Paraview. Additionally, to reduce the memory consumption to import the stl, the dendrite was split into 16 equal parts along the z-axis (top segment shown below in Figure 51).

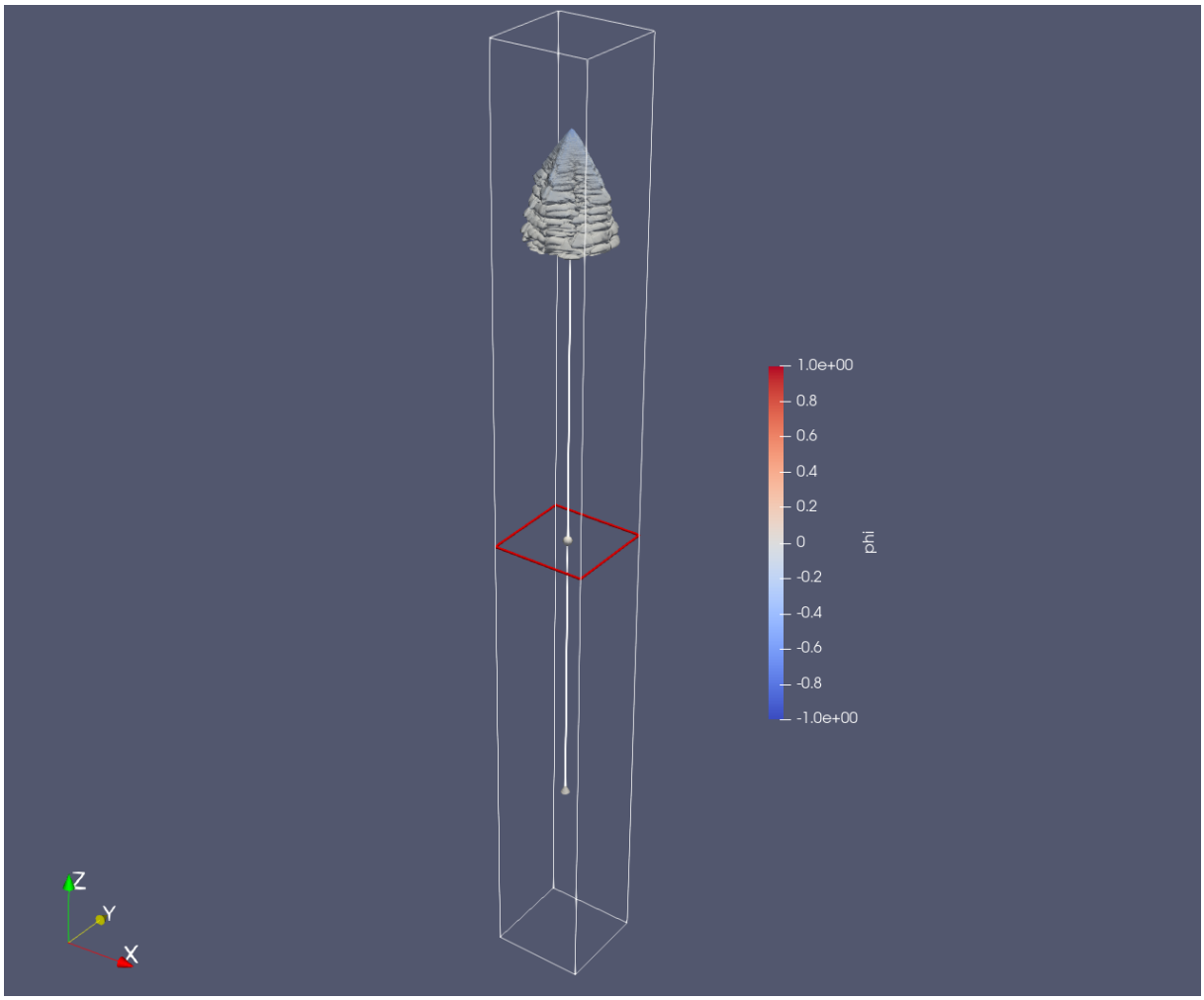


Figure 51: Dendrite split into segments (top segment isolated).

The import continued to have the same error. Therefore, it was concluded that algorithmically inserting the results into the simulation is the most suitable method for performing the calculation of phi. This was attempted without full completion, where the algorithm was included in the appendix (Code A 2). For demonstration, an “stl” of a cat (Figure 52) was converted to points, which can be interpolated into a machine-readable format for the LB code.

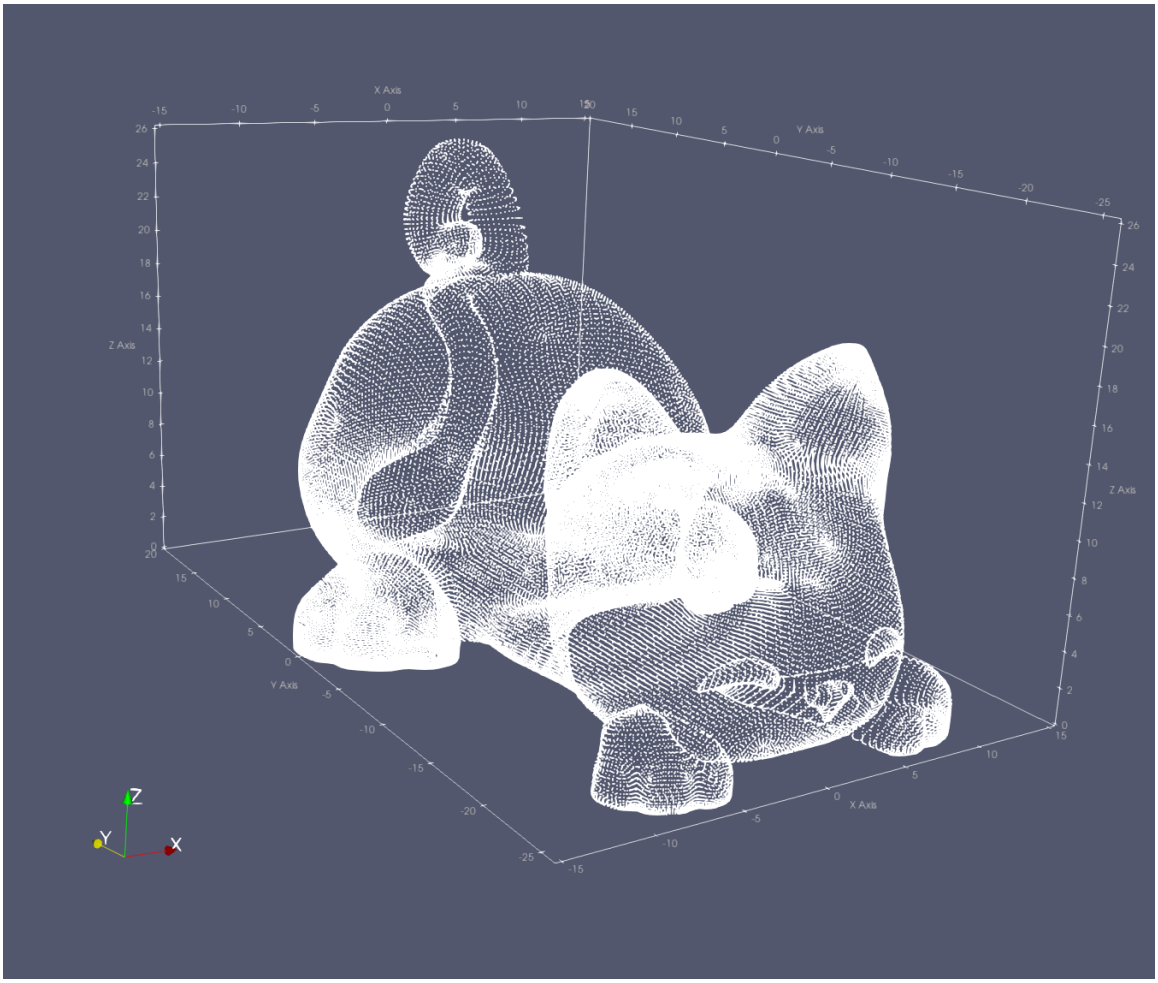


Figure 52: “Stl” cat conversion to points for LB flow analysis.

Alternatively, the stl conversion was completed using another opensource voxelization software called “binvox” [240], which used a simplified polygon model using volumetric technique [249]. The application produces a bounding box automatically, while the output can be specified in vtk format. This is effective for importing directly into the LB algorithm. This process needs more validation to be implemented; however, it shows great promise to automate the implementation of flow analysis using LB method.

CHAPTER 4

Conclusion

The applications of simulating the evolution of microstructure during solidification are as diverse as the observable features. The large-scale simulations require extensive computing power; therefore, a combination of hardware and numerical techniques are necessary to enable scaling. Employing the most efficient numerical methods together with communication techniques over a large number of processing units, the research into solidification simulations resulted in innovative methodologies enabling investigation of various features during alloy solidification.

To enable efficient simulation of solidification microstructure, various numerical methods have been developed to improve the simulation results over larger domains. Cellular automaton, with its simplicity and computational efficiency, was featured for simulating dendrite growth in large 2D and 3D domains. Phase field, being the most popular, has been utilized for reproducing the physics more accurately. Although more computationally demanding compared to CA, PF has been used in the largest dendrite growth simulation to this date. Direct interface tracking and level set methods are less popular for large-scale simulations. Scaling with dendrite needle network method allows for analytical and coarse-grained models to be used in combination.

Several manufacturing and materials processing techniques can take advantage of the prediction capabilities offered by large-scale simulations of solidification microstructure. The modeling approaches still have a lot of future improvements to innovate upon, with promising developments in machine learning and computing power.

In addition, the research performed in this thesis provides much opportunity for future development.

Utilizing Cal State LA's GPU server, large-scale simulations were performed by simulating both the morphology and the transport phenomena. The results were compared using different parameters. The simulated dendrite was then used to measure the permeability of the dendrite using the lattice Boltzmann (LB) method. This was also performed using a simulated equiaxed dendrite comparing the LB results with COMSOL simulation results. This validation provided agreeably similar results for equiaxed dendrite permeability within a relative domain size.

Future Developments

There are many potential considerations to improve in the development of this research. First of which, the study of permeability can be improved by measuring the evolution of the permeability in a growing dendritic network. This can be accomplished by combining the outputs from the first code (PFLB) and the last code (LB) to calculate the permeability at different stages. This can be further enhanced using a regression technique to scale.

An interesting concept of utilizing machine learning for simulating grain growth has potential for development in the future. In 2017, Hu et al. [241] used a CA method, along with machine learning, to simulate the grain and pore growth in aluminum alloys. The back-propagation neural network (BPNN) was used to create a correlation between the solidification parameters and pore growth; however, the domain was restricted to a size of 200 x 200 μm. Machine learning has a unique potential to assist in the modeling of dendritic features and has been implemented in the prediction of secondary arm

spacing in aluminum alloys [242]. While deep learning has been used for prediction of porosity defects in aluminum alloys [243], applications with simulations have been limited. Most recently, Hu et al. [244] has used recurrent neural networks to accelerate PF predictions. By comparing different dimensionality-reduction methods, such as linear (principal component analysis (PCA) [245]) and nonlinear embedding (isometric feature mapping (Isomap)) [246] and Uniform Manifold Approximation and Projection (UMAP) [247] techniques, the latent space can preserve PF input parameters. The autocorrelation based PCA proved to be the most efficient, while a computation speedup of 3x was able to be implemented using recurrent neural network (RNN) models with fewer number of cells and a gating mechanism, such as gated recurrent unit (GRU) [248] or long short-term memory (LSTM) [249]. Figure 53 shows the implementation of the RNN model. This model can be utilized in application with our current codebase to enable predictions. Future development shows promise with integration of mesoscale simulation systems.

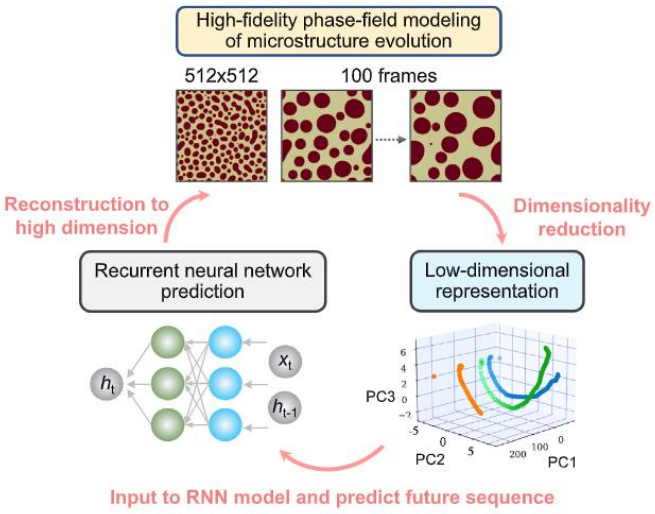


Figure 53. RNN model to predict microstructural evolution in latent space for PF modeling [244].

Another potential area for development is implementation of remelting in the simulations. The evolution of dendrite remelting consists of 4 stages, while the last stage consists of fragmentation [250] that enables further discretion of large-scale realistic simulations. An interesting phenomenon, where secondary arms remelt at the roots, detaching from the primary trunk, and moving toward unsolidified melt can be better understood in larger domains [251]. The combination of improved physics models and computational algorithms with thermodynamic databases would enable calculation of multicomponent phase equilibria, allowing for more reliable simulations for real-world industrial applications. Remelting with the study of permeability can provide higher predictability for post-processing techniques for reducing defects, such as freckling.

REFERENCES

[1] P. D. Lee and J. D. Hunt, *Hydrogen Porosity In Directional Solidified Aluminium-Copper Alloys: In Situ Observation*, Acta Metallurgica Inc, 1997.

[2] A. K. Dahle and D. H. Stjohn, "Rheological Behavior of the Mushy Zone and its Effect on the Formation of Casting Defects during Solidification," *Acta mater*, vol. 47, no. 1, pp. 31–41, 1998, doi: 10.1016/S1359-6454(98)00342-5

[3] V. Alexiades and J. R. Cannon, "Free Boundary Problems in Solidification of Alloys," *SIAM Journal on Mathematical Analysis*, vol. 11, no. 2, pp. 254–264, 1980, doi: 10.1137/0511025.

[4] Wolfgang Pfeiler, *Alloy physics_ a comprehensive reference*, Wiley-VCH, (2007).

[5] T. Aneesh *et al.*, "Exploring Casting Defects of AA7075 Alloy in the Gravity Die Casting Simulation of an IC Engine Block," *Proceedings of the Institution of Mechanical Engineers, Part E: Journal of Process Mechanical Engineering*, vol. 0, no. 0, p. 09544089211073296, doi: 10.1177/09544089211073296.

[6] L. Xiang *et al.*, "Mechanical Properties and Microstructure of Thin-Walled Al-Cu Alloy Casting," *Strength of Materials*, Jun. 2022, doi: 10.1007/s11223-022-00406-2.

[7] J. Sertucha and J. Lacaze, “Casting Defects in Sand-Mold Cast Irons—An Illustrated Review with Emphasis on Spheroidal Graphite Cast Irons,” *Metals*, vol. 12, no. 3. MDPI, Mar. 01, 2022. doi: 10.3390/met12030504.

[8] R. D. Gupta, N. Sharma, R. Khanna, S. Sharma, and S. Dayal, “A study of process optimization of carburetor die casting process,” *Mater Today Proc*, Jan. 2022, doi: 10.1016/j.matpr.2022.01.153.

[9] Y. Li, J. Liu, W. Huang, and S. Zhang, “Microstructure related analysis of tensile and fatigue properties for sand casting aluminum alloy cylinder head,” *Eng Fail Anal*, vol. 136, Jun. 2022, doi: 10.1016/j.engfailanal.2022.106210.

[10] R. Ali *et al.*, “Elimination of solidification shrinkage defects in the casting of aluminum alloy,” *Journal of Mechanical Science and Technology*, vol. 36, no. 5, pp. 2345–2353, May 2022, doi: 10.1007/s12206-022-0416-z.

[11] M. Jolly and L. Katgerman, “Modelling of defects in aluminium cast products,” *Prog Mater Sci*, vol. 123, Jan. 2022, doi: 10.1016/j.pmatsci.2021.100824.

[12] T. S. Srivatsan, T. S. Sudarshan, and K. Manigandan, *Manufacturing Techniques for Materials : Engineering and Engineered*, CRC Press, 2018

[13] E. Gabriel *et al.*, “Open MPI: Goals, Concept, and Design of a Next Generation MPI Implementation,” Springer, Berlin, Heidelberg, 2004, doi: 10.1007/978-3-540-30218-6_19

[14] W. Gropp, W. D. Gropp, E. Lusk, A. Skjellum, and A. D. F. E. E. Lusk, *Using MPI: portable parallel programming with the message-passing interface*, vol. 1. MIT press, 1999.

[15] "MPICH," https://www.mpich.org/ (accessed Feb. 18, 2023).

[16] "Intel MPI Library," https://www.intel.com/content/www/us/en/developer/tools/oneapi/mpi-library.html#gs.46ggfe (accessed Feb. 18, 2023).

[17] "MVAPICH," https://mvapich.cse.ohio-state.edu/faq/ (accessed Feb. 18, 2023).

[18] "Open MPI," https://www.open-mpi.org/ (accessed Feb. 18, 2023).

[19] "NVIDIA Developer," https://developer.nvidia.com/cuda-zone (accessed Feb. 18, 2023).

[20] J. Cheng, M. Grossman, and T. McKercher, *Professional CUDA c programming*. John Wiley & Sons, 2014.

[21] S. Cook, *CUDA programming: a developer's guide to parallel computing with GPUs*. Newnes, 2012.

[22] J. Sanders and E. Kandrot, *CUDA by example: an introduction to general-purpose GPU programming*. Addison-Wesley Professional, 2010.

[23] T. Plewa, T. Linde, and V. G. Weirs, "Adaptive mesh refinement-theory and applications," Springer Science & Business Media, 2004.

[24] E. Dorari, M. Eshraghi, and S. D. Felicelli, "A multiple-grid-time-step lattice Boltzmann method for transport phenomena with dissimilar time

scales: Application in dendritic solidification," *Appl Math Model*, vol. 62, pp. 580–594, Oct. 2018, doi: 10.1016/j.apm.2018.06.023.

[25] S. Sakane, T. Takaki, M. Ohno, Y. Shibuta, and T. Aoki, "Acceleration of phase-field lattice Boltzmann simulation of dendrite growth with thermosolutal convection by the multi-GPUs parallel computation with multiple mesh and time step method," *Model Simul Mat Sci Eng*, vol. 27, no. 5, May 2019, doi: 10.1088/1361-651X/ab20b9.

[26] D. Schneider, "The Exascale Era is Upon Us: The Frontier supercomputer may be the first to reach 1,000,000,000,000,000,000 operations per second," *IEEE Spectrum, Volume 59, Issue 1*, pp 34–35, 2022, doi: 10.1109/MSPEC.2022.9676353

[27] H. Ji, W. Daughton, J. Jara-Almonte, A. Le, A. Stanier, and J. Yoo, "Magnetic reconnection in the era of exascale computing and multiscale experiments," *Nature Reviews Physics*, vol. 4, no. 4, pp. 263–282, 2022, doi: 10.1038/s42254-021-00419-x.

[28] N. Zhang, "Moore's Law is dead, long live Moore's Law. " May 2022.

[29] J. Yanagimoto, D. Banabic, M. Banu, and L. Madej, "Simulation of metal forming – Visualization of invisible phenomena in the digital era," *CIRP Annals*, Jun. 2022, doi: 10.1016/j.cirp.2022.05.007.

[30] A. Brasoveanu, M. Moodie, and R. Agrawal, "Textual evidence for the perfunctoriness of independent medical reviews," in *CEUR Workshop Proceedings*, CEUR-WS, 2020, pp. 1–9. doi: 10.1145/nnnnnnn.nnnnnnn.

[31] S. Watanabe and T. Aoki, “Large-scale flow simulations using lattice Boltzmann method with AMR following free-surface on multiple GPUs,” *Comput Phys Commun*, vol. 264, Jul. 2021, doi: 10.1016/j.cpc.2021.107871.

[32] S. Sakane, T. Aoki, and T. Takaki, “Parallel GPU-accelerated adaptive mesh refinement on two-dimensional phase-field lattice Boltzmann simulation of dendrite growth,” *Comput Mater Sci*, vol. 211, p. 111507, Aug. 2022, doi: 10.1016/j.commatsci.2022.111507.

[33] H. Y. Schive, J. A. ZuHone, N. J. Goldbaum, M. J. Turk, M. Gaspari, and C. Y. Cheng, “GAMER-2: A GPU-accelerated adaptive mesh refinement code - Accuracy, performance, and scalability,” *Mon Not R Astron Soc*, vol. 481, no. 4, pp. 4815–4840, Dec. 2018, doi: 10.1093/MNRAS/STY2586.

[34] E. Miyoshi *et al.*, “Ultra-large-scale phase-field simulation study of ideal grain growth,” *NPJ Comput Mater*, vol. 3, no. 1, Dec. 2017, doi: 10.1038/s41524-017-0029-8.

[35] S. Sakane *et al.*, “Multi-GPUs parallel computation of dendrite growth in forced convection using the phase-field-lattice Boltzmann model,” *J Cryst Growth*, vol. 474, pp. 154–159, Sep. 2017, doi: 10.1016/j.jcrysgro.2016.11.103.

[36] W. Wang, P. D. Lee, and M. McLean, “A model of solidification microstructures in nickel-based superalloys: Predicting primary dendrite spacing selection,” *Acta Mater*, vol. 51, no. 10, pp. 2971–2987, Jun. 2003, doi: 10.1016/S1359-6454(03)00110-1.

[37] C. A. Gandin and M. Rappaz, “A coupled finite element-cellular automaton model for the prediction of dendritic grain structures in solidification processes,” *Acta Metallurgica Et Materialia*, vol. 42, no. 7, pp. 2233–2246, 1994, doi: 10.1016/0956-7151(94)90302-6.

[38] Ch.-A. Gandin, J.-L. Desbiolles, M. Rappaz, P. Thé, and T. Voz, “A Three-Dimensional Cellular Automaton-Finite Element Model for the Prediction of Solidification Grain Structures,” *Metallurgical and Materials Transactions A: Physical Metallurgy and Materials Science*, May 1999, doi: 10.1007/s11661-999-0226-2

[39] M. Eshraghi, M. Hashemi, B. Jelinek, and S. D. Felicelli, “Three-dimensional lattice Boltzmann modeling of dendritic solidification under forced and natural convection,” *Metals (Basel)*, vol. 7, no. 11, Nov. 2017, doi: 10.3390/met7110474.

[40] M. Jegatheesan and A. Bhattacharya, “A model for predicting the effects of buoyancy driven convection on solidification of binary alloy with nanoparticles,” *Int J Heat Mass Transf*, vol. 182, Jan. 2022, doi: 10.1016/j.ijheatmasstransfer.2021.121916.

[41] S. Sakane, T. Takaki, M. Ohno, T. Shimokawabe, and T. Aoki, “GPU-accelerated 3D phase-field simulations of dendrite competitive growth during directional solidification of binary alloy,” in *IOP Conference Series: Materials Science and Engineering*, Institute of Physics Publishing, Jun. 2015. doi: 10.1088/1757-899X/84/1/012063.

[42] W. Sun, R. Yan, Y. Zhang, H. Dong, and T. Jing, “GPU-accelerated three-dimensional large-scale simulation of dendrite growth for Ti6Al4V alloy based on multi-component phase-field model,” *Comput Mater Sci*, vol. 160, pp. 149–158, Apr. 2019, doi: 10.1016/j.commatsci.2018.12.051.

[43] T. Shimokawabe, T. Takaki, T. Endo, N. Maruyama, T. Aoki, and S. Matsuoka, “Peta-scale Phase-Field Simulation for Dendritic Solidification on the TSUBAME 2 . 0 Supercomputer Categories and Subject Descriptors,” *Science And Technology*, 2010, doi: 10.1145/2063384.2063388.

[44] B. Jelinek, M. Eshraghi, S. Felicelli, and J. F. Peters, “Large-scale parallel lattice Boltzmann-cellular automaton model of two-dimensional dendritic growth,” *Comput Phys Commun*, vol. 185, no. 3, pp. 939–947, 2014, doi: 10.1016/j.cpc.2013.09.013.

[45] M. Eshraghi, B. Jelinek, and S. D. Felicelli, “Large-Scale Three-Dimensional Simulation of Dendritic Solidification Using Lattice Boltzmann Method,” *Jom*, vol. 67, no. 8, pp. 1786–1792, 2015, doi: 10.1007/s11837-015-1446-0.

[46] S. Sakane, T. Takaki, M. Ohno, T. Shimokawabe, and T. Aoki, “GPU-accelerated 3D phase-field simulations of dendrite competitive growth during directional solidification of binary alloy,” in *IOP Conference Series: Materials Science and Engineering*, 2015. doi: 10.1088/1757-899X/84/1/012063.

[47] A. Kao, I. Krastins, M. Alexandrakis, N. Shevchenko, S. Eckert, and K. Pericleous, “A Parallel Cellular Automata Lattice Boltzmann Method for Convection-Driven Solidification,” *Jom*, vol. 71, no. 1, pp. 48–58, 2019, doi: 10.1007/s11837-018-3195-3.

[48] T. Takaki *et al.*, “Phase-field-lattice Boltzmann studies for dendritic growth with natural convection,” *J Cryst Growth*, vol. 474, pp. 146–153, Sep. 2017, doi: 10.1016/j.jcrysgro.2016.11.099.

[49] S. A. Nabavizadeh, M. Eshraghi, S. D. Felicelli, S. N. Tewari, and R. N. Grugel, “The Marangoni convection effects on directional dendritic solidification,” *Heat and Mass Transfer/Waerme- und Stoffuebertragung*, vol. 56, no. 4, pp. 1329–1341, Dec. 2020, doi: 10.1007/s00231-019-02799-4.

[50] A. Kao, N. Shevchenko, M. Alexandrakis, I. Krastins, S. Eckert, and K. Pericleous, “Thermal dependence of large-scale freckle defect formation,” *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 377, no. 2143, 2019, doi: 10.1098/rsta.2018.0206.

[51] A. Yamanaka, T. Aoki, S. Ogawa, and T. Takaki, “GPU-accelerated phase-field simulation of dendritic solidification in a binary alloy,” *J Cryst Growth*, vol. 318, no. 1, pp. 40–45, Mar. 2011, doi: 10.1016/J.JCRYSGRO.2010.10.096.

[52] A. Yamanaka, T. Aoki, S. Ogawa, and T. Takaki, “GPU-accelerated phase-field simulation of dendritic solidification in a binary alloy,” in *Journal of Crystal Growth*, Mar. 2011, pp. 40–45. doi: 10.1016/j.jcrysgro.2010.10.096.

[53] T. Aoki, T. Watanabe, Y. Itakura, “Overview of TSUBAME3.0, Green Cloud Supercomputer for Convergence of HPC, AI and Big-Data ChainerMN: Scalable Distributed Deep Learning Framework Creation of Chemical Structures Aimed for Drug Design, Facilitated by Efficient GPU Computing,” *TSUBAME e-Science Journal*, vol. 16, 2017.

[54] B. Jelinek, M. Eshraghi, S. Felicelli, and J. F. Peters, “Large-scale parallel lattice Boltzmann-cellular automaton model of two-dimensional dendritic growth,” *Comput Phys Commun*, vol. 185, no. 3, pp. 939–947, Mar. 2014, doi: 10.1016/j.cpc.2013.09.013.

[55] M. M. Strout, B. Kreaseck, and P. D. Hovland, “Data-flow analysis for MPI programs,” in *Proceedings of the International Conference on Parallel Processing*, 2006, pp. 175–184. doi: 10.1109/ICPP.2006.32.

[56] “Supercomputing at ORNL,” *https://www.ornl.gov/directorate/ccsd/supercomputing-ornl*, (accessed Feb. 18, 2023).

[57] T. Baer, “Comparison of Scheduling Policies and Workloads on the NCCS and NICS XT4 Systems at Oak Ridge National Laboratory,” CUG 2009 Proceedings, 2009.

[58] M. Eshraghi, B. Jelinek, and S. D. Felicelli, “Large-Scale Three-Dimensional Simulation of Dendritic Solidification Using Lattice

Boltzmann Method," *JOM*, vol. 67, no. 8, pp. 1786–1792, Aug. 2015, doi: 10.1007/s11837-015-1446-0.

[59] T. Shimokawabe *et al.*, "Peta-scale Phase-Field Simulation for DendriticSolidification on the TSUBAME 2.0 Supercomputer," 2011.

[60] F. D. Witherden, B. C. Vermeire, and P. E. Vincent, "Heterogeneous computing on mixed unstructured grids with PyFR," *Comput Fluids*, vol. 120, pp. 173–186, Oct. 2015, doi: 10.1016/j.compfluid.2015.07.016.

[61] J. Hötzer, A. Reiter, H. Hierl, P. Steinmetz, M. Selzer, and B. Nestler, "The parallel multi-physics phase-field framework PACE3D," *J Comput Sci*, vol. 26, pp. 1–12, May 2018, doi: 10.1016/j.jocs.2018.02.011.

[62] C. Icenhour *et al.*, "Multi-physics object oriented simulation environment (moose)," Idaho National Lab.(INL), Idaho Falls, ID (United States), 2018.

[63] M. S. Alnaes *et al.*, "The FEniCS Project Version 1.5," http://fenicsproject.org, (accessed Feb. 18, 2023).

[64] S. Balay, W. D. Gropp, L. Curfman, M. Barry, and F. Smith, "Efficient Management of Parallelism in Object-Oriented Numerical Software Libraries," Modern Software Tools for Scientific Computing, pp 163–202, 1997, doi: 10.1007/978-1-4612-1986-6_8

[65] A. Rollett and J. Gruber, "Mesoscale Microstructure Simulation Project (MMSP)." Carnegie Mellon Computational Materials Science, 2007.

[66] M. Parashar and J. C. Brownet, "Systems Engineering For High Performance Computing Software: The HDDA/DAGH Infrastructure For Implementation Of Parallel Structured Adaptive Mesh," Structured

Adaptive Mesh Refinement (SAMR) Grid Methods, pp 1–18, 2000, doi: 10.1007/978-1-4612-1252-2_1

[67] P. J. Lu, M. C. Lai, and J. S. Chang, “A Survey of High-Performance Interconnection Networks in High-Performance Computer Systems,” *Electronics (Switzerland)*, vol. 11, no. 9, May 2022, doi: 10.3390/electronics11091369.

[68] M. Apel *et al.*, “Microstructure Modeling,” Handbook of Software Solutions for ICME, First Edition, 2017, doi: 10.1002/9783527693566.ch3.

[69] D. Cagigas-Muñiz, F. Diaz-del-Rio, J. L. Sevillano-Ramos, and J. L. Guisado-Lizar, “Efficient simulation execution of cellular automata on GPU,” *Simul Model Pract Theory*, vol. 118, Jul. 2022, doi: 10.1016/j.simpat.2022.102519.

[70] E. Dorari, “Modeling Dendritic Solidification Under Melt Convection Using Lattice Boltzmann and Cellular Automaton Methods,” Ph.D. dissertation, The University of Akron, 2017.

[71] E. Wesner, A. Choudhury, A. August, M. Berghoff, and B. Nestler, “A phase-field study of large-scale dendrite fragmentation in Al-Cu,” *J Cryst Growth*, vol. 359, no. 1, pp. 107–121, Nov. 2012, doi: 10.1016/j.jcrysgro.2012.08.036.

[72] Y. Zhang, J. Zhou, Y. Yin, X. Shen, and X. Ji, “Multi-GPU implementation of a cellular automaton model for dendritic growth of binary alloy,” *Journal of Materials Research and Technology*, vol. 14, pp. 1862–1872, Sep. 2021, doi: 10.1016/j.jmrt.2021.07.095.

[73] A. Saad, C. A. Gandin, M. Bellet, N. Shevchenko, and S. Eckert, "Simulation of Channel Segregation During Directional Solidification of In—75 wt pct Ga. Qualitative Comparison with In Situ Observations," *Metall Mater Trans A Phys Metall Mater Sci*, vol. 46, no. 11, pp. 4886–4897, Nov. 2015, doi: 10.1007/s11661-015-2963-8.

[74] N. Moelans, B. Blanpain, and P. Wollants, "An introduction to phase-field modeling of microstructure evolution," *CALPHAD*, vol. 32, no. 2, pp. 268–294, Jun. 2008, doi: 10.1016/j.calphad.2007.11.003.

[75] D. Tourret, H. Liu, and J. LLorca, "Phase-field modeling of microstructure evolution: Recent applications, perspectives and challenges," *Prog Mater Sci*, vol. 123, Jan. 2022, doi: 10.1016/j.pmatsci.2021.100810.

[76] J. Segurado, R. A. Lebensohn, and J. Llorca, "Computational Homogenization of Polycrystals," Advances in Applied Mechanics, Elsevier, 2018, doi: 10.1016/bs.aams.2018.07.001

[77] S. Chen, B. Merriman, S. Osher, and P. Smereka, "A Simple Level Set Method for Solving Stefan Problems," Journal of Computational Physics, vol. 135, 1997, doi: 10.1006/jcph.1997.5721

[78] R. Kobayashi, "Modeling and numerical simulations of dendritic crystal growth," *Physica D*, vol. 63, no. 3–4, pp. 410–423, Mar. 1993, doi: 10.1016/0167-2789(93)90120-P.

[79] T. Takaki, "Phase-field modeling and simulations of dendrite growth," in *ISIJ International*, Iron and Steel Institute of Japan, 2014, pp. 437–444. doi: 10.2355/isijinternational.54.437.

[80] R. Kobayashi, “A numerical approach to three-dimensional dendritic solidification,” *Exp Math*, vol. 3, no. 1, pp. 59–81, 1994, doi: 10.1080/10586458.1994.10504577.

[81] N. Provatas, N. Goldenfeld, and J. Dantzig, “Adaptive Mesh Refinement Computation of Solidification Microstructures Using Dynamic Data Structures,” Journal of Computational Physics, vol. 148, 1999, doi: 10.1006/jcph.1998.6122

[82] C. C. Chen, Y. L. Tsai, and C. W. Lan, “Adaptive phase field simulation of dendritic crystal growth in a forced flow: 2D vs 3D morphologies,” *Int J Heat Mass Transf*, vol. 52, no. 5–6, pp. 1158–1166, Feb. 2009, doi: 10.1016/j.ijheatmasstransfer.2008.09.014.

[83] Y. Lu, C. Beckermann, and J. C. Ramirez, “Three-dimensional phase-field simulations of the effect of convection on free dendritic growth,” *J Cryst Growth*, vol. 280, no. 1–2, pp. 320–334, Jun. 2005, doi: 10.1016/j.jcrysgro.2005.03.063.

[84] J. H. Jeong, N. Goldenfeld, and J. A. Dantzig, “Phase field model for three-dimensional dendritic growth with fluid flow,” *Phys Rev E Stat Phys Plasmas Fluids Relat Interdiscip Topics*, vol. 64, no. 4, p. 14, 2001, doi: 10.1103/PhysRevE.64.041602.

[85] T. Takaki, A. Yamanaka, Y. Higa, and Y. Tomita, “Phase-field model during static recrystallization based on crystal-plasticity theory,” in *Journal of Computer-Aided Materials Design*, Dec. 2007, pp. 75–84. doi: 10.1007/s10820-007-9083-8.

[86] Y. Shibuta, M. Ohno, and T. Takaki, “Solidification in a Supercomputer: From Crystal Nuclei to Dendrite Assemblages,” *JOM*, vol. 67, no. 8, pp. 1793–1804, Aug. 2015, doi: 10.1007/s11837-015-1452-2.

[87] J. A. Warren and W. J. Boettinger, “Prediction Of Dendritic Growth And Microsegregation Patterns In A Binary Alloy Using The Phase-Field Method,” Acta Metallurgica et Materialia, vol. 43, PP 689-703, 1995, doi: 10.1016/0956-7151(94)00285-P.

[88] T. Suzuki, M. Ode, G. Kim, and W. T. Kim, “Phase-field model of dendritic growth,” Journal of Crystal Growth, vol. 237-239, pp. 125-131, 2002, doi: 10.1016/S0022-0248(01)01891-7.

[89] T. Takaki, T. Shimokawabe, M. Ohno, A. Yamanaka, and T. Aoki, “Unexpected selection of growing dendrites by very-large-scale phase-field simulation,” *J Cryst Growth*, vol. 382, pp. 21–25, 2013, doi: 10.1016/j.jcrysgro.2013.07.028.

[90] A. Choudhury, K. Reuther, E. Wesner, A. August, B. Nestler, and M. Rettenmayr, “Comparison of phase-field and cellular automaton models for dendritic solidification in Al-Cu alloy,” *Comput Mater Sci*, vol. 55, pp. 263–268, Apr. 2012, doi: 10.1016/j.commatsci.2011.12.019.

[91] B. Echebarria, R. Folch, A. Karma, and M. Plapp, “Quantitative phase-field model of alloy solidification,” *Phys Rev E Stat Phys Plasmas Fluids Relat Interdiscip Topics*, vol. 70, no. 6, p. 22, 2004, doi: 10.1103/PhysRevE.70.061604.

[92] K. _ S. T. S.- Manigandan, *Manufacturing Techniques for Materials Engineering and Engineered*, vol. ch 15.

[93] L. Beltran-Sanchez and D. M. Stefanescu, "A Quantitative Dendrite Growth Model and Analysis of Stability Concepts," *Metallurgical and Materials Transactions A*, vol. 35, pp. 2471–2485, 2004.

[94] M. F. Zhu and D. M. Stefanescu, "Virtual front tracking model for the quantitative modeling of dendritic growth in solidification of alloys," *Acta Mater*, vol. 55, no. 5, pp. 1741–1755, Mar. 2007, doi: 10.1016/j.actamat.2006.10.037.

[95] G. Tryggvason, A. Esmaeeli, and N. Al-Rawahi, "Direct numerical simulations of flows with phase change," *Comput Struct*, vol. 83, no. 6–7, pp. 445–453, Feb. 2005, doi: 10.1016/j.compstruc.2004.05.021.

[96] D. Juric and G. T. Tryggvason, "A Front -Tracking Method for Dendritic Solidification," *Journal of Computational Physics*, vol. 123, pp. 127-148, 1996, doi: 10.1006/jcph.1996.0011.

[97] P. Zhao and J. C. Heinrich, "Front-tracking finite element method for dendritic solidification," *J Comput Phys*, vol. 173, no. 2, pp. 765–796, Nov. 2001, doi: 10.1006/jcph.2001.6911.

[98] Y. T. Kim, N. Goldenfeld, and J. Dantzig, "Computation of dendritic microstructures using a level set method," *Phys Rev E Stat Phys Plasmas Fluids Relat Interdiscip Topics*, vol. 62, no. 2 B, pp. 2471–2474, 2000, doi: 10.1103/physreve.62.2471.

[99] F. Gibou, R. Fedkiw, R. Caflisch, and S. Osher, “A Level Set Approach for the Numerical Simulation of Dendritic Growth,” *Journal of Scientific Computing*. Vol. 19, pp. 183-199, 2002, doi: 10.1023/A:1025399807998.

[100] L. Tan and N. Zabaras, “A level set simulation of dendritic solidification of multi-component alloys,” *J Comput Phys*, vol. 221, no. 1, pp. 9–40, Jan. 2007, doi: 10.1016/j.jcp.2006.06.003.

[101] M. Corbit, “The Cornell Theory Center,” *IEEE Computational Science and Engineering*, vol. 1, no. 4, pp. 10–13, 1994, doi: 10.1109/99.338766.

[102] L. Tan and N. Zabaras, “Modeling the growth and interaction of multiple dendrites in solidification using a level set method,” *J Comput Phys*, vol. 226, no. 1, pp. 131–155, Sep. 2007, doi: 10.1016/j.jcp.2007.03.023.

[103] S. Chen, B. Merriman, S. Osher, and P. Smereka, “A simple level set method for solving Stefan problems,” *J Comput Phys*, vol. 135, no. 1, pp. 8–29, 1997.

[104] S. Osher, R. Fedkiw, and K. Piechor, “Level set methods and dynamic implicit surfaces,” *Appl. Mech. Rev.*, vol. 57, no. 3, pp. B15–B15, 2004.

[105] L. Tan and N. Zabaras, “A level set simulation of dendritic solidification with combined features of front-tracking and fixed-domain methods,” *J Comput Phys*, vol. 211, no. 1, pp. 36–63, Jan. 2006, doi: 10.1016/j.jcp.2005.05.013.

[106] D. Tourret and A. Karma, “Multiscale dendritic needle network model of alloy solidification,” *Acta Mater*, vol. 61, no. 17, pp. 6474–6491, Oct. 2013, doi: 10.1016/j.actamat.2013.07.026.

[107] D. Tourret and A. Karma, “Three-dimensional dendritic needle network model for alloy solidification,” *Acta Mater*, vol. 120, pp. 240–254, Nov. 2016, doi: 10.1016/j.actamat.2016.08.041.

[108] L. Sturz *et al.*, “ISS-Experiments Of Columnar-To-Equiaxed Transition In Solidification Processing,” *141st TMS 2012 Annual Meeting and Exhibition*, 2012.

[109] B. Bellón *et al.*, “Multiscale prediction of microstructure length scales in metallic alloy casting,” *Acta Mater*, vol. 207, Apr. 2021, doi: 10.1016/j.actamat.2021.116686.

[110] D. Tourret, L. Sturz, A. Viardin, and M. Založnik, “Comparing mesoscopic models for dendritic growth,” in *IOP Conference Series: Materials Science and Engineering*, Institute of Physics Publishing, Jun. 2020. doi: 10.1088/1757-899X/861/1/012002.

[111] T. Isensee and D. Tourret, “Convective effects on columnar dendritic solidification – A multiscale dendritic needle network study,” *Acta Mater*, vol. 234, p. 118035, Aug. 2022, doi: 10.1016/j.actamat.2022.118035.

[112] H. KYOGOKU and T.-T. IKESHOJI, “A review of metal additive manufacturing technologies: Mechanism of defects formation and simulation of melting and solidification phenomena in laser powder bed fusion process,” *Mechanical Engineering Reviews*, vol. 7, no. 1, pp. 19-00182-19–00182, 2020, doi: 10.1299/mer.19-00182.

[113] J. H. K. Tan, S. L. Sing, and W. Y. Yeong, “Microstructure modelling for metallic additive manufacturing: a review,” *Virtual Phys Prototyp*, vol. 15, no. 1, pp. 87–105, Jan. 2020, doi: 10.1080/17452759.2019.1677345.

[114] H. Yin and S. D. Felicelli, “Dendrite growth simulation during solidification in the LENS process,” *Acta Mater*, vol. 58, no. 4, pp. 1455–1465, Feb. 2010, doi: 10.1016/j.actamat.2009.10.053.

[115] Y. Yu, Y. Li, F. Lin, and W. Yan, “A multi-grid Cellular Automaton model for simulating dendrite growth and its application in additive manufacturing,” *Addit Manuf*, vol. 47, Nov. 2021, doi: 10.1016/j.addma.2021.102284.

[116] R. Shi, S. A. Khairallah, T. T. Roehling, T. W. Heo, J. T. McKeown, and M. J. Matthews, “Microstructural control in metal laser powder bed fusion additive manufacturing using laser beam shaping strategy,” *Acta Mater*, vol. 184, pp. 284–305, 2020, doi: 10.1016/j.actamat.2019.11.053.

[117] C. Noble *et al.*, *ALE3D: An Arbitrary Lagrangian-Eulerian Multi-Physics Code*. 2017. doi: 10.13140/RG.2.2.28136.88325.

[118] M. A. Groeber and M. A. Jackson, “DREAM.3D: A Digital Representation Environment for the Analysis of Microstructure in 3D” *Integrating Materials and Manufacturing Innovation*, vol. 3, pp. 56-72, 2014.

[119] R. Shi, S. A. Khairallah, T. T. Roehling, T. W. Heo, J. T. McKeown, and M. J. Matthews, “Microstructural control in metal laser powder bed fusion additive manufacturing using laser beam shaping strategy,” *Acta Mater*, vol. 184, pp. 284–305, Feb. 2020, doi: 10.1016/j.actamat.2019.11.053.

[120] Y. Lian, Z. Gan, C. Yu, D. Kats, W. K. Liu, and G. J. Wagner, “A cellular automaton finite volume method for microstructure evolution during additive manufacturing,” *Mater Des*, vol. 169, May 2019, doi: 10.1016/j.matdes.2019.107672.

[121] S. M. Elahi, R. Tavakoli, A. K. Boukellal, T. Isensee, I. Romero, and D. Tourret, “Multiscale simulation of powder-bed fusion processing of metallic alloys,” Mar. 2022, doi: 10.1016/j.commatsci.2022.111383.

[122] N. Raghavan *et al.*, “Localized melt-scan strategy for site specific control of grain size and primary dendrite arm spacing in electron beam additive manufacturing,” *Acta Mater*, vol. 140, pp. 375–387, Nov. 2017, doi: 10.1016/j.actamat.2017.08.038.

[123] D. Walton and B. Chalmers, “The origin of the preferred orientation in the columnar zone of ingots,” *Transactions of the American Institute of Mining and Metallurgical Engineers*, vol. 215, no. 3, pp. 447–457, 1959.

[124] Y. Z. Zhou, A. Volek, and N. R. Green, “Mechanism of competitive grain growth in directional solidification of a nickel-base superalloy,” *Acta Mater*, vol. 56, no. 11, pp. 2631–2637, Jun. 2008, doi: 10.1016/j.actamat.2008.02.022.

[125] N. D’Souza, M. Ardakani, A. Wagner, B. A. Shollock, and M. Mclean, “Morphological aspects of competitive grain growth during directional solidification of a nickel-base superalloy, CMSX4,” *J Mater Sci*, vol. 37, pp. 481–487, 2002.

[126] A. Wagner, B. A. Shollock, and M. McLean, “Grain structure development in directional solidification of nickel-base superalloys,” *Materials Science and Engineering A*, vol. 374, no. 1–2, pp. 270–279, Jun. 2004, doi: 10.1016/j.msea.2004.03.017.

[127] C. Yang, Q. Xu, and B. Liu, “GPU-accelerated three-dimensional phase-field simulation of dendrite growth in a nickel-based superalloy,” *Comput Mater Sci*, vol. 136, pp. 133–143, Aug. 2017, doi: 10.1016/j.commatsci.2017.04.031.

[128] S. Raghavan, G. Singh, S. Sondhi, and S. Srikanth, “Construction of a pseudo-binary phase diagram for multi-component Ni-base superalloys,” *CALPHAD*, vol. 38, pp. 85–91, Sep. 2012, doi: 10.1016/j.calphad.2012.04.003.

[129] C. Yang, Q. Xu, and B. Liu, “GPU-accelerated three-dimensional phase-field simulation of dendrite growth in a nickel-based superalloy,” *Comput Mater Sci*, vol. 136, pp. 133–143, Aug. 2017, doi: 10.1016/j.commatsci.2017.04.031.

[130] T. Takaki, S. Sakane, M. Ohno, Y. Shibuta, T. Shimokawabe, and T. Aoki, “Large-scale phase-field studies of three-dimensional dendrite competitive growth at the converging grain boundary during directional solidification of a bicrystal binary alloy,” *ISIJ International*, vol. 56, no. 8, pp. 1427–1435, 2016, doi: 10.2355/isijinternational.ISIJINT-2016-156.

[131] T. Takaki, R. Rojas, M. Ohno, T. Shimokawabe, and T. Aoki, “GPU phase-field lattice Boltzmann simulations of growth and motion of a binary alloy

dendrite," in *IOP Conference Series: Materials Science and Engineering*, Institute of Physics Publishing, Jun. 2015. doi: 10.1088/1757-899X/84/1/012066.

[132] Y. Feng, M. Založnik, B. G. Thomas, and A. B. Phillion, "Meso-scale simulation of liquid feeding in an equiaxed dendritic mushy zone," *Materialia (Oxf)*, vol. 9, Mar. 2020, doi: 10.1016/j.mtla.2020.100612.

[133] C. Dussert, G. Rasigni, M. Rasigni, J. Palmari, and A. Llebaria, "Minimal spanning tree: A new approach for studying order and disorder," *Phys. Rev. B*, vol. 34, 1986, doi: 10.1103/PhysRevB.34.3528.

[134] D. Tourret *et al.*, "From Solidification Processing to Microstructure to Mechanical Properties: A Multi-scale X-ray Study of an Al-Cu Alloy Sample," *Metall Mater Trans A Phys Metall Mater Sci*, vol. 48, no. 11, pp. 5529–5546, Nov. 2017, doi: 10.1007/s11661-017-4302-8.

[135] T. Takaki, S. Sakane, M. Ohno, Y. Shibuta, T. Shimokawabe, and T. Aoki, "Primary arm array during directional solidification of a single-crystal binary alloy: Large-scale phase-field study," *Acta Mater*, vol. 118, pp. 230–243, Oct. 2016, doi: 10.1016/j.actamat.2016.07.049.

[136] J. Hui, R. Tiwari, X. Wu, S. N. Tewari, and R. Trivedi, "Primary Dendrite Distribution and Disorder during Directional Solidification of Pb-Sb Alloys," *Metallurgical and Materials Transactions A: Physical Metallurgy and Materials Science*, vol. 33, 2002, doi: 10.1007/s11661-002-0337-5.

[137] S. N. Tewari, Y.-H. Weng, G. L. Ding, and R. Trivedi, "Cellular Array Morphology during Directional Solidification," *Metallurgical and Materials*

Transactions A: Physical Metallurgy and Materials Science, vol. 33, 2002, doi: 10.1007/s11661-002-0224-0.

[138] J. Strickland *et al.*, "On the nature of hexagonality within the solidification structure of single crystal alloys: Mechanisms and applications," *Acta Mater*, vol. 200, pp. 417–431, Nov. 2020, doi: 10.1016/j.actamat.2020.09.019.

[139] K. Pandit, S. R. Upadhyay, and S. N. Tewari, "Effect of cross-section-change induced advective flow on the primary dendrite array morphology of hypoeutectic Pb-Sb alloys during directional solidification," *J Cryst Growth*, vol. 502, pp. 19–29, Nov. 2018, doi: 10.1016/j.jcrysgro.2018.09.005.

[140] T. Takaki, S. Sakane, M. Ohno, Y. Shibuta, and T. Aoki, "Phase-field study on an array of tilted columnar dendrites during the directional solidification of a binary alloy," *Comput Mater Sci*, vol. 203, Feb. 2022, doi: 10.1016/j.commatsci.2021.111143.

[141] J. A. Dantzig and M. Rappaz, *Solidification Engineering Sciences Materials 2nd Edition Revised & Expanded*, Materials, 2016.

[142] H. B. Dong and P. D. Lee, "Simulation of the columnar-to-equiaxed transition in directionally solidified Al-Cu alloys," *Acta Mater*, vol. 53, no. 3, pp. 659–668, Feb. 2005, doi: 10.1016/j.actamat.2004.10.019.

[143] S. A. Nabavizadeh, M. Eshraghi, and S. D. Felicelli, "Three-dimensional phase field modeling of columnar to equiaxed transition in directional

solidification of Inconel 718 alloy," *J Cryst Growth*, vol. 549, Nov. 2020, doi: 10.1016/j.jcrysgro.2020.125879.

[144] J. D. Hunt, "Steady state columnar and equiaxed growth of dendrites and eutectic," *Materials Science and Engineering*, vol. 65, no. 1, pp. 75–83, 1984, doi: 10.1016/0025-5416(84)90201-5.

[145] M. Gkmann, R. Trivedi, and T. v Kurz, "A Nucleation ahead of the advancing interface in directional solidification," *Materials Science and Engineering: A*, vol. 226-228, pp. 763-769, 1997, doi: 10.1016/S0921-5093(97)80081-0

[146] L. Nastac, "An efficient 3D stochastic model for predicting the columnar-to-equiaxed transition in alloy 718," in *IOP Conference Series: Materials Science and Engineering*, Institute of Physics Publishing, Jun. 2015. doi: 10.1088/1757-899X/84/1/012089.

[147] H. B. Dong, X. L. Yang, P. D. Lee, and W. Wang, "Simulation of equiaxed growth ahead of an advancing columnar front in directionally solidified Ni-based superalloys," *Proceedings of the 2003 International Symposium On Liquid Metals,* 2003.

[148] X. Xu, W. Zhang, and P. D. Lee, "Tree-Ring Formation during Vacuum Arc Remelting of INCONEL 718: Part II. Mathematical Modeling," *Metallurgical and Materials Transactions A*, vol. 33, pp. 1805–1815, 2002, doi: 10.1007/s11661-002-0189-z.

[149] P. A. Geslin, C. H. Chen, A. M. Tabrizi, and A. Karma, "Dendritic needle network modeling of the Columnar-to-Equiaxed transition. Part I: two

dimensional formulation and comparison with theory," *Acta Mater*, vol. 202, pp. 42–54, Jan. 2021, doi: 10.1016/j.actamat.2020.10.009.

[150] B. S. P Neuman, "Theoretical Derivation of Darcy's Law," Acta Mechanica, vol. 25, pp. 153–170, 1977, doi: 10.1007/BF01376989.

[151] T. Campanella, C. Charbon, and M. Rappaz, "Influence of permeability on the grain refinement induced by forced convection in copper-base alloys," *Scr Mater*, vol. 49, no. 10 SPEC., pp. 1029–1034, 2003, doi: 10.1016/S1359-6462(03)00446-9.

[152] D. Bernard, Ø. Nielsen, L. Salvo, and P. Cloetens, "Permeability assessment by 3D interdendritic flow simulations on microtomography mappings of Al-Cu alloys," *Materials Science and Engineering A*, vol. 392, no. 1–2, pp. 112–120, Feb. 2005, doi: 10.1016/j.msea.2004.09.004.

[153] B. Böttger, C. Haberstroh, and N. Giesselmann, "Cross-Permeability of the Semisolid Region in Directional Solidification: A Combined Phase-Field and Lattice-Boltzmann Simulation Approach," *JOM*, vol. 68, no. 1, pp. 27–36, Jan. 2016, doi: 10.1007/s11837-015-1690-3.

[154] M. S. Bhat, D. R. Poirier, and J. C. Heinrich, "Permeability for cross flow through columnar-dendritic alloys," *Metallurgical and Materials Transactions B*, vol. 26, no. 5, pp. 1049–1056, 1995.

[155] E. Khajeh and D. M. Maijer, "Permeability of dual structured hypoeutectic aluminum alloys," *Acta Mater*, vol. 59, no. 11, pp. 4511–4524, Jun. 2011, doi: 10.1016/j.actamat.2011.03.074.

[156] R. Berger *et al.*, “Permeability measurements of 3d microstructures generated by phase field simulation of the solidification of an al-si alloy during chill casting,” *Metals (Basel)*, vol. 11, no. 12, Dec. 2021, doi: 10.3390/met11121895.

[157] Y. Mitsuyama, T. Takaki, S. Sakane, Y. Shibuta, and M. Ohno, “Permeability tensor for columnar dendritic structures: Phase-field and lattice Boltzmann study,” *Acta Mater*, vol. 188, pp. 282–287, Apr. 2020, doi: 10.1016/j.actamat.2020.02.016.

[158] H. J. White, “Book Review : Flow of Gases Through Porous Media. P. C. Carman. New York, Academic Press, 1956. ix + 182 pages, illust. Price $6.00,” *Textile Research Journal*, vol. 29, no. 1, p. 98, Jan. 1959, doi: 10.1177/004051755902900112.

[159] J. C. Heinrich and D. R. Poirier, “Convection modeling in directional solidification,” in *Comptes Rendus - Mecanique*, May 2004, pp. 429–445. doi: 10.1016/j.crme.2004.02.001.

[160] Y. Natsume, D. Takahashi, K. Kawashima, E. Tanigawa, and K. Ohsasa, “Quantitative model to determine permeability for columnar dendritic structures,” *ISIJ International*, vol. 53, no. 5, pp. 838–847, 2013, doi: 10.2355/isijinternational.53.838.

[161] E. Khajeh and D. Maijer, “Physical and numerical characterization of the near-eutectic permeability of aluminum–copper alloys,” *Acta Materialia - ACTA MATER*, vol. 58, pp. 6334–6344, Nov. 2010, doi: 10.1016/j.actamat.2010.07.055.

[162] C. Puncreobutr, A. B. Phillion, J. L. Fife, and P. D. Lee, “Coupling in situ synchrotron X-ray tomographic microscopy and numerical simulation to quantify the influence of intermetallic formation on permeability in aluminium-silicon-copper alloys,” *Acta Mater*, vol. 64, pp. 316–325, Feb. 2014, doi: 10.1016/j.actamat.2013.10.044.

[163] T. Takaki, S. Sakane, M. Ohno, Y. Shibuta, and T. Aoki, “Permeability prediction for flow normal to columnar solidification structures by large–scale simulations of phase–field and lattice Boltzmann methods,” *Acta Mater*, vol. 164, pp. 237–249, Feb. 2019, doi: 10.1016/j.actamat.2018.10.039.

[164] A. Ludwig, A. Kharicha, C. Hölzl, J. Domitner, M. Wu, and T. Pusztai, “3D Lattice Boltzmann flow simulations through dendritic mushy zones,” *Eng Anal Bound Elem*, vol. 45, pp. 29–35, 2014, doi: 10.1016/j.enganabound.2014.01.015.

[165] B. Böttger, C. Haberstroh, and N. Giesselmann, “Cross-Permeability of the Semisolid Region in Directional Solidification: A Combined Phase-Field and Lattice-Boltzmann Simulation Approach,” *JOM*, vol. 68, no. 1, pp. 27–36, Jan. 2016, doi: 10.1007/s11837-015-1690-3.

[166] Y. Natsume, D. Takahashi, K. Kawashima, E. Tanigawa, and K. Ohsasa, “Quantitative model to determine permeability for columnar dendritic structures,” *ISIJ International*, vol. 53, no. 5, pp. 838–847, 2013, doi: 10.2355/isijinternational.53.838.

[167] H.-J. Diepers, C. Beckermann, and I. Steinbach, “Simulation of Convection and Ripening In A Binary Alloy Mush Using The Phase-Field Method,” *Acta Materialia*, vol. 47, pp. 3663-3678, 1999, doi: 10.1016/S1359-6454(99)00239-6

[168] S. G. R. Brown, J. A. Spittle, D. J. Jarvis, and R. Walden-Bevan, “Numerical determination of liquid flow permeabilities for equiaxed dendritic structures,” *Acta Materialia*, vol. 50, pp. 1559-1569, 2002, doi: 10.1016/S1359-6454(02)00014-9.

[169] X. ning Meng, L. Cui, Y. han Shi, and M. yong Zhu, “A CA-LBM model for simulating dendrite growth with forced convection,” *Journal of Iron and Steel Research International*, vol. 28, no. 8, pp. 997–1008, Aug. 2021, doi: 10.1007/s42243-021-00587-1.

[170] C. Vázquez-López *et al.*, “Influence of dendrite arm spacing on the thermal conductivity of an aluminum-silicon casting alloy,” *J Mater Res*, vol. 15, no. 1, pp. 85–91, 2000, doi: 10.1557/JMR.2000.0016.

[171] A. Zhang, J. Du, Z. Guo, Q. Wang, and S. Xiong, “A Phase-Field Lattice-Boltzmann Study on Dendritic Growth of Al-Cu Alloy Under Convection,” *Metallurgical and Materials Transactions B: Process Metallurgy and Materials Processing Science*, vol. 49, no. 6, pp. 3603–3615, Dec. 2018, doi: 10.1007/s11663-018-1418-1.

[172] T. Takaki, S. Sakane, M. Ohno, Y. Shibuta, and T. Aoki, “Large–scale phase–field lattice Boltzmann study on the effects of natural convection on dendrite morphology formed during directional solidification of a binary

alloy," *Comput Mater Sci*, vol. 171, Jan. 2020, doi: 10.1016/j.commatsci.2019.109209.

[173] E. Dorari, M. Eshraghi, and S. D. Felicelli, "Buoyancy induced flow and dendritic pattern formation during directional solidification," *Unpublished*.

[174] D. J. Hebditch and J. D. Hunt, "Observations Of Ingot Macrosegregation On Model Systems.," *Metall Trans*, vol. 5, no. 7, pp. 1557–1564, 1974, doi: 10.1007/BF02646326.

[175] H. Ge, F. Ren, J. Li, X. Han, M. Xia, and J. Li, "Four-Phase Dendritic Model for the Prediction of Macrosegregation, Shrinkage Cavity, and Porosity in a 55-Ton Ingot," *Metall Mater Trans A Phys Metall Mater Sci*, vol. 48, no. 3, pp. 1139–1150, Mar. 2017, doi: 10.1007/s11661-016-3910-z.

[176] D. Samanta and N. Zabaras, "Numerical study of macrosegregation in Aluminum alloys solidifying on uneven surfaces," *Int J Heat Mass Transf*, vol. 48, no. 21–22, pp. 4541–4556, Oct. 2005, doi: 10.1016/j.ijheatmasstransfer.2005.06.008.

[177] C. Zhang, D. Shahriari, A. Loucif, H. Melkonyan, and M. Jahazi, "Influence of thermomechanical shrinkage on macrosegregation during solidification of a large-sized high-strength steel ingot," *International Journal of Advanced Manufacturing Technology*, vol. 99, no. 9–12, pp. 3035–3048, Dec. 2018, doi: 10.1007/s00170-018-2695-1.

[178] F. Ren *et al.*, "Simulation of Macrosegregation and Shrinkage Cavity in an Al-4.5 Wt Pct Cu Ingot Using a Four-Phase Model," *Metall Mater Trans A*

Phys Metall Mater Sci, vol. 49, no. 12, pp. 6243–6254, Dec. 2018, doi: 10.1007/s11661-018-4892-9.

[179] Z. Q. Han, R. W. Lewis, and B. C. Liu, "Modelling of the thermosolutal convection and macrosegregation in the solidification of an Fe-C binary alloy," *Int J Numer Methods Heat Fluid Flow*, vol. 17, no. 3, pp. 313–321, Jan. 2007, doi: 10.1108/09615530710730175.

[180] H. Zhao, Z. Zhang, Y. Bai, B. Li, and M. Gao, "Numerical and Experimental Study on the Direct Chill Casting of Large-Scale AA2219 Billets via Annular Coupled Electromagnetic Field," *Materials*, vol. 15, no. 5, Mar. 2022, doi: 10.3390/ma15051802.

[181] Y. Cao *et al.*, "On the Mechanism of Steel Homogenization via Rare Earth Addition: Experimental Characterization and Numerical Simulation," *Metallurgical and Materials Transactions B: Process Metallurgy and Materials Processing Science*, vol. 53, no. 3, pp. 1858–1874, Jun. 2022, doi: 10.1007/s11663-022-02496-4.

[182] C. Wang, Z. Liu, and B. Li, "A three-phase volume-averaged solidification model considering the growth direction of columnar crystal axis," *Int J Heat Mass Transf*, vol. 194, p. 122974, Sep. 2022, doi: 10.1016/j.ijheatmasstransfer.2022.122974.

[183] F. Wang, I. Tzanakis, D. Eskin, J. Mi, and T. Connolley, "In situ observation of ultrasonic cavitation-induced fragmentation of the primary crystals formed in Al alloys," *Ultrason Sonochem*, vol. 39, pp. 66–76, Nov. 2017, doi: 10.1016/j.ultsonch.2017.03.057.

[184] A. C. FOWLER, “The Formation of Freckles in Binary Alloys,” *IMA J Appl Math*, vol. 35, no. 2, pp. 159–174, Sep. 1985, doi: 10.1093/imamat/35.2.159.

[185] S. D. Felicelli, D. R. Poirier, and J. C. Heinrich, “Modeling Freckle Formation in Three Dimensions during Solidification of Multicomponent Alloys,” *Metallurgical and Materials Transactions B: Process Metallurgy and Materials Processing Science*, vol 29, 1998, doi: 10.1007/s11663-998-0144-5.

[186] G. Zimmermann *et al.*, “Fragmentation-driven grain refinement in directional solidification of AlCu10wt-% alloy at low pulling speeds,” *Acta Mater*, vol. 126, pp. 236–250, Mar. 2017, doi: 10.1016/j.actamat.2016.12.063.

[187] T. Campanella, C. Charbon, and M. Rappaz, “Grain Refinement Induced by Electromagnetic Stirring: A Dendrite Fragmentation Criterion,” *Metallurgical and Materials Transactions A*, vol. 35, 2004, doi: 10.1007/s11661-004-0064-1.

[188] H. J. Dai *et al.*, “Grain selection during solidification in spiral grain selector,” in *Proceedings of the International Symposium on Superalloys*, Minerals, Metals and Materials Society, 2008, pp. 367–374. doi: 10.7449/2008/superalloys_2008_367_374.

[189] C. H. Ng, M. J. Bermingham, L. Yuan, and M. S. Dargusch, “Towards β-fleck defect free additively manufactured titanium alloys by promoting the

columnar to equiaxed transition and grain refinement," *Acta Mater*, vol. 224, Feb. 2022, doi: 10.1016/j.actamat.2021.117511.

[190] N. Ren, J. Li, C. Panwisawas, M. Xia, H. Dong, and J. Li, "Insight into the sensitivities of freckles in the directional solidification of single-crystal turbine blades," *J Manuf Process*, vol. 77, pp. 219–228, May 2022, doi: 10.1016/j.jmapro.2022.03.019.

[191] A. Kao, N. Shevchenko, M. Alexandrakis, I. Krastins, S. Eckert, and K. Pericleous, "Thermal dependence of large-scale freckle defect formation," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 377, no. 2143, Apr. 2019, doi: 10.1098/rsta.2018.0206.

[192] H. Xing, X. Dong, D. Sun, and Y. Han, "Anisotropic lattice Boltzmann-phase-field modeling of crystal growth with melt convection induced by solid-liquid density change," *J Mater Sci Technol*, vol. 57, pp. 26–32, Nov. 2020, doi: 10.1016/j.jmst.2020.05.011.

[193] J. Cui, B. Li, Z. Liu, F. Qi, J. Xu, and J. Zhang, "Comparative investigation on ingot evolution and product quality under different arc distributions during vacuum arc remelting process," *Journal of Materials Research and Technology*, vol. 18, pp. 3991–4006, May 2022, doi: 10.1016/j.jmrt.2022.04.058.

[194] R. Ananth and W. N. Gill B', "Dendritic growth in microgravity and forced convection," Journal of Crystal Growth, vol. 179, pp. 263-276, 1997, doi: 10.1016/S0022-0248(97)00143-7

[195] H.-J. Diepers, C. Beckermann, and I. Steinbach, “Simulation Of Convection And Ripening In A Binary Alloy Mush Using The Phase-Field Method,” Acta Materialia, vol. 47, pp. 3663-3678, 1999, doi: 10.1016/S1359-6454(99)00239-6

[196] C. Beckermann, H.-J. Diepers, I. Steinbach, A. Karma, and X. Tong, “Modeling Melt Convection in Phase-Field Simulations of Solidification,” Journal of Computational Physics, vol. 154, pp. 468-496, 1999, doi: 10.1006/jcph.1999.6323

[197] R. T. Tönhardt and G. Amberg, “Phase-field simulation of dendritic growth in a shear flow,” *Journal of Crystal Growth*, vol. 194, issue 3, 1998, doi: 10.1016/s0022-0248(98)00687-3

[198] C. C. Chen and C. W. Lan, “Efficient adaptive three-dimensional phase field simulation of free dendritic growth under natural convection,” *J Cryst Growth*, vol. 312, no. 8, pp. 1437–1442, Apr. 2010, doi: 10.1016/j.jcrysgro.2009.11.008.

[199] R. Tö and G. Amberg, “Simulation of natural convection effects on succinonitrile crystals,” *Phys. Rev. E*, vol. 62, 199, doi: 10.1103/PhysRevE.62.828.

[200] N. Al-Rawahi and G. Tryggvason, “Numerical simulation of dendritic solidification with convection: Three-dimensional flow,” *J Comput Phys*, vol. 194, no. 2, pp. 677–696, Mar. 2004, doi: 10.1016/j.jcp.2003.09.020.

[201] L. Yuan and P. D. Lee, “Number 5 Citation Lang Yuan and Peter D Lee,” *Modelling Simul. Mater. Sci. Eng*, vol. 18, p. 55008, 2010, doi:

10.1088/0965-0393/18/5/055008/meta?casa_token=PP4iUNB3fekAAAAA:cYHlqx9XxmZ6pawnjhKunzbsD2e37kvJ0n.

[202] A. Jakhar, A. Bhattacharya, P. Rath, and S. K. Mahapatra, "Combined Effect of Thermal Anisotropy and Forced Convection on the Growth of Binary Alloy Equiaxed Dendrites," *J Therm Sci Eng Appl*, vol. 11, no. 5, Oct. 2019, doi: 10.1115/1.4042587.

[203] T. Takaki, S. Sakane, M. Ohno, Y. Shibuta, and T. Aoki, "Large–scale phase–field lattice Boltzmann study on the effects of natural convection on dendrite morphology formed during directional solidification of a binary alloy," *Comput Mater Sci*, vol. 171, Jan. 2020, doi: 10.1016/j.commatsci.2019.109209.

[204] S. Sakane, T. Takaki, M. Ohno, Y. Shibuta, and T. Aoki, "Two-dimensional large-scale phase-field lattice Boltzmann simulation of polycrystalline equiaxed solidification with motion of a massive number of dendrites," *Comput Mater Sci*, vol. 178, Jun. 2020, doi: 10.1016/j.commatsci.2020.109639.

[205] S. G. Kim, D. I. Kim, W. T. Kim, and Y. B. Park, "Computer simulations of two-dimensional and three-dimensional ideal grain growth," *Phys Rev E Stat Nonlin Soft Matter Phys*, vol. 74, no. 6, 2006, doi: 10.1103/PhysRevE.74.061605.

[206] T. Takaki, T. Hirouchi, Y. Hisakuni, A. Yamanaka, and Y. Tomita, "Multi-phase-field model to simulate microstructure evolutions during dynamic

recrystallization," *Mater Trans*, vol. 49, no. 11, pp. 2559–2565, Nov. 2008, doi: 10.2320/matertrans.MB200805.

[207] M. Ohno, S. Tsuchiya, and K. Matsuura, "Formation conditions of coarse columnar austenite grain structure in peritectic carbon steels by the discontinuous grain growth mechanism," *Acta Mater*, vol. 59, no. 14, pp. 5700–5709, Aug. 2011, doi: 10.1016/j.actamat.2011.05.045.

[208] A. Zhang, S. Meng, Z. Guo, J. Du, Q. Wang, and S. Xiong, "Dendritic Growth Under Natural and Forced Convection in Al-Cu Alloys: From Equiaxed to Columnar Dendrites and from 2D to 3D Phase-Field Simulations," *Metallurgical and Materials Transactions B: Process Metallurgy and Materials Processing Science*, 2019, doi: 10.1007/s11663-019-01549-5.

[209] D. Sun, M. Zhu, S. Pan, and D. Raabe, "Lattice Boltzmann modeling of dendritic growth in a forced melt convection," *Acta Mater*, vol. 57, no. 6, pp. 1755–1767, Apr. 2009, doi: 10.1016/j.actamat.2008.12.019.

[210] A. A. Mohamad, *Lattice Boltzmann Method Fundamentals and Engineering Applications with Computer Codes Second Edition*, Springer Nature, 2019, doi: 10.1007/978-1-4471-7423-3

[211] V. Thomée, *Galerkin Finite Element Methods for Parabolic Problems*, Springer Berlin, 2006, doi: 10.1007/3-540-33122-0.

[212] J. Dongarra, A. Lumsdaine, R. Pozo, and K. Remington, "IML++ v. 1.2," *Iterative methods library reference guide, NIST Washington, ORNL, Univ. of Notre Dame and Univ. of Tennessee at Knoxville*, 1995.

[213] M. J. M. Krane, D. R. Johnson, and S. Raghavan, “The development of a cellular automaton-finite volume model for dendritic growth,” *Appl Math Model*, vol. 33, no. 5, pp. 2234–2247, May 2009, doi: 10.1016/j.apm.2008.06.007.

[214] M. F. Zhu and C. P. Hong, “A three-dimensional cellular automaton model for dendritic growth in multi-component alloys,” Acta Materialia, vol 60, issue 5, pp. 2249-2257, 2012 doi: 10.1016/j.actamat.2011.12.045.

[215] S. Shahane, N. Aluru, P. Ferreira, S. G. Kapoor, and S. P. Vanka, “Finite volume simulation framework for die casting with uncertainty quantification,” *Appl Math Model*, vol. 74, pp. 132–150, Oct. 2019, doi: 10.1016/j.apm.2019.04.045.

[216] H. S. Udaykumar, R. Mittal, and P. Rampunggoon, “Interface tracking finite volume method for complex solid-fluid interactions on fixed meshes,” *Commun Numer Methods Eng*, vol. 18, no. 2, pp. 89–97, Feb. 2002, doi: 10.1002/cnm.468.

[217] A. Karma and J. Rappel, “Quantitative phase-field modeling of dendritic growth in two and three dimensions,” *Phys. Rev. E*, vol. 57, 1998, doi: 10.1103/PhysRevE.57.4323.

[218] R. Li, Y. Di, and T. Tang, “A General Moving Mesh Framework in 3D and its Application for Simulating the Mixture of Multi-Phase Flows,” *Communications in Computational Physics*, vol. 3, No. 3, pp. 582-602, 2007.

[219] H. Wang, R. Li, and T. Tang, “Efficient computation of dendritic growth with r-adaptive finite element methods,” *J Comput Phys*, vol. 227, no. 12, pp. 5984–6000, Jun. 2008, doi: 10.1016/j.jcp.2008.02.016.

[220] G. H. Farrahi, “Finite element simulation of residual stresses in laser heating Application of Homotopy Analysis Method (HAM) in studying MEMS behaviour View project Plim Project PSI View project,” *The 7th Iranian Aerospace Society Conference*, 2008.

[221] J. Yang and F. Wang, “3D finite element temperature field modelling for direct laser fabrication,” *International Journal of Advanced Manufacturing Technology*, vol. 43, no. 11–12, pp. 1060–1068, Aug. 2009, doi: 10.1007/s00170-008-1785-x.

[222] Q. Qin, S. Shang, D. Wu, and Y. Zang, “Comparative analysis of bulge deformation between 2D and 3D finite element models,” *Advances in Mechanical Engineering*, vol. 2014, 2014, doi: 10.1155/2014/942719.

[223] V. E. Alexopoulou, E. L. Papazoglou, P. Karmiris-Obratański, and A. P. Markopoulos, “3D finite element modeling of selective laser melting for conduction, transition and keyhole modes,” *J Manuf Process*, vol. 75, pp. 877–894, Mar. 2022, doi: 10.1016/j.jmapro.2022.01.054.

[224] F. Varnik and D. Raabe, “Chaotic flows in microchannels: a lattice Boltzmann study,” *Mol Simul*, vol. 33, pp. 583–587, 2006.

[225] Varnik, F., Dorner, D., & Raabe, D, “Roughness-induced flow instability: A lattice Boltzmann study,” *Journal of Fluid Mechanics*, vol.573, pp. 191-209. doi:10.1017/S0022112006003715

[226] F. Varnik and D. Raabe, “Scaling effects in microscale fluid flows at rough solid surfaces,” *Model Simul Mat Sci Eng*, vol. 14, pp. 857–873, 2006.

[227] A. K. Gunstensen and D. H. Rothman, “Lattice-Boltzmann studies of immiscible two-phase flow through porous media,” *J Geophys Res Solid Earth*, vol. 98, no. B4, pp. 6431–6441, 1993, doi: https://doi.org/10.1029/92JB02660.

[228] G. R. Mcnamara and G. Zanetti ’, “Use of the Soltzmann Equation to Simulate Lattice-Gas Automata,” Phys. Rev. Lett., vol. 61, doi: 10.1103/PhysRevLett.61.2332.

[229] R. Rojas, T. Takaki, and M. Ohno, “A phase-field-lattice Boltzmann method for modeling motion and growth of a dendrite for binary alloy solidification in the presence of melt convection,” *J Comput Phys*, vol. 298, pp. 29–40, Oct. 2015, doi: 10.1016/j.jcp.2015.05.045.

[230] D. Medvedev, F. Varnik, and I. Steinbach, “Simulating mobile dendrites in a flow,” in *Procedia Computer Science*, Elsevier B.V., 2013, pp. 2512–2520. doi: 10.1016/j.procs.2013.05.431.

[231] S. Zhang, C. Li, and R. Li, “Cellular automata-lattice Boltzmann simulation of multi-dendrite motion under convection based on dynamic grid technology,” *Mater Today Commun*, vol. 31, Jun. 2022, doi: 10.1016/j.mtcomm.2022.103342.

[232] B. Jelinek, M. Eshraghi, S. Felicelli, and J. F. Peters, “Large-scale parallel lattice Boltzmann-cellular automaton model of two-dimensional dendritic

growth," *Comput Phys Commun*, vol. 185, no. 3, pp. 939–947, Mar. 2014, doi: 10.1016/j.cpc.2013.09.013.

[233] D. Medvedev, F. Varnik, and I. Steinbach, "Simulating mobile dendrites in a flow," in *Procedia Computer Science*, Elsevier B.V., 2013, pp. 2512–2520. doi: 10.1016/j.procs.2013.05.431.

[234] A. Subhedar, I. Steinbach, and F. Varnik, "Modeling the flow in diffuse interface methods of solidification," *Phys Rev E Stat Nonlin Soft Matter Phys*, vol. 92, no. 2, Aug. 2015, doi: 10.1103/PhysRevE.92.023303.

[235] "Open Phase," www.openphase.de. (accessed Feb. 18, 2023).

[236] Y. Ren, Z. Liu, Y. Pang, X. Wang, and Y. Xu, "Lattice Boltzmann simulation of phase change and heat transfer characteristics in the multi-layer deposition," *Applied Mathematics and Mechanics (English Edition)*, vol. 42, no. 4, pp. 553–566, Apr. 2021, doi: 10.1007/s10483-021-2720-7.

[237] S. Sakane, T. Aoki, and T. Takaki, "Parallel-GPU AMR implementation for phase-field lattice Boltzmann simulation of a settling dendrite," *Comput Mater Sci*, vol. 211, p. 111542, Aug. 2022, doi: 10.1016/j.commatsci.2022.111542.

[238] C. A. De Moura and C. S. Kubrusly, "The courant–friedrichs–lewy (cfl) condition," *AMC*, vol. 10, no. 12, 2013.

[239] S. A. Nabavizadeh, M. Eshraghi, S. D. Felicelli, S. N. Tewari, and R. N. Grugel, "Effect of bubble-induced Marangoni convection on dendritic solidification," *International Journal of Multiphase Flow*, vol. 116, pp. 137–152, Jul. 2019, doi: 10.1016/j.ijmultiphaseflow.2019.04.018.

[240] F. S. Nooruddin and G. Turk, “Simplification and Repair of Polygonal Models Using Volumetric Techniques,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 9, no. 2, pp. 191-205, April-June 2003, doi: 10.1109/TVCG.2003.1196006.

[241] Y. Hu *et al.*, “CA method with machine learning for simulating the grain and pore growth of aluminum alloys,” *Comput Mater Sci*, vol. 142, pp. 244–254, Feb. 2018, doi: 10.1016/j.commatsci.2017.09.059.

[242] A. Dong and L. Nastac, “Prediction of Secondary Dendrite Arm Spacing in Al Alloys Using Machine Learning,” *Metallurgical and Materials Transactions B: Process Metallurgy and Materials Processing Science*, vol. 52, no. 4, pp. 2395–2403, Aug. 2021, doi: 10.1007/s11663-021-02183-w.

[243] F. Nikolić, I. Štajduhar, and M. Čanađija, “Casting Defects Detection in Aluminum Alloys Using Deep Learning: a Classification Approach,” *International Journal of Metalcasting*, 2022, doi: 10.1007/s40962-022-00777-x.

[244] C. Hu, S. Martin, and R. Dingreville, “Accelerating phase-field predictions via recurrent neural networks learning the microstructure evolution in latent space,” *Comput Methods Appl Mech Eng*, vol. 397, p. 115128, Jul. 2022, doi: 10.1016/j.cma.2022.115128.

[245] H. Abdi and L. J. Williams, “Principal component analysis,” *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 2, no. 4. pp. 433–459, Jul. 2010. doi: 10.1002/wics.101.

[246] C. Orsenigo and C. Vercellis, “An effective double-bounded tree-connected Isomap algorithm for microarray data classification,” *Pattern Recognit Lett*, vol. 33, no. 1, pp. 9–16, Jan. 2012, doi: 10.1016/j.patrec.2011.09.016.

[247] L. McInnes, J. Healy, and J. Melville, “UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction,” *arXiv*, 2020, doi: 10.48550/arXiv.1802.03426

[248] K. Cho *et al.*, “Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation,” *arXiv*, 2014, doi: 10.48550/arXiv.1406.1078

[249] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Comput*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[250] N. Ren, J. Li, N. Bogdan, M. Xia, and J. Li, “Simulation of dendritic remelting and fragmentation using coupled cellular automaton and Eulerian multiphase model,” *Comput Mater Sci*, vol. 180, Jul. 2020, doi: 10.1016/j.commatsci.2020.109714.

[251] Z. Guo, J. Mi, S. Xiong, and P. S. Grant, “Phase field simulation of binary alloy dendrite growth under thermal- and forced-flow fields: An implementation of the parallel-multigrid approach,” *Metallurgical and Materials Transactions B: Process Metallurgy and Materials Processing Science*, vol. 44, no. 4, pp. 924–937, Aug. 2013, doi: 10.1007/s11663-013-9861-5.

APPENDIX

Code A 1

```
# Permeability Conversion
 # This code inputs an csv of data points that represents a permeable solid for the LB code
import numpy as np
import pandas as pd
import csv

# READ CSV
import os
path = 'C:\\<PATH>\\Documents'
output_file = os.path.join(path,'vel.csv')

df = pd.read_csv (output_file, index_col = [0])
print (df)



# importing the csv library
import csv

# opening the csv file by specifying
# the location
# with the variable name as csv_file
```

```
with open(<PATH>/vel.csv') as csv_file:

    # creating an object of csv reader
    # with the delimiter as ,
    csv_reader = csv.reader(csv_file, delimiter = ',')

    # list to store the names of columns
    list_of_column_names = []

    # loop to iterate through the rows of csv
    for row in csv_reader:

        # adding the first row
        list_of_column_names.append(row)

        # breaking the loop after the
        # first iteration itself
        break

# printing the result
print("List of column names : ",
      list_of_column_names[0])
```

```
#df.set_index(['v_x', 'v_y','v_z'])


for x, y in df.iteritems():
  print(x)
  print(y)


#Parameters
mew = 0.001 #Pa/s
g_p = 100   #Pa/m


#v_g = k*g_p/mew
# solving for k
#  k = v_g*mew/g_p
#    = v_bar_x*mew/(dp/dx)


#Average velocity


for i in range (len(v_x)):


    v_bar_x = sum(v_x)/i
```

```
for i in range (len(v_y)):
    v_bar_y = sum(v_y)/i

for i in range (len(v_z)):
    v_bar_z = sum(v_z)/i

for i in range (len(v_y)):
    k_xx =  v_bar_x*mew/(dp/dx)
    k_xy =  v_bar_x*mew/(dp/dy)
    k_xz =  v_bar_x*mew/(dp/dz)

    k_yx =  v_bar_y*mew/(dp/dx)
    k_yy =  v_bar_y*mew/(dp/dy)
    k_yz =  v_bar_y*mew/(dp/dz)

    k_zx =  v_bar_z*mew/(dp/dx)
    k_zy =  v_bar_z*mew/(dp/dy)
    k_zz =  v_bar_z*mew/(dp/dz)
    print(k_xx,k_xy,k_xz,k_yx,k_yy,k_yz,k_zx,k_zy,k_zz)
#K = [k_xx, k_xy, k_xz; k_yx, k_yy, k_yz; k_zx, k_zy, k_zz]

#np.savetxt('scores.csv', K, delimiter=',', fmt='%s')
```

Code A 2

In [12]:

#Steps to take an stl input and create a bounding box for the LB code

#1 - Create bounding box for stl part - import box to stl part (using insert)

#2- make negative from using insert->features->combine->subtract

#3 - Convert from Binary to ASCII using code below (1)

#4 - Convert ASCII STL to CSV using stl-to-csv.exe or .py

#5 - Make coordinate integers and positive values using code below (2)

#6 - calculate phi by taking difference of csvs using code below (3)

In [10]:

#!/usr/bin/python

```
from struct import unpack
file = "<PATH>/tree1"
fileNew = file + "ASCII"
f = open(file + ".stl","rb")

header = f.read(80)
numfacets = unpack("<I",f.read(4))[0]



with open(fileNew + ".stl", 'w') as fw:
```

```
fw.write("solid stock" + "\n")
for i in range(1,numfacets):
    a1 = unpack("<f",f.read(4))[0]
    a2 = unpack("<f",f.read(4))[0]
    a3 = unpack("<f",f.read(4))[0]

    v11 = unpack("<f",f.read(4))[0]
    v12 = unpack("<f",f.read(4))[0]
    v13 = unpack("<f",f.read(4))[0]

    v21 = unpack("<f",f.read(4))[0]
    v22 = unpack("<f",f.read(4))[0]
    v23 = unpack("<f",f.read(4))[0]

    v31 = unpack("<f",f.read(4))[0]
    v32 = unpack("<f",f.read(4))[0]
    v33 = unpack("<f",f.read(4))[0]

    attribs = unpack("<H",f.read(2))
    fw.write("facet normal " + str(a1) + " " + str(a2) +" " + str(a3) + "\n")
    fw.write("  outer loop" + "\n")
    fw.write("    vertex " + str(v11) + " " + str(v12) + " " + str(v13) + "\n")
    fw.write("    vertex " + str(v21) + " " + str(v22) + " " + str(v33) + "\n")
```

```
            fw.write("    vertex " + str(v31) + " " + str(v32) + " " + str(v33) + "\n")
            fw.write("  endloop" + "\n")
            fw.write("endfacet" + "\n")
        fw.write("endsolid stock" + "\n")
```

In [13]:

```
import pandas as pd
df = pd.read_csv (r'<PATH>tree2ASCII-table.csv',index_col=False)
df2=pd.DataFrame([df["x_coord"]-df["x_coord"].min(),
df["y_coord"]-df["y_coord"].min(),
df["z_coord"]-df["z_coord"].min()])
df3 = df2.round(0).astype(int).transpose()
df3 = df3.assign(frac0='1')
df3.to_csv("C:/<PATH>/tree2c.csv", index=False)
```

In [17]:

```
import pandas as pd
df4 = pd.read_csv (r'C:\<PATH>\tree1ASCII-table.csv',index_col=False)
df5=pd.DataFrame([df["x_coord"]-df["x_coord"].min(),
df4["y_coord"]-df4["y_coord"].min(),
df4["z_coord"]-df4["z_coord"].min()])
df6 = df5.round(0).astype(int).transpose()
df6 = df6.assign(frac0='1')
df6.to_csv("<PATH>/tree1c.csv", index=False)
```

In [15]:

print(df3.shape)

(13005, 4)

In [18]:

print(df6.shape)

(14517, 4)