Formation Mechanism and Computational Modelling of Isle of Rum Plagioclase Stellates

by Steven Zhang

Thesis submitted to the

Faculty of Graduate and Postdoctoral Studies

In partial fulfillment of the requirements

For the Ph.D. degree in Earth Sciences

Department of Earth Sciences

University of Ottawa

© Steven Zhang, Ottawa, Canada, 2013

Abstract

We propose a hypothesis and a numerical model for the formation of branching plagioclase textures visible at both macroscopic (∼cm to ∼m) and microscopic scale within melagabbro of the Isle of Rum, Scotland, based on macroscopic, microscopic observations and relevant geological history. The plagioclase crystals are typically linked as twins and form meshes of planar stellate structures (m-scale) with a large range in geometrical organization from patchy to radiating. Evidence of macroscopic crystal aggregation and alignment is attributed to interfacial free energy minimization at the microscopic scale during growth. Accordingly, a binary immiscible Lattice Boltzmann model was developed to simulate diffusion of simplified plagioclase in the melt phase. Isothermal phase transitions modelled via first order chemical reactions are subsequently coupled with stochastic dynamics at the crystal growth front to simulate energy minimization processes including twinning during crystallization in an igneous environment. The solid phase and the liquid phase are coupled with a temporal flexibility that sets the overall ratio between the rate of diffusion and chemical enrichment in the liquid state and the rate of crystallization. The parameter space of the model is explored extensively, followed by a reasonable transcription of physical parameters and an estimation of other parameters to construct realistic simulation scenarios yielding synthetic plagioclase stellates.

The results are presented, analyzed and discussed. They appear to be in reasonable qualitative agreement with observations, and several aspects of the natural stellates such as the stellate spacing and long continuous stretches of plagioclase with epitaxial junctions seem to be in reasonable quantitative agreement with observations.

Acknowledgements

Many appreciations and thanks to Anthony D. Fowler and Ivan L'Heureux, for their ongoing support, encouragement and mentoring.

Special thanks to André E. Lalonde, for his personal and professional support since before this project began.

Thanks to my family for their shelter and support. Without them, graduate life can become much harder. I am especially thankful to my grandparent, whose scientific inspirations, insights and endless love since my birth has undoubtedly shaped my path to this thesis.

Thanks to my friends (including as a subset, but certainly not limited to: {Jay, Ben, Sev, Jean-Michel, Julian, Alex}), who share with me and each other, the goals of exploring and learning, appreciating, critical thinking and relaxing.

Special thanks to Julie Bourdeau, for all moments in life is better when shared.

Contents

1 Introduction **1**

2 Geological Background and Observations **4**

2.1 Geological History of the Isle of Rum 4

2.2 Plagioclase Stellate Patterns 12

2.3 Experimental Methods 27

2.3.1 Microscopy 32

2.3.2 Electron Back Scatter Diffraction 43

2.4 Brief Discussion 48

2.5 The Crucial Bits 52

3 Theory **55**

3.1 Genesis of the Stellate Pattern: A Working Hypothesis 55

3.1.1 Necessary Conditions 55

3.1.2 A Little Story 56

3.1.3 Alternative Ideas 63

3.1.4 Uniqueness of Plagioclase Stellates 65

3.2 Nucleation and Spinodal Decomposition, Crystallization in a Melt 67

3.2.1 Choice of Models 79

3.3 Lattice Boltzmann Scheme: A Primer 82

3.3.1 Introduction to the Lattice Boltzmann Method 82

3.3.2 The Boltzmann Equation 83

3.3.3 The Lattice Boltzmann Equation 88

3.3.4 A Functional Lattice Boltzmann Model 91

3.3.5 Immiscible Lattice Boltzmann Model . . . 100
3.4 Stochastics . . . 107

4 Model **121**
4.1 Phase Space Implementation and Parameters . . . 121
4.1.1 Coupling of Liquid to Solid . . . 136
4.2 Coding Illustration . . . 141
4.3 Simulation Requirements . . . 146
4.4 Output Methods . . . 151

5 Exploration of Parameter Space **155**
5.1 Liquid State Parameters . . . 155
5.1.1 Viscosity Parameter ω . . . 156
5.1.2 Surface Tension Parameter α_i . . . 160
5.1.3 Global Plagioclase Enrichment Parameter R . . . 163
5.1.4 Interfacial Region Mixing Length Parameter β_0 . . . 165
5.2 Solid State Parameters . . . 168
5.2.1 Gaussian Parameters b, σ . . . 168
5.2.2 Anisotropy Parameter Table $p2$ . . . 173
5.2.3 Threshold Parameters a_1, a_2 . . . 178
5.2.4 Perturbation Parameters A, B, C . . . 181
5.2.5 Neighbourhood Size Parameter n . . . 185
5.3 Liquid-Solid Coupling Parameters . . . 187
5.3.1 Solid State to Liquid State Parameter q . . . 187

6 Application to Plagioclase Stellates, Analysis and Discussion **193**
6.1 Towards Stellate Growth Scenarios . . . 193
6.2 Realistic Stellate Growth Scenarios . . . 202
6.3 Analysis of Results . . . 217

7 Conclusion **244**

References **252**

A Appendix on Diffusion **253**

A.1 Classical Derivation 254

A.2 Microscopic Perspective 256

A.3 Solutions to the Diffusion Equation 258

A.3.1 Analytical Solution to the Diffusion Equation 259

A.4 Comparison of Simulation with Analytical Solution 264

B Simulation Source Code **272**

B.1 LBM.f90 274

B.2 Streaming.f90 281

B.3 Collision.f90 283

B.4 Collision_Regular.f90 290

B.5 Phase.f90 292

B.6 Setup.f90 296

B.7 Flagging.f90 305

B.8 INOUT.f90 308

List of Figures

2.1 Location of the Isle of Rum. 5

2.2 Coarse geological map of Rum. 7

2.3 Finger structures in the CS. 10

2.4 Replacement structures in the ELS. 11

2.5 Sample area of the CS. 13

2.6 Nodular structures in the stellates. 14

2.7 Typical picture of two stellates. 15

2.8 Anatomy of plagioclase stellates. 16

2.9 Sub-radial stellate texture. 16

2.10 Medium sized radial stellate texture. 17

2.11 Patchy stellate texture - 1/3. 18

2.12 Patchy stellate texture - 2/3. 19

2.13 Patchy stellate texture - 3/3. 20

2.14 Sub-radial, loosely organized to patchy stellate texture. 21

2.15 CS layers, vertical exposed sections 1/2. 22

2.16 CS layers, vertical exposed sections 1/2. 23

2.17 Two different layers exhibiting stellate plagioclase. 24

2.18 Large radiating stellate texture. 25

2.19 Connections between stellate systems. 26

2.20 Rock section sampled by Labonté. 29

2.21 Ternary diagrams of plagioclase, olivine and clinopyroxenes. 30

2.22 EMP backscatter image showing zoning in plagioclase. 31

2.23 Typical clinopyroxene rich thin-section. 33

2.24 Typical stellate bearing thin-section. . . . 34
2.25 Twin lamellae of plagioclase in stellate bearing thin-section. . . . 35
2.26 Typical stellate bearing thin-section. . . . 36
2.27 Typical stellate bearing thin-section - Rum D'. . . . 37
2.28 Typical stellate bearing thin-section with slip boundaries illustrated. . . . 38
2.29 Chromite band from magma mixing. . . . 39
2.30 Embayed chromite and olivine crystals. . . . 39
2.31 Plagioclase phenocryst inside of olivine crystal. . . . 40
2.32 Plagioclase phenocryst inside of olivine crystal. . . . 41
2.33 Plagioclase crystal encapsulating another plagioclase crystal. . . . 42
2.34 EBSD data of olivine alignment of Rum-D. . . . 44
2.35 Plagioclase twins under EBSD. . . . 45
2.36 Olivine alignment. . . . 46
2.37 Plagioclase twins under EBSD. . . . 47

3.1 Illustration of hypothesis, macroscopic phenomena. . . . 61
3.2 Illustration of hypothesis, microscopic phenomena. . . . 62
3.3 Classical phase diagram. . . . 69
3.4 Amplification of fluctuation ΔC. . . . 70
3.5 Spinodal decomposition stages. . . . 71
3.6 Comparison of classical and non-classical decomposition. . . . 76
3.7 Illustration of DnQm nomenclature. . . . 92
3.8 Illustration of the collision operator. . . . 93
3.9 Illustration of the streaming process. . . . 95
3.10 Illustration of no-slip bounce-back boundary condition. . . . 100
3.11 Illustration of immiscible multi-phase collision operator. . . . 106
3.12 Illustration of under-determined cell. . . . 113
3.13 Illustration of epitaxy algorithm. . . . 114
3.14 Illustration of growth algorithm. . . . 116
3.15 Illustration of combined phase transition rule. . . . 117

4.1 Effect of b on $f(\rho 1)$. . . . 129

4.2 Effects of σ on $f(\rho 1)$. 129
4.3 Simultaion block diagram. 141
4.4 Routines in the LBM portion. 142
4.5 Results of comparison of precision-related compiler optimizations. 145
4.6 Run time vs. simulation size. 148

5.1 Results of synchronous sweep of ω_i. 158
5.2 Results of differential sweep of ω_i. 159
5.3 Results of sweep of α. 162
5.4 Results of sweep of R. 164
5.5 Results of sweep of β_0. 167
5.6 Results of sweep of σ and growth rate. 170
5.7 Results of sweep of b. 171
5.8 Results of sweep of σ with concentration gradient. 172
5.9 Choice of diagonal attachment coefficients. 174
5.10 Results of sweep of Asr. 175
5.11 Results of sweep of Asr on the navigation of channels. 176
5.12 Results of sweep of a_2. 180
5.13 Results of sweeps of A, B. 182
5.14 Results of sweep of C. 184
5.15 Results of sweep of n. 186
5.16 Illustration of the steps utilized to obtain $\langle\lambda\rangle$ values. 188
5.17 FFT of $\langle\lambda\rangle$ versus Δt_l. 189
5.18 Results of sweep of q. 192

6.1 Results of sweep of $\Sigma\Delta t_{liquid}$. 197
6.2 Results of sweep of σ. 200
6.3 Realistic stellate patterns - 1/2. 207
6.4 Realistic stellate patterns - 2/2. 208
6.5 Realistic stellate patterns. Collision operator interleave disabled. 211
6.6 Histograms of concentrations of ρ_1 and ρ_2. 216
6.7 Histogram of cell types of a synthetic stellate. 218

6.8 Radial correlation function of a synthetic stellate. 218
6.9 Large-radius average angular correlation of synthetic stellates. 220
6.10 Small-radius angular correlation of synthetic stellates, $Asr = 1$. 221
6.11 Small-radius angular correlation of synthetic stellates, $Asr = 8$. 222
6.12 Small-radius angular correlation of synthetic stellates, $Asr = 20$. 223
6.13 Small-radius angular correlation of synthetic stellates, $Asr = 32$. 224
6.14 Plotted image of the horizontal solid structure for $Asr = 1$. 225
6.15 Plotted image of the horizontal solid structure for $Asr = 8$. 226
6.16 Plotted image of the horizontal solid structure for $Asr = 20$. 227
6.17 Plotted image of the horizontal solid structure for $Asr = 32$. 228
6.18 FFT results of sweep of β_0. 230
6.19 Plotted image of the synthetic stellate solid. $\beta_0 = 0.05$. 231
6.20 Plotted image of the synthetic stellate solid. $\beta_0 = 0.10$. 232
6.21 Plotted image of the synthetic stellate solid. $\beta_0 = 0.15$. 233
6.22 Plotted image of the synthetic stellate solid. $\beta_0 = 0.20$. 234
6.23 Plotted image of the synthetic stellate solid. $\beta_0 = 0.25$. 235
6.24 Plotted image of the synthetic stellate solid. $\beta_0 = 0.30$. 236
6.25 FFT results of sweep of R. 237
6.26 Plotted image of the synthetic stellate solid. $R = 0.25$. 238
6.27 Plotted image of the synthetic stellate solid. $R = 0.30$. 239
6.28 Plotted image of the synthetic stellate solid. $R = 0.35$. 240
6.29 Plotted image of the synthetic stellate solid. $R = 0.40$. 241
6.30 Plotted image of the synthetic stellate solid. $R = 0.45$. 242
6.31 Plotted image of the synthetic stellate solid. $R = 0.50$. 243

A.1 Illustration of elementary volume and flux vectors 254
A.2 Illustration of data sampling pattern over domain. 265
A.3 Data matching at $t = 100\Delta t$ 266
A.4 Data matching at $t = 1000\Delta t$ 267
A.5 Data matching at $t = 10000\Delta t$ 268
A.6 Data matching at $t = 30000\Delta t$ 269

A.7 Data matching at $t = 60000\Delta t$ 270
A.8 Data matching at $t = 100000\Delta t$ 271

Chapter 1

Introduction

Donaldson et al. (1973) first described the existence of stellate plagioclase as "macro-spherulites" on the surfaces of peridotitic and allivalitic (chiefly olivine and anorthite, with rare pyroxene) rocks in the Central Series of the Isle of Rum, Scotland. Stellate plagioclase is a term defined by Fowler et al. (2005) to describe an apparent linkage of rays of plagioclase crystals in mafic igneous layered intrusions. Donaldson et al. (1973) further pointed out that the plagioclase stellates "...are inconsistent with the much applied adcumulus crystal growth hypothesis which dominates current thinking on the derivation of ultrabasic rocks from basaltic magmas". Although the stellate pattern itself exhibits a large variety of behaviours, the most striking and typical cases which probably gave rise to its name can be seen in figures 2.7 and 2.10. The texture is brought into relief via differential weathering. Since Donaldson et al. (1973), numerous geological studies of Rum including the Central Series have been conducted, but none have touched on the subject of the stellate plagioclase. From 2000 to 2002, Fowler and Labonté mapped the known occurrences of stellate plagioclase and sampled accordingly. Petrological, mineralogical and geochemical studies were carried out in an attempt to characterize and understand the relationship between the olivine alignment, stellate plagioclase and the Central Series. It was concluded by Labonté (2004) that the texture is not appropriately "spherulitic" and the orientation of the platy olivines coincided with the plane of igneous layering but the formation of stellate plagioclase was not understood although hypotheses were proposed including viscous fingering and flow. The source of preferential, intra-layer plagioclase enrichment was not explained.

In this thesis, a source for the intra-layer plagioclase enrichment is proposed along with

a hypothesis that seeks to explain the entire formation process of the stellate plagioclase. In addition a supporting numerical model is proposed and the results are analyzed and discussed.

This thesis has five main chapters: the first detailing the geological observations relevant to the stellate textures; the second introducing the theory used in the simulations; the third describing various aspects of the model and the last two presenting the results and supporting discussions.

Chapter 2 (the geological observations) briefly describe the relevant past efforts to understand the Rum igneous system and the history of the discovery, exploration, observation and research on the Rum stellates. The results presented are an amalgamation of past and current efforts and encompasses several spatial scales from regional to outcrop, from hand sample to thin-section, and finally to micro-analytical techniques such as electron backscatter diffraction and electron micro-probe analysis pertaining to the stellate textures.

Chapter 3 (the theory chapter) proposes a novel hypothesis to explain the formation of the stellate patterns based on inter-layer melt heterogeneities (solid and/or liquid) and intra-layer layered igneous intrusion specific interactions. Some topics associated with the hypothesis are then summarized and discussed. The hypothesis is subsequently distilled to synthesize a corresponding numerical simulation proposition based on a combination of Lattice Boltzmann method and stochastics that captures the essentials of the underlying igneous environment and the hypothesis. This is followed by the introduction of the Boltzmann equation, which is gradually discretized in spatial and temporal coordinates, as well as in velocity space to eventually produce the Lattice Boltzmann equation. The equilibrium distribution function is derived and a collision operator which mimics natural particle collisions is adopted. Thereafter, the single component model is extended to a two-component model by appropriately amending the collision operator which distinguishes the two components via the existence of an non-zero interfacial energy. Subsequently, a stochastic algorithm which governs the phase transition process is proposed that uses local concentration of liquid phases and neighbourhood structural orientation information to determine the growth and epitaxy of the solid state. Lastly, some feedbacks are introduced between the solid state and the liquid state.

Chapter 4 (the model chapter) explains all the parameters used, establishes the known stability limits of the model and discusses the lattice parameters and their (if extant) physical equivalents. The coupling of the liquid to solid sub-models is explored and some considerations

are proposed. Subsequently, the schematic form of the model (block diagram) is presented, along with some considerations to initial conditions and compiler optimizations. The computer specific requirements of the simulation, along with some performance data and discussions are then presented. Finally, the output methods that are used to generate the images presented in the next chapter are described along with some discussions on the run-to-run consistency of simulations.

Chapter 5 (the first of the results chapters), each parameter is independently varied to cover the necessary and/or possible range. Whenever it is not possible to cover an entire range of parameter values (such as when the parameter has no clear bounds), a reasonable subset of the parameter is proposed with logical arguments. The results are presented, and if necessary, discussed and further analyzed. Thereafter, some parameters whose values are not known for the relevant system that strongly affect the result are explored via direct simulation to establish some possible values.

Chapter 6 (the last results chapter and discussions), the known physical parameter ranges are translated into lattice parameters and combined with hypothetical possible values of other parameters to simulate a set of realistic growth scenarios that result in stellate-like patterns in two dimensions. The results are then analyzed in several ways and discussed.

Appendix 1 includes some model validation-specific materials. The derivations of the diffusion equation, the analytical solution of the diffusion equation and finally the comparison of numerical results to the analytical results are presented in the first appendix.

Appendix 2 includes the simulation source code and a brief explanation.

As will be seen, the development of the stellate plagioclase texture requires a proper composition of the parent melt, an optimal melt heterogeneity, and some hydrodynamic inter-layer and/or intra-layer interactions.

Chapter 2

Geological Background and Observations

2.1 Geological History of the Isle of Rum

The Isle of Rum is a sparsely populated island located off the West coast of Scotland (figure 2.1) lying on a ridge of Precambrian rocks separating two Mesozoic basins (Binns et al., 1974). It was formed approximately $(63 - 51)$ million years ago by a volcanic system associated with the Mid Atlantic Ridge during the opening of the North Atlantic (Upton, 1988) and coincides with a zone of widespread lithospheric thinning during the late Palaeozoic and the Mesozoic (Thompson and Gibson, 1991). The interval of igneous activity contributing to the modern day Rum complex is thought to have been restricted to < 3Ma between $(61 - 58)$Ma. As the volcanic activity subsided, prolonged erosion to date has revealed an igneous layered complex at the historic core of a volcano.

Prior to the onset of volcanic activity, Rum principally consisted of Pre-Tertiary rocks made of a thick (> 2.5km) succession of WNW-dipping sandstones of the Proterozoic Torridon Group. Archaean - early Proterozoic (Lewisian) gneisses underlie the intra-basin ridge but are only exposed within the Rum Central Complex (Emeleus et al., 1996). Mesozoic rocks are limited to $\sim$ 50m of Triassic sandstones in the northwest Rum and faulted slivers of Lower Jurassic sandstones, limestones and shales preserved on the southeast edge of the central complex (Emeleus et al., 1996). Evidence suggests faulting of the Torridonian beds including movement on the Long Loch Fault (the prominent N-S fault bisecting the island, see figure 2.2), before

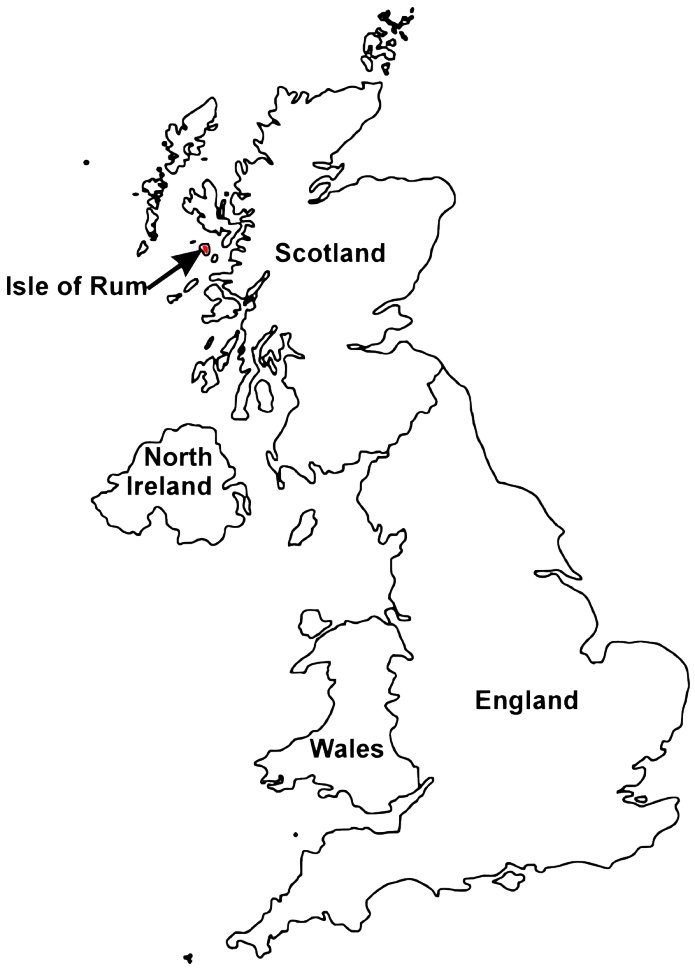


Figure 2.1 – Geographical location of Isle of Rum. Figure modified after Fowler (2000).

emplacement of the central complex (Emeleus et al., 1996).

As summarized by Emeleus et al. (1996), the igneous and tectonic events during the Palaeocene proceeded as follows:

(1) Igneous activity commenced with the eruption of basalt flows of the Eigg Lava Formation best observed on the neighbouring Isle of Eigg (Smith, 1985) (Emeleus, 1994). On Rum, these are restricted to faulted relics associated with the Jurassic sediments.

(2) Intrusion of sparse basaltic dykes and sills into the Torridonian beds probably post-dates 1 as suggested by Emeleus et al. (1996). However, they could have occurred close enough in time to be synchronous, (Emeleus et al., 1996).

(3) Early activity in the central complex initiated the Main Ring Fault, when Lewisian gneiss and basal members of the Torridon Group were brought to high levels within the complex. This was accompanied by tilting of country rocks and by the formation of the Welshman's Rock and

Mullach Ard Faults when masses of sandstone slid off a dome that was forming over the central complex, (Emeleus et al., 1996).

(4) Subsidence on the Main Ring Fault led to caldera formation, the intrusion of igneous breccias, the intrusion and extrusion (as ash flows) of rhyodacites, and the formation of breccias and mega-breccias, probably by collapse of caldera walls. Igneous breccias contain abundant blocks, in some cases exceeding 100m in diameter, of coarse gabbro, together with peridotite clasts, indicating mafic plutonic intrusions were already present at this early stage (Emeleus, 1996), (Emeleus et al., 1996).

(5) Intrusion of the Western Granite and other granitic rocks followed shortly after 4.

(6) Renewed uplift on the Main Ring Fault is recorded in the complex faulting found in the Southern Mountains Zone and the faulted contacts on the north of the Western Granite.

(7) A change from dominantly silicic to basaltic magmatism was marked by the intrusion of cone-sheets and radial dykes, and NW-trending dykes of the regional dyke swarm. The minor intrusions are predominantly of tholeiitic and transitional to mildly alkaline basalts (Forster (1980), McClurg (1982)). The cone-sheets focus at a depth of - 2.5km below central Rum, about 0.5 km southeast of Barkeval, a prominent peak in the central region of the island (Emeleus, 1996).

(8) Near-surface mafic magmatism reached its peak with the emplacement of plutonic basic and ultrabasic rocks that compose the Layered Suites (Eastern and Western Layered Series, and Central Series), together with an array of peridotite, gabbro and dolerite plugs. This involved the intrusion of large volumes of primitive basaltic magma. Rocks of the Layered Suite form the core of the central complex and are excellently exposed over about $35km^2$.

(9) Cessation of igneous activity in the central complex was followed by vigorous subaerial erosion of the complex and its surroundings. This was coeval with eruption of basaltic and more evolved flows of the Canna Lava Formation, which were fed from sources outside Rum.

(10) A few basalt dykes intrude the Canna Lava Formation. They include the only examples of normally-magnetized Palaeocene rocks from Rum and its surroundings (Mussett et al., 1996).

(11) A north-south fault system (the Long Loch Fault) bisects the complex. Dextral movement along the Long Loch Fault affected the Layered Suite and earlier rocks but the actual date of movement is otherwise poorly constrained.

In the above summary, events (3)-(6) are referred to as Phase I of the Central Complex in

figure 2.2, or Stage 1 in Emeleus et al. (1985). Event (8) is Phase II in figure 2.2 or stage 2 in Emeleus et al. (1985).

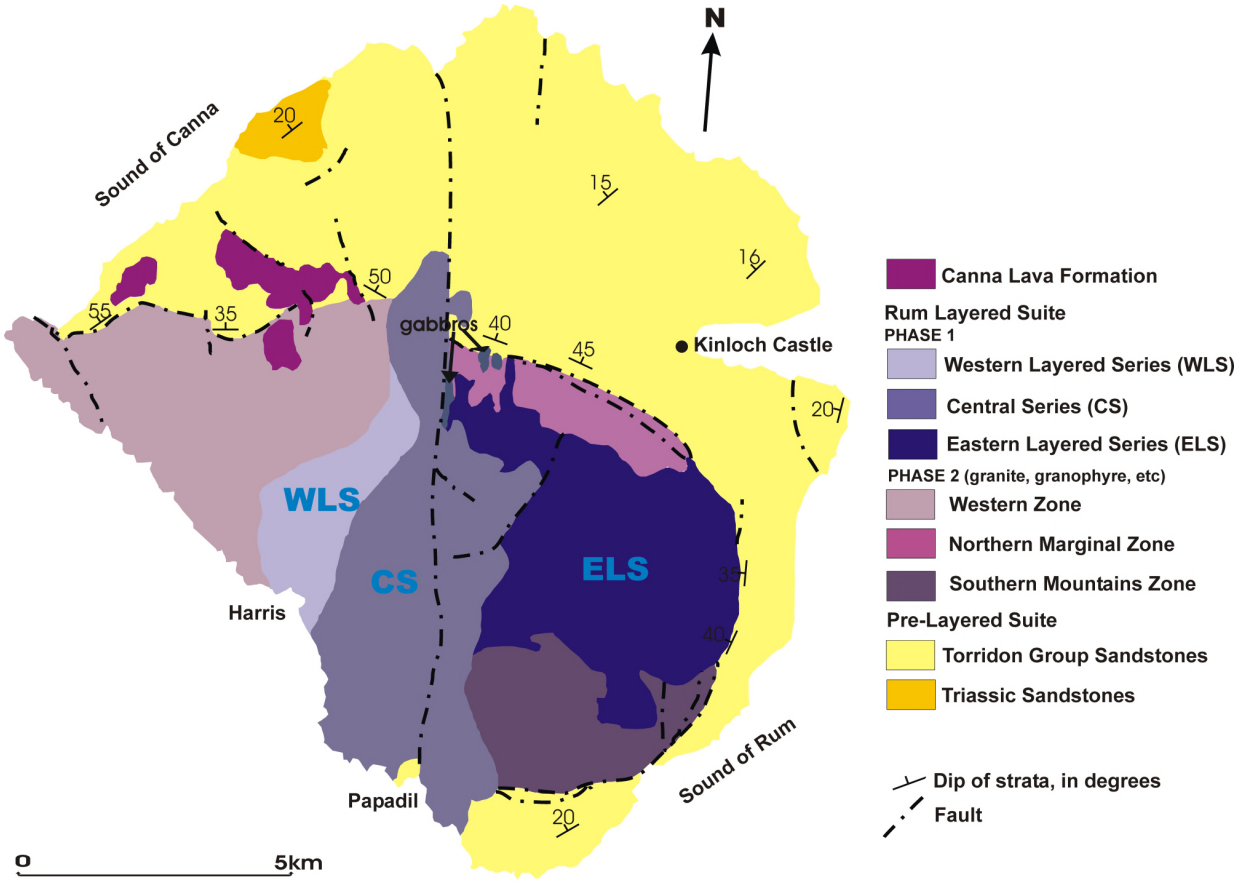


Figure 2.2 – Coarse geological map of Isle of Rum. Figure modified from Labonté (2004).

The Central Series postdate the Western and Eastern Series. While no clear genetic relationship of these layers has been established, the younger Central Series is interpreted by the author to represent a successively replenished and withdrawn magma chamber. Emeleus et al. (1996) independently interpreted the Central Series and reached at the same conclusion. They also point out their current view that the Eastern and Western Layered Series are lateral equivalents, with a slightly different structural aspect and somewhat older than the CS, (Emeleus et al., 1996). The CS is also referred to as the Central Layered Series (CLS) to retain naming consistency with the neighbouring layered series.

Although near the contact margins of the Layered Suites, the dip outwards are typically $\sim 20^\circ - 30^\circ$ to occasionally 47° (Emeleus et al., 1996), the dip at the centers are typically

much shallower, around $\sim 10^\circ - 15^\circ$ for the Eastern Layered Series; $\sim 5^\circ - 10^\circ$ for the Western Layered Series (Emeleus et al., 1996) and $\sim 7^\circ$ (Fowler, 2000) for the Central Layered Series where the stellate plagioclase outcrop.

Within the Eastern Layered Series, the exposed sequence is divided into sixteen macrorhythmic units, each broadly defined by a feldspathic peridotite base and an allivalitic (troctolitic) or rarely a gabbroic top, the thicknesses of which range from a few tens of meters to $\sim$ 100m (Emeleus et al., 1996). The proportion of peridotite to allivalite (troctolite) ranges from $\sim 95:5$ to $\sim 15:85$, decreasing from west to east (Volker and Upton, 1990). In thinly layered locations with small cm-scale layers, such as Unit 13, Hallival, anorthositic intraclasts can be seen, while in other locations with somewhat thicker layers, density-graded layering with distinct feldspar-rich tops are common (Emeleus et al., 1996).

The Western Layered Series contains the lower Harris Bay Member dominated by layered gabbro ($\sim$ 130m thick) with well developed olivine crescumulate textures ('harrisite')(Emeleus et al., 1996). Harrisite texture is named from this locality of Harris Bay. It is formed when the bottom of the magma chamber is significantly cooler than the injected melt and preferentially allows a fast rate of growth after initial nucleation in a direction mostly perpendicular to the direction of layering and parallel with local thermal gradient. It is seen in the lower layers of the Central and Western Layered Series as exposed at Harris Bay. There is no observed in situ layered gabbro in the Western Layered Series although detached blocks apparently derived from Western Layered Series occur within the Central Series (Emeleus et al., 1996).

The Central Series consists of large sheet like mega-blocks of layered allivalite (as named by Emeleus et al. (1996)), with some on the order of hm, surrounded either by heterolithic feldspathic peridotite breccias or relatively massive and homogeneous feldspathic peridotite (Emeleus et al., 1996). Numerous deformation structures such as loading, slumping and melanges and flame-structures were observed, in particular where less basic layers are overlain by more basic layers (Emeleus et al., 1996). In places where lithologies are contrasting, peridotite layers have been inferred to have intruded and become inter-folded with gabbro with very incomplete mixing (Renner and Palacz (1987), Volker and Upton (1990)). In the Central Series, peridotite can be seen fingering upward into adjacent layers perpendicular to layer contacts on the meter and submeter scales and independent of local dip (Butcher et al. (1985), Morse et al. (1987)) (see figure 2.3). Large pipes of peridotites, up to a meter in diameter and several meters in height were

described from gabbros on Trallval where they are discordant to gabbro layering (Volker (1983), Volker and Upton (1990)). Extensive cross cutting intrusive relationships are present throughout the Layered Suite including large scale gabbro and peridotite plugs and sheets (Emeleus, 1994) and smaller bodies of homogeneous peridotite commonly cross-cut, intrude, thermally erode and replace layered gabbro (Emeleus et al., 1996), see figure 2.4.



Figure 2.3 – Finger structures common within the Central Series, northwest of Long Loch. Fingers project upwards from peridotite into weakly-layered feldspathic peridotite. Hammer shaft length = 40cm. Image modified from Emeleus et al. (1996).

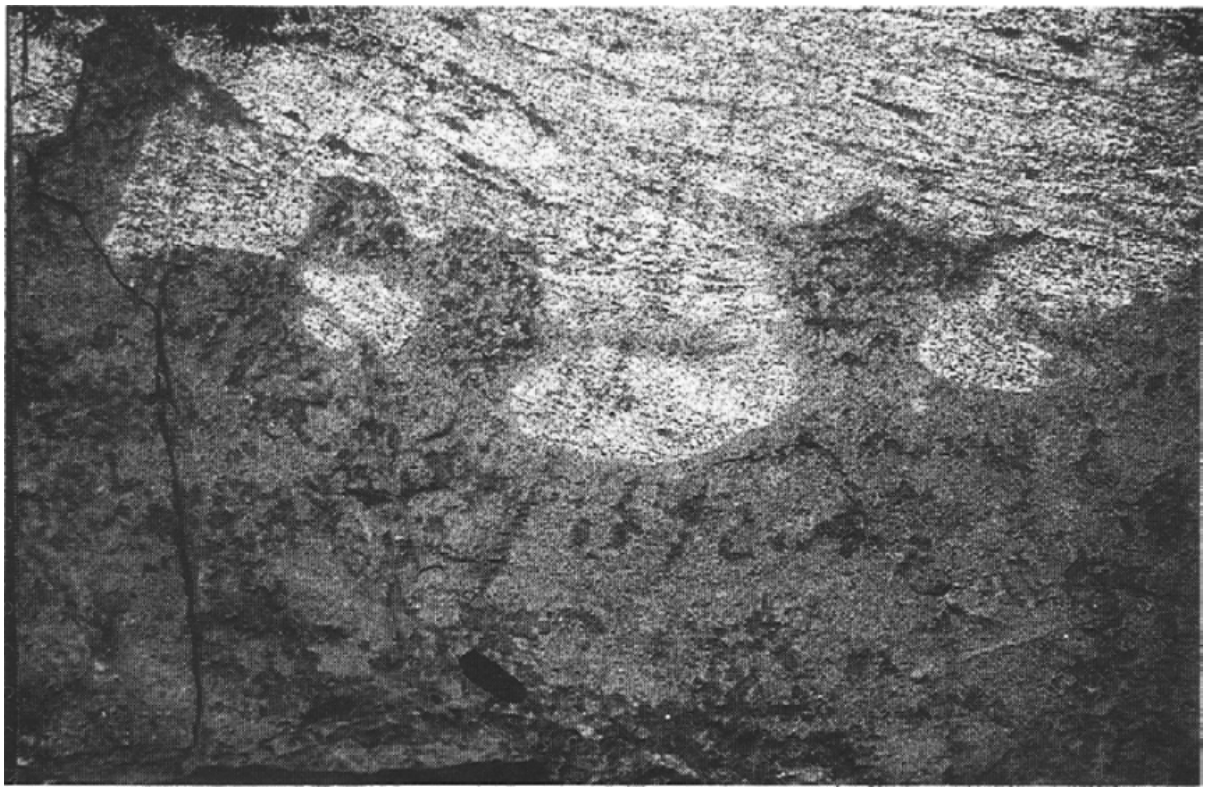


Figure 2.4 – Replacement structures seen in Unit 7 of Emeleus et al. (1996), Eastern Layered Series. Gabbro above has been replaced by irregular, lobate protrusions of peridotite from the top of the lower peridotite sheet. Lens cap at the base of photograph = 5 cm. Image modified from Emeleus et al. (1996).

2.2 Plagioclase Stellate Patterns

Within the Central Series, near the top of the exposed layers adjacent to Loch An-Dornabac (a small lake located at West-Central Rum) are outcroppings of layered peridotites and/or feldspathic peridotites/gabbro, (approximate Latitude: 56.993461°, Longitude: −6.352239°, for a rough idea of the sampled area, see figure 2.5.). The top surfaces of these layers have been subjected to extensive weathering producing a cm-scale relief (see figure 2.6) of dm to m scale rays of branching plagioclase (stellate plagioclase, or stellates) that were first described by Donaldson et al. (1973). A typical stellate is seen in figure 2.7. A typical anatomy of a radial plagioclase stellate is given in figure 2.8.

This macroscopic texture are made of networks of plagioclase rays that are radial, sub-radial to patchy in horizontal sections parallel to and on the top surface of individual layers (sub-radial see figures 2.9 and 2.14, radial texture see figure 2.10, patchy see figures 2.11, 2.12 and 2.13). On the vertically exposed sections, no radial symmetry can be seen (in fact, no visually discernible symmetries). There are also numerous plagioclase rays forming complex networks, but without any apparent centers or larger scale structure visible. Two such typical textures are shown in figures 2.15 and 2.16 as well as the exposed vertical surfaces of 2.17. However, the area and frequency of exposed vertical surfaces is much less than that of horizontal surfaces, and an in-depth investigation of the third dimension of the stellates on a large scale is nearly impossible.

In addition, plagioclase stellates seem to occur on the top surface of every observable layer (made accessible by erosion) with roughly the same branch diameters regardless of the layer thickness or layer order. In general, the stellates of one layer are not known to be linked to the stellates of an adjacent layer since layer parting surfaces separate easily. However, more detailed field studies would be needed to establish an exacting relationship between stellates of different layers. Stellates within the same layer are known to have shared branches and cluster like behaviour (see figure 2.10), long extended branches and intersections of branches with neighbouring stellates (see figures 2.18, 2.19). Some branches cross over, passing through or seemingly originating from nodular structures each containing mono-crystalline poikilitic clinopyroxene (see figure 2.6 and sub-section 2.3.1).

The stellate systems represented by individual occurrences can be roughly classified based on their radial distribution (symmetry); size of overall visible occurrence (size) and the degree of

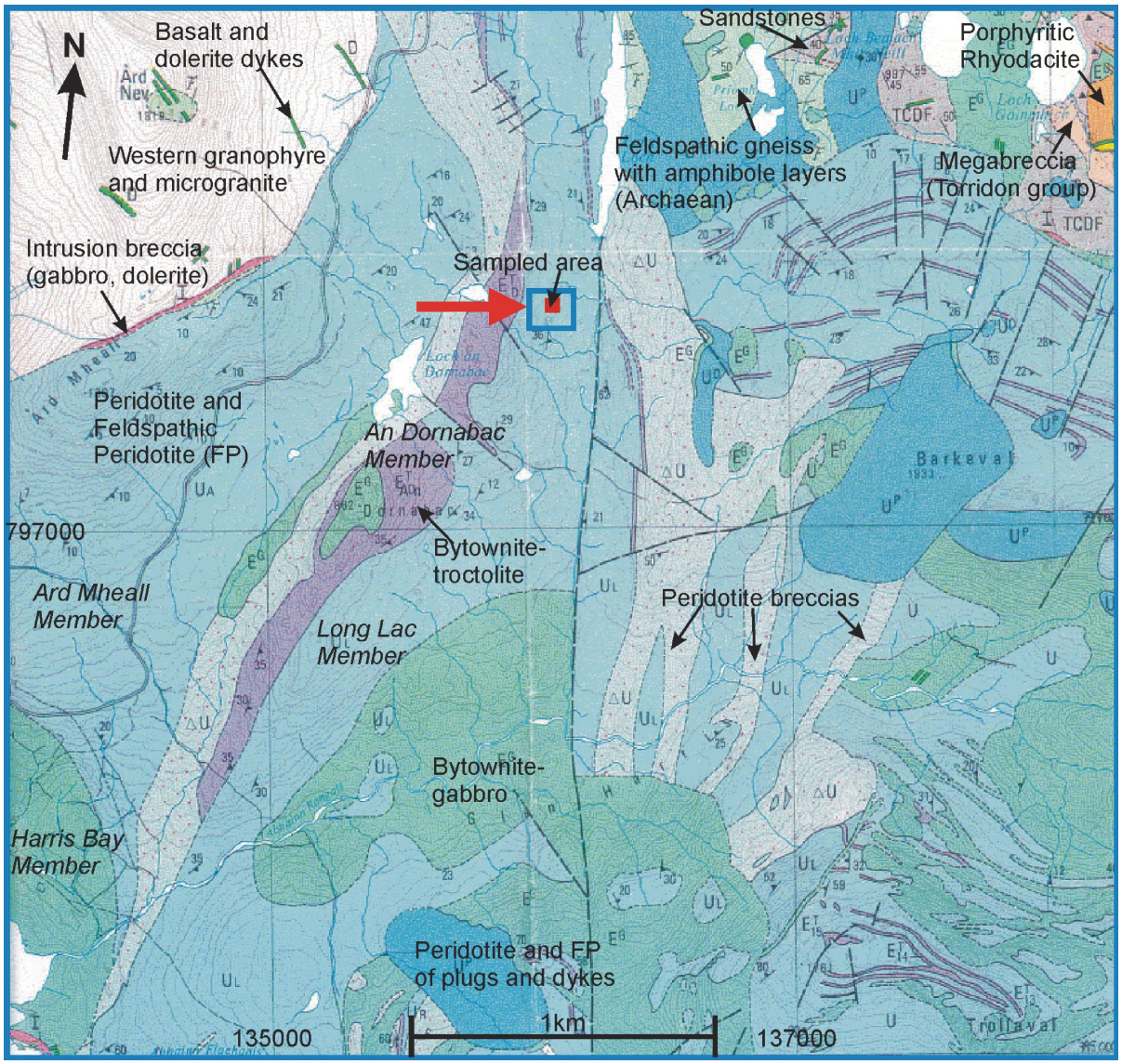


Figure 2.5 – Sampled area of Central Series of Isle of Rum. Figure modified from Labonté (2004), originally from regional map produced by Department of Geography and Science Topography, University of Glasgow & Scottish Natural Heritage.

branching (complexity). Symmetry in our context shall refer to a range of stellate organizations varying from high symmetry as exemplified by the near-radial distribution of stellates seen in figure 2.10 to the asymmetrical radial distribution of stellates seen in figure 2.9 to the seemingly dis-organized, patchy stellates with no apparent symmetry (e.g. figure 2.11). The size of the stellates is evaluated by the equivalent two-dimensional, circular area with a radius as defined by the stellates rays, often equivalent to the aerial extent of a stellate occurrence for near-radial systems. The complexity of a stellate system is inferred from the number of daughter branches

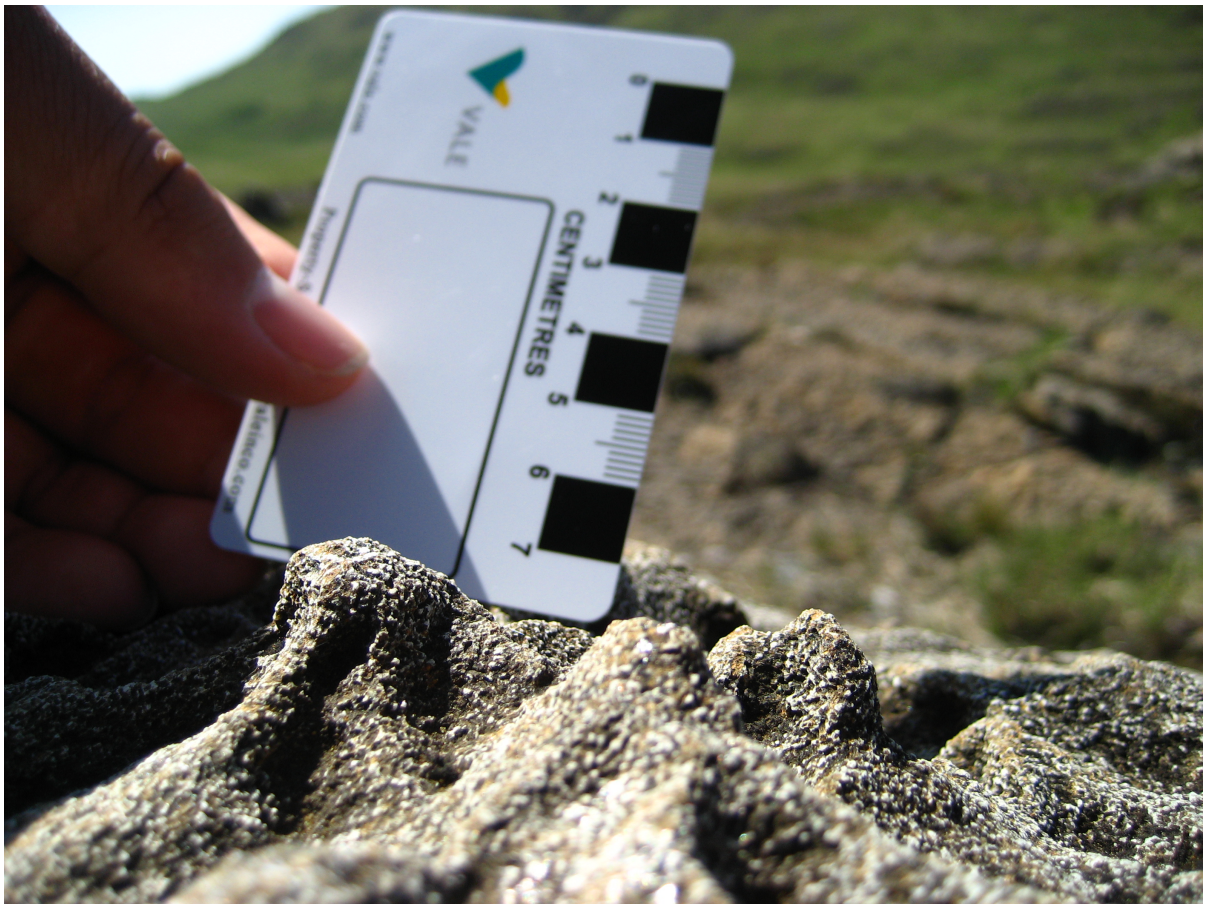


Figure 2.6 – Nodular structures in the stellates, and typical vertical relief due to weathering.

originating from a principle branch. Larger systems have the possibility of many daughter branches from a single parent branch.

While the classification scheme itself may seem somewhat obvious, its application is surely not. In general, there are occasional connecting arms of stellates to other stellates; and a very complex, almost (but not exactly) fractal-like type of branching along individual arms. Some of these systems are on the order of meters or possibly more, limited only by our ability to accurately gauge the physical constraints of such a system, see figure 2.18 and the upper right stellate clusters of figure 2.10. It is far less obvious where some systems terminate than where they apparently originate. For non-symmetric systems, such as patchy textures or the systems visible from the sides of the exposed layers, any determination would be impossible without physically dissecting the layers. This presents a two fold difficulty. Firstly, the system is not uniformly weathered and/or exposed and/or intact in many of its assumed extremities such as figure 2.11, making the tracing of stellates particularly difficult past certain sections.

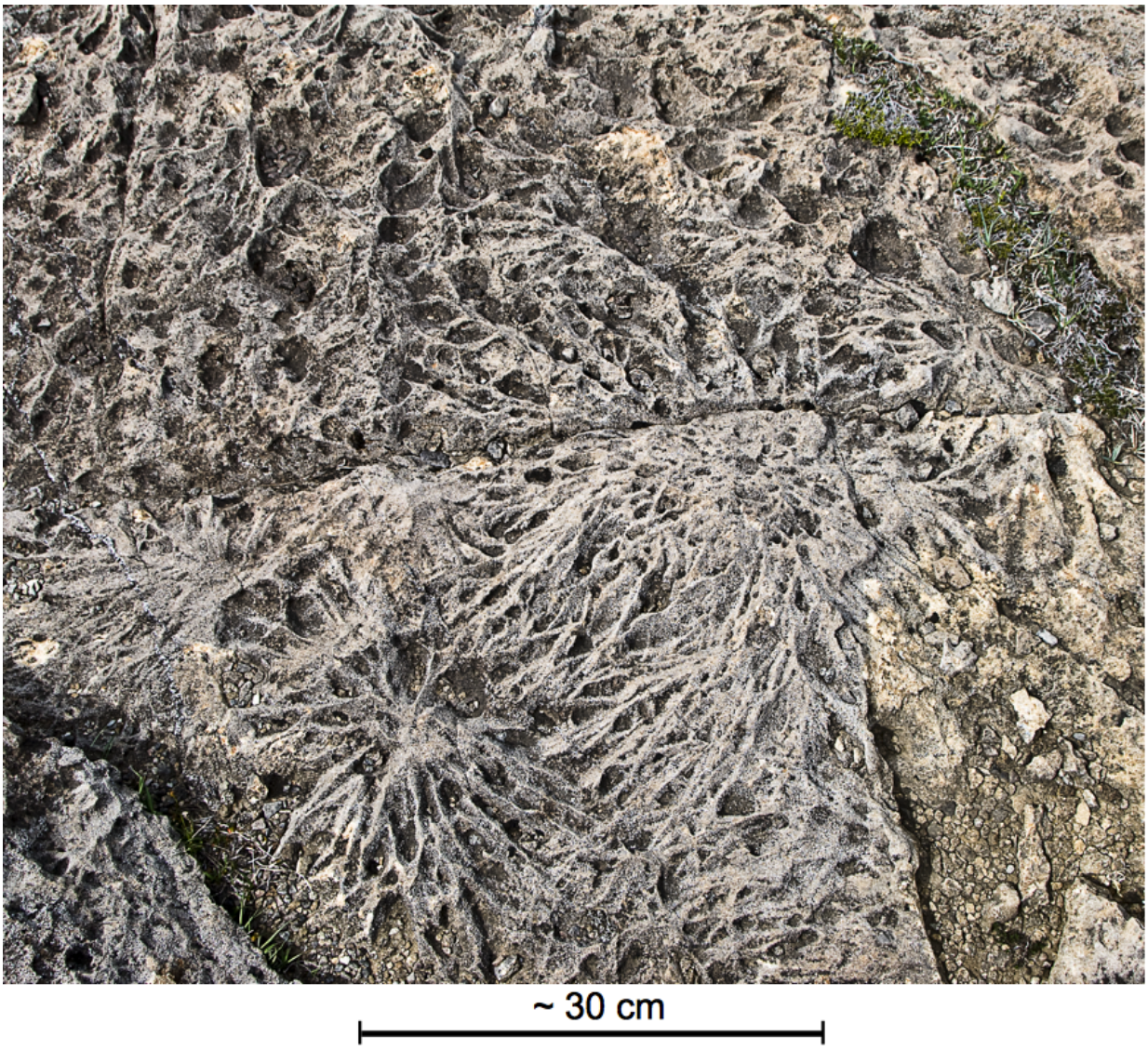


Figure 2.7 – Typical picture of plagioclase stellates. Two distinct stellates are seen in figure, one bottom left and another bigger one middle right.

Secondly, the stellates at the extremities are often found to connect to other systems such as figure 2.19, or overlap other systems such as figure 2.10, making the determination of the exact end points difficult. It is because of these constraints that when speaking of system "size" that we shall adopt the meaning: the physical extent of the system excluding the connections, where connections means relatively few branches of plagioclase that seemingly originate from more than one system (apparently simultaneously) in 2D. In addition, where the system is radial in most directions, the "size" shall mean the radial extent assuming near-radial symmetry, and excluding the connections where possible.

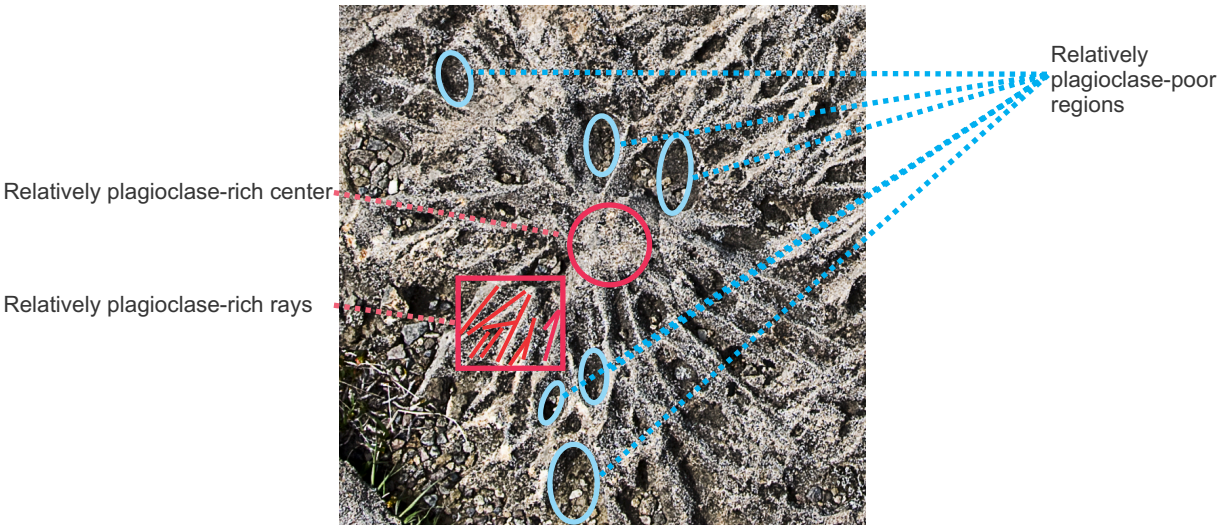


Figure 2.8 – Anatomy of typical plagioclase stellates. Plagioclase stellate taken from figure 2.7.



Figure 2.9 – Sub-radial texture. Small system size and medium complexity. Scale card is $\sim$ 8cm long.

Based on the above scheme of characterization, medium sized occurrences (sub-meter scale) tend to be high in symmetry, medium in complexity. Small sized occurrences (dm to sub-



Figure 2.10 – Medium sized radiating texture. Cross cutting stellates near scale card, overlapping stellates further right. Medium to low complexity. Scale card is $\sim$ 8cm long.

meter scale) are usually low in symmetry to no apparent symmetry and low complexity. Large occurrences (meter to possibly decametre scale) can exhibit high to low symmetry, and high complexity, such as the system(s) in figure 2.18. Some large occurrences cross cut smaller adjacent stellates or sometimes have very extended branches reaching several meters in size or more and can apparently connect multiple stellates, e.g. figure 2.19.

The variation of the size and symmetry of the stellates superficially appear to correlate with the West-East direction along the series of outcrops. However, this could be an apparent variation not intrinsic to the stellates due to the preferential weathering of higher ground exposed layers versus the lower ground less well exposed layers.

Although the igneous layered rocks in the Central Series containing these macroscopic textures are bulk mapped and understood to be peridotites (e.g. Labonté (2004)), the textures seem to occur in aerial patches much larger than the individual grain sizes of olivine and with a frequency that can be entirely dominating in large swaths (in some areas depending on scale



Figure 2.11 – Patchy stellate. No global symmetry and low complexity. Size not inferred due to rock fragmentation. Shoe for scale. Shoe size is 12 per US sizing ($\sim$ 10cm width-wise).

of observation, the rock appears to be nearly-anorthositic). In fact, in Labonté (2004), the maximum difference of modal plagioclase composition in 15 sequentially produced thin-sections from a piece of stellate bearing rock measuring $\sim$ 0.5m long was 5.2%. Thus statistically valid sampling of bulk rock studies would require large numbers of samples for each layer, spatially homogeneous in such a way as to remove the large variation in feldspar content in final modal analysis and pin-point a proper igneous name. Even though the area is well mapped and studied by some (e.g. Emeleus et al. (1996), Labonté (2004), Fowler (2000)), such methodology was not used to analyze the bulk rock modality, if only because such a process would be unfeasible, see section 2.3. However, it is not necessary to invoke any particular naming schemes for the study of plagioclase stellates since their patterns are apparently observable on any well-weathered surface of any igneous layers of the study area in the Central Series to the best of our determination and do not appear to correlate with bulk rock composition for the observable instances of well weathered surfaces. For example, in 2.17, the lighter top layer and the more basic lower layer both clearly exhibit weakly-organized vertical textures associated with planar plagioclase



Figure 2.12 – Patchy stellate. No global symmetry and low complexity. Size not inferred due to rock fragmentation. Camera in pouch for scale. Camera pouch size is $\sim$ 10cm.

stellates when viewed from the side.



Figure 2.13 – Patchy stellate. No global symmetry and low complexity. Medium size, although overall size difficult to estimate due to rock fragmentation. Scale card is $\sim$ 8cm long.



Figure 2.14 – Sub-radial to patchy stellate. Low symmetry and low complexity. Small size. Scale card is $\sim$ 8cm long.

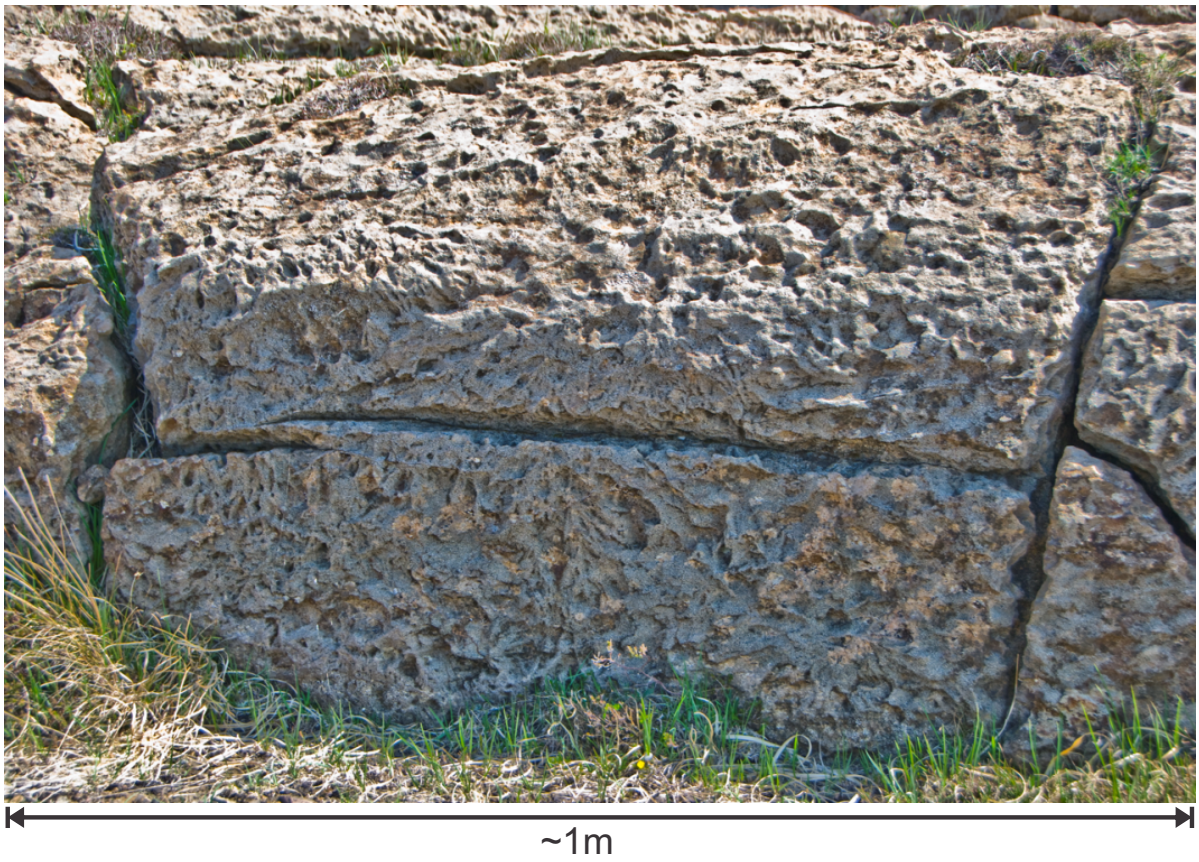


Figure 2.15 – Typical side texture, no overall organization or symmetry. Low complexity.

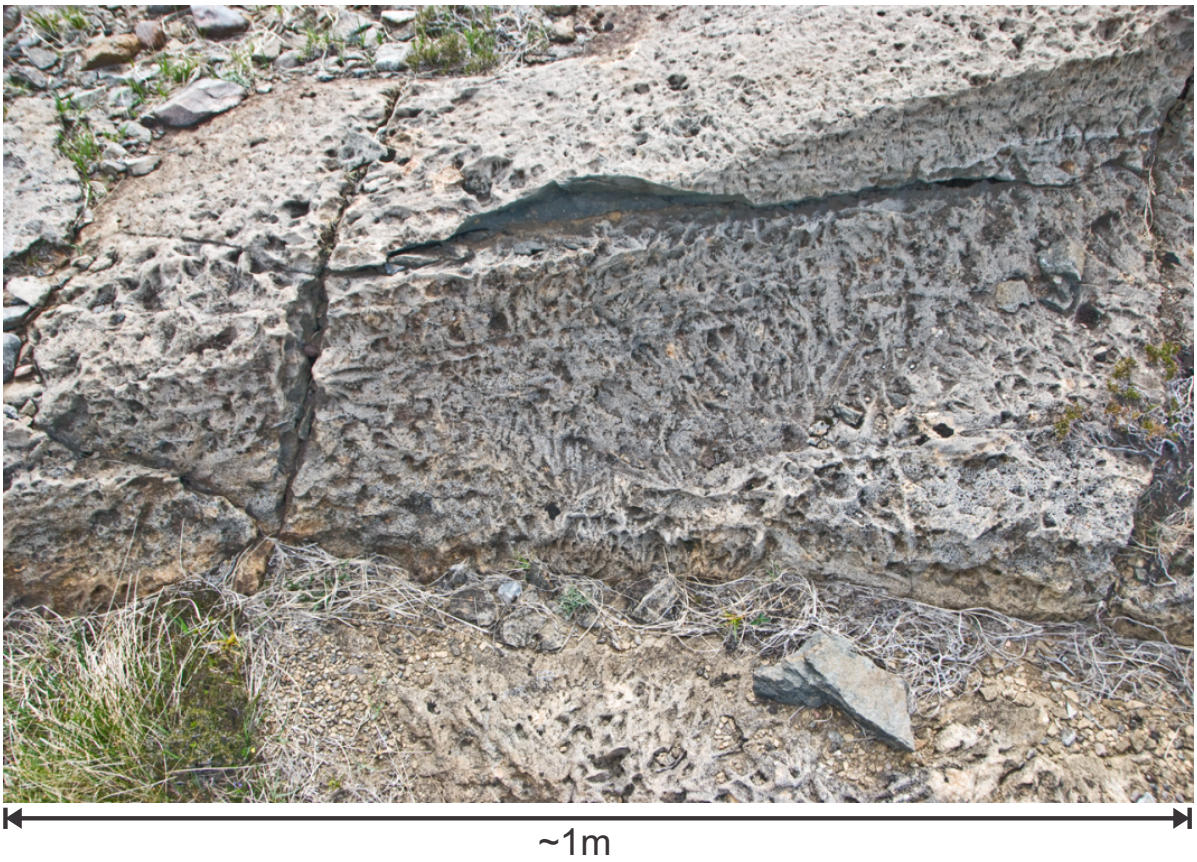


Figure 2.16 – Typical side texture, no discernable overall organization or symmetry. Low complexity.



Figure 2.17 – Two different layers both exhibiting plagioclase networks viewable from the side. The measurement of the exact length of the goat was not attempted (probably $\sim$ 1m in length). No goats were harmed.



Figure 2.18 – Large radiating texture. High complexity. Exact system size probably bigger than FOV. Scale card ∼8 cm long.

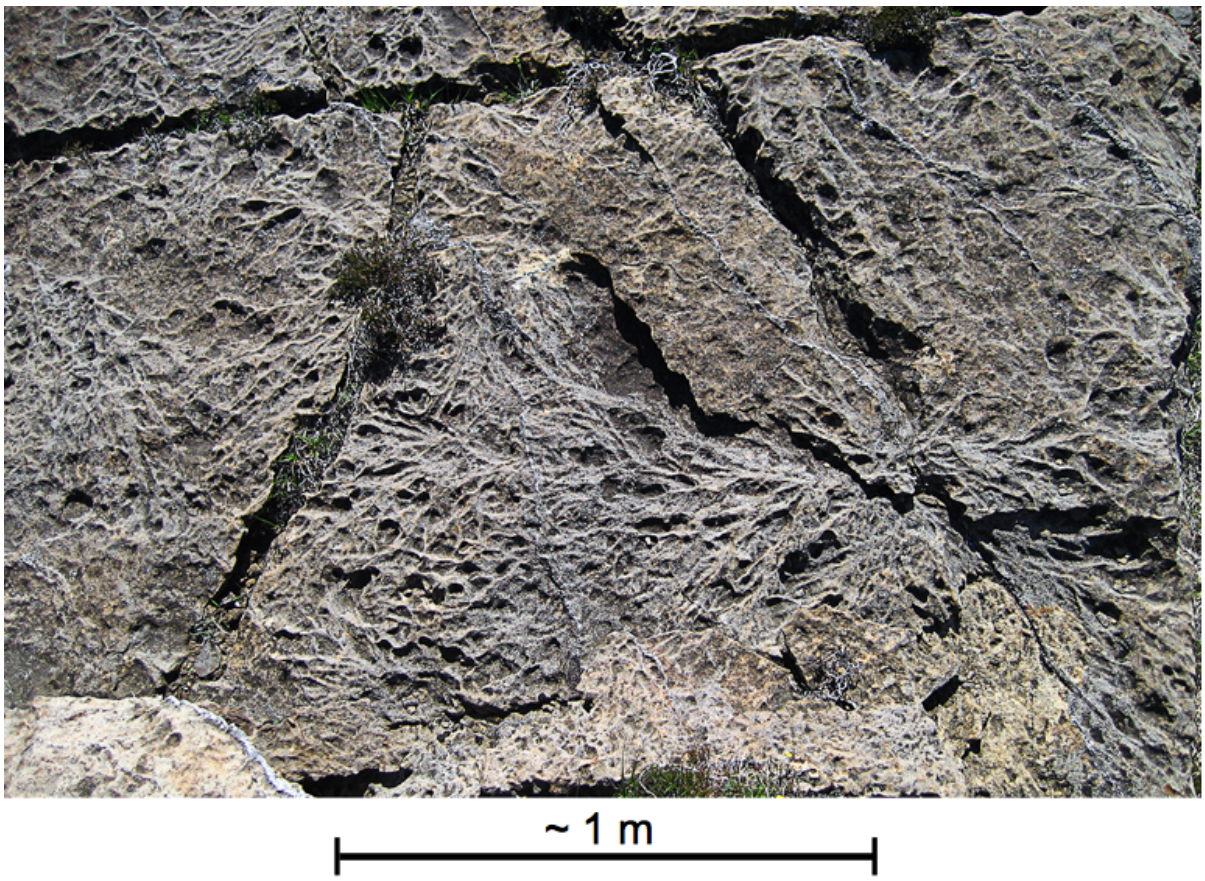


Figure 2.19 – Connections between individual stellate systems. Cluster on the right has connections to the clusters on the left. It is not discernible how many exact stellate clusters there are in total within the image.

2.3 Experimental Methods

Since the Isle of Rum is part of the Small Isles National Scenic Area; a Special Protection Area for Birds; a Site of Special Scientific Interest and a Special Area of Conservation, geological sampling requires permits and carries strict limits. As such in 2000 and 2002, the area was observed, mapped and sampled during the field work of Fowler (2000) and Labonté (2004) over a period of several days each. The numerous specimens ranging from 0.1 to several kilograms in mass were all from exposed outcrops of the layered intrusions of the Central Layered Series in the vicinity of Loch An-Dornabac. A wide range of acquisition techniques were employed including cm-scale drill cores and hammers. In 2008, another observation-oriented field work conducted by the author and assistant Julie Bourdeau yielded a multitude of high resolution images and a few additional samples.

In the studies of 2000 and 2002, one main focus was to characterize, classify and map the rocks. An additional task was the attempt to understand the relationship between preferred grain orientations of olivine to the macroscopic stellate ray orientations. Most of the samples were acquired from the top exposed surfaces of the Central Series near Loch An-Dornabac, with a sample area covering that of later visits of Zhang and Bourdeau for the preparation of this thesis, see figure 2.5. The studies in 2008 by Zhang and Bourdeau focused on the plagioclase stellates where as the earlier studies were more general. There was an emphasis on photo-documentation of the outcrops and investigating some basic field relationships such as the relative occurrences of stellates and types of stellates. This study was conducted without a geological sampling permit, and such, only loose segments were collected. Following the study in 2008, the photos were reviewed and a classification scheme of the stellates based on size, symmetry and complexity was proposed. Additional electron microprobe (EMP) work and thin-section microscopy were carried out to confirm some previous mineralogical findings.

Following the 2000 to 2002 field studies, several standard analytical procedures were carried out on a diversity of samples including thin-section optical microscopy, EMP, back-scattered electron imaging (BSE), crystal size distributions analysis (CSD) and electron backscatter diffraction analysis (EBSD). An interesting aspect of the laboratory work during the researches of Labonté (2004) was the serial sectioning of a large (~ 0.5 m in the longest direction) planar sheet of plagioclase stellate bearing rock (with a patchy texture and somewhat sparse distribu-

tion of plagioclase stellate rays) as seen in figure 2.20. Due to the serial sectioning technique employed, most of the stellate rays were actually cut in a plane orthogonal to the ray propagation plane and therefore, less than an ideal amount of plagioclase stellate behaviours were observed. The average modal composition of the gabbro resulting from 15 thin sections produced from the serial sectioning process is 65.6% olivine, 14.8% pyroxenes, 14.6% plagioclase, 2.3% iddingsite, 1.1% chromite-spinel, 1.0% serpentine and/or chlorite and 0.5% phlogopite. The highest observed plagioclase content is 16.9%. Since the plagioclase rays in the rock used by Labonté is not dense compared to some locations (e.g. middle to right portion of figure 2.10 or center of figure 2.18), the modal concentration of plagioclase may be much higher in other stellate bearing rocks. In addition, with the analysis of thin-sections resulting from the serial sectioning technique, it is affirmed that the olivine grains are clustered and aligned in shape and crystallographic preferred orientations (SPO and CPO) inside the layers with a definite foliation plane parallel to the layer contacts or layer surfaces, see figure 2.36.

Although the studies of 2000 and 2002 found no clear or conclusive relationship of the alignment of olivine to plagioclase stellates, numerous geochemical results did establish the most probable (mode) An content of the plagioclase to be An70 - An80 over a wide range of exposed layers with lesser amounts of lower An content, ∼An50 in the vicinity of Loch An-Dornabac, see Labonté (2004). The distribution of the An content of plagioclase in sampled rocks was determined to be highly non-uniform over the entire An content interval. At the mode An content ∼An75, the distribution contains a relatively clustered set of data points with no samples possessing An content above ∼An80. Between about $\mathrm{An}(70 - 55)$ is a set of fairly sparse, and much more uniform set of samples. The frequency of high An content plagioclase is also much higher than that of the relatively low An content plagioclase in sampled rocks. In contrast to the plagioclase, the compositional variability of olivine and clinopyroxene are much less diverse, see figure 2.21.

There were some very sparse and faint (both under microscopy and EMP) zones of the plagioclase in the samples acquired, see figure 2.22. The zoning appears to be oscillatory.

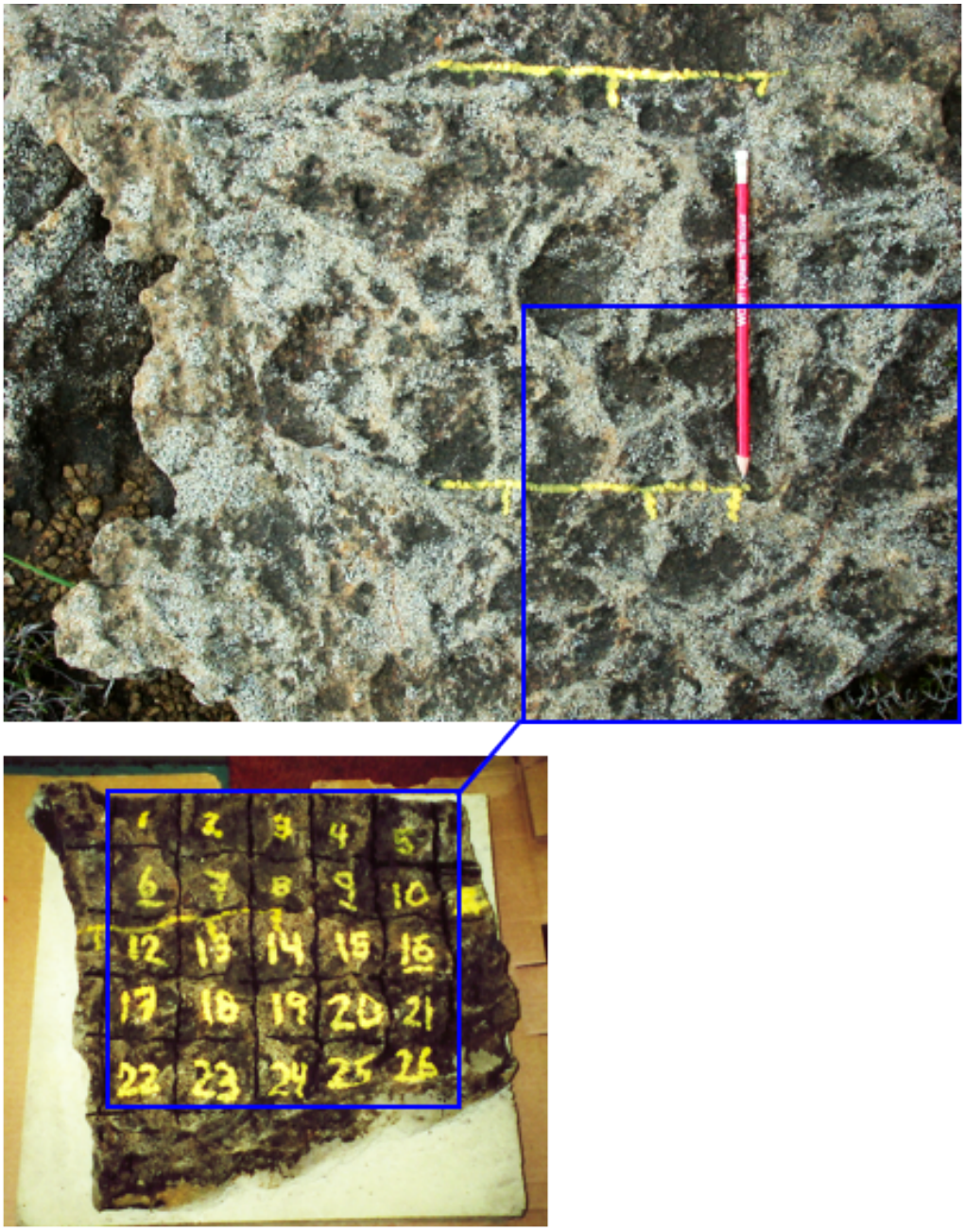
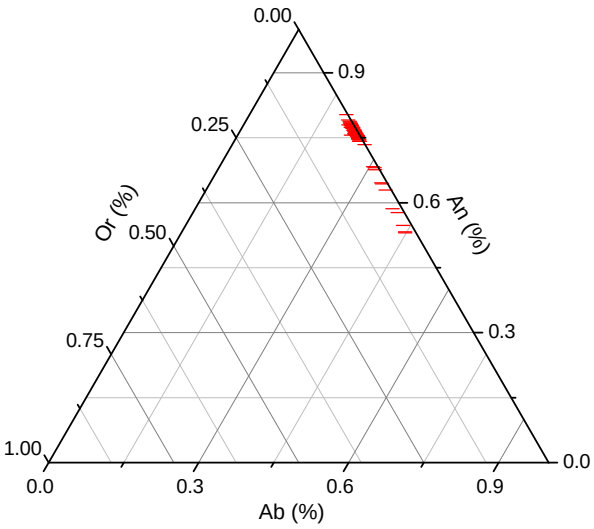
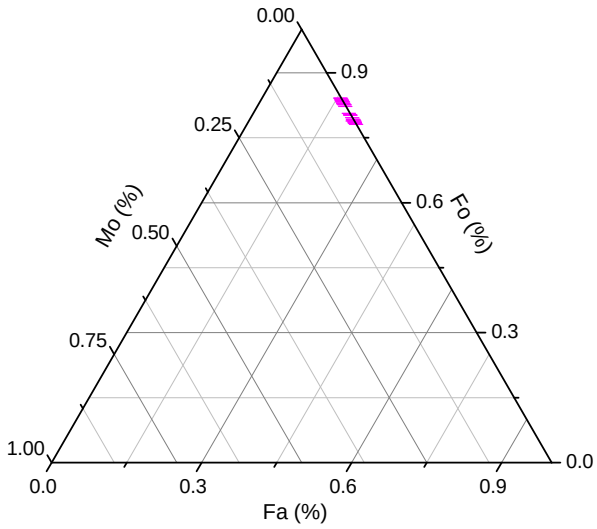


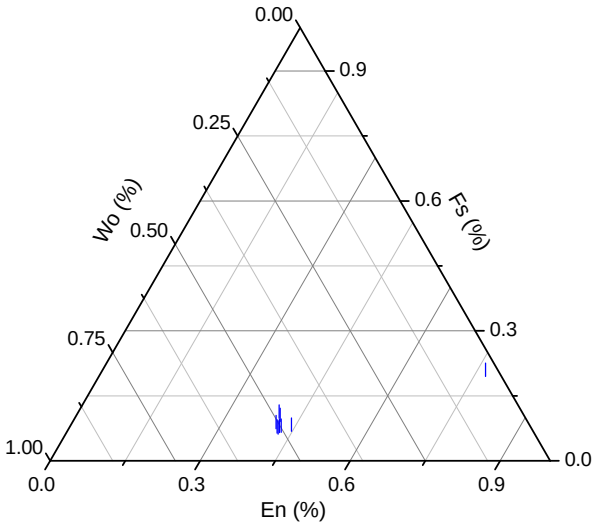
Figure 2.20 – Rock section utilized by Labonté (2004) and subsequent serial-sectioning. Figure modified from Labonté (2004).



(a) End-member composition of plagioclase.



(b) End-member composition of olivine.



(c) End-member composition of clinopyroxene.

Figure 2.21 – Ternary diagrams of plagioclase, olivine and clinopyroxenes compositions. Data from (Labonté, 2004).

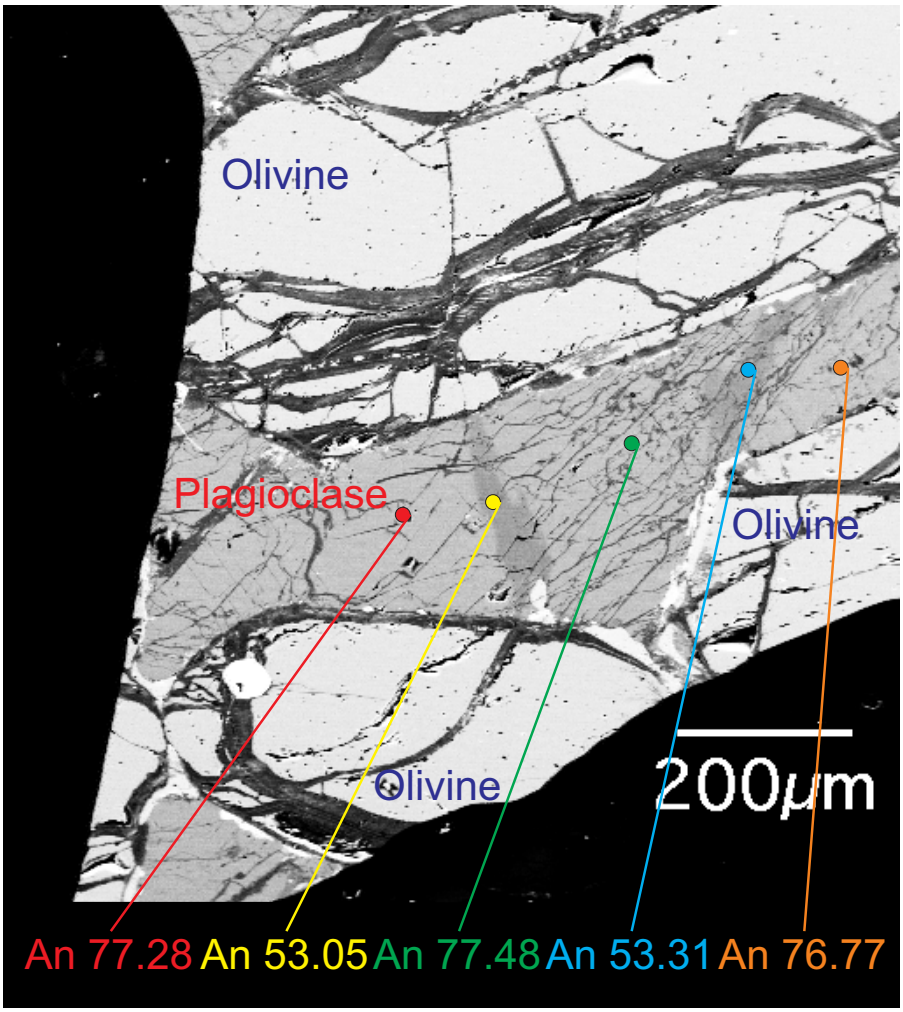


Figure 2.22 – EMP backscatter image of plagioclase and olivine showing oscillatory zoning in the centre grain of plagioclase. Figure modified from (Labonté, 2004).

2.3.1 Microscopy

Without exception, all the thin-sections were prepared from samples taken from the Central Series. The set of thin-sections used in the analyses consists of the all the available thin-sections, including the thin-sections of Labonté (2004), Fowler's collection, and from the author's collection. Based on field observations, it is expected that the mineral compositions of thin-sections will vary depending on the source/site of actual sampling (i.e., whether the source was a stellate branch, or in between stellate branches, or a nodular structure).

Microscopy confirms the above expectations in that the thin-sections can be classified modally: principally olivine and plagioclase, or clinopyroxene. The modal composition correlates with spatial heterogeneities of the rock sampled; i.e., samples taken from the branches of the stellates typically show an olivine and plagioclase-rich composition while samples from the interstitial regions are typically clinopyroxene-rich with some olivine (see figure 2.23), samples taken from the nodular structures (see figure 2.6) display a relatively clinopyroxene-rich composition.

In olivine and plagioclase-dominated thin-sections, the olivine is typically found to be euhedral to subhedral and preferentially elongated (prismatic). Occasionally the olivine is found to be embayed, see figure 2.30. The olivine is typically found to be aligned parallel with the plane of igneous layering as expected from compaction. The plagioclase is typically aligned with the *c*-axis in the local direction of stellate propagation, inferred by twin lamellae and EBSD pole identifications. The plagioclase is subhedral to anhedral and encapsulates the olivine and other minerals. The typical size of the olivine crystals is ∼mm. This dwarfs in comparison to the large linear dimensions of the plagioclase crystals, which in many thin-sections appear to extend beyond the confines of the thin-section's length. A rough estimate on the overall length of the plagioclase crystals by comparison with the olivine crystals and the overall length of the thin-section indicates that the plagioclase crystals may be several orders of magnitude longer than the olivine crystals, as expected from field observations. A few perfectly euhedral pieces of plagioclase can be seen in some thin-sections, as exceptions. Other studies of the bulk properties of minerals in the Layered Suite such as Emeleus et al. (1996) concluded that for the series accessible by constant erosion, the olivines are largely unzoned and where encapsulated by plagioclase are less magnesian than cumulate olivines. There is also evidence for post-cumulate interactions in the form of preferential chemical influences from above and

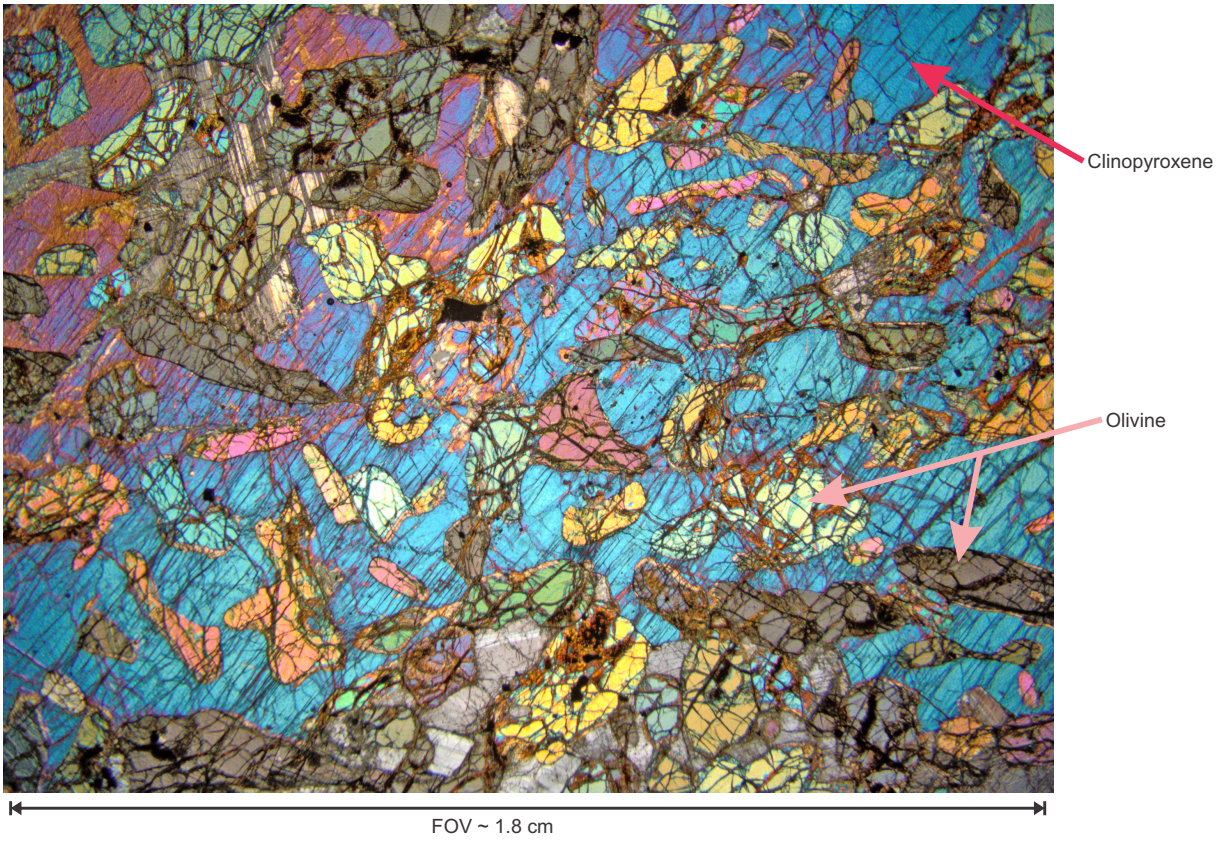


Figure 2.23 – Typical clinopyroxene rich thin-section in crossed-polarized light. Thin-section made from interstitial regions between stellate rays.

below layers in the regions close to layer contacts, see Emeleus et al. (1996). In addition, a good correlation exists between the magnesium content of clinopyroxene and olivine which seems to suggest post-cumulus interactions and chemical re-homogenization (Emeleus et al., 1996).

Typical plagioclase textures in stellate bearing samples are given in figures 2.24 and 2.26. The twin lamellae are explicitly illustrated as an example for figure 2.24 in figure 2.25. In these two thin-sections, we can clearly see twinning of plagioclase crystals, mostly albite twins and possibly some Carlsbad twins. Lamellae which run perpendicular to the albite twins appear to be pericline twins. Some stress-induced slips are shown in figure 2.27 with the slip boundaries explicitly illustrated in figure 2.28. It is possible that some of the twinning observed are due to thermal and/or mechanical stresses either during growth or subsequent periods.

In general, there are few microscopically observable zoned plagioclase crystals in the sample set. Zoned plagioclase crystals tend to be small and anhedral. These smaller zoned plagioclase crystals are typically not crystallographically-linked (as shown by extinction positions) to other

pieces of similar plagioclase crystals, and occasionally seem to link to larger crystals without zoning. Larger crystals of plagioclase and euhedral crystals do not typically show zoning under microscopy. Even under EMP (of a few probed samples), chemical zoning is not generally observed, except where they are physically attached, or connected to a smaller zoned plagioclase. In other studies, such as McClurg (1982) and Volker (1983), extensive zonings of both complex and reverse types were documented for plagioclase in the Eastern Layered Series (which has been the focal point of studies of the Rum Layered Series for the last few decades, see Emeleus et al. (1996)). In their study, the plagioclase core compositions range from An90-An79 with some zones down to An50. No other detailed studies of plagioclase zones are available as summarized by (Emeleus et al., 1996).

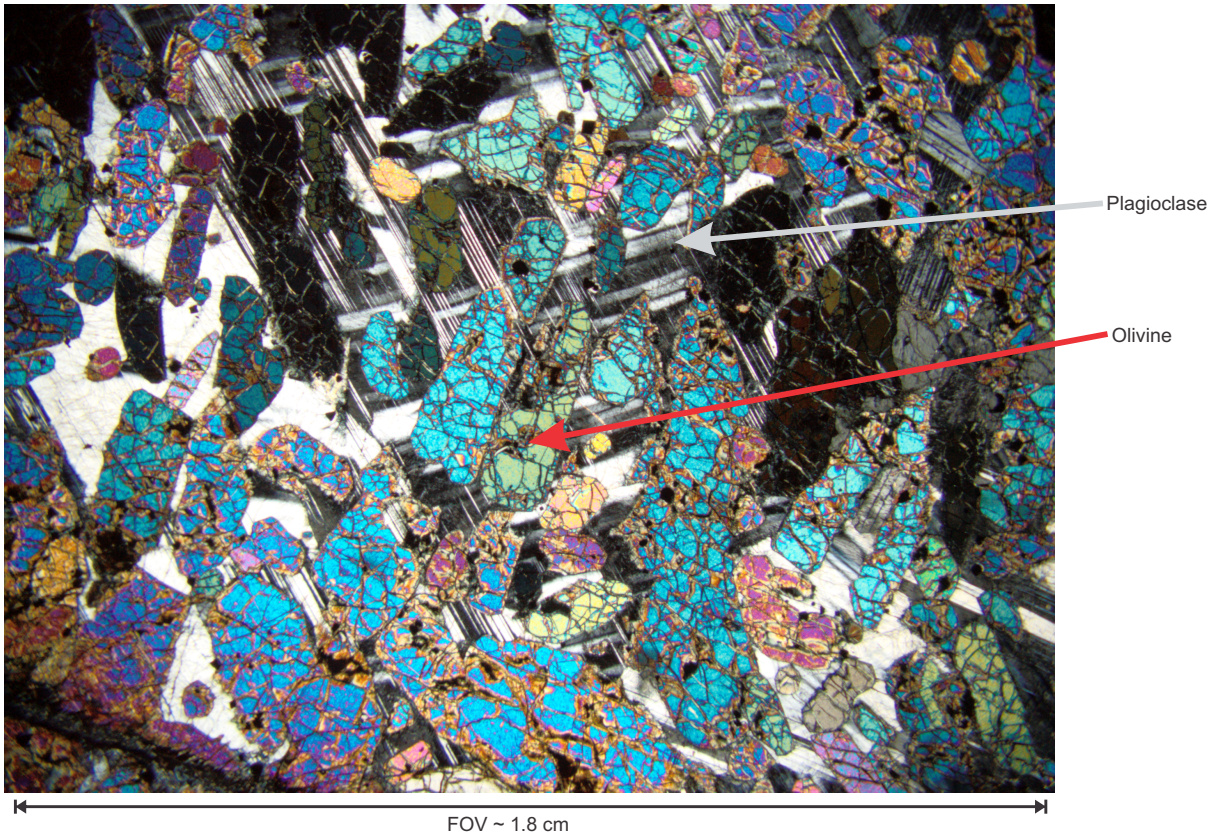


Figure 2.24 – Typical stellate bearing thin-section. Sample from Labonté (2004), section T15. Polarized light microscopy analysis and interpretations by author. Notice the twinned plagioclase crystal(s) occurring interstitially to the olivine. The thicker lamellae which run nearly perpendicular to albite twin lamellae appear to be pericline twins.

Many of the thin-sections with long sweeping-extinction plagioclase crystals show a large variety of twinning with some cm-long thin-sections containing only 2 crystals oriented length-

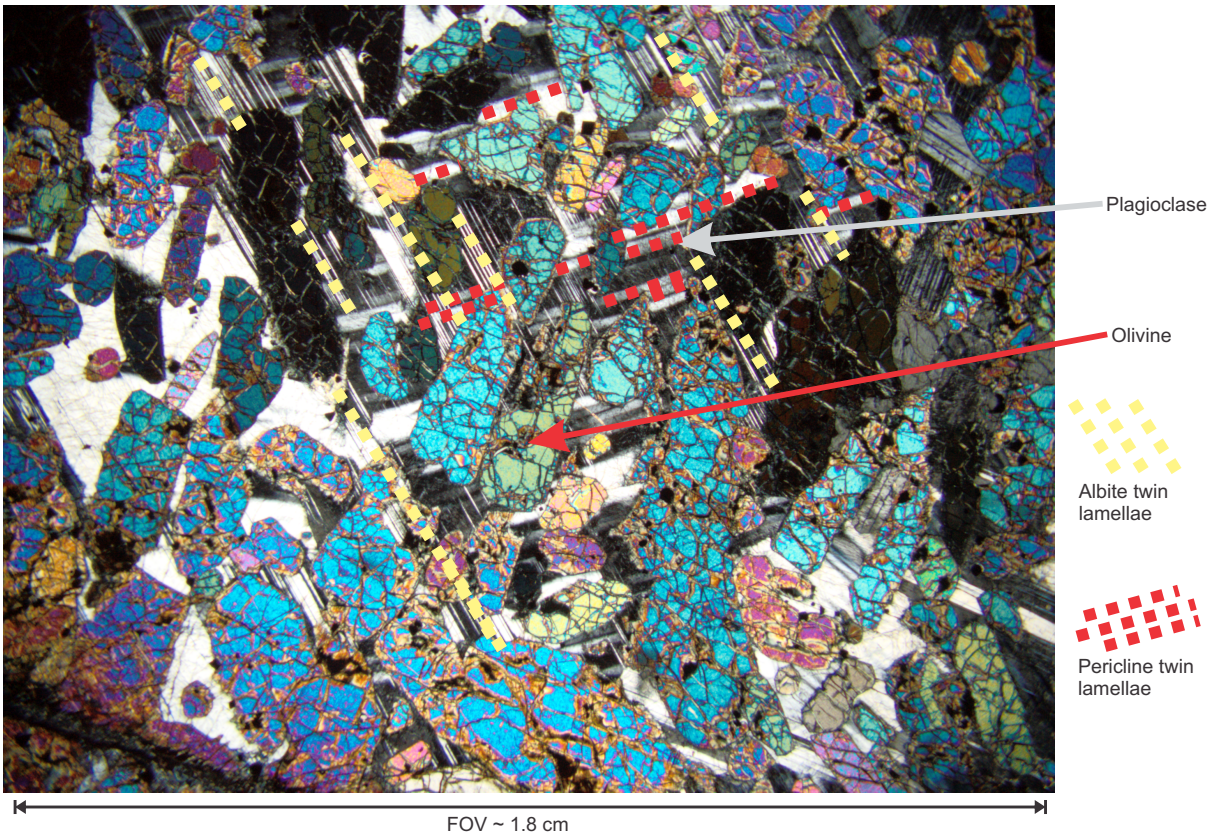


Figure 2.25 – Demonstration of plagioclase zoning in figure 2.24. Most prominent albite twin lamellae are overlain by yellow dotted lines while for pericline twin lamellae, red dotted lines are used.

wise twinned according to the Carlsbad rule (with smaller lamellae of albite twins, and possibly other twin types intra-lamellae). Additionally, in apparently polycrystalline-plagioclase thin-sections, it is frequently found that the crystals of plagioclase exhibit sweeping or unison extinction traceable across some thin-sections length-wise. This implies crystal to crystal structural connectivity, in three dimensions, beyond the confines of the relatively two dimensional thin-section.

Chromite is typically seen as isolated crystals and occasionally embayed, see figure 2.30, and rarely as thin layers (∼mm scale) that separate (generally) olivine-rich and feldspar-rich areas. In thin-section T19 (figure 2.29), a chromite band separates a plagioclase-rich layer and a olivine-rich layer. The thickness of the chromite band is mm-scaled. The physical location of the sample which produced the thin-section T19 is approximately 50 meters south of the most abundant outcroppings of plagioclase stellates within the sampled area in the CLS.

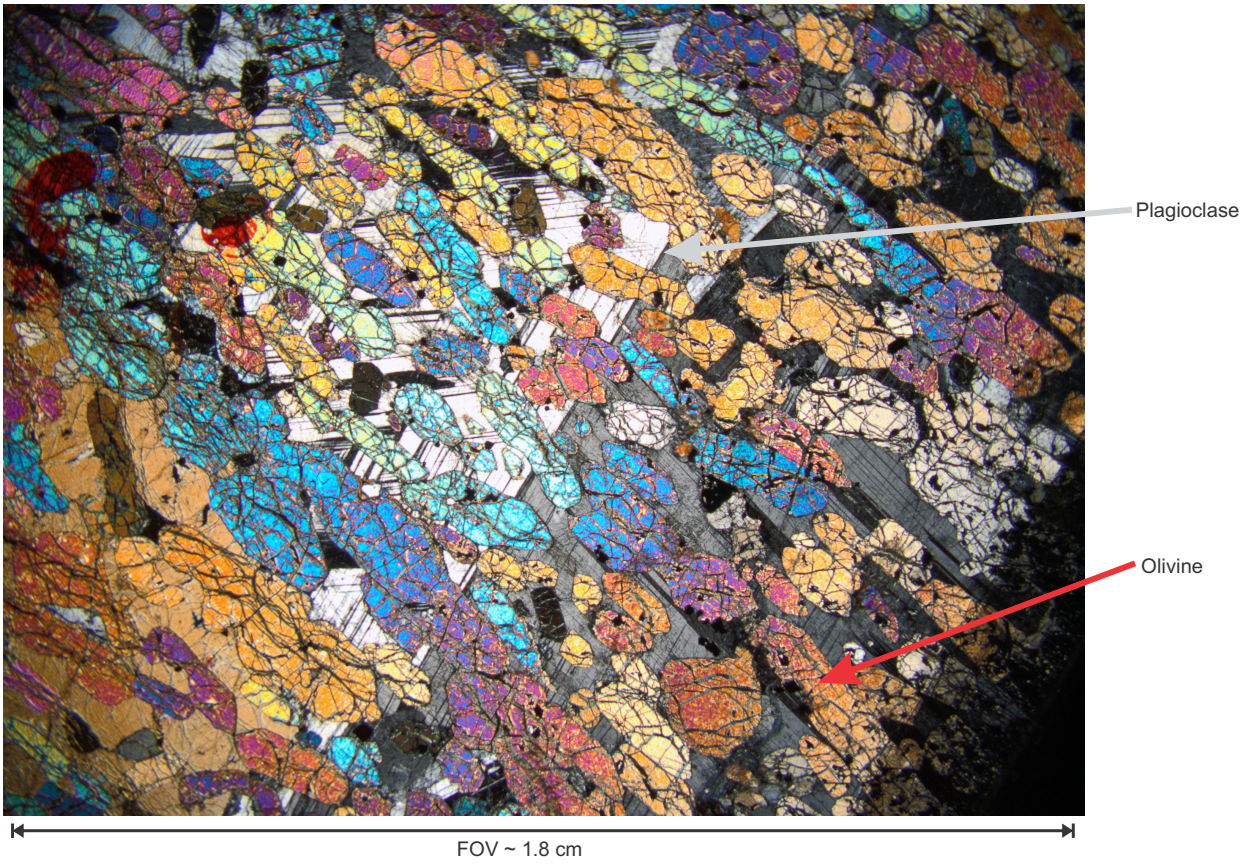


Figure 2.26 – Typical stellate bearing thin-section. Sample from Labonté (2004), section T15. Polarized light microscopy analysis and interpretations by author. The plagioclase interstitially to the olivine in the thin-section is twinned with a combination of albite and Carlsbad, possibly complex albite-Carlsbad and pericline. One long twin plane is visible in the center of the section, marked by the sharp transition of not-in-extinction plagioclase (white, mostly left of the figure) and partial-extinction plagioclase (grey, mostly right of the figure). This twin plane possibly defines a Baveno twin law, as there are numerous lamellae of albite twins at an oblique angle to this plane. Some of the acute angles between lamellae in the upper left twin member of the suspected Baveno twin are possibly prismatic twins.

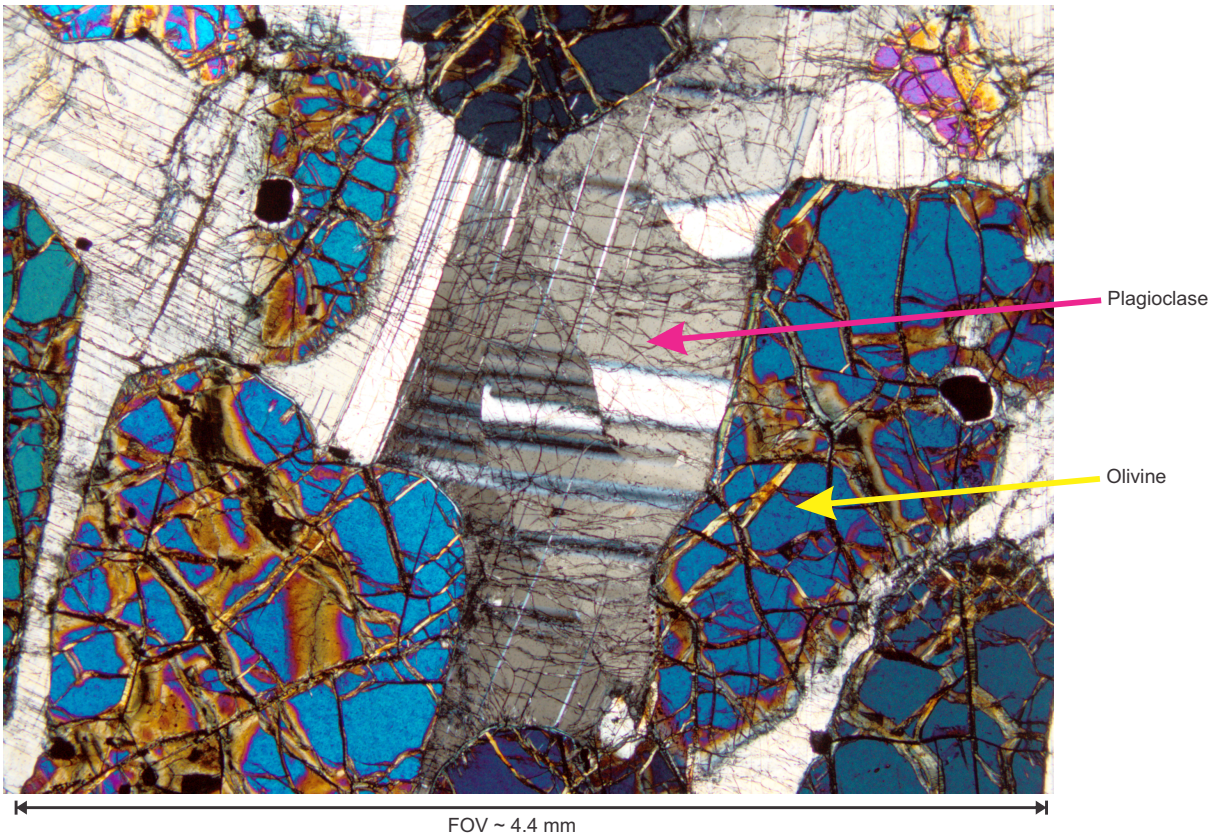


Figure 2.27 – Typical stellate bearing thin-section. Section label Rum D' ("D prime"). Notice the twinned plagioclase crystal(s) occurring interstitially to the olivine with both albite and Carlsbad twin laws. The thicker lamellae which run nearly perpendicular to albite twin lamellae appear to be pericline twins. Near the arrow mark for the plagioclase crystal, there are slip boundaries present that seems to cut both the thicker pericline lamellae and the albite twin lamellae.

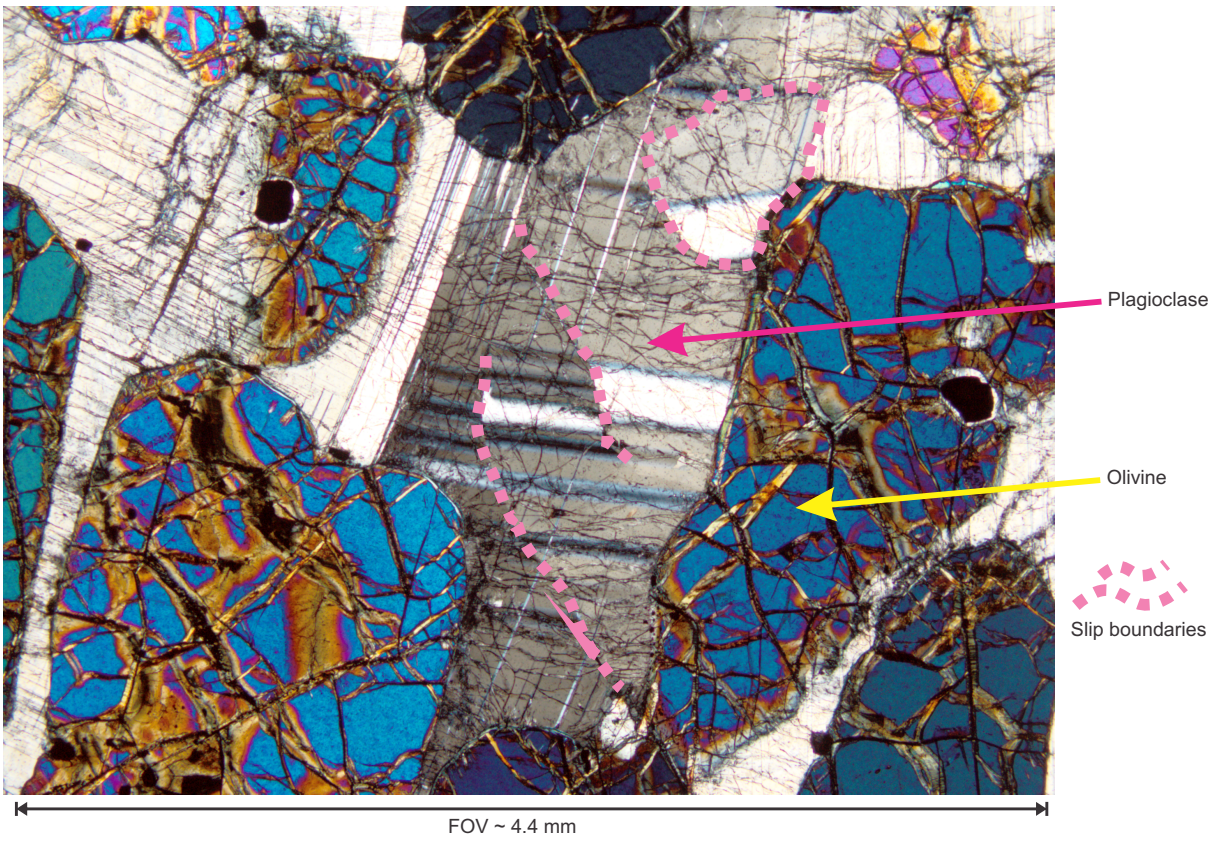


Figure 2.28 – Figure 2.27 with slip boundaries illustrated. The contours of the obvious slip boundaries visible are highlighted in pink dotted lines.

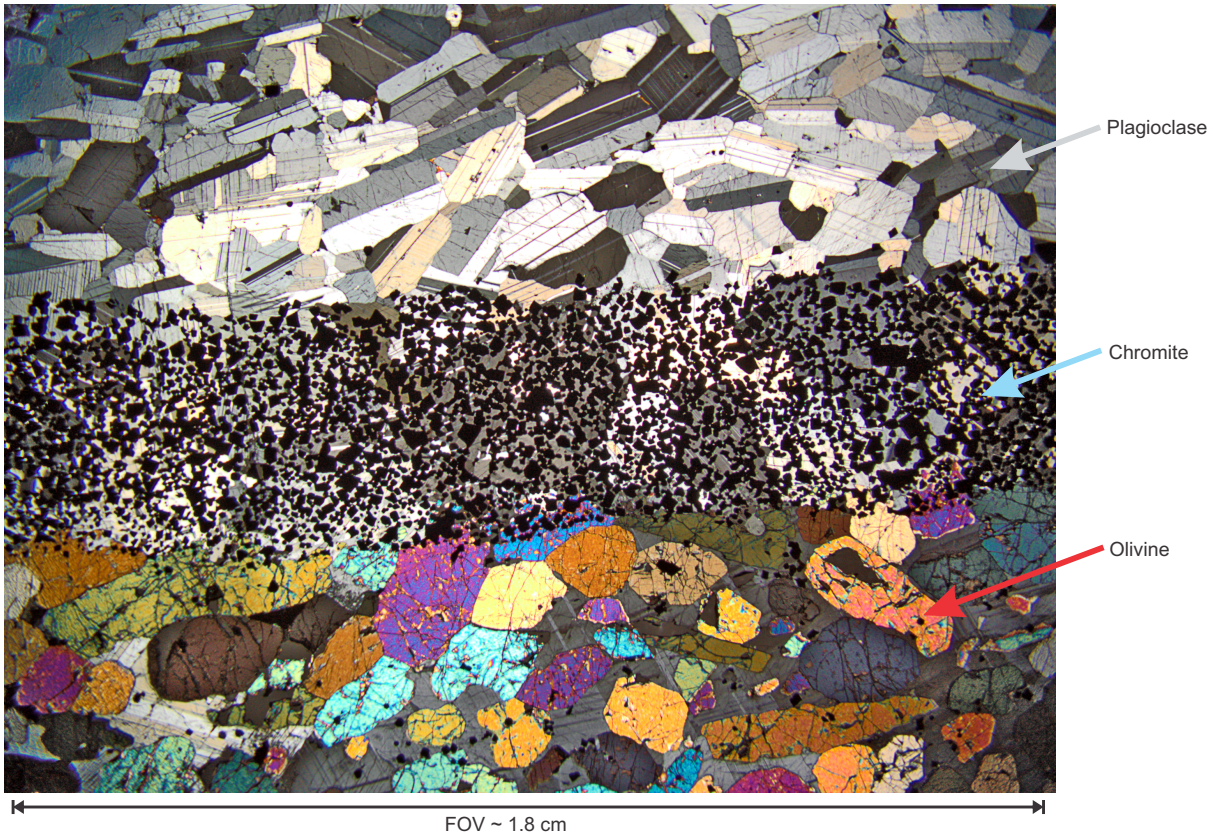


Figure 2.29 – Chromite band in thin-section T19. Magma mixing between residual melt of previous layer and mafic injection of new layer produces sudden chromite precipitation. Section made by drilling into stellate bearing layers.

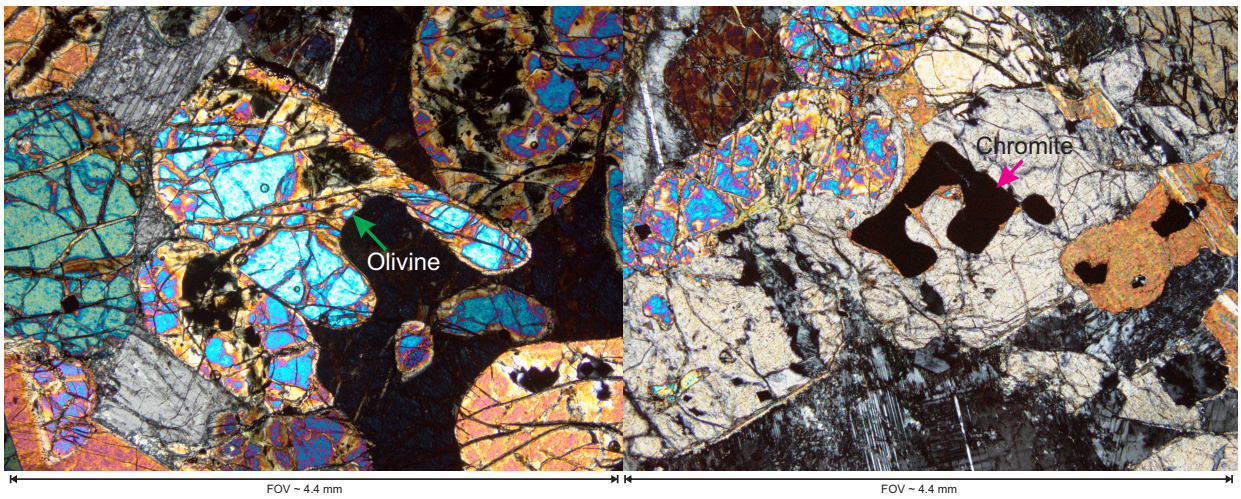


Figure 2.30 – Embayed chromite [center mineral in right picture] and olivine crystals [center mineral in left picture] under crossed-polarized light. Chromite taken from thin-section BK; olivine taken from thin-section T15. Section from Labonté (2004), polarized light microscopy analysis and interpretations by author.

Some thin-sections produced from horizontal sections of the layers parallel to the stellate-plane such as in figures 2.31 and 2.32, contain single large grains of olivine encapsulating other phases, frequently plagioclase. As can be seen in figure 2.31 and 2.32, each plagioclase crystal inside the respective olivine is multiply zoned. Plagioclase enclosing other plagioclase crystals are also common throughout the sample set, an example being given in figure 2.33. The encapsulated plagioclase in both of these cases are typically well rounded, indicating re-melting/resorption interactions with the magma. In some rare cases, the encapsulated plagioclase occur perfectly euhedrally when encapsulated by plagioclase crystals in the stellate rays.

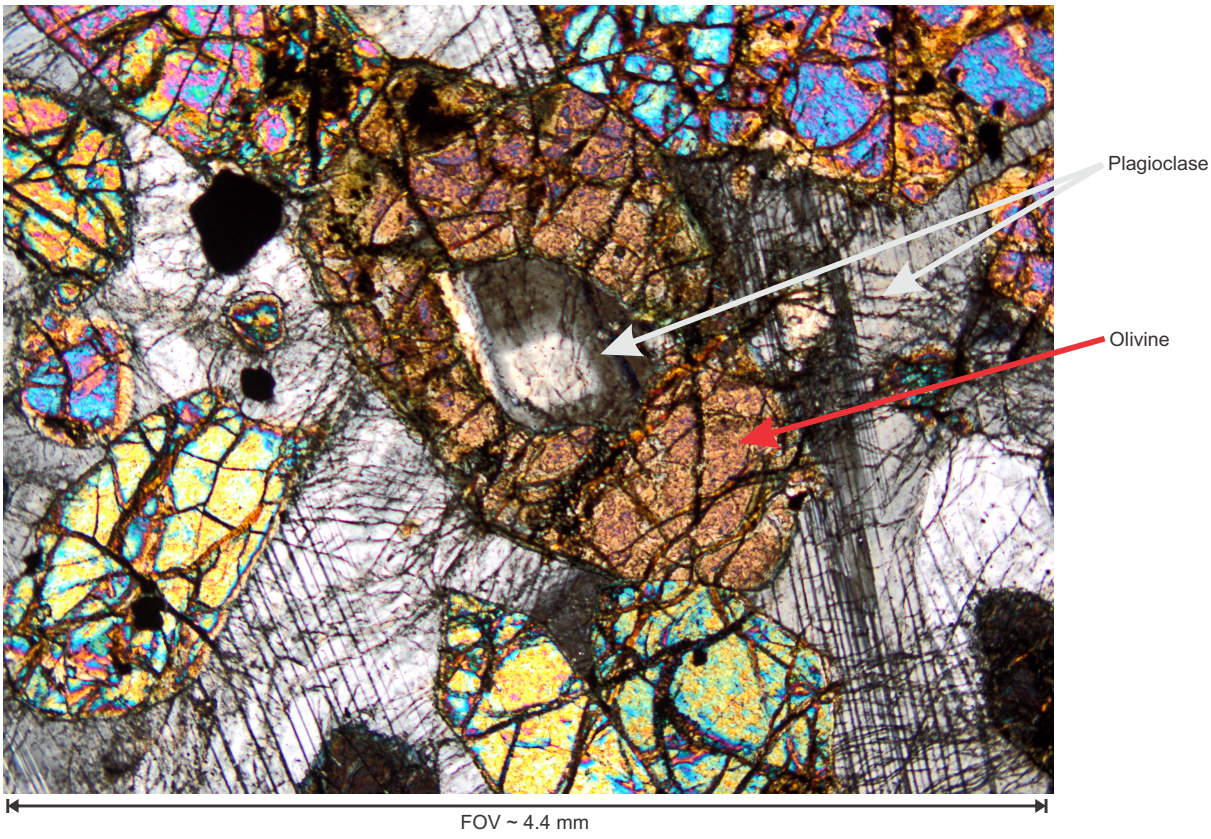


Figure 2.31 – Olivine encapsulating plagioclase phenocryst. The plagioclase was likely dredged or plucked away from the previous layer by the injection of a new layer and subsequently developed two distinct compositional zones, possibly due to different magmatic environments. The olivine crystal then nucleated on the plagioclase and completely encapsulated it. The un-zoned but twinned plagioclase outside the center olivine likely crystallized interstitially later.

Other minerals such as intercumulus orthopyroxene, kaersutite, phlogopite are found in the rocks of the Layered Suite (Emeleus et al., 1996) but are not seen and/or studied here under

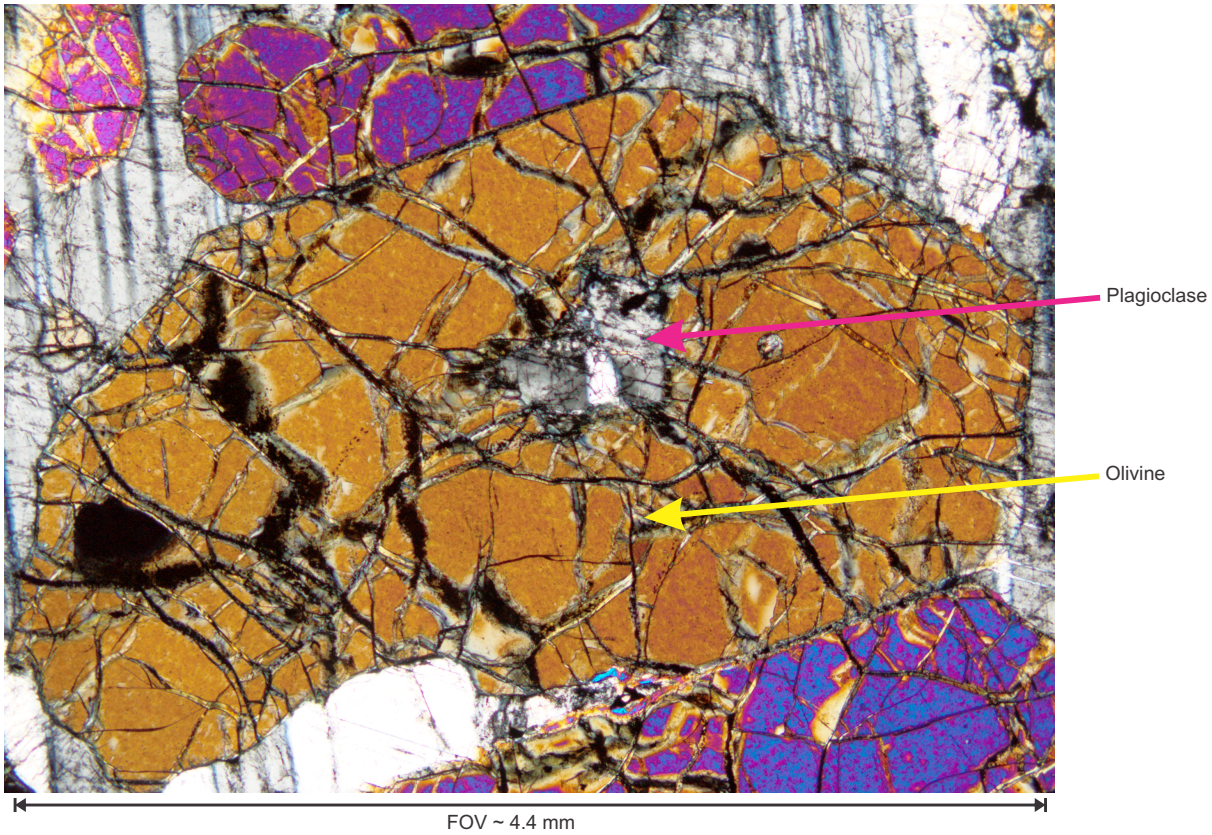


Figure 2.32 – Olivine encapsulating plagioclase phenocryst. The plagioclase crystal shows a central zone flanked by two side zones.

the scope of this thesis.

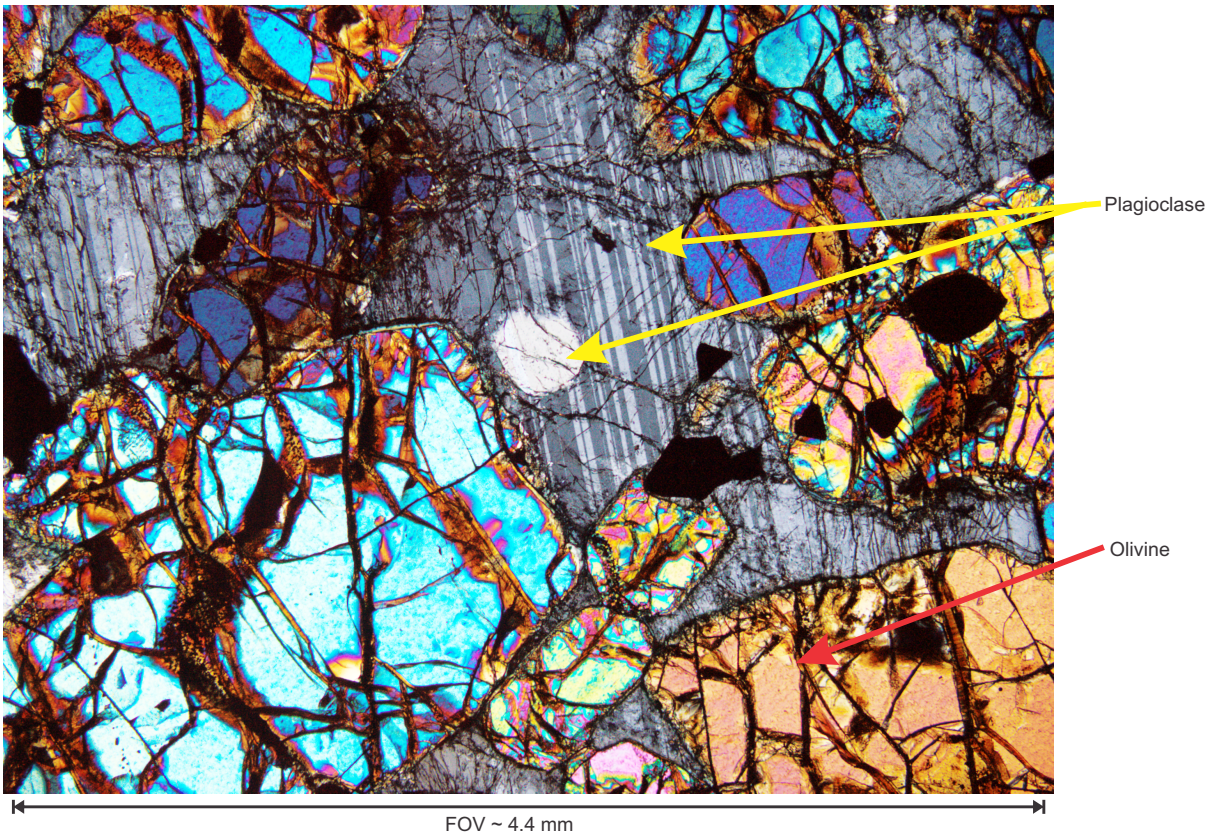


Figure 2.33 – Plagioclase crystal encapsulating a rounded, smaller plagioclase crystal. The smaller plagioclase is seen in lighter grey while the interstitial plagioclase is seen in darker shades of grey and albite-twinned.

2.3.2 Electron Back Scatter Diffraction

Labonté in 2002 in collaboration with Dr. Michael Cheadle of the Department of Geology and Geophysics at the University of Wyoming performed several EBSD scans on the RUM-D thin-section along with several other thin-sections. The original intent was to investigate the preferred crystallographic and shape alignments of olivine with respect to the local spatial coordinates by mapping them using phase and crystal orientation discrimination provided by the EBSD. In general, it was concluded that the olivine exhibits strong crystallographic and shape-preferred orientations on vertical sections with a foliation plane parallel to the average plane as defined by igneous layer contacts. A representative set of data from a serially sectioned piece of stellate bearing gabbro with a layer thickness between ~ 1 dm to < 2 dm is shown in figure 2.36.

The EBSD results presented in figures 2.34, 2.35 are from the thin-section Rum-D prepared by grinding horizontally into one of the plagioclase rays in the plane of stellate propagation and thus is orthogonal to the previously presented thin-section 15A (figure 2.36). The thin-sections are scanned in each case under EBSD producing data representing pole orientations of each electron beam surveyed site along the thin-section. One set of such data is produced for each mineral phase, utilizing EBSD's phase discrimination ability. These sets of data were subsequently coordinate transformed into the crystallographic orientations indicated in each figure. The Miller indices (top left) of each sub-figure indicate the crystal plane normals (poles), with respect to which the data points are plotted. The transformation coefficients Fx, Fy, Fz are at the top right of each subset of figures. Therefore, the second column of the sub-figures suggests a strong crystallographic (b-axis) alignment of olivine crystals in the thin-sections under consideration. This is consistent with EBSD data from orthogonally-prepared thin-sections, microscopy and field observations.

In the RUM-D thin-section, a thin-section dominated by olivine, two large plagioclase crystals are Carlsbad-twinned and encapsulate all the olivine and clinopyroxene. The plagioclase crystals are visible in figure 2.35. The average cell parameters (since plagioclase exhibits phase domains and is incongruent at An75) for An75 plagioclase are $a = 0.818\text{nm}, b = 1.287\text{nm}, c = 0.710\text{nm}$ and $\alpha = 93.6°, \beta = 116.2°, \gamma = 90.6°$ (Smith and Brown, 1988). The Carlsbad rule is a twin axis rule around the c-axis and produces a negligible spread in the b-axes but a large difference in the spatial orientations of the a-axes.

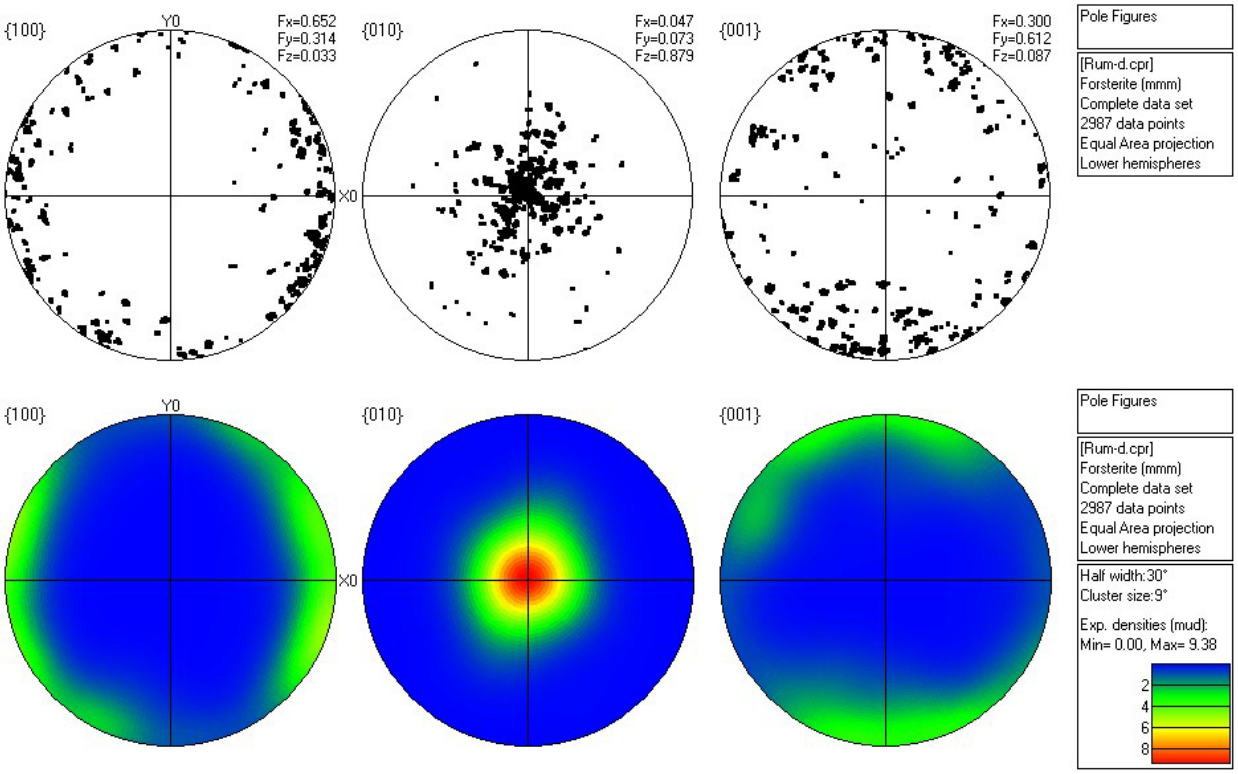


Figure 2.34 – EBSD Data of olivine alignment from Rum-D. A strong alignment of the b-axis (pole) is seen. Data courtesy of Dr. Michael Cheadle and Anthony Fowler.

Thus, the angle between a and a' of the crystals 1 and 2 forms the bipolar $< 100 >$ pole plots in the EBSD scans. A quick calculation shows that $\varphi \equiv a \wedge a' \approx 180° - [2(180° - \beta)] = 52.4°$. Therefore, the EBSD data in figure 2.35 seem to suggest a Carlsbad twin with an a-axes twin angle $\approx 52.4°$. This angle is consistent with a Carlsbad twin of plagioclase based on the optical microscopy and EMP data constraining the average plagioclase composition to be $\sim$An(70-75). In addition, the a-axes are seen to twist helically along the long direction of the thin-section. An illustration for the interpretation of the plagioclase inside Rum-D with respect to the thin-section orientation is shown below in figure 2.37. The analysis of the EBSD results of plagioclase and the subsequent interpretation are by the author.

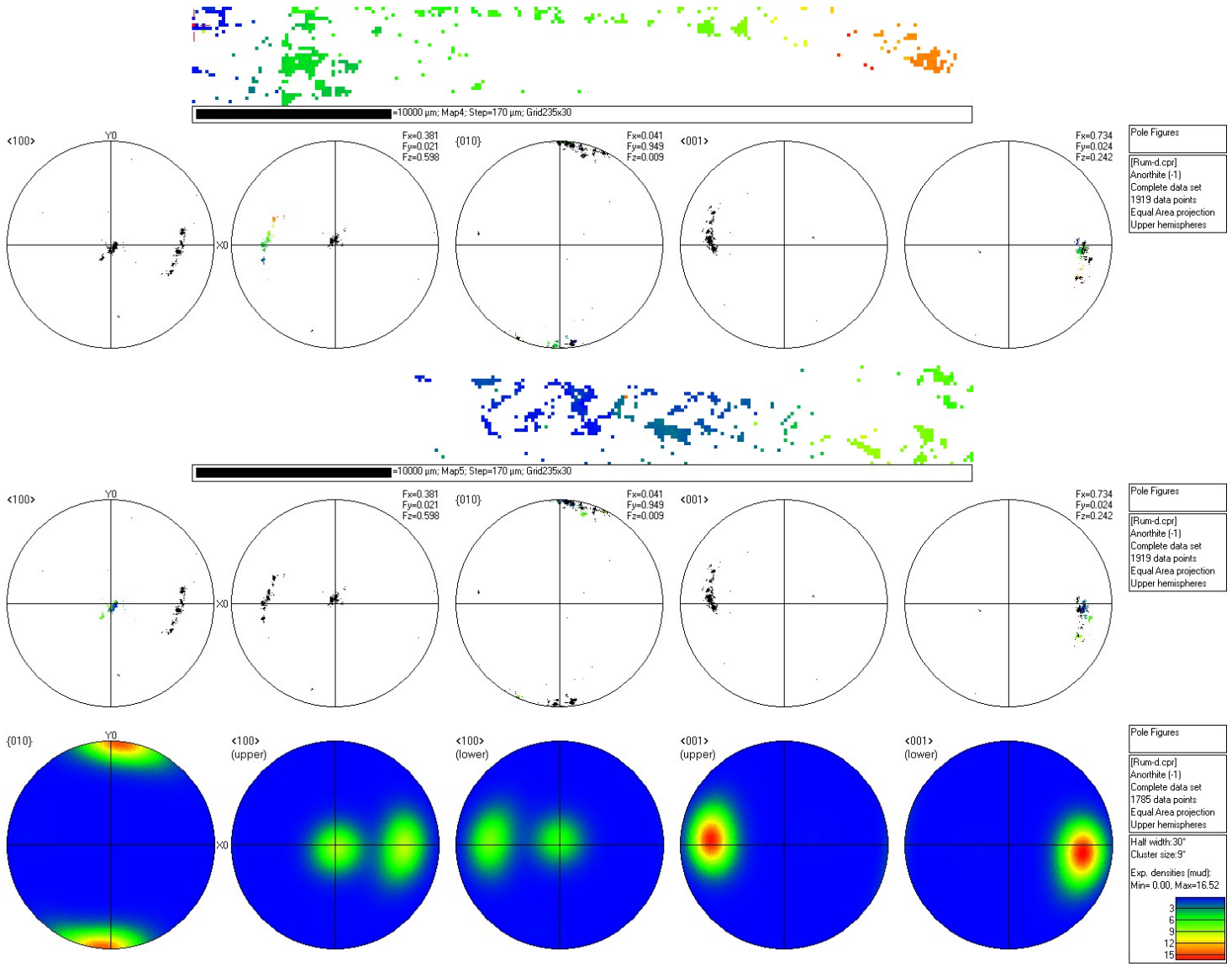


Figure 2.35 – EBSD Data of Plagioclase from Rum-D. Notice in particular the bipolar clustering of a-axis at a well defined angle, suggestive of crystallographic relationships between the crystals in the thin-section. Data courtesy of Dr. Michael Cheadle and Anthony Fowler.

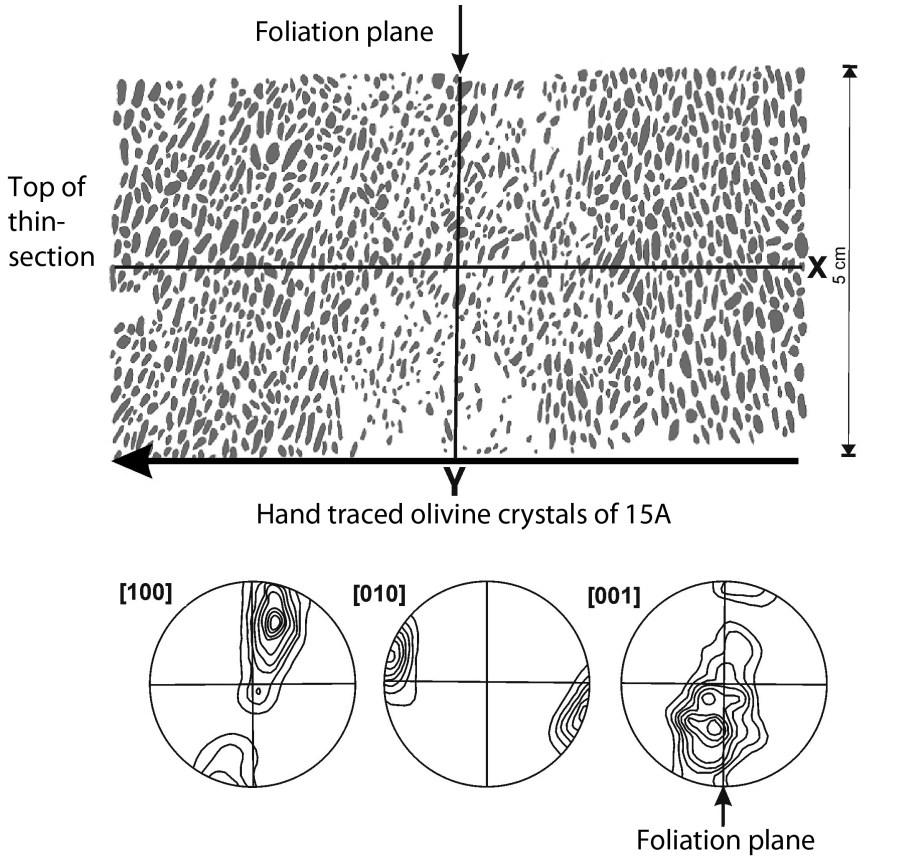


Figure 2.36 – Typical olivine alignment with respect to spatial directions defined by the serially sectioned rock slab. Direction Y indicates the depth direction of the rock slab. Direction X indicates a direction parallel to the rock slab. The top direction as indicated is the weathered surface of the rock exhibiting the stellate patterns. Figure modified from Labonté (2004).

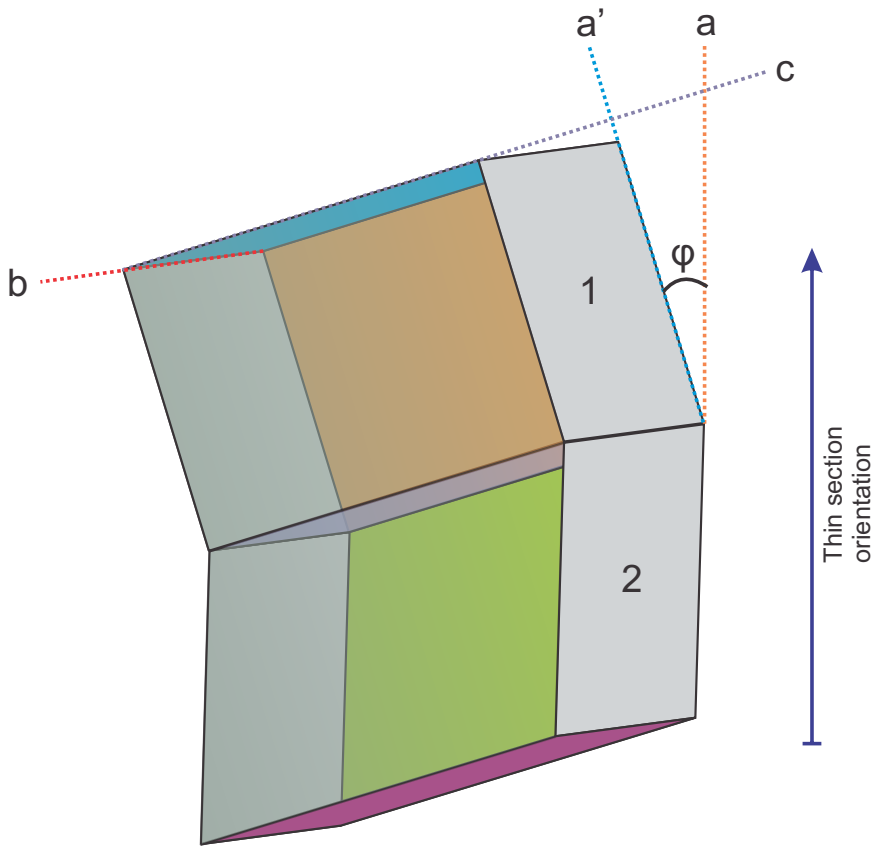


Figure 2.37 – Unit-cell layout interpretation of EBSD data from Rum-D. Two members of the Carlsbad twin are labelled 1 and 2. The angle φ corresponds to the angle of deviation of the a-axes. The arrow points in the up direction indicating the normal to the thin-section plane.

2.4 Brief Discussion

A high frequency of plagioclase twins is observed along with a large variety of twins using polarized light microscopy. The Carlsbad, Carlsbad-albite and albite twins (as well as other lamellae-type, simple and complex twins in plagioclase excluding pericline twins) are known as high temperature, primary growth twins. Genesis requires crystallization at relatively high temperatures associated with an intrusive igneous environment or laboratory equivalent, and the frequency of twins in a given crystal correlates with the rate of crystal growth and inversely with post-crystallization annealing time (Xu et al., 1997). The observation of a variable frequency of twins may support the idea that in some places, the plagioclase growth rate was fast, and in other locations it was rather slow, likely due to variable melt saturation. However, in general, the stellate rays that contain plagioclase crystals with minimal zoning are observable only along stellate rays and are of an oscillatory nature with a variable frequency of twinning. This constrains the growth rate of plagioclase as typically slow enough with respect to diffusive chemical transport but fast enough to produce multiple, dense twins in the stellate rays. The lack of significant amounts of zoning and sharp zone boundaries can also be attributed to a later reheating event, such as re-injections that caused chemical homogenization. However, the preservation of narrow oscillatory zones ($\sim \mu$m) precludes any major reheating events after the crystallization of stellate plagioclase. Such a scenario corroborates findings summarized by Emeleus et al. (1996). Given that the plagioclase in the stellate rays exhibit an immature cumulate texture, a somewhat fast pace of growth with subsequent homogenizations seems plausible. The undulatory extinction of a single plagioclase crystal seems to indicate a gradual twisting of the crystallographic axis and/or stress related strain. This can be caused by a gradual compositional change during growth, length-wise and/or mechanical stress during compaction of the layer after crystallization. The compositional difference is not observed with EMP, perhaps due to chemical homogenization after crystal growth.

The observation of a high degree of connectivity is interesting. Such connectivity was not obvious during the field observations. Therefore, it is entirely possible that the number of connections per crystal of plagioclase in stellate systems from the Central Series study area is sufficiently high such that the majority of the plagioclase crystals are connected to other plagioclase crystals via a stellate branch, across the expanse of the layers. It would appear

that the rocks contain a sponge-like structure of plagioclase with a planar radial symmetry in many places where a center, rich in plagioclase, is found close to top layer contact planes. Studies of gabbros rich in plagioclase from other locations (e.g., Philpotts et al. (1999)) have also reached the same conclusion excluding the radial symmetry, with the additional discovery of higher tensile strength for plagioclase-network bearing gabbros under near-melting conditions. Obviously the connectivity of plagioclase interstitially to the olivine and clinopyroxenes depends largely on the total percentage of plagioclase in the rock. Therefore, the stellate sections are probably where the connectivity is highest.

Isolated occurrences of chromite are fairly typical of mafic igneous rocks, Emeleus et al. (1996) summarized the chromite bands' genesis as either settled to the layer boundaries initially in a cumulate process or preferential crystallization at the boundaries of dissimilar melts due to a combination of chemical/thermal gradients such as mixing of silicic rocks that are cooler with newly injected mafic magma. The chromite band observed in figure 2.29 most likely formed when the plagioclase crystallizing mush layer was overlain by a new layer of mafic magma. The sudden contact at the margins between the two magmas allowed some chemical mixing confined to the margins. In this region, metal oxides are unstable in the melt phase and preferentially precipitates into a thin layer (e.g., chromite bands). The onset of precipitation is sudden compared to nominal crystallization rates and thus a sharp contrast develops highlighting the contact margins between the magmas. The plagioclase in this section shows small, unconnected individual grains with some zoning and larger twin domains/lamellae compared to the stellate-originated plagioclase. If the plagioclase in this thin-section grew less rapidly than the plagioclase in the stellate rays, then the relative lack of a high frequency of twinning and a somewhat more mature cumulate texture as indicated by typically larger contact angles and more angular contacts between crystals, would seem to corroborate this idea. The plagioclase crystals in this thin-section were probably already partially crystallized when a more mafic melt infiltrated and changed the local phase equilibria such that mixing occurred near the contact margins. In some cases, embayed textured olivine and chromite are observed as seen in figure 2.30, indicating the relatively fast growth of both minerals when the melt was injected. This implies that the previous layer had cooled somewhat when the new layer was injected, prompting the embayed growth behaviour of chromite. If the temperature range of chromite and olivine embayed growth behaviour are known, then we would be able to infer the temperature contrast

between the injected melt and the previous layer.

In instances where olivine completely encapsulates plagioclase (e.g. figure 2.31), the olivine clearly nucleated on the plagioclase. Since this process requires solid state plagioclase at the onset of olivine crystallization, the formation mechanism requires a melt of mafic compositions to come into close contact with a melt which had already been crystallizing plagioclase. Presumably the plagioclase was dredged up by the injecting mafic melt or plucked away from the top of the layer hosting the plagioclase (perhaps by layer-scaled convection) and then used as a site for olivine heterogeneous nucleation. The temperature change during this process can probably be quantified by the relative An content of the zones of the plagioclase, if we assume the two different compositional zones are due to crystallization in two different environments. The change in temperature affects the chemical composition of the melt, since it alters the mineral-melt stability. Therefore, the minerals of our thin-sections can be divided into minerals which crystallized from the original melt in the layer, and minerals which crystallized via late mixing due to injection(s). For plagioclase crystals in the thin-sections, presumably the long and large plagioclase crystals were crystallizing within a given layer when the new injection(s) occurred, rapidly changing the stability of the crystallizing plagioclase compositions and allowing zoning to occur. The smaller crystals of plagioclase commonly encapsulated and zoned are likely due to injection events, as well the occurrence of chromite bands. When the melt temperature and chemistry shifts sufficiently toward higher temperatures and more mafic compositions, plagioclase crystallization ceased, explaining the relatively small size of the zoned plagioclase crystals and rounded-anhedral shapes. These pre-existing crystals eventually served as centers of heterogeneous nucleation for other minerals from the new mafic melt, resulting in nearly euhedral olivines completely enclosing small zoned plagioclase crystals as seen in figures 2.31 and 2.32.

It can be concluded based on microscopic textural evidence that although there is some variation in the magmatic environment in the Central Series, the magmatic environment was fairly static during crystallization of the stellate-plagioclase as exemplified by the lack of flow alignment, long stretches of continuous interstitial plagioclase and a typical lack of zoning. There is also sufficient evidence, such as immature cumulate texture, abundant frequency and types of twinning and a general lack of zoning, to support that the plagioclase in the stellate rays grew faster than cumulate plagioclase and much slower than required for zoning. Additionally, frequent reheating, re-injection events and chemical homogenization (chemical composition

correlation between layers, lack of mineral compositional variations) occurred on a time scale much shorter than the time scale required to crystallize the plagioclase in the stellate rays. The evidence correlates well with the findings of Emeleus et al. (1996) and combines to paint a picture of a slowly cooled and crystallized environment with few primary minerals punctuated by re-injection events.

2.5 The Crucial Bits

In the anticipation of the construction of a working hypothesis, we need to take into account all of the fundamental observations from and prior to our research on the Rum stellates. There are many different macroscopic and microscopic observations. It is necessary to summarize and itemize the observations deemed crucial from the above sections.

On the largest spatial scale, there are considerable chemical inhomogeneities represented by the preferential occurrences of plagioclase stellate clusters. In some locations, there are abundant stellate patterns observable on the outcrops, while in the vicinity there are outcrops with either poorly developed or non-existent stellate patterns. This spatial variation is three-dimensional to the best of our observation. On a given exposed layer, there are planar 2D variations in the occurrences of plagioclase stellate clusters. This variation is probably on the order of meter squared, i.e., the chemical inhomogeneities vary on a meter scale in linear dimensions. The other dimension is of course depth. This direction has a per layer to multiple layer variation component, where as we documented above, the stellates do not seem to penetrate more than a single layer and the vertical exposed surfaces do not show a pattern similar to what is found on the top horizontal surfaces. This limits the stellate patterns' depth to be relatively shallow, at less than or at very most, equals to the thickness of a given layer of intrusion, typically ∼dm. The top units of the Central Series examined, best exposed near the top of the hill show the most abundant and well developed stellate patterns; while the lower layers preferentially show progressively less developed patterns, to no apparent patterns visible. This can be interpreted two ways: (1) the stellate patterns seems to be concentrated on higher layers, and the higher layers possess a larger ratio of plagioclase versus other minerals; (2) the lower units are less weathered and masked due more abundant local vegetation. Nevertheless, on some well exposed lower layer rock surfaces, there are obvious small patchy stellate textures with no apparent symmetry and very low apparent complexity. The general transition from apparently patchy behaviour to the higher complexity, larger size and higher symmetry behaviour follows a somewhat West-East trend along the slope of the hill, although this trend could also correlate with the preferential outcrop exposure and weathering.

On the microscopic spatial scale, there are also chemical inhomogeneities. Where there are rays of the stellate patterns, there are less per unit volume of other minerals such as olivine

and clinopyroxene and vice versa. This spatial variation is also three dimensional, but the scale of this variation is on the order of centimeters, at most decimeters. This variation can be seen with the photographs presented earlier (e.g., figures 2.9, 2.19).

On this scale, one general observation that is crucial to infer the kinetic regime of the crystal growth process is the lack of diffusively limited behaviour. This can be most appropriately deduced by considering the distinct and general lack of zoning, particularly oscillatory and complex types in the plagioclase crystals that make up the bulk of stellate rays. Plagioclase growth is sensitive to diffusively limited behaviours of the melt since it is part of a miscible system (Ab-An) and is also a tectosilicate. The zoning pattern of the plagioclase crystals is of particular interest to petrologists and volcanologists due to its ability to reveal thermal and chemical aspects of the primary growth environment in an igneous system. The plagioclase also does not show any diffusively controlled behaviour such as skeletal/lath-like or dendritic growth at the microscopic scale; on the contrary it is full-bodied and occasionally euhedral. This general lack of zoning and diffusively-controlled behaviour is observed both with polarized light microscopy and EMP. Some occurrences of euhedral plagioclase (sometimes embedded in the stellate plagioclase) seems to also imply a slow rate of growth and thus slow rate of cooling during the growth of plagioclase. All the evidence suggests that the temperature of the parent system for the range of temperatures associated with crystallization of plagioclase was slowly varied, and can be considered to be close to equilibrium. In the context of the layered intrusions of Rum, particularly for the thinner layers (where the heat loss would be more rapid), this likely implies a very slow rate of heat loss and a fairly constant temperature during the crystallization of the stellate plagioclase aided by the periodic reheating effect of re-injection events.

Beyond the observations intrinsic to the plagioclase, there are other important observations, such as the mechanical fluid mixing signatures of finger, flame, slumping and replacement structures, the chemical fluid mixing signatures of rapid phase precipitation (e.g., of chromite), the preferential alignment of olivine with the foliation plane parallel to the layers and seemingly reverse crystallization sequences where olivine mantles zoned and anhedral plagioclase xenocryst, all indicative of magma mingling and partial-mixing.

In summary, Rum represents a slowly cooled, periodically reheated/re-injected magmatic environment with intervals of undisturbed cooling and very little layer-wide internal flow (when crystallizing the bulk of the observed minerals). Any proposition seeking to explain the stellate

textures must reflect this environmental setting.

Chapter 3

Theory

3.1 Genesis of the Stellate Pattern: A Working Hypothesis

3.1.1 Necessary Conditions

No theories of the formation of the Rum stellate textures have been proposed to date. One of the first publications on the discovery of the Rum stellates was C.H. Donaldson's in 1973 (Donaldson et al., 1973). They noticed the macroscopic radial systems of plagioclase stellates and coined the name "poikilo-macrospherulitic". The naming scheme, as they clarify, highlights the interstitial nature of plagioclase, and the apparently spherulitic nature of the rays, but on a macroscopic scale not typically seen in spherulites. They conjectured several conditions for the formation of these stellate textures including the existence of a feldspathic peridotite magma; post-depositional regeneration of plagioclase and olivine as radiate and harrisitic growths; changes in water pressure or fractional crystallization of olivine; independent mobility of a liquid with plagioclase chemical composition and nucleation-crystallization in the liquid column prior to layer injection event. Several other ideas have been discussed in private communications, most notably synneusis, nucleation-limited crystallization and convective flow. Each idea independently does not satisfactorily explain the texture. See section 3.1.3 for detailed discussions.

A working theory must be able to explain satisfactorily the crucial details of the stellate patterns presented in section 2.5. In particular, it must address the rise of large scale inhomogeneous distribution of plagioclase rich clusters, and the rise of small scale inhomogeneous

occurrences of the rays of a stellate, as well as the non-diffusively limited behaviour as exemplified by very seldom oscillatory zonings in large plagioclase crystals in the stellate rays (even under EMP) and nearly constant An composition over a large spatial range in the plagioclase crystals. The theory should also fit with the layered igneous environment of the Rum complex, specifically that the layerings arose through repetitive magma injections into incompletely crystallized layers under apparently near-isothermal conditions. It should also naturally generalize to explain the genesis of radial patterns vs. the more spongey, foam like patterns observed in non-Rum sourced gabbros such as the fine mesh-like networks described by Philpotts et al. (1999).

3.1.2 A Little Story

To propose the theory of stellate formation in the Rum complex, let us first set the scene and bring back to vivid imagination what was happening at Rum during its volcanically active years of the past.

Somewhere deep under the ocean, a very basic magma intrudes into the surrounding country rock, oozing along the fault now known as the Long Loch Fault. Where the Lewisian gneiss contacts the Torridonian sandstone, a discontinuity allows the magma to pool in thin sill-like bodies (Emeleus et al., 1996). Periodic replenishments of the magma along the same conduit yielded layers upon layers of decimeter to meter scale intrusions, each layer pushing through the previous one at the site of magma entry point and likely overlaying the previous partially solidified layers. The environment surrounding the intrusions is highly insulated; the source is periodically replenished, allowing the intrusions to remain at high temperatures, likely near the liquidus temperature for plagioclase for many of the layers. The newer layers are frequently deposited while the previous layer(s) are still mushy and only partially solidified.

Shortly after the intrusion of a layer, the crystals of olivine (nesosilicates) begin to settle from the melt to the bottom of the layer, adding to the cumulate mass of denser minerals. Where the olivine crystallizes, the residual melt in the vicinity necessarily becomes preferentially enriched in plagioclase and clinopyroxene chemical components. As the temperature lowers gradually towards sub-liquidus temperatures for clinopyroxenes and plagioclase, their respective chemical components and short range order bonds within the melt become unstable energetically. This is required for chemical/structural differentiation. As the different phases in the melt which are

sub-liquidus begin to differentiate from the rest of the melt via surface tension (immiscibility), spinodal decomposition/nucleation-growth occurs. This process is aided by the presence of crystallized/crystallizing minerals such as olivine which exposes surfaces to the melt state. In the vicinity of olivine, the melt is chemically differentiated by both gravity and surface tension. By these processes, the top of that layer becomes more concentrated in plagioclase-rich melt, which at lower temperatures begin to polymerize (into tectosilicate structures) possibly later than, or overlapping with the polymerization and crystallization of clinopyroxenes (chain silicates). While no exact rates have been established for cooling and crystallization, some rough rate estimates cooling at $(1-10)^\circ$ C per year (Emeleus et al., 1996) and crystallizing slowly ($0.1-1.0$ m per year) (Emeleus et al., 1996), though it is logically conceivable and highly probable that in more felsic patches, the rates of cooling and crystallization are much lower than the bulk overall rates and fluctuate with the injection of new layers, since the rate of cooling of any given layer is proportional to the temperature gradient, and re-injections reheat the previous layer(s).

The addition of a new layer of magma pushing through the previous layer and eventually over-riding the top layer (other mechanisms occurring in tandem such as compaction are possible, without affecting the hypothesis, by contributing to inter-layer interactions in a heterogeneity-enhancing way, see Sparks and Huppert (1984)) causes gravitationally driven Rayleigh-Taylor instabilities if the previous layer is sufficiently liquid in bulk or in patches, whereby the denser injected basic magma on top fingers through the relatively lighter but more viscous (due to polymerization) plagioclase and clinopyroxene-enriched mush (e.g., figure 2.3). This inversion occurs periodically when a new layer is added before the previous layer has finished crystallizing. The plagioclase-rich upwelling fluid is heated and re-homogenized chemically by the new layer. Since the fingering and spinodal decomposition/nucleation-growth process is not spatially uniform on the decameter to decimeter scale, local regions rich in plagioclase and clinopyroxene fluids appear and enlarge. Since it is possible that some clinopyroxene and plagioclase crystals have already partially crystallized before the onset of plagioclase crystallization, the upward convection dredges up chunks of partially crystallized plagioclase and clinopyroxene crystals embedded in polymerized and enriched fluids. This introduction of more basic melt interacts with the existing mush of crystallized olivine, clinopyroxene and largely uncrystallized plagioclase. It then yields a multitude of evidence discussed in Chapter 2 such as finger struc-

tures, replacement structures, flame structures, slumping, etc. The rapidly cooling nature of the new layer and higher viscosity of the more felsic patches inverted from below combined with a lack of large-scaled flow prevents thorough mixing of the residual melt of the previous layer and the new layer. However, it is very likely, at least for some layers and periods between injection events, the temperature was high enough to delay the onset of plagioclase crystallization and partially homogenize clinopyroxenes. In places where the mixing does occur to some degree, and the residual melt is of the correct chemistry, the phase stability in the melt is changed significantly, and chromite precipitates out of the melt and forms thin (mm - cm) bands in some layers as seen in 2.3.1. This precipitation changes the melt composition away from mafic to more silicious. In regions (or layers) where mixing is thorough enough to produce chromite bands, the composition is on average more silicious while in regions (or layers) where mixing is poor, the precipitation of chromite and concentration of plagioclase compatible melt is highly variable spatially, and neither (of chromite and plagioclase) forms complete layers but rather patches.

The new layer undergoes the same process of chemical differentiation with the presence of already chemically differentiated regions from the previous layer. The presence of chemical inhomogeneities at sub-liquidus temperatures favours surface tension-induced phase separation, whereby the chemically homogeneous domains of different chemical compositions held together by short-range forces enlarge. These regions are partially crystallized to form a mesh with the centers located at the most enriched regions. The partially crystalline nature of these regions allows them to better resist the loading effects of new injection layers and retain their shape. With the injection of yet another layer, the inversion process of the lighter plagioclase enriched melt (residual melt) repeats leaving behind interstitially crystallized plagioclase which radiate away from the most enriched regions. When the re-injection is timed such that the old layer has a partially crystallized stellate network with a large enough fraction of residual fluid remaining, the inversion process allows both a large chemical inhomogeneity to exist in the new layer and a stellate texture in the old layer.

The crystallization of other minerals like clinopyroxenes distributed around the plagioclase-enriched pockets tends to slow the rate of cooling in its vicinity, due to the latent heat of crystallization and also speeding up the crucial diffusion process and perhaps retarding the nucleation and crystallization processes for plagioclase. These pockets of dominantly plagioclase

chemistry melt/mush over time merge with other randomly migrating pockets of similar compositions via occasional channels through the layer(s) while crystallizing plagioclase crystals slowly. When the crystallization process for plagioclase begins, it likely does so near the most concentrated regions and close to country rock contacts (which are cooler), and thus the centers of each stellate system where the initial nucleation process is more likely to contribute to a stable population of nuclei. These centers of enrichment may also already contain nuclei or small chunks of plagioclase crystals. Where the layer is devoid of these enrichment centers, the nucleation sites are more random. However, the plagioclase crystals can not grow appreciably until the melt is cool enough and sufficient fractionation has occurred. Where these dredged up minerals from the previous layers are bathed in the mafic melt, they serve as nucleating centers of other minerals such as olivines and clinopyroxenes in the new layer. Finally when the temperature and chemistry of the new layer is correct for plagioclase crystallization, the inverted residual melt centers serve as the initial crystallization fronts and radiate outward into the established regions and channels of plagioclase enriched melt. As the crystallization of plagioclase progresses, the temperature of the system becomes more constant, perhaps even completely isothermal in large volumes surrounding the immediate crystallization regions due to the large latent heat of crystallization of plagioclase (Getson, 2006). Regions enriched in plagioclase and not immediately neighbouring heat sinks such as country rocks can certainly be considered to be macroscopically isothermal during this part of the process (Getson, 2006).

As a crystal nucleates on an existing crystal, the direction of growth is constrained by the crystallography of the crystal and the orientation of the plagioclase-rich fluid channel. Plagioclase is highly anisotropic, and thus the growth rate is highly direction dependent. When the crystal and the channel(s) are sufficiently unaligned, the overall growth rate stalls. If the surrounding melt is sufficiently enriched, epitaxial nucleation processes occur at a higher rate than the crystallization rate allowing a new crystal with a more favourable orientation with respect to the liquid channel to nucleate on the existing crystal or impurities and a new crystal is formed, linked to the old crystal via this surface relationship. Should the nucleus region be an existing crystal, it is more likely the new crystal is oriented in such a way as to both lower the interfacial energy between itself and the existing crystal, and pointing in a direction allowing further growth since the existing crystal has stalled. Unique to plagioclase, and in particular incongruent unit cell type plagioclase around An(70-80), twinning is possible in all directions

and some twin types (Carlsbad, Carlsbad-Albite) produce minimal additional energy barrier at the interface (Xu et al., 1997), and thus the energy cost for twinning as a means to change crystal growth directions to realign with enriched liquid channel is relatively small. And so the plagioclase zig-zags along the preferentially enriched and polymerized channels until it exhausts its supply, or a new layer is overlain. The overall growth rates are low since the layered series of Rum are well isolated and insulated, and in addition periodically reheated due to replenishments from deeper reservoirs (see Chapter 2).

A set of illustrations in cartoon style have been created to illustrate these processes, see figures 3.1 for macroscopic aspects mentioned and 3.2 for microscopic effects using outputs from the eventual model.

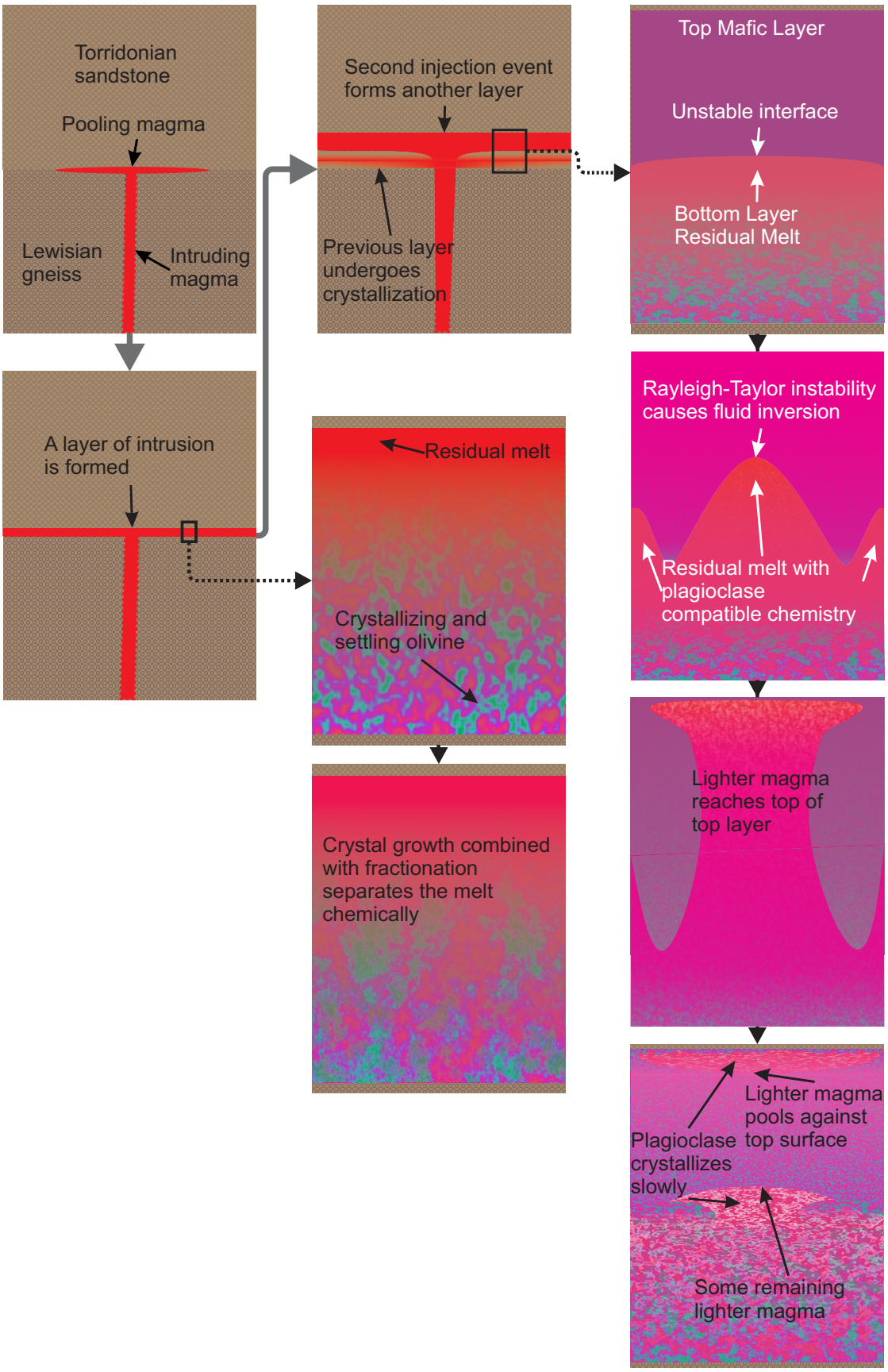


Figure 3.1 – Illustration of events precluding microscopic scale phenomena.

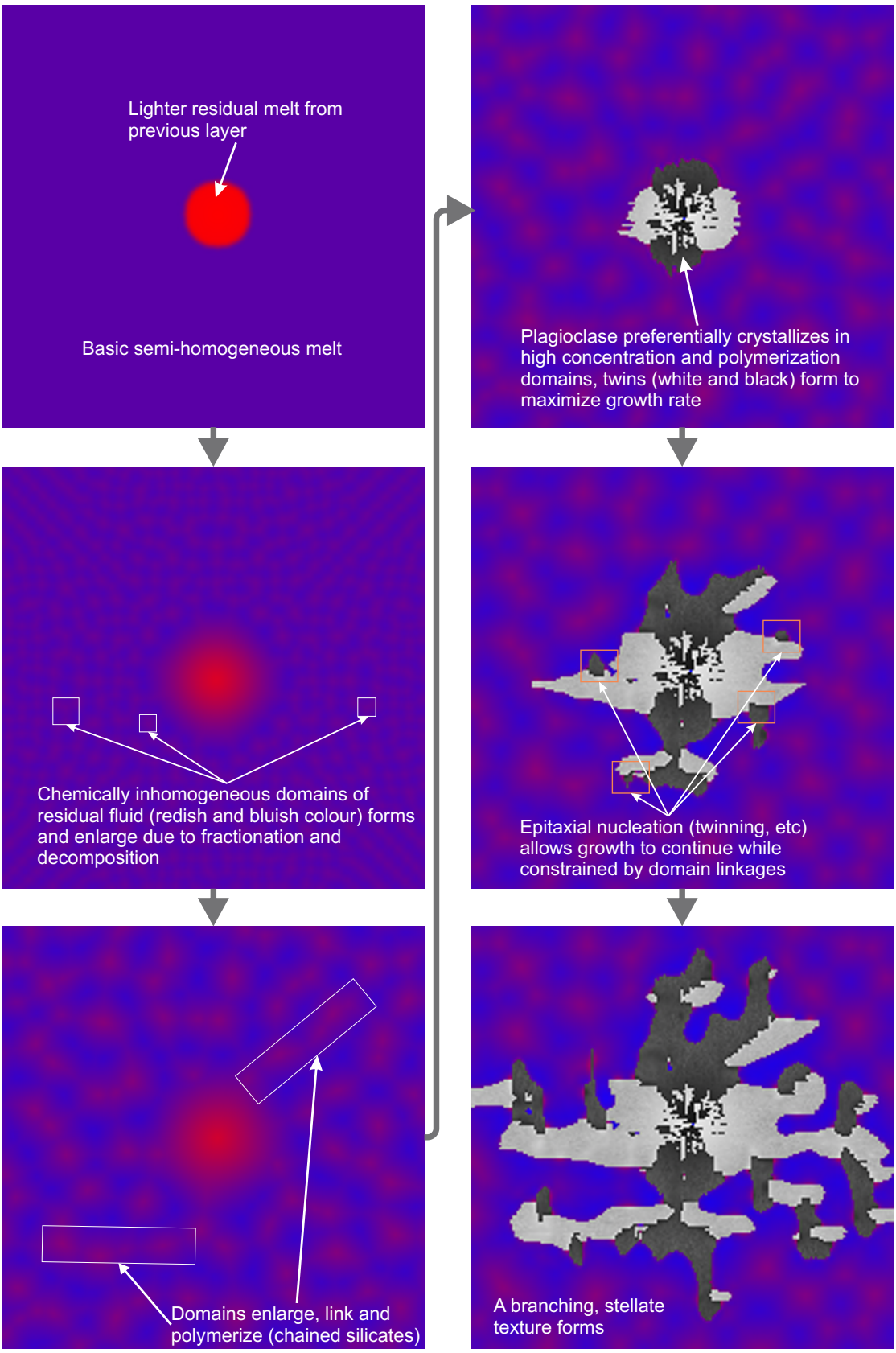


Figure 3.2 – Illustration of microscopic phenomena. Images illustrate a horizontal section of the model melt and solid state corresponding to a thin horizontal slice of an igneous layer from the Central Series.

3.1.3 Alternative Ideas

Several other ideas were discussed as contributors to the formation of the Rum stellates. The main ones were synneusis, nucleation-limited crystallization and convective (or other) flow.

Synneusis is a self-assembly of smaller crystals in suspension in the melt, resulting in (typically) clumps of feldspars. The idea, when applied to the plagioclase stellates would suggest that smaller sub-grains of plagioclase self-assembled radially outwards from some central point, to great dimensions. This hypothetical scenario has a few problems. One such problem is the direct continuity of plagioclase crystals that form the rays. In synneusis, crystals attach to each other with (usually) random orientations, not necessarily always crystal symmetries. Thus the resulting union of feldspars should rarely exhibit epitaxial relations that are considered primary growth twins. Another problem is that the general appearance of plagioclase as interstitial on the ∼mm scale suggests that if synneusis did occur, the plagioclase crystals would have to be smaller than the interstitial spacing. The existence of long, continuous stretches of plagioclase as a single crystal, much longer and larger than the interstitial spacing contradicts this idea. If we reduce the size of the plagioclase crystals initially involved in the synneusis process that yielded the stellate patterns to much smaller than the interstitial spacing, then in the limit that the crystals approach the size of stable nuclei, this idea approaches our hypothesis, in that epitaxially nucleated and grown plagioclase on existing plagioclase surfaces allows the plagioclase crystals to navigate the melt heterogeneities and form a range of stellate patterns.

Nucleation-limited crystallization is a process whereby the density of nuclei, typically due to kinetic reasons (e.g. melt being held at a particular temperature favouring the formation of fewer nuclei but faster growth), is low per unit volume. Therefore the crystals that nucleate are able to grow without competition from neighbouring growths until they become quite large. When this idea is applied to the plagioclase stellates, it would seem to suggest that the stellates are the results of low nuclei density in the melt, and when the stellates are radial, the center of nucleation would coincide with the center of the radial stellate pattern. A typical thin-section study of the stellate plagioclase suggests that there are numerous locations where plagioclase can suitably nucleate heterogeneously. Where the plagioclase surrounds an existing plagioclase such as in figure 2.33, the opportunity to nucleate is obvious. In fact, with so many existing surfaces in the melt presented by (typically) small crystals (not necessarily of plagioclase, but also clinopyroxenes and olivine) and layer contacts, it is hard to logically conceive an environment

where nucleation of plagioclase would be difficult. However, the idea that the centers of radially-symmetric plagioclase stellates are likely the centers of nucleation events for those particular stellates, coincides with our hypothesis for the most probable nucleation site for such systems, although it is not an absolute necessity in our hypothesis (in that it is possible to generate the same patterns without central and singular nucleation events).

Convective flow occurs when thermal/chemical gradients cause density contrasts, which then combined with gravity, allows bulk quantities of materials to move in a concerted fashion. This idea is already incorporated in the hypothesis on the macroscopic scale, as a candidate for the surface-preferential distribution of plagioclase. However, it is not the entire hypothesis, because it lacks the necessary explanations for the microscopic observations.

3.1.4 Uniqueness of Plagioclase Stellates

Perhaps the most often raised question is the uniqueness of Rum plagioclase stellates. Several supporting premises should be stated prior to addressing this question. Number one, the Rum plagioclase stellates were only discovered ~ 40 years ago, and most of that time span resulted in traditional studies of Rum with the patterns neglected. Number two, the number of researchers in pattern formation and self-assembly (geological or otherwise) is a trifle compared to longer-established disciplines and areas, and it was not until the full development of chaos theory, around the same time as the discovery of the Rum plagioclase stellates that pattern formation became sub-fields in most physical sciences. Number three, the amount of layered igneous intrusions around the world that satisfies: a proper amount of weathering with a shallow dip of the igneous layering plane (so as to maximize horizontal exposure), a minimal amount of alteration, cataclysmic events and sufficient erosional exposure is small. Perhaps the Isle of Rum, and to some extent the adjacent isles and the Skaargaard intrusion are the only possible candidates. Number four, the hypothesis for the formation of the Rum plagioclase stellates also puts specific restrictions on the magmatic environment in the genesis of the patterns.

Having stated the most important premises to the question, the answer of the uniqueness in general lies probabilistically. Is it possible to see plagioclase stellates in other intrusions ? If we generalize the stellate patterns from a range of no-symmetry to high-symmetry occurrences (depending on the strength of intra-layer to inter-layer contributions to heterogeneity), with the property that the plagioclase crystals are interstitial to the other minerals and necessarily bigger (volumetrically) in three dimensions compared to other minerals, then the answer is most assuredly yes. Any gabbro with interstitial plagioclase that exhibits the above-mentioned properties would be an example. The lack of weathering may not bring into relief the traces of plagioclase connections. This is probably what Philpotts et al. (1999) observed. In personal communications, it has been discussed that Skaargaard also exhibit some plagioclase stellate rays. However, due to the lack of weathering, they are not easily visible and require viewing at a specific angles depending on the orientation of the sun's rays.

Generalizing the criterion in the hypothesis for the formation of the plagioclase stellates, it is conceivable that similar conditions are present in other geological (or non-geological) situations resulting in a macroscopically assembled pattern from microscopic kinetics. If this holds true, then the occurrence of plagioclase stellates would be a specific application of a more general

type of pattern formation mechanism. In this context, the question of uniqueness would not connote "unique" but rather "where".

3.2 Nucleation and Spinodal Decomposition, Crystallization in a Melt

The classical theory of nucleation considers large atomistic fluctuations, stoichiometrically identical to the final solid in a metastable region of a macroscopic free energy (e.g., Gibbs, usually approximated by Helmholtz free energy in materials sciences) to overcome the sum in reduction of bulk (volume) energy and a gain in free surface energy as a new pseudo-unit cell (reactant) is added to the existing proto-nucleus (product). Since the bulk energy is a function of the volume of the product, it varies as the third power of the linear dimensions of the product whereas the surface energy is a function of the surface of the product and thus varies as the second power of the linear dimensions. As can be expected, when the product reaches a critical size where the further gain in surface energy is more than compensated by the loss in bulk energy of the phase transition, crystallization then proceeds. A very good in-depth derivation from first principles can be found in *Kinetic Theory in the Earth Sciences*, see Lasaga (1998). The large atomistic fluctuations required by classical theory of nucleation lies in the formation of an initial cluster whose size typically varies between orders of $10 - 1000$ unit cells depending on specific conditions. The classical theory of nucleation is derived with the reaction product (proto-nucleus) in equilibrium with the macroscopic phase surrounding it. The properties (such as surface tension and chemistry) of the proto-nucleus does not change while it is forming. The period of time which precedes the establishment of the steady-state nucleation rate is typically referred to as the induction period. The microscopic affinity of constituents in the phase prior to the onset of crystallization is not considered. Post crystallization events such as Ostwald ripening are invoked when necessary as additional processes independent of classical nucleation.

Spinodal decomposition on the other hand, is a newer theory, originally based on thermodynamic grounds which explains the spontaneous separation due to small fluctuations of two or more phases in a mixture which is unstable energetically. The term "spinodal" literally describes the limit of stability of a mixture. Where the mixture is unstable, phase separation (ordering) occurs via spinodal decomposition. It is a generalized idea of the apparent spontaneous separation of phases, typically referred to as "unmixing". This instability can be induced by super-cooling, introduction of a new chemical species, or physical disturbances or any other number of modifications to the local state variables. At the onset of instability, the

phases which originally were mixed homogeneously spontaneously separate into stable phases via surface tension (more appropriately interfacial surface tension) via hydrodynamics and/or diffusion. Several different formulations exist to describe spinodal decomposition, such as the Cahn-Hiliard equation, which also describes Ostwald ripening (as a specific case of spinodal decomposition, where chemicals are transported to cluster in a homophase fashion). Unlike nucleation, spinodal decomposition has no dependency on large atomistic fluctuations. The types of fluctuations needed to initiate spinodal decomposition are always present at temperatures above absolute zero and thus no real activation energy is needed to proceed, (e.g., see Kostorz (2001)). Spinodal decomposition occurs in liquid phases as well as in solid-solid phase transitions, such as in feldspars, where solid state re-ordering occurs to separate phases which differ in structure as the feldspar is gradually cooled. In K-feldspars, this typically results in patchy, inhomogeneous textures and lamella; in plagioclase feldspars, this results in unit-cell domains, modulation and super-structures whereby the unit-cells break down into lamella of potassic and albitic domains with sharp boundaries. The main macroscopic requirement for spinodal decomposition to occur is interfacial surface tension. The interfacial (non-zero) surface tension of multiple types of chemical compositions in a mixture (melt or solute, or otherwise) serves to drive apart chemically dissimilar phases due to internal interactions of identical phases and non-identical phases. The microscopic equivalent of this description is that molecular bonds established in a mixture between different chemical species have different affinities with general preferences of one species to itself over the other participating species in the chemical mixture. In general, the surface tension requirement of spinodal decomposition is better satisfied as polymer sizes and structures becomes bigger and more complex. This is due to the cumulative mismatch, which for larger polymer and structure sizes propagates the local mismatches along the entire structure, and particularly so if the structure is inflexible.

In thermodynamic variables, the free energy of a mixture (usually Gibbs or Helmholtz free energy) should exhibit a downward pointing (second derivative of the free energy function is less than zero) curvature, see figure 3.3. The result of spinodal decomposition is an initial coagulation (via uphill diffusion) of mutually attractive species (this need not be stoichiometrically the same as any solid state materials resulting from the eventual phase transition) to form an overall percolation network of entangled pathways (channels) of inhomogeneities (e.g., see Kostorz (2001)). This transitions to a second stage whereby these channels gradually collapse and form

larger domains (droplets). These droplets then eventually merge with neighbours through slow bulk diffusion (if the phase only allows diffusion] or faster convection (if the phase allows hydrodynamics) to yield very large domains that survive at the expense of smaller domains (Ostwald ripening, since the large inertial mass of the fluid domains reduces the rate of hydrodynamics, if the phase allows hydrodynamics), see, for example, Bray (2003). The process can be illustrated for a small initial composition fluctuation inside the spinodal region of the phase diagram, see figure 3.4. For illustration of these stages, see figure 3.5.

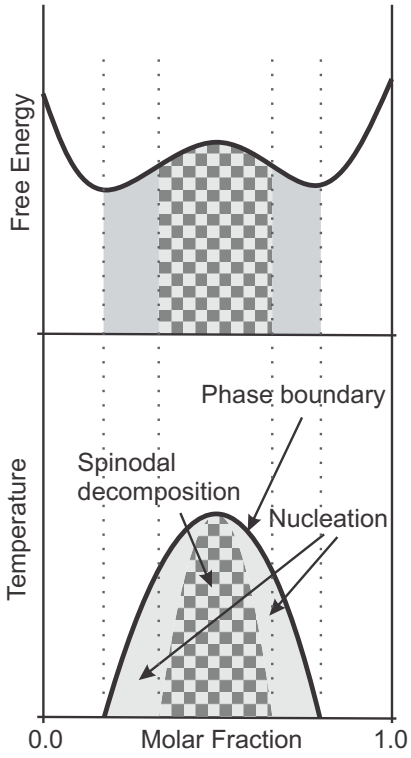


Figure 3.3 – Hypothetical binary free energy diagram and phase diagram showing both spinodal decomposition and nucleation regions. In the two nucleation regions (shaded grey) to the right and left of the spinodal curve defined and illustrated as the boundary of the cross-hatched region, the more concentrated phase becomes over-saturated and leaves the liquid mixture via nucleation and growth. In the spinodal decomposition region (cross hatched), both fluids become over-saturated and leave the mixture via spontaneous unmixing.

As per free energy-stability arguments, it is expected that a (binary) mixture will either decompose spontaneously if the mixture is energetically unstable, or nucleate a new phase and

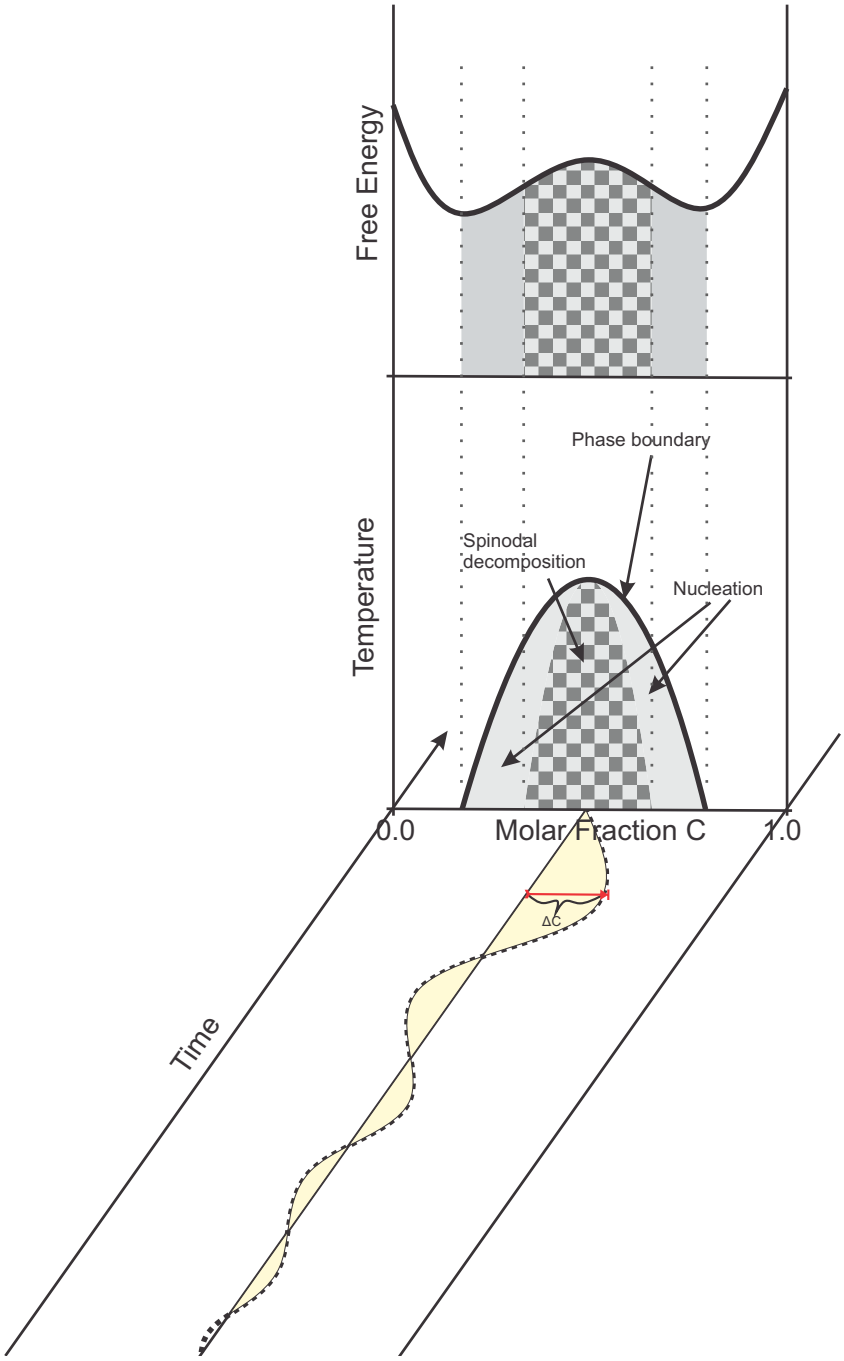


Figure 3.4 – Illustration of amplification of small ΔC with time inside the spinodal region as a function of time. ΔC is a compositional (heterophase) fluctuation. Once the fluctuation becomes large enough to reach outside of the spinodal region, nucleation takes over as the dominant mechanism (homophase).

coarsen if the mixture is metastable. The two paths for one set of conditions are mutually exclusive with a sharp changeover transition. For multi-saturated mixtures (or melts), the phase transformation of each species needs to be considered, and its effects on the remaining mix-

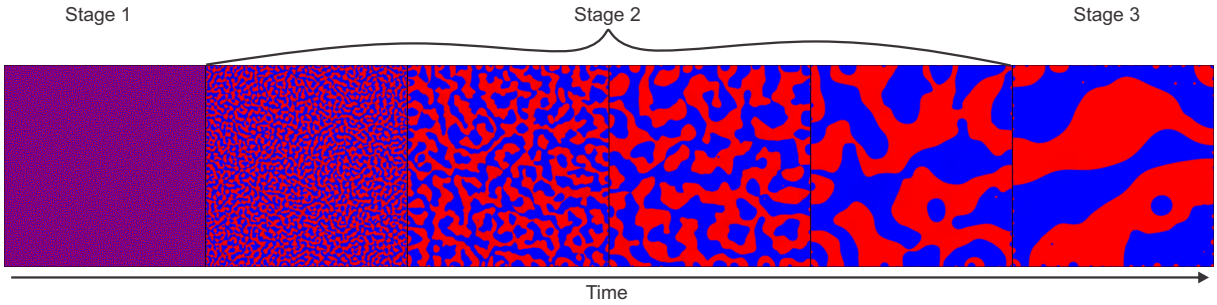


Figure 3.5 – Illustration of spinodal decomposition in a binary fluid mixture. Stage 1 is called the diffusive regime, wherein early formed macroscopic interfaces gradually develop due to uphill diffusion. Stage 2, called viscous hydrodynamics, occurs at intermediate times when the coarsening of the system is controlled by viscous hydrodynamics. Stage 3, late times, when the inertia of individual domains dominate the coarsening behaviour, it is appropriately called inertial hydrodynamics. Terminology of stages is from detailed analysis of binary fluid mixture spinodal dynamics by Bray (2003).

ture taken into account. It is entirely possible that the phase transformation of one phase in the mixture leads to, or is coupled either via spinodal decomposition or nucleation to another phase transformation. Examples of this type of reaction can be found in fractionally crystallized scenarios in the systems Anorthite-Forsterite-SiO_2, Diopside-Forsterite-SiO_2 and Fayalite-Leucite-SiO_2, see Morse (1980). The continued fractionation of olivine in those systems usually leaves the residual melt metastable/unstable and the system can separate spontaneously due to spinodal decomposition (provided the system is initially prepared in the proper chemical stoichiometry). In melts, the most commonly observed phase transition phenomenon is frequently attributed to nucleation and crystallization (liquid-solid phase transition), although direct evidence of classical nucleation as a transient phenomenon is non-existent for complex chemistries and unit-cells and theoretical predictions of nucleation dynamics using classical theories are poor for all but a few systems. For the example systems mentioned above, both spinodal decomposition (liquid-liquid phase transition) and crystallization can be observed, although usually as observably separate events if one carefully controls the thermal and chemical aspects of the system. It is much less clear if and how spinodal decomposition can be coupled with nucleation and growth in igneous melts. Much of this uncertainty comes from a relative lack of experimental and theoretical investigations. In other, closely related areas of science such as materials sciences and polymer/chemical physics, a much larger amount of research output is available. In general, they tend to unanimously support the idea of coupled decomposition to nucleation and growth in the sense that a decomposition stage primes the mixture with inhomo-

geneities to allow the larger fluctuations needed for nucleation (e.g., see Kostorz (2001)). The initial decomposition does not necessarily proceed to generate spatially large domains, since nucleation and growth processes usually take over when compositions of the domains gradually become suitable. In some instances, where the nucleation barrier is thought to be large (the critical nucleus is physically big), the phase transition process is enhanced by concentration of a specific phase, and nuclei form inside the separated domains. In other cases, where the nucleation barrier is largely in the creation of a new surface, the nuclei form at the boundaries between phase domains. Hence in this sense, spinodal decomposition and nucleation events are intrinsically idealizations, i.e., end-members of phase transitions both of which contribute (in multi-saturated melts) to the final solid product. The phase transition process in this case would be liquid-liquid: an amorphous ordering of chemicals, then liquid-solid: transition from amorphous ordering to structured ordering, then possibly further solid-solid, coarsening and further structural ordering.

Historically, as Cahn (1969) originally pointed out, even back in the 19^{th} century Gibbs understood the compatibility of classical nucleation with spinodal decomposition. In approaching the spinodal, if one accepts that the original nucleus is not necessarily compositionally in equilibrium with the macroscopic phase surrounding it, but instead, certain differences are to be amplified due to changing surface tension as the nucleus gradually acquires a structure, then a non-classical theory can be formulated which encompasses the two schools of thought. Therefore, a nucleus can be stoichiometrically different from the final solid phase, and can be seen as an intermediate phase with a gradual transition from the stoichiometry of the macroscopic phase undergoing phase transformation to the final solid and in this process, acquire some form of structural ordering. In reality, this transition should be more apparent for chemically complex solids where numerous elements must come together to form a single unit-cell. Each chemical component of a phase has a characteristic atomic size and charge, and therefore, a different mobility in a mixture. Conceptually, fluctuations in the macroscopic phase necessarily exist at the per atom level first, then any coupled fluctuations on a molecular scale or larger. Even for a relatively large nucleus, perhaps on the order of ~ 1000 unit cells (for silicate melts), the linear dimensions of such a nucleus is only ~ 10 unit cells. The electronic structure of such a small solid is a very poor approximation of that of the larger solid, in that each unit cell involved is a very small distance away from a surface of dangling bonds. For such a structure to eventually become

identical to the final solid structure, the nucleus must eventually adopt spatial order. This highly non-classical nucleation process with a distinct structural transition process is being studied by several groups including non-classical nucleation studies of pre-nucleation amorphous clusters by De Yoreo (2012), computational studies of amorphous nucleation byDemichelis (2012) and simulation of non-classical crystallizations by Gale (2012). However, theories regarding realistic nucleation kinetics (non-classical nucleation) are not as easily established on the basis of free energy thermodynamics, since the changing composition and structure removes the convenience in describing the free energy of the nucleus. Therefore, one of the approaches to modern nucleation theories is based on the microscopic, cluster-kinetics approach. Binder is one of the first to propose non-classical nucleation theories, such as summarized in Kostorz (2001). Other formulations also exist, for example, a theory formulated without the the constraint of compositional equilibrium was proposed by Schmelzer et al. (2010), and *spinodal decomposition assisted crystallization in binary mixtures* by Mitra and Muthukumar (2010). These ideas are not entirely new. The concept of a ramified droplet as coagulation of a concentrated region that gradually undergoes structural rearrangement to compactify and yield a nucleus in the classical sense has long since been discussed (e.g., Klein and Unger (1983), Unger and Klein (1984), Heermann and Klein (1983)). These types of theories supports the experimental and simulation data from all different areas of science concerned with phase transitions.

A full treatise on the development of spinodal decomposition and nucleation was given by Binder in 1991, in the book *Materials Science and Technology*, republished in 2013 (Binder and Fratzl, 2013), although plenty of published journals on this subject matter including Binder's own predate this by more than a decade. He argued that the apparent incompatibility of spinodal decomposition with nucleation largely stems from the definition of free energy for systems near criticality. Free energy of mixtures is typically derived by choosing a macroscopic observational scale, over which, the system is coarse-grained to define a constraint free energy. In the case of spinodal decomposition, one needs to choose a coarse-grain size that is larger than the interfacial width of the two-phase (or more) region and small compared to the wavelengths of the fluctuations of the macroscopic phase. This requirement in the scaling of the constraint free energy function for the two-phase region prevents the construction of a unique spinodal. Therefore, the sharp transition of the spinodal to nucleation behaviour is physically meaningless. The mean-field formulation of kinetics of spinodal decomposition also necessitates unrealistic

assumptions. By making assumptions of infinitely-weak and infinitely-long-ranged forces, a mean-field approximation can be obtained that facilitates the location of a fictitious spinodal, still dependant on the choice of the coarse-graining scale, typically irrespective of the real physical system by choice of assumption. This mean-field spinodal and mean-field type, sharp transition (singular) behaviour does not occur in real mixtures, apart from highly idealized systems. The smooth transition from spinodal decomposition to nucleation depends on ΔG in the phase transition where G is a free energy (i.e., Gibbs free energy, often approximated by Helmholtz free energy in materials sciences). In the region $\Delta G \sim kT$, where k is Boltzmann's constant, spinodal decomposition occurs, and in the region where $\Delta G \gg kT$ nucleation occurs. The transition between these two regions is smooth and a gradual change in behaviour occurs as the rate of nucleation exceeds the rate of decomposition. This gradual change in thermal phase transitions usually limits the size of domains attainable with initial decomposition, as nucleation and growth of a new phase will likely take over at a characteristic domain size. Binder also argues that in most experimental analysis of the spinodal boundaries, the extrapolations of the extent of the boundary at near critical temperatures and critical mixtures are not physically meaningful and possibly ambiguous, (e.g., Binder and Fratzl (2013), Kostorz (2001)). The most current summary of this type of phase transformation can be seen in detail in the text *Phase Transitions in Materials*, (Kostorz, 2001).

Therefore, the actual behaviour in systems not at the physical extremes adequately described by either classical nucleation or "classical" spinodal decomposition usually exhibits a range depending on composition, temperature and cooling rate. In summary, the phenomenological description of miscibility and immiscibility largely depends on the scale of observation: from the size of the molecules (in molecular systems) themselves to the scale of interest of the occurrence of the phenomena, (illustrations of this newer and the older treatments, see figures 3.3 and 3.6). The definition of miscible to immiscible depends on the choice of scale of observation for near critical phenomena and hence is particularly important for igneous melt systems. A cursory application of these findings to igneous systems suggests that the contribution of spinodal decomposition is critically important in igneous melts. The initial pre-nucleation stage (induction period) is dominated largely by spinodal decomposition (or spinodal nucleation), and the subsequent crystallization of species should follow its respective enrichments in the melt. In the late stage, the heterogeneous melt structure due to early stage decomposition might transi-

tion into large domains, crystalline or amorphous depending on the situation (Ostwald ripening) feeds existing growths sites until solid state-only effects ensue, where Ostwald ripening is still very important. The effects of heterogeneities in the early melt may also have a profound effect on the kinetics of phase separation. Already crystallized species expose surfaces to the melt, which may exhibit preferential interactions (of a decomposition nature) with different phases in the melt. Effects such as surface-directed concentration waves (Jones et al., 1991) and wetting layers (Dietrich, 1988) can be significant in early melt evolution while for late stage melt evolution, the effects of a propagating wetting layer and domain growth competition are important (Krausch (1995), Puri and Frisch (1997), Binder (1998)).

The existence and acceptance of such non-classical theories in other fields outside of geology is starkly contrasting with the present state of understanding of melt crystallization in igneous systems, where as far as research literature is concerned, immiscibility is a much more fringe, less investigated and accepted idea, despite the direct applicability of theories and conclusions from adjacent fields such as materials sciences and polymer chemical physics. Some of these difficulties are obvious: the high temperature environment of igneous melts and slow crystallization of some minerals hinders direct observation; while other difficulties are less obvious: the smaller polymer sizes and typically higher viscosities compared to polymer mixtures in the afore-mentioned sciences implies that necessarily, reduced possibilities of direct observation of immiscibility, due to sluggish kinetics. However, one can immediately relate the polymerization in igneous melts to that of synthetic polymers and alloys and deduce that, the much smaller molecular scales of materials sciences (metallurgical) and the much larger molecular scales of polymer sciences nicely encapsulate the extremes of natural molecular sizes in igneous systems. The behaviour of igneous systems in this regard, must necessarily fall in between those extremes. Some of the remaining difficulties in the acceptance of non-classical theories seem to stem from a nearly dogmatic adherence to mean-field approximations, and its incorrect, singular behaviour at the spinodal.

Currently, the most well accepted theory of melt crystallization for igneous systems incorporates only the classical theory of nucleation and growth combined with the experimental phase diagrams to explain melt evolution past the onset of crystallization with much of the sections of the phase diagrams remaining unexplored. The result of this, of course, distills into the well known Bowen reaction series, and the incorporation of partitioning effects such as fractional

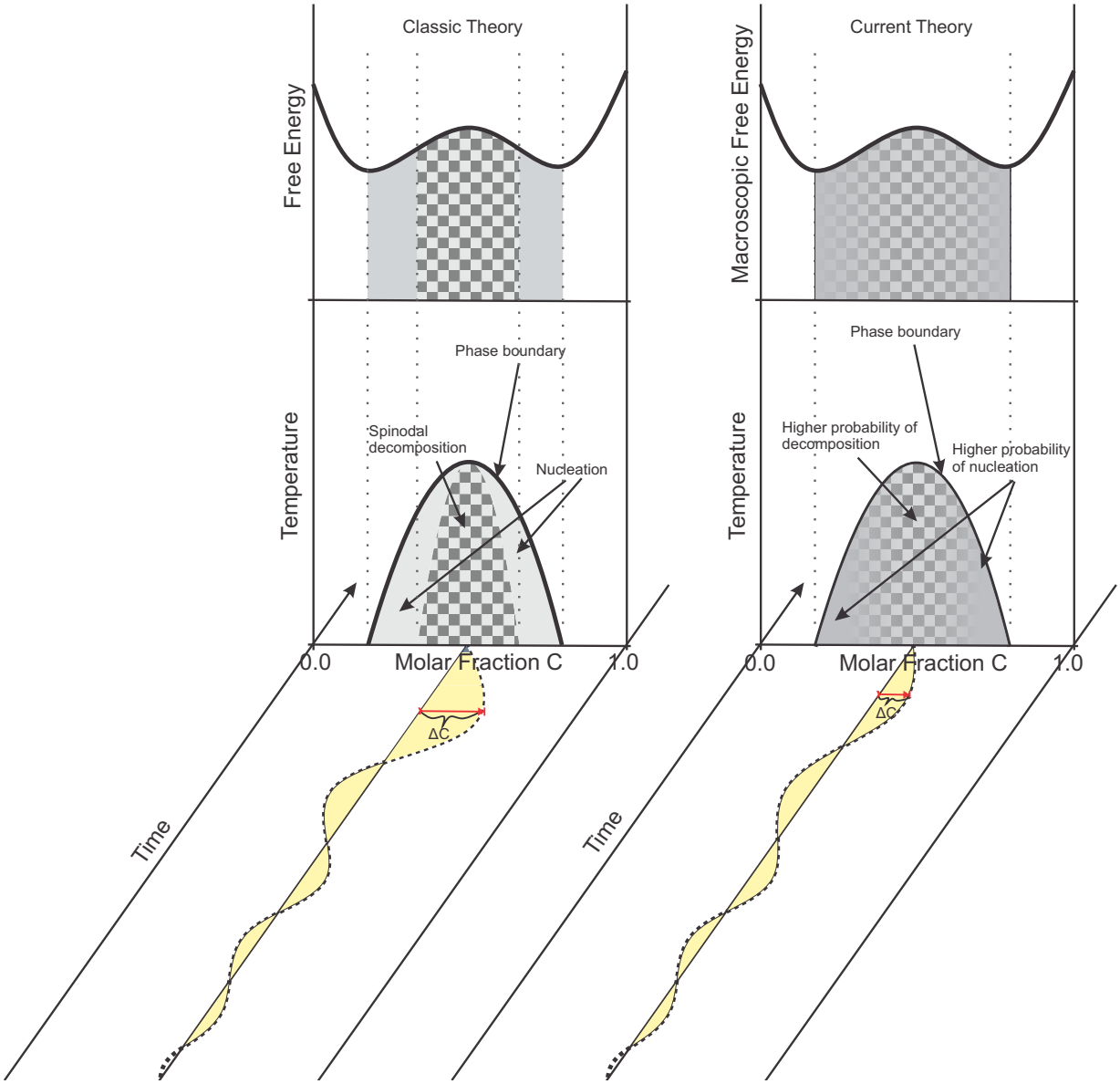


Figure 3.6 – Illustration of classical nucleation and spinodal decomposition (unmixing with nucleation) in comparison with the most current theory of spinodal nucleation. Notice that in the classic case, unmixing and nucleation are separate, distinct scenarios while in the current theory, the transition between the two is smooth, and no distinction is to be made.

crystallization, ionic partitioning, and additional effects such as mixing and melting of country rocks. In this picture, the melt migrates along a silica enrichment trend whereby, generally speaking, the residual melt becomes progressively depleted in elements needed to make shorter polymers such as monomer silicates (i.e., olivine), chained polymer silicates (pyroxenes and amphiboles), sheet polymers (micas) and framework polymers (i.e., feldspars, etc). Since many of these different structures and lengths of polymers occur in the melt in tandem, the crystalliza-

tion sequence of minerals is adjusted to fit experimental results. The effects of surface tension and its implications to spinodal decomposition and subsequent coupling with nucleation and crystallization are almost without exception ignored.

This traditional picture of melt crystallization being rather incomplete and disjunct, is unable to explain many natural and experimental observations, such as the existence of spherulites when a melt is rapidly quenched, and droplet emulsions and fine bandings found in volcanic glasses and experimental equivalents. The role of such liquid phase amorphous clusters, domains and channels and nucleation-crystallization is completely missing. Nor is it able to explain some fine layer sequences in igneous layered intrusions such as the Skaergaard intrusion (e.g., Holness et al. (2007)). It is clear from some studies such as Veksler et al. (2007), Veksler et al. (2010), Holness et al. (2011) and Parsons et al. (2008) that surface tension inside the melt plays a major role in the evolution of the melt, particularly nucleation and ripening. Veksler et al. (2010) directly measured the interfacial tensions between Fe-rich and silica-rich fluids and apparently observed non-classical (i.e., incomplete separation) spinodal decomposition type of phase separation. They estimated natural basalt unmixing interfacial tensions to be about $(5-8)$ mN/m which agrees with prior calculations by Hammel (1967). As a reference, the interfacial surface tension between water and vegetable oil is $(30-32)$ mN/m at room temperature. In all of the experimental studies mentioned above, the authors observed μm sized emulsions (in colloidal suspension) with sub-μm sized channel networks of phases. Complete separation of Fe-rich and silica-rich liquids is rather slow naturally, but can be accomplished with a large enough crucible and centrifugation, see (Veksler et al. (2010), Veksler et al. (2010)).

To generalize the ideas of nucleation and growth with the ideas of spinodal decomposition for a multi-saturated polymer melt, we arrive at a new picture of melt crystallization for igneous systems. In this new framework, a perfectly homogeneous melt is gradually cooled and due to surface tension between different types of polymers inside the melt, small inhomogeneous chemical domains begin to form and enlarge to μm - mm sized (by simulation and natural samples) domains and channels (otherwise known as ramified droplets). If quenched, the channels quickly collapse and the remaining domains can generate spherulites and emulsions. If not quenched, these domains and channels will very slowly enlarge progressing through the initial stages of spinodal decomposition. This percolation structure depends on the bulk proportions of the separable components in the melt. When further cooled, the heterogeneous domains will

become centers which facilitate the nucleation process by supplying a range of stoichiometries needed to construct the unit-cells. Once a non-classical proto-nucleus is large enough, structural reconfigurations occur to transition amorphous nuclei into structured nuclei. This describes the kinetic processes that occur during the ill-understood nucleation induction period and combines what is known of phase transformations and glass structures. Therefore, the processes that takes place during the induction period is crucially important in determining the spatial distribution of nuclei and at least the initial crystallization rates that follow. It is expected that these processes hold important implications on the current approach to the texture, partitioning and fluid dynamics, as applied to geology in general, but particularly petrology and volcanology. During the growth period, the presence of the crystals' surface to adjacent melt typically increases the local interfacial tension (between compatible and non-compatible chemical domains) by presenting a structured surface and thus reducing the ability of that surface to accommodate the dangling bonds presented by the other chemicals and attracting more compatible chemicals. This locally increases (at constant temperature) the rate of melt decomposition. If a crystal grows in the melt this way slowly enough, or the melt is periodically reheated, there will always be channels of enriched areas attached to the crystal (such as extended wetting layers) and the crystal should never experience diffusion limited growth. The first stage where nucleation is enhanced by liquid-liquid spinodal decomposition is the first feedback in a melt. The last stage where the growing crystal enhances the liquid-liquid spinodal decomposition is the second feedback in a melt. In general, the effects of these feedbacks depend on the thermodynamic state of the melt including parameters such as temperature and pressure, which controls the viscosity and surface tension and thus spinodal decomposition rate and the chemical potential allowing phase transformations. These effects also depend on the original composition of the melt, which past the onset of crystallization can be a function of time and space in the sense of fractional crystallization, decomposition and external interferences such as mingling and mixing. The latter solid-solid transformations due to decomposition occur in a similar fashion, if the solid products are subsequently cooled sufficiently slowly, but the transport process is only diffusive.

In this more complete picture of melt crystallization, the chemical inhomogeneities preferentially tapped by the slowly growing plagioclase serve to explain the formation of the stellate texture. The restriction of the texture to top surfaces of these layers can be explained by grav-

itational gradation and melt fractionation. The centers of some of these radial stellate systems can be explained by Rayleigh-Taylor instability (or other differential fluid motions) driven fluid upwellings during liquid-liquid interactions of layers.

In essence, the proposed theory of formation of Rum stellate texture can be divided into three parts:

(1) Decomposition, nucleation and crystallization coupled with gravity chemically differentiates the melt in any given layer.

(2) Repetitive addition of more layers enhance the chemical inhomogeneities due to fluid motions.

(3) Anisotropic crystallization of plagioclase constrained by chemical inhomogeneities and crystal orientation with some feedback into the melt facilitating further spinodal decomposition (and nucleation in the form of heterogeneous nucleation).

(4) Heterogeneous nucleation and subsequent, repetitive epitaxial growth allows slow adaptation of growth fronts to melt inhomogeneities.

3.2.1 Choice of Models

To investigate these effects numerically for the stellate textures found in the Rum igneous complex, the above mentioned ideas need to be carefully examined and all important aspects which are expected to contribute to the formation of the stellate textures should be captured by the model. These aspects include:

(1) The plagioclase enriched centers in the parent melt due to inter-layer interactions.

(2) The spinodal/nucleation decomposition of the melt.

(3) The subsequent crystallization.

(4) Inhomogeneity guided crystallization with orientation-anisotropic growth rates.

(5) Heterogeneous epitaxy and growth.

The igneous layers as found in Rum are thin and the texture is largely two dimensional, parallel to the top surfaces of the layering. In addition, the main logical effect of (1) is to introduce large spots that are not necessarily circular (dm-m scaled) of chemical inhomogeneities. We can effectively forgo this process and simply initiate our model melt with pre-existing chemical heterogeneities. Therefore our fluid model can and shall be two dimensional. The jump from two dimensions to three does not yield significantly more information but does allow additional

connections between enriched channels at the expense of much more computation time and reduction of simulation grid size. This will be dealt with later in the next chapter.

Number (2) needs to be simulated using a fluid model capable of multi-phase transport and spontaneous genesis of heterogeneities. Since the transition from spinodal decomposition to nucleation is entirely smooth, the only difference between the two pertaining to the generation of the heterogeneities is the rate (and therefore the total time required to develop a fixed amount of melt heterogeneities). The rate of spinodal decomposition is much quicker than the crystallization process due to nucleation alone, since crystallization involves the diffusion of chemicals and structural ordering, while spinodal decomposition only depends on the hydrodynamics and diffusion. However, for a method to generate the intra-layer heterogeneities, spinodal decomposition is chosen. In addition to the standard spinodal decomposition (which is typically formulated with respect to constant temperature), we need a method to control the rate of and spatial extent of spinodal decomposition. Doing so allows us to fully compare the textures generated when we investigate the rates of crystallization versus decomposition.

The nucleation stage in (3) is not of a great deal of interest to us. We make the assumption a-priori that the geometry of stellate plagioclase textures is not related to this transient phenomenon. Therefore, nucleation for us will be greatly simplified. An additional argument for the forgoing of the nucleation stage is that the spatial scale for the nucleation process is much too small to be captured by a macroscopic fluid dynamic simulation. Given that the main purpose of nucleation is to allow (liquid-solid) decomposition, the effects of nucleation can be adequately captured by just (2), if we can control the extent of separation in the liquid state.

The subsequent growth and interplay of enriched melt channel to crystal growth direction alignment in (4)+(5) obviously needs a solid phase-only model that is capable of chemical reactions with direct coupling to the fluid phase.

Therefore, the summary of the simulation requirements is as follows:

(1)+(2) are simulated by a multi-phase capable, two dimensional fluid model which may need additional control over the extent of spinodal decomposition.

(3) does not need an explicit model.

(4)+(5) are simulated by a solid phase model.

There are many models which are capable of simulating separable fluids and chemical reactions. Due to the flexibility and novel value of Lattice Boltzmann simulations, it was decided

very early in this research that it will be the underlying fluid model of our choice. For the solid phase model, the initial approach was to use a continuous phase transformation in the Lattice Boltzmann model based on first order chemical reactions (rate of reaction is proportional to the driving force to the first power). However, after much consternation, we settled on a discrete, stochastic model similar to Metropolis algorithm-based models.

3.3 Lattice Boltzmann Scheme: A Primer

3.3.1 Introduction to the Lattice Boltzmann Method

The Lattice Boltzmann method (LBM) is in essence an a-priori statistical refinement on the Lattice gas method (LGM).

Lattice gas methods are somewhat modern alternatives (∼1980s) to traditional computational fluid dynamics which employ discretized versions of the continuity and Navier-Stokes equations. The lattice gas method does not directly make use of these equations. Instead, a "lattice" of moving particles and some very specific rules governing local collisions and streaming (moving of the particles) make up the bulk of standard LGM simulations.

An intrinsic nature of LGM is their rather microscopic description of a fluid via directly simulating individual particles (which could be atoms, molecules, unit cells, etc). As such, for a fixed amount of computational resources, the effective scale of the simulation is either small, or the amount of noise per simulation trial is large. To overcome these issues, either a large amount of computational resources are needed to simulate large spatially extended systems, or employ extensive averages from many trials (realizations) of the same simulation scenario.

Following the development of LGM, floating point computational capabilities gradually increased and streaming of real numbers instead of integers along a fixed lattice became possible in real time, the Lattice Boltzmann method became a physical possibility. The Lattice Boltzmann method is a fluid dynamics simulation technique where density distribution functions represented by real numbers are streamed along a lattice with a collision operator more closely resembling that of the collision term in the Boltzmann equation.

As with LGM, LBMs are capable of simulating in a more microscopic fashion, most of the behaviours of real fluids including turbulence, multi-phase flows, etc, but with much higher computational accuracy per lattice size. Thus, LBM compared with LGM can scale up much better on the macroscopic scale at the expense of inability to scale down to small microscopic scale. The presence of continuous distributions also alleviates the noisy dynamical nature of LGM.

As suggested by its name, LBM stems from a spatial, temporal and velocity discretized version of the Boltzmann equation. The Boltzmann equation was originally derived by Ludwig Boltzmann to describe the statistical distribution of one particle in rarefied gas under time

evolution (Morse, 1969). It is a (very powerful) form of microscopic kinetic equation. Since its inception, the Boltzmann equation has been successfully used to describe some properties of magnetic, plasmic, fluid dynamic and solid state systems and is considered to be one of the most important equations governing non-equilibrium statistical mechanics. In the context of gas or fluid transports, many physical properties such as viscosity, conductivity and diffusion coefficients can be derived using the Boltzmann equation if a few detailed microscopic properties of the system under examination are known.

The Lattice Boltzmann method relies in large part on the Boltzmann kinetic equation. It solves the Boltzmann equation to any desired degree of accuracy numerically and can be easily implemented in modern computers, particularly large computer clusters and super-computers. Unlike the parent Boltzmann equation, the Lattice Boltzmann equation can be solved for a large amount of case scenarios, which currently includes thermal and high Reynolds situations.

3.3.2 The Boltzmann Equation

Although various derivations of the Boltzmann equation are possible, one such derivation can be had by considering an ideal gas (or fluid) of local density $\rho \equiv Nf(\mathbf{r}, \mathbf{p}, t)$ where $\mathbf{r}, \mathbf{p}$ are the vector quantities of the spatial coordinate and momentum coordinate in phase space, respectively and t is time. The N is the total number of particles at the location under consideration and $f(\mathbf{r}, \mathbf{p}, t)$ is a probability distribution function (probability density function) that gauges the probability of finding 1 particle in the above mentioned location in phase space as a function of time. Then the total derivative of ρ with respect to time yields 3 terms:

$$\frac{d\rho}{dt} = \left(\frac{\partial \rho}{\partial t}\right)_{\mathbf{r},\mathbf{p}} + \left(\frac{\partial \rho}{\partial \mathbf{r}}\right)_{\mathbf{p},t} \cdot \frac{d\mathbf{r}}{dt} + \left(\frac{\partial \rho}{\partial \mathbf{p}}\right)_{\mathbf{r},t} \cdot \frac{d\mathbf{p}}{dt} + O(\geq 2). \tag{3.1}$$

In the above equation, $O(\geq 2)$ includes all 2nd and higher order total differentials in the standard Landau order notation. These expanded, 1st order terms can then be examined. The first term $\left(\frac{\partial \rho}{\partial t}\right)_{\mathbf{r},\mathbf{p}}$ is the time only dependence of the local concentration, with the location coordinates and the momentum (and thus energy) coordinates fixed. It is a one dimensional derivative with respect to time only.

The second term contains two derivatives, a partial derivative of the density with respect to the 3-dimensional position coordinate, and the time derivative of the said position coordinate.

Since the two terms are both vectorial quantities, the multiplication between them is appropriately a dot product $\cdot$. The first derivative term is simply the positional gradient of ρ in vector form and the second derivative term is velocity which when written in terms of momentum and combined with the first term yields:

$$\left(\frac{\partial \rho}{\partial \mathbf{r}}\right)_{\mathbf{p},t} \cdot \frac{d\mathbf{r}}{dt} = \nabla_{\mathbf{r}} \rho \cdot \frac{\mathbf{p}}{m}. \tag{3.2}$$

The third and last term also contains two derivatives, one of the partial derivative of ρ with respect to the momentum coordinate and one of momentum with respect to time. The first derivative term is the momentum gradient of ρ in vector form and the second term is the change of momentum, which in the presence of a conservative external field represented by a potential energy function ϕ is the force exerted by that field, or the negative positional gradient of the energy function. Hence:

$$\left(\frac{\partial \rho}{\partial \mathbf{p}}\right)_{\mathbf{r},t} \cdot \frac{d\mathbf{p}}{dt} = -\nabla_{\mathbf{p}} \rho \cdot \nabla_{\mathbf{r}} \phi. \tag{3.3}$$

Neglecting terms of Landau order 2 and above, since they approach zero faster than dt approaches zero, ie.,

$$\lim_{dt \to 0} \frac{O(dt^n)}{dt} = 0, n \geq 2, \tag{3.4}$$

then, inserting the first term and the expanded terms in equations 3.2 and 3.3 brings us the entire equation:

$$\frac{d\rho}{dt} = \left(\frac{\partial \rho}{\partial t}\right) + \nabla_{\mathbf{r}} \rho \cdot \frac{\mathbf{p}}{m} - \nabla_{\mathbf{p}} \rho \cdot \nabla_{\mathbf{r}} \phi. \tag{3.5}$$

Invoking our definition of ρ, and dividing by N this equation rightfully transforms into one of the standard forms of the Boltzmann equation:

$$\frac{df}{dt} = \left(\frac{\partial f}{\partial t}\right) + \nabla_{\mathbf{r}} f \cdot \frac{\mathbf{p}}{m} - \nabla_{\mathbf{p}} f \cdot \nabla_{\mathbf{r}} \phi. \tag{3.6}$$

Here we have dropped the subscripts for clarity indicating that the first term involving the derivative of the function f with respect to t at constant $\mathbf{r}, \mathbf{p}$ is implied.

More insight can be gained if we examine more closely what the equation is describing. The

equation states that the total rate of change in time of the probability density function describing a particle's motion is influenced by the intrinsic rate of change of the particle's probability density function with respect to time, modified by the local concentration differential induced flow and further modified by the force established by an external field.

At the per particle level, the total rate of change in time of the probability density function at a location $(\mathbf{r}, \mathbf{p})$ with a small infinitesimal volume $d\mathbf{r}d\mathbf{p}$ is simply the net difference of particles entering this space, and particles leaving this space. Denote the incoming portion of particles entering this infinitesimal phase space from elsewhere $Q_+(\mathbf{r}, \mathbf{p}, t)$ and conversely, the total particles leaving this space $Q_-(\mathbf{r}, \mathbf{p}, t)$. Then the net change (flux) is $Q_+(\mathbf{r}, \mathbf{p}, t) - Q_-(\mathbf{r}, \mathbf{p}, t)$. The particles leaving at the particle level is due to collisions of particles in this space. Only 2 body collisions are considered here since we hereby invoke the dilute gas assumption making higher order collisions increasingly unlikely. A pair of particles originally on trajectories $(\mathbf{r}_{1,2}, \mathbf{p}_{1,2})$ collide and become particles on new trajectories $(\mathbf{r}_{3,4}, \mathbf{p}_{3,4})$. The collision cross section σ_c is the small area cover surrounding a particle orthogonal to the incident trajectory which describes the effective size of the particles for considerations of inter-particle collisions. The overall effect has been likened to, by Philip M. Morse, as the incident bullet on a target of area σ_c (Morse, 1969). The particles $(1, 2)$ involved in collisions have probability density $f(\mathbf{r}_{1,2}, \mathbf{p}_{1,2}, t)$ to be found given by their momenta. The pair in a time dt moves with mutual (with respect to each other) velocity $2v_m$ where v_m is the center of mass velocity, or the mutual velocity of the two particles and is equals to p_m/m_c in the center of mass frame, where m_c is the reduced mass (i.e., $m_1 m_2/(m_1 + m_2)$). Thus the joint probability density of this pair of particles can be written as $f^{(2)}(\mathbf{r}_1, \mathbf{r}_2, \mathbf{p}_1, \mathbf{p}_2, t)$. Considering the collisions of two particles occur in the cross section specified by σ_c, then the coordinates $\mathbf{r}_1, \mathbf{r}_2$ are considered equal in the context of this two body collision process (the two particles collide at the same physical place), and thus $\mathbf{r}_1 \approx \mathbf{r}_2 \equiv \mathbf{r}$. In a time dt, the total probability of collision of particles 1 and 2 is found by multiplying the joint probability density $f^{(2)}(\mathbf{r}, \mathbf{p}_1, \mathbf{p}_2, t)$ by the mutual velocity p_m/m_c by the differential collision cross section area $d\sigma_c$ and the differential phase volume $d\mathbf{r}d\mathbf{p}_1 d\mathbf{p}_2$. This yields

$$P(f_{1,2}) = \frac{p_m}{m_c} f^{(2)}(\mathbf{r}, \mathbf{p}_1, \mathbf{p}_2, t) d\mathbf{r}d\mathbf{p}_1 d\mathbf{p}_2 d\sigma_c dt. \tag{3.7}$$

If we assume that the probability density functions of the particles 1 and 2 are inde-

pendent, or equivalently the particles motions are not correlated in time prior to collision, then the joint probability density $f^{(2)}(\mathbf{r}, \mathbf{p}_1, \mathbf{p}_2, t)$ can be split up such that $f^{(2)}(\mathbf{r}, \mathbf{p}_1, \mathbf{p}_2, t) = f(\mathbf{r}, \mathbf{p}_1, t) f(\mathbf{r}, \mathbf{p}_2, t)$. This is the so called *Molecular Chaos* hypothesis or *closure assumption*. While this hypothesis preserves the forward moving nature of time and maximization of entropy, it is not entirely correct. Every particle collision is in fact a starting point of correlations between two particles during which some energy transfer occurred. Therefore, a system which only progresses through time with collisions between particles will have inter-particle correlations to some degree, of prior collisions between pairs, triples, etc. If we ignore the correlations, we ignore the collisional history of the system; but it is in this ignorance, that we have established an artificially forward arrow of time and force the entropy to increase monotonically with time. Nevertheless, time-symmetric reversibility is not usually necessary in kinetics and only theoretically so in thermodynamics. Combining the above yields

$$P(f_1, f_2) = \frac{p_m}{m_c} f(\mathbf{r}, \mathbf{p}_1, t) f(\mathbf{r}, \mathbf{p}_2, t) d\mathbf{r} d\mathbf{p}_1 d\mathbf{p}_2 d\sigma_c dt. \tag{3.8}$$

Hence, the total rate of particles being lost to a small differential volume element in phase space $d\mathbf{r}d\mathbf{p}_1$ with $\int d\mathbf{p}_2$ target particle momentum space is:

$$Q_-(\mathbf{r}, \mathbf{p}_1, t) d\mathbf{r} d\mathbf{p}_1 dt = N dt d\mathbf{r} d\mathbf{p}_1 \int d\sigma_c \int \frac{p_m}{m} f(\mathbf{r}, \mathbf{p}_1, t) f(\mathbf{r}, \mathbf{p}_2, t) d\mathbf{p}_2. \tag{3.9}$$

In the right hand side of the above equation 3.9, $N \int p_m/m f(\mathbf{r}, \mathbf{p}_2, t) d\mathbf{p}_2 \int d\sigma_c$ is the frequency of all collisions of particles of momentum $\mathbf{p}_1$ with any other momenta, and thus the inverse of the mean free time $(1/t_c)$. It is only dependent on the magnitude of the momentum (p) of the incident particles. Thus,

$$Q_-(\mathbf{r}, \mathbf{p}, t) = \frac{f(\mathbf{r}, \mathbf{p}, t)}{t_c(p)}. \tag{3.10}$$

The Q_+ term is a little more elusive since it accounts for all the particles that scatter into the region of interest, and thus we know nothing of their original momenta. However, for states converging to equilibrium at a quasistatic pace or for situations near equilibrium, it can not be too far from Q_-. In fact, it is the term which ensures as time goes on to infinity $(t \to \infty)$, that

the distribution approaches the equilibrium distribution ($f(\mathbf{r},\mathbf{p},t) \to f_0(\mathbf{r},\mathbf{p})$). Hence,

$$Q_+(\mathbf{r},\mathbf{p},t) \simeq \frac{f_0(\mathbf{r},\mathbf{p})}{t_c(p)}. \tag{3.11}$$

The Boltzmann equation 3.6, combined with the explicit form of the collision term $Q(\mathbf{r},\mathbf{p},t) = Q_+(\mathbf{r},\mathbf{p},t) - Q_-(\mathbf{r},\mathbf{p},t)$ then becomes:

$$\left(\frac{\partial f}{\partial t}\right) + \nabla_{\mathbf{r}} f \cdot \frac{\mathbf{p}}{m} - \nabla_{\mathbf{p}} f \cdot \nabla_{\mathbf{r}} \phi \simeq \frac{1}{t_c}\left[f_0(\mathbf{r},\mathbf{p}) - f(\mathbf{r},\mathbf{p},t)\right]. \tag{3.12}$$

In the above, we have substituted every term for which we can obtain an explicit expression for. The term t_c is also frequently called the relaxation time and gauges the time scale at which particles with magnitude of momentum p approaches equilibrium. One can deduce that as p becomes larger, the target particle suffers more collisions in a given time interval, and thus relaxes preferentially faster to equilibrium conditions.

As the system described by the Boltzmann equation locally relaxes toward its equilibrium state, the collision term contributes less and less. Eventually, $\lim_{t\to+\infty} Q(\mathbf{r},\mathbf{p},t) \simeq 0$. This is to be expected, since at equilibrium, the contribution into a particular differential volume in phase space (or else) should equal to the contribution outwards from the differential volume, minus some small fluctuations, of course. The fluctuations are small enough locally so that effectively, at a macroscopic scale in local equilibrium conditions, f does not depend on t, and thus $\frac{\partial f}{\partial t} \simeq 0$ as well.

Thus, at equilibrium, the Boltzmann equation becomes:

$$\nabla_{\mathbf{r}} f \cdot \frac{\mathbf{p}}{m} - \nabla_{\mathbf{p}} f \cdot \nabla_{\mathbf{r}} \phi \simeq 0, \tag{3.13}$$

which can be decoupled in spatial and momentum coordinates if $f(\mathbf{r},\mathbf{p}) = f_r(\mathbf{r}) f_p(\mathbf{p})$. $f_p(\mathbf{p}) = (2\pi mkT)^{-3/2} e^{-p^2/2mkT}$ is the Maxwell distribution and $\nabla_{\mathbf{p}} f = -(\mathbf{p}/mkT) f_r f_p$.

Substitution of the above back into equation 3.13 produces:

$$\nabla_{\mathbf{r}} f_r = -(f_r/kT)\nabla_{\mathbf{r}}\phi. \tag{3.14}$$

Thus the solution to this equation is an exponential of the form $f_r(\mathbf{r}) = Be^{-\phi/kT}$ where B is a constant of normalization. Multiplying this and the 3-dimensional Maxwell distribution

$f_p(\mathbf{p})$ produces

$$f(\mathbf{r},\mathbf{p}) = \frac{B}{(2\pi mkT)^{3/2}} e^{\left[-\frac{1}{kT}\left(\frac{p^2}{2m}+\phi(\mathbf{r})\right)\right]}, \tag{3.15}$$

which is known as the Maxwell-Boltzmann distribution. The solution can be interpreted as stating that the equilibrium probability to find a particle at $(\mathbf{r},\mathbf{p})$ is determined by its total energy $(p^2/2m)+\phi(\mathbf{r}) = H(\mathbf{r},\mathbf{p})$ where $H(\mathbf{r},\mathbf{p})$ is the sum of total kinetic and potential energy in phase space, and thus the phase space Hamiltonian of the particle. The larger the value of H, the less probable that particles with that energy level will be found in the system. The smaller the absolute temperature T, the more influence the H will exert over the distribution of particles. The scalar factor B is adjusted so that the integral of $f(\mathbf{r},\mathbf{p})$ over its entire domain is unity. Notice the prior derivations all consider a 3-dimensional space, where the normalization factor in the Maxwell distribution in front of the exponential has an exponent of $-3/2$. In general, the derivation is valid for D-dimensional space (or phase space) and the exponent becomes $-D/2$ instead.

3.3.3 The Lattice Boltzmann Equation

To obtain the Lattice Boltzmann equation from the Boltzmann equation, we first recognize the left hand side of the Boltzmann equation 3.12 as the 1st order expanded total time derivative of $f(\mathbf{r},\mathbf{p},t)$, i.e.,

$$\frac{f[\mathbf{r}+(\mathbf{p}/m)dt, \mathbf{p}-(\nabla_{\mathbf{r}}\phi)dt, t+dt] - f[\mathbf{r},\mathbf{p},t]}{dt} \simeq \left(\frac{\partial f}{\partial t}\right) + \nabla_{\mathbf{r}} f \cdot \frac{\mathbf{p}}{m} - \nabla_{\mathbf{p}} f \cdot \nabla_{\mathbf{r}}\phi. \tag{3.16}$$

We can rewrite the entire Boltzmann equation in a more discretization friendly manner

$$f[\mathbf{r}+(\mathbf{p}/m)dt, \mathbf{p}-(\nabla_{\mathbf{r}}\phi)dt, t+dt] - f[\mathbf{r},\mathbf{p},t] \simeq \frac{1}{t_c}\left[f_0(\mathbf{r},\mathbf{p}) - f(\mathbf{r},\mathbf{p},t)\right] dt. \tag{3.17}$$

In computer simulations, the lattice is conceptually a large multi-dimensional matrix which has no intrinsic spatial or temporal units. Hence, all units associated with spatial and temporal measures have to be scaled to the interest of the simulator and usage at hand. We begin to non-

dimensionalize the equation and drop the momentum coordinate term since we do not have an explicit form of ϕ. With an explicit form of ϕ, we can non-dimensionalize the energy units and thus obtain a non-dimensionalization of the momentum term. To also prepare for the eventual Lattice Boltzmann model we will introduce in more specific detail, we also rewrite the equation to transform it from the phase space to Euclidean space where $\mathbf{r} \rightarrow \mathbf{x}'$ and $\mathbf{p}/m \rightarrow \mathbf{v}'$. This sequence of modifications leads us to

$$f(\mathbf{x}' + \mathbf{v}'dt', t' + dt') - f(\mathbf{x}', t') \simeq \frac{1}{t'_c}[f_0(\mathbf{x}') - f(\mathbf{x}', t')]dt', \tag{3.18}$$

where a prime $'$ is used to designate quantities which are dimensionalized where both dimensionalized and un-dimensionalized quantities exist with the same symbol. Next we introduce a characteristic length scale L and a velocity scale U. With these new scaling factors, various parameters of the equation above becomes

$$\begin{aligned} \mathbf{x} &= \frac{\mathbf{x}'}{L}; \\ \mathbf{v} &= \frac{\mathbf{v}'}{U}; \\ t &= \frac{t'U}{L}; \\ t_c &= \frac{t'_c U}{L}; \\ dt &= \frac{U}{L}dt'. \end{aligned} \tag{3.19}$$

Substituting the set of equations 3.19 into equation 3.18 yields the non-dimensionalized form

$$f(\mathbf{x} + \mathbf{v}dt, t + dt) - f(\mathbf{x}, t) \simeq \frac{1}{t_c}[f_0(\mathbf{x}) - f(\mathbf{x}, t)]dt. \tag{3.20}$$

To proceed to discretization of the above equation, we allow a few finite quantizations in both spatial and temporal approximations. The quantized equivalents of the continuous variables are shown below

$$\begin{aligned} \mathbf{x} &\rightarrow \mathbf{x}_i; \\ \mathbf{v} &\rightarrow \mathbf{v}_i; \\ t &\rightarrow t_j; \\ dt &\rightarrow \Delta t, \end{aligned} \tag{3.21}$$

where indices numerate over the levels of quantization. For time, the index j can theoretically extend to infinity. With these substitutions, the equation becomes

$$f(\mathbf{x}_i + \mathbf{v}_i \Delta t, t_j + \Delta t) - f(\mathbf{x}_i, t_j) \simeq \frac{\Delta t}{t_c}[f_0(\mathbf{x}_i) - f(\mathbf{x}_i, t_j)]. \tag{3.22}$$

In the above, the term $\Delta t/t_c$ is the ratio of time step size in the simulation to the time between collisions in the physical system being simulated. Since the time step is fixed but t_c is a continuous spectrum, t_c is usually approximated by a few discrete values (usually one). Therefore, a discretized version denoted by τ is used in the final equation. Since relaxation times can be very short (in less viscous systems) or very long (in very viscous systems), and its values sweep a large range over the positive real numbers to infinity, the inverse of τ is usually used in coding for convenience of parametrization. Therefore $\omega = \Delta t/\tau$ where ω is the lattice collision frequency. Approximations to t_c leading to realistic values of τ (and subsequently ω) will be discussed shortly as we delve into the specifics of LBMs'.

If in the Lattice Boltzmann equation we combine indices i and drop a few implied terms in the brackets and use the relaxation parameter ω, we can reduce this to a much cleaner and often used format:

$$f_i(t_{j+1}) \simeq f_i(t_j) + \omega[f_i^0 - f_i(t_j)]. \tag{3.23}$$

In the above equation, the term f_i^0 refers to the equilibrium distribution function f in the direction i. It is important to set proper lattice spacing to time such that $\Delta \mathbf{x}_i/\Delta t = \mathbf{v}_i$ where the lattice spacing is $\Delta \mathbf{x}_i = \mathbf{x}_{i+1} - \mathbf{x}_i$ to preserve proper dynamics.

The Lattice Boltzmann equation can be interpreted heuristically as follows: the total kinetic

evolution of some microscopic system can be separated into two aspects - spatial and temporal, the spatial portion at every time step is contributed by the distribution function directions which "stream" there, plus the local temporal relaxation towards equilibrium via "collision". The spatial evolution step (which is non-local) is carried out explicitly independently of the temporal evolution (which is local). From now on, we shall refer to the spatial evolution as streaming, and the temporal evolution as collision. This is a characteristic shared/inherited with LGM. In fact, there is a direct correspondence with the governing lattice gas equation to the lattice Boltzmann equation, however that will not be delved into here.

3.3.4 A Functional Lattice Boltzmann Model

To construct a working LBM, a few choices must be made regarding the type of lattice, the dimensionality and number of quantizations of velocity, the collision operator(s) and the scaling factors.

In the Lattice Boltzmann equation, the position parameter is usually a vector of two dimensions or higher, but usually no more than three. The quantization of velocity however is much more diverse. Typically, the greater number of discrete velocities, the more accurate the simulation. However, additional velocities introduce extra memory usage during run time and become particularly significant at higher lattice dimensions. Regardless of choice, lattices used for LBM carry over the nomenclature from LGM for designating the choice of lattice and dimensionality. DnQm is the commonly used nomenclature where n is the number of physical dimensions to be represented, and m is the number of direction vectors per node along which mass is streamed and stored. To better illustrate this, observe figure 3.7.

If the number of directions per node includes diagonal directions, then the lattice is a multi-speed model. In the above illustration, the D2Q9 model is a two speed model while the D2Q5 model is a single speed model. The rest position does not have a speed (actually, the speed is zero) and is not involved in the streaming process, only the collision process.

A proper, functional Lattice Boltzmann model first and foremost incorporates the crucial kinetic equation derived earlier. This equation as discussed earlier, governs the time and spatial evolution of our model. The one remaining bit of detail not discussed above, although alluded to in the derivation of the Boltzmann equation (the collision operator part) is that the mean time between collisions t_c in the Boltzmann equation, or τ in the Lattice Boltzmann equation,

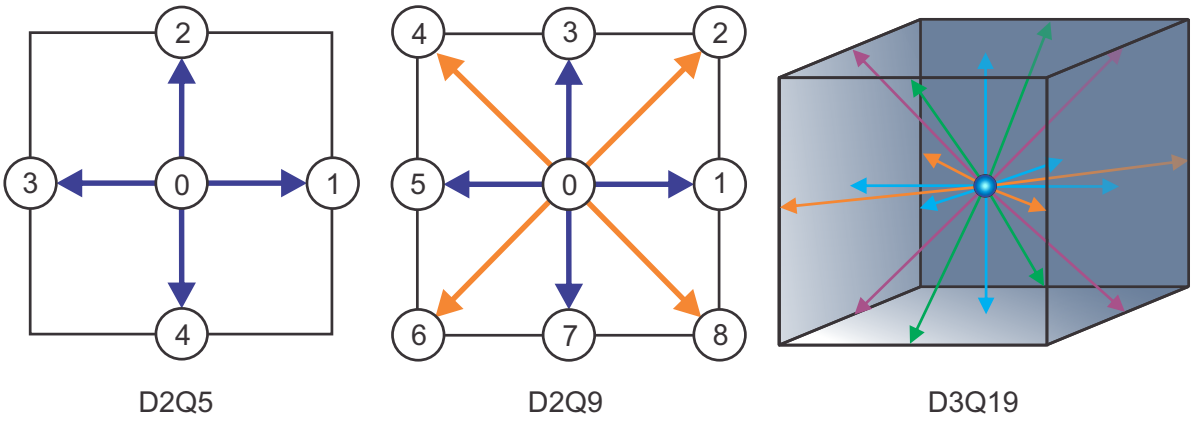


Figure 3.7 – Illustration of DnQm nomenclature and velocity vectors which stream mass. Two lattices above are in two dimensions, thus $D2$ while one has 5 vectors along which mass streams and the other has 9, thus $Q5$ and $Q9$ respectively. The 3-dimensional lattice shown here has 19 directions, thus D3Q19.

generally depends on the momentum distribution of the particles and thus is not a single valued scalar. We did not explicitly discretize this parameter when deriving the Lattice Boltzmann equations, for the reason that there are numerous ways to dissect the mean time between collisions, and all of them are equally valid pending the ultimate use of the model.

One such approximation to the continuum spectrum of relaxation times in the Boltzmann equation is of no surprise, the single relaxation time approximation whereby the time between collisions is replaced by its single, mean value: $\tau \approx \langle t_c(p) \rangle$. Our model of choice will make use of this type of collision operator, and thus, one can consider the ω term in the governing kinetic equation 3.23 to be of this single valued scalar. Also named after its inventors in abbreviation, the single mean valued relaxation time collision operator is known as the BGK (Bhatnagar-Gross-Krook) collision operator (Bhatnagar et al., 1954). It was also simultaneously independently proposed by Welander (Welander, 1954). Lattice Boltzmann models employing this type of collision operator are sometimes referred to as Lattice BGK models, or LBGK. In fact, we are not done with the adoption of the collision operator(s). It should be noted that the choice of τ directly affects the stability of the LBM since in reality, t_c is dictated by the dynamics of the system while the choice of τ is much more arbitrary. There are no entropic principles (such as the H-theorem, Sterling and Chen (1996)) which govern the behaviour of a LBM (BGK or otherwise), and therefore, LBMs are subject to instabilities. For a discussion on how τ shall be chosen to avoid instabilities, see the chapter on model specifics (4.1). For an

illustration on how the collision process works, see figure 3.8.

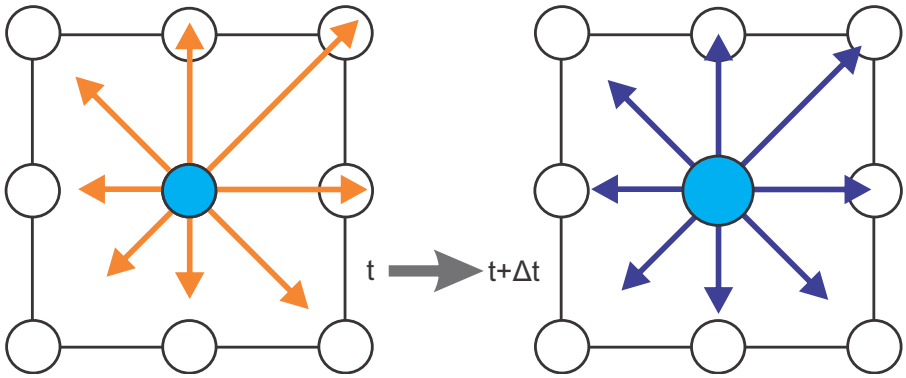


Figure 3.8 – Collision process illustrated for the D2Q9 lattice. The distributions after the collision process at time $t + \Delta t$ are adjusted by the collision operator. The amount of adjustment in each vector depends on the amount of mass in that vector at time t and the value of $\tau, \Delta t$. The adjust is made towards local equilibrium such that vectors with over-equilibrium quantities of mass are adjusted downward, weighted by τ, and vice versa.

Secondly, a choice of the lattice type, dimensionality and number of speeds is required. In the early days of fluid dynamical simulations when memory and processing power were great constraints to the simulation, many choices were not made lightly. The choice of a lattice type, for example, reveals that aspect, where the grids originally popular for fluid simulations were hexagonal (such as the D2Q6/7 models, also know as the various FHP models named after its creators Frisch, Hasslacher and Pomeau) to reduce memory usage since it is the lowest symmetry grid with sufficient symmetry to produce isotropic behaviour for lattice tensors up to 4th rank (unlike the earlier HPP models named after its creators Hardy, de Pazzis and Pomeau or D2Q4 models) (Wolf-Gladrow, 2005a) but without the extra memory needs of the higher m numbered DnQm models. However, it was not possible to extend the hexagonal models to three dimensions as the only regular polytope (a geometric object with flat sides in any number of dimensions, in two dimensions, the word polygon is more frequently used) that spans three dimensions with no voids is the cube, hexagonal lattices eventually gave way to DnQm types of models where the cube is the underlying symmetry. One additional caveat of non-square model meshes is the earlier onset of numerical instability (Wolf-Gladrow, 2005a). The progress of computational power and the adoption of LBM and three dimensional simulations then cemented this transition. We will only refer to DnQm types of cube based models and all subsequent discussions will be made accordingly.

In a multi-speed model, the linear Euclidean distance between two adjacent nodes that are

not diagonally related, such as directions $(1, 3, 5, 7)$ in D2Q9 grid illustration in figure 3.7 (nd) has a unit distance in the model, say Δx; while distances diagonally, such as directions $(2, 4, 6, 8)$ in the D2Q9 illustration in figure 3.7 (d) nodes has a distance $\sqrt{2}\Delta x$. Since in one unit time step Δt, the streaming distributions travel to their nearest nodes following their paths (see figure 3.9), the diagonal directions must necessarily possess a faster linear velocity to complete their streaming process in the same time step as the shorter distanced nodes. To rephrase this in the point of view of velocities then, if $\mathbf{c}_i$ are the velocities in direction i of a given multi-speed grid with a fixed, dimensionalized scalar speed $\pm c$, then the grid velocities can be given as $\mathbf{c}_i$ multiplied by the eigenvector $\mathbf{e}^i \equiv (e^i_x, e^i_y)$ of direction i to yield

$$\begin{aligned}
\mathbf{c}_0 &= c(e^0_x, e^0_y) = c(0,0) = (0,0);\\
\mathbf{c}_1 &= c(e^1_x, e^1_y) = c(+1,0) = (c,0);\\
\mathbf{c}_2 &= c(e^2_x, e^2_y) = c(+1,+1) = (c,c);\\
\mathbf{c}_3 &= c(e^3_x, e^3_y) = c(0,+1) = (0,c);\\
\mathbf{c}_4 &= c(e^4_x, e^4_y) = c(-1,+1) = (-c,c);\\
\mathbf{c}_5 &= c(e^5_x, e^5_y) = c(-1,0) = (-c,0);\\
\mathbf{c}_6 &= c(e^6_x, e^6_y) = c(-1,-1) = (-c,-c);\\
\mathbf{c}_7 &= c(e^7_x, e^7_y) = c(0,-1) = (0,-c);\\
\mathbf{c}_8 &= c(e^8_x, e^8_y) = c(+1,-1) = (c,-c).
\end{aligned} \tag{3.24}$$

Equivalently, for a model with only one diagonal direction (such as D2Q9), the equations can be put into a more compact form

$$\begin{aligned}
\mathbf{c}_0 &= (0,0),\\
\mathbf{c}_{nd} &= (\pm c, 0), (0, \pm c),\\
\mathbf{c}_d &= (\pm c, \pm c).
\end{aligned} \tag{3.25}$$

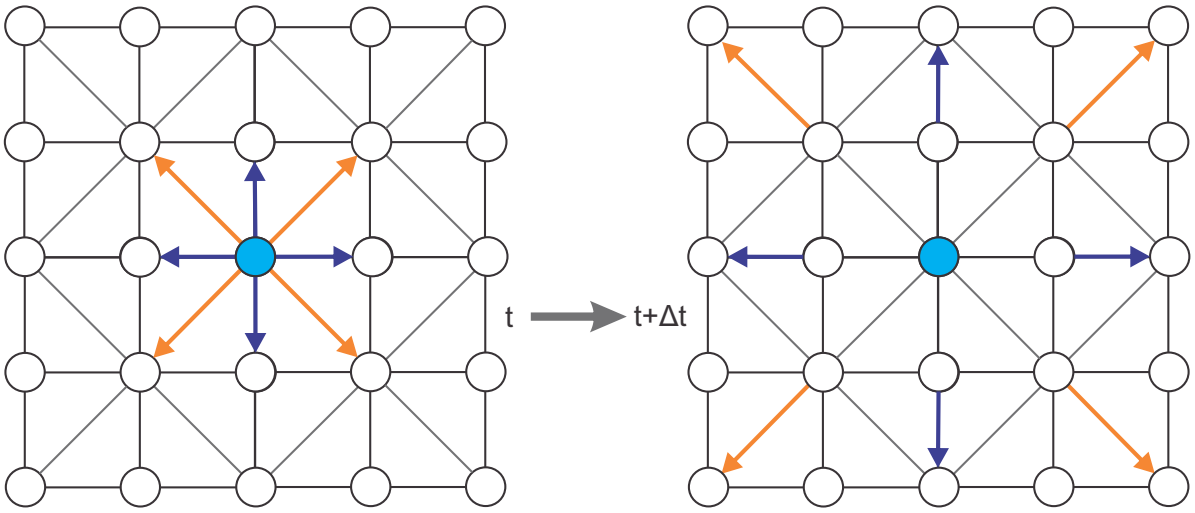


Figure 3.9 – Streaming process illustrated for one cell in the D2Q9 lattice. The cell in the center prior to streaming at time t contains all 8 non-zero velocity vectors. During the streaming process, the vectors in each direction i are moved one $\mathbf{c}_i \times \Delta t$ distance away, into the adjoining neighbour cells. For clarity, the neighbouring cells' vectors are not illustrated, but the streaming effect is the same.

Knowing the above, the total mass density and velocity per site can be calculated as

$$\begin{aligned} \rho(\mathbf{x},t) &= \sum_i f_i(\mathbf{x},t), \\ \mathbf{u}(\mathbf{x},t) &= \frac{\sum_i \mathbf{c}_i f_i(\mathbf{x},t)}{\rho(\mathbf{x},t)}. \end{aligned} \tag{3.26}$$

An equilibrium distribution function f_i^0 can then be constructed via approximations and discretization in dimensional form (non-dimensionalization will occur later when lattice velocity $\mathbf{c}$ will have a functional form in terms of kT, m) from the Maxwell distribution $F_v(\mathbf{v}_f) = (m/2\pi kT)^{D/2} e^{-m{v_f}^2/2kT}$ (introduced earlier but in momentum space instead of velocity space, the two differs by substitution $\mathbf{p} = \mathbf{v}_f m$ and normalization factor, D is the number of spatial dimensions the particles distribute over).

The microscopic fluid velocity $\mathbf{v}_l$ on the lattice has the mean value $\langle \mathbf{v}_l \rangle \equiv \mathbf{u}$ [macroscopic fluid velocity, same as in equation 3.26]. Thus a fluctuations term $\mathbf{v}_f$ about the mean (thermally distributed intrinsic velocity of underlying particles) is $\mathbf{v}_f = \mathbf{v}_l - \mathbf{u}$. For a fluid density of ρ_0, our D-dimensional Maxwell distribution can be rewritten as

$$
\begin{aligned}
F(\mathbf{v}_f) &= \rho_0 (m/2\pi kT)^{D/2} e^{-m(\mathbf{v}_l - \mathbf{u})^2/2kT} \\
&= \rho_0 (m/2\pi kT)^{D/2} e^{-m(\mathbf{v}_l^2 - 2\mathbf{v}_l \cdot \mathbf{u} + \mathbf{u}^2)/2kT}
\end{aligned} \tag{3.27}
$$

Separating the velocities in the exponential into two parts with a part containing the macroscopic velocity and the part containing only microscopic lattice direction velocity and labelling them yields

$$
\begin{aligned}
F(\mathbf{v}_f) &= F_{\mathbf{v}_l} \times F_{\mathbf{u}} && (3.28) \\
F_{\mathbf{v}_l} &= \rho_0 (m/2\pi kT)^{D/2} e^{-m(\mathbf{v}_l^2)/2kT} && (3.29) \\
F_{\mathbf{u}} &= (m/2\pi kT)^{D/2} e^{-m(-2\mathbf{v}_l \cdot \mathbf{u} + \mathbf{u}^2)/2kT} && (3.30)
\end{aligned}
$$

To obtain the equilibrium distribution function, we assume the macroscopic velocity of the fluid $\mathbf{u}$ locally at each site is much smaller than lattice grid velocities $\mathbf{v}_l$. Thus, at local equilibrium, $\mathbf{u}$ becomes a perturbation about $\mathbf{v}_l$ and thus $\mathbf{v}_f - \mathbf{v}_l = -\mathbf{u} \approx 0$. Expanding $F(\mathbf{v}_f)$ around $\mathbf{v}_l$ yields an approximate equilibrium distribution function valid for low flow velocities.

$$
\begin{aligned}
F(\mathbf{v}_f) &= F(\mathbf{v}_l) + (\mathbf{u} \cdot \nabla_{\mathbf{u}}) F(\mathbf{v}_f)|_{\mathbf{v}_f = \mathbf{v}_l} + \frac{1}{2} \mathbf{u} \cdot [\mathbf{u} \cdot \nabla_u (\nabla_u F(\mathbf{v}_f))] \,|_{\mathbf{v}_f = \mathbf{v}_l} + O(\mathbf{u}^3) \\
&= F_{vl} \left[1 + \frac{m}{kT} (\mathbf{u} \cdot \mathbf{v}_l) + \left(\frac{m}{kT} \right)^2 \frac{(\mathbf{u} \cdot \mathbf{v}_l)^2}{2} - \frac{m}{kT} \frac{\mathbf{u}^2}{2} \right],
\end{aligned} \tag{3.31}
$$

which is the general form of the equilibrium distribution function. To obtain the coefficient F_{vl}, we consider if the macroscopic velocity $\mathbf{u}$ is zero, then the only part that remains is the coefficient F_{vl}, and thus it is appropriately the zero velocity distribution function or sometimes just "weights". These dictate the relative proportions of mass being streamed at various velocities on the lattice when the macroscopic velocity is not present.

To get the complete discrete equilibrium distribution f^0, a discrete set of zero velocity distribution function $f_i^0(0)$ must replace F_{vl} (discretization of the velocities in the equilibrium distribution function), and the lattice velocity $\mathbf{c}$ must replace $\mathbf{v}_l$. For this discretization to

be realistic, they must obey in some way, the form of our established equilibrium distribution function (Maxwellian). Thus the zero velocity equilibrium distribution function must match in the shape to that of the Maxwell distribution F_{vl}. This can be accomplished in a few different ways, but here we will show the moments method to obtain one of many sets of solutions. For simplicity, we rename the $f_i^0(0)$ into w_i after the "weights" interpretation of this term in the expansion.

At equilibrium, the distribution functions W_i in D2Q9 lattice satisfy, for each scalar v_l, $-\infty < v_l < \infty$ in vector $\mathbf{v}_l$

$$\sum_i W_i = \int_{\mathbf{v}_l} F_{vl} d\mathbf{v}_l = \rho_0 \tag{3.32}$$

$$\sum_i c_{i\alpha} c_{i\beta} W_i = \int_{\mathbf{v}_l} v_\alpha v_\beta F_{vl} d\mathbf{v}_l = \frac{kT}{m} \delta_{\alpha\beta} \rho_0 \tag{3.33}$$

$$\sum_i c_{i\alpha} c_{i\beta} c_{i\delta} c_{i\gamma} W_i = \int_{\mathbf{v}_l} v_\alpha v_\beta v_\delta v_\gamma F_{vl} d\mathbf{v}_l = \left(\frac{kT}{m}\right)^2 (\delta_{\alpha\beta}\delta_{\gamma\delta} + \delta_{\alpha\gamma}\delta_{\beta\delta} + \delta_{\alpha\delta}\delta_{\beta\gamma}) \rho_0 \tag{3.34}$$

where $\delta_{\alpha\beta} = 1$ if $\alpha = \beta$ else $\delta_{\alpha\beta} = 0$ is the Kronecker delta. All odd order moments (not shown above) vanish due to the symmetry of the Maxwell distribution (and the lattice) around $\mathbf{v} = 0$.

Solving the above equations yields the normalized zero-velocity equilibrium distribution of the D2Q9 lattice. They are listed in equation table 3.37 in correspondence with the D2Q9 lattice illustrated in figure 3.7

$$W_0 = \frac{4}{9}\rho_0; \tag{3.35}$$

$$W_{1,3,5,7} = \frac{1}{9}\rho_0; \tag{3.36}$$

$$W_{2,4,6,8} = \frac{1}{36}\rho_0. \tag{3.37}$$

In general, the real density ρ_0 may be scaled and non-dimensionalized to yield a lattice density $\rho*$. In terms of this lattice density, the non-dimensionalized weights w_i can be written as

$$w_0 = \frac{4}{9}\rho*; \tag{3.38}$$

$$w_{1,3,5,7} = \frac{1}{9}\rho*; \tag{3.39}$$

$$w_{2,4,6,8} = \frac{1}{36}\rho*. \tag{3.40}$$

The lattice speed c which is also known as the speed of sound on the lattice (the speed at which the information is propagated on the lattice) is then

$$c = \sqrt{\frac{3kT}{m}}. \tag{3.41}$$

The characteristic velocity scale introduced earlier U for the generalized non-dimensionalization is typically taken to be equals to c.

Since for our lattice, the microscopic velocity is $c_i = c \times \mathbf{e}^i$ (see equation set 3.24), substituting our choice of c_i from above into equation 3.31 and replacing F_{vl} by our discrete zero velocity distribution function w_i, the entire discrete, dimensionless equilibrium distribution function is

$$f_i^0 = w_i \left[1 + \frac{3\mathbf{u}\cdot\mathbf{c}_i}{c^2} + \frac{9(\mathbf{u}\cdot\mathbf{c}_i)^2}{2c^4} - \frac{3\mathbf{u}^2}{2c^2}\right]. \tag{3.42}$$

The entire collision operator is usually denoted by $\mathbf{\Omega} = (\Omega_1, \Omega_2, ..., \Omega_m)$ where m is the number of lattice directions. For direction i, the kinetic equation can be written in terms of Ω_i as

$$\Omega_i[f_i(t_j)] = \omega[f_i^0 - f_i(t_j)]. \tag{3.43}$$

The entire lattice Boltzmann kinetic equation can be rewritten, one last time as

$$f_i(t_{j+1}) \simeq f_i(t_j) + \Omega_i[f_i(t_j)],$$

To ensure linear stability of the model, the grid velocity (speed of sound in the grid) must be chosen such that the fastest advection speed is less than the grid velocity ($\mathbf{c} \gg \mathbf{u}$ as per our assumption to obtain equilibrium distribution function). I.e., the fastest advection speed in the model must be subsonic. Without this constraint, for macroscopic velocities large enough on the

lattice, one can create negative mass populations, and the simulation quickly becomes unstable. Instabilities in the LBM are largely contributed by the choice of the equilibrium distribution function. Other choices can be made when the simulation necessitates them. For our purposes, the above choice is sufficient.

We are now armed with the most crucial ingredients of a standard LBM with a kinetic equation, a specific lattice of choice (D2Q9) and an equilibrium distribution which conforms with a known equilibrium distribution on the D2Q9 lattice. Note that the coefficients of the equilibrium distribution functions (weights) are not unique. There is a continuum set of real numbers located in an island of stability of the LBM phase space. In fact, there are several such islands of stability yielding several sets of equilibrium distribution weights.

One can retrieve the Navier-Stokes equations from both the Boltzmann and the Lattice Boltzmann equations by expanding in two time scales, a diffusive time-scale and a convective time-scale and an additional assumption that the diffusive time-scale is much smaller than the convective time-scale. We will not delve into this here, but just state some of the crucial results of the expansion. In particular, the kinematic shear viscosity (not dimensionless) for the D2Q9 lattice can be shown to be (eg., Wolf-Gladrow (2005a))

$$\nu = \frac{kT}{m}\left(\frac{1}{\omega} - \frac{1}{2}\right)\Delta t' = \frac{c^2}{3}\left(\frac{1}{\omega} - \frac{1}{2}\right)\Delta t' = \frac{c^2}{3}\left(\tau - \frac{\Delta t'}{2}\right). \tag{3.44}$$

ν is related to lattice viscosity, which is dimensionless. The specifics on the parameters will be dealt with in the next chapter.

Lastly, the boundaries of the simulation area are solid structures which are representations of the containers of a liquid. The conditions for streaming for cells touching the boundaries are different in that some of the streaming functions are reflected in the case of bounce-back boundary conditions, or simply reset or altered in the case of other types of boundary conditions. For our purposes, a simple bounce-back boundary (no-slip type) will suffice. In particular, the no-slip bounce back boundary condition is employed. In this process, vectors which stream mass into the boundary are reflected 180° after one time step. The vectors which stream into the boundary are bounced back, such that at any given time, there are equal amounts of particles moving in directions into and out of the boundary. Notice that in this scheme, the velocity of the liquid parallel to the boundary surface at the boundary is equal to zero. This process is

illustrated in figure 3.10.

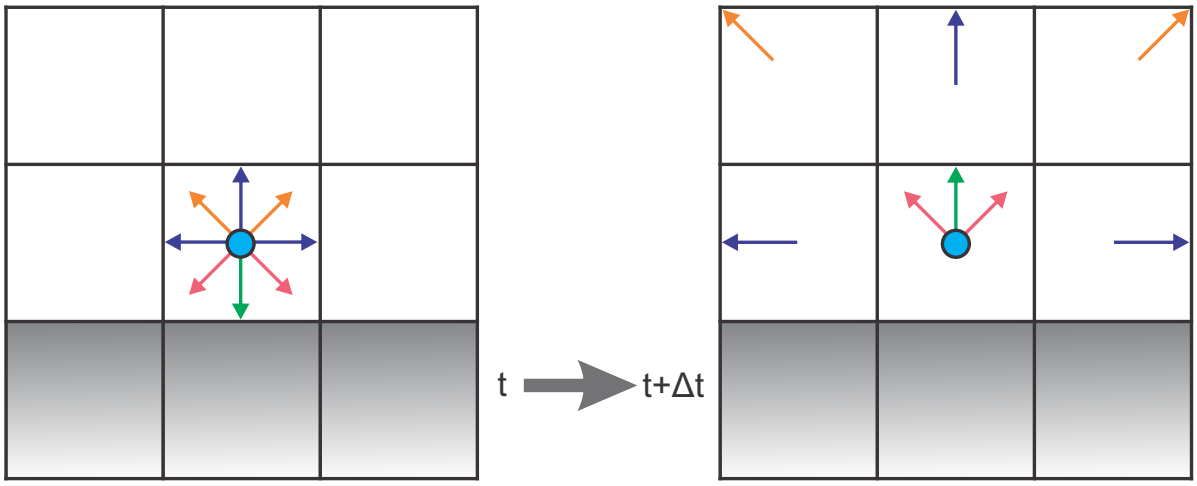


Figure 3.10 – No-slip bounce-back boundary conditions illustrated for the D2Q9 lattice. The boundary is formed by solid nodes occupying the bottom 1/3rd of the lattice cells as shown in shaded-gradient. Notice that mass vectors which stream into adjacent cells of non-boundary types are streamed normally while mass vectors which stream into the boundary are reflected. The pink vectors (diagonal vectors pointing toward the boundary before streaming) bounce back, finishing in the same cell where they originated (diagonal vectors pointing away from the boundary after streaming). In a free-slip bounce back boundary, the pink vectors streams into the left and right neighbouring cells of the illustrated cell in addition to direction reversal.

The boundary condition is the last necessary ingredient for the operation of our standard LBM. This model can be tested and verified to ensure numerical implementation correctness and accuracy. One such test uses the diffusive properties of the model. For a brief introduction to the theoretical aspects of diffusion, analytical solutions to the diffusion problem posed, comparison of analytical to experimental results, see appendix A.

3.3.5 Immiscible Lattice Boltzmann Model

In addition to the normal ingredients of a LBM, our model requires a few more modifications. One such modification is in the liquid state, simulated by the LBM. Whereas the standard LBM simulates identical particles (of a fluid or gas) and is sufficient for isothermal, single chemical component situations, it is insufficient for our scenario. As discussed earlier in the hypothesis section, the melt being simulated undergoes spontaneous phase separation near and below liquidus temperatures. Our model must be able to simulate this spontaneous phase separation. It can be accomplished with the use of a modified LBM - immiscible LBM (ILBM).

In general, the immiscible LBM simulates 2 or more types of chemical components which

exhibit a non-zero mutual surface tension. At the macroscopic level, the surface tension is a result of the free surface energy at the interface between two dissimilar continuous media (or of one and vacuum). It is a function of the surface structure due to the bonding on the microscopic level. A species of condensed matter tends to attract like surface structures and repel dissimilar ones as in the example of glass and water versus water and oil. In a melt, as temperature approaches that of the liquidus, numerous chemical bonds begin to establish, and short range order is created. The differences in short range order and bulk chemical properties serve to decompose the melt spontaneously into two (or more) phase domains in which one type (or more) somewhat molten equivalent modal compositions exist depending on the temperature. The lower the temperature, the stronger the tendency of these domains to acquire exact mineral stoichiometries. These regions repel unlike regions and enlarge by merging with neighbouring regions.

To distinguish fluids that resemble the plagioclase composition (in chemistry, physical properties and structural properties) from the mixture (more appropriately a mush) of other solids and fluids, we need a model with at least 2 types of fluids in that they differ in macroscopic properties. They both obey the kinetic equation as derived earlier in the general sense that they both stream and collide. In addition, to simulate the interaction between them, an additional collision step is required to simulate the effects of cross-fluid type collisions. Whereas the effect of the single fluid collision (single-phase) operators is to bring the local distribution closer to equilibrium by redistributing local distribution quantities, the effect of the cross-fluid collision (multi-phase) operator is to separate the mixture of two fluids at a given site based on surface tension.

If we denote the phases by k, the single-phase collision operator as $\Omega 1_i^k$ and the newly proposed multi-phase collision operator as $\Omega 2_i^k$, then the immiscible kinetic equation for LBM can be written as

$$f_i^k(t_{j+1}) = f_i^k(t_j) + \Omega 1_i^k + \Omega 2_i^k. \tag{3.45}$$

Since we want to encourage phase separation, the explicit form of $\Omega 2_i^k$ must preferentially redistribute grid mass quantities (mass) with the local phase gradient. So if the local gradient points in the direction of phase k, then the local mass of type k shall be assigned to the grid

directions that will distribute the mass to the direction of higher k concentrations, and the same for other species and so on. The collision operator must also obey the local conservation of mass and momentum ($\Omega 1_i^k$ already obeys this as per derivation), i.e.,

$$\sum_i \Omega 2_i^k = 0; \tag{3.46}$$

$$\sum_i \Omega 2_i^k \mathbf{c}_i = 0. \tag{3.47}$$

The operator $\Omega 2_i^k$ is composed of two parts $N1_i^k$ and $N2_i^k$. $N1_i^k$ preferentially assigns mass with the local phase gradient by adding mass to grid directions weighed by relative alignment to the gradient. The local concentration gradient can be approximated discretely. For the two-phase flow we are interested in, this can be

$$\mathbf{F}(\mathbf{x}) = \sum_{i=1}^{8} \mathbf{e}^i \sum_{j=0}^{8} \left[f_j^1(\mathbf{x} + \mathbf{e}^i) - f_i^2(\mathbf{x} + \mathbf{e}^i) \right], \tag{3.48}$$

which is an iterative sum of mass in all directions, over the nearest neighbour neighbourhood of the site in consideration, multiplied by the eigenvectors $\mathbf{e}^i$. With the gradient, a collision operator can be constructed to perturb the local mass to preferentially favour directions i with a good alignment to the gradient (i.e., maximize $|\mathbf{e}^i \cdot \mathbf{F}|$) to promote separation.

A collision operator of this type was proposed by Grunau et al. (1993). However, Reis and Phillips (2007) improved on the collision operator and were able to obtain a form of the multi-phase collision operator which produced the correct form of the continuum equations. We will only consider this particular form of the partial multi-phase collision operator given by

$$N1(f_i^k) = \frac{\alpha_k}{2} |\mathbf{F}| \left[w_i \frac{(\mathbf{e}^i \cdot \mathbf{F})^2}{|\mathbf{F}|^2} - B_i \right], i = 0, ..., 8. \tag{3.49}$$

The coefficient α_k is linearly proportional to the surface tension and the coefficients B_i are

given by

$$B_i = \begin{cases} -\frac{4}{27}\rho*, & i = 0, \\ \frac{2}{27}\rho*, & i = 1,3,5,7, \\ \frac{5}{108}\rho*, & i = 2,4,6,8. \end{cases} \tag{3.50}$$

The surface tension parameter α_k can be tuned to match real physical values. A discussion on this parameter and how it will be tuned will be given in the next chapter on model details.

This collision operator serves to add mass in directions with a component parallel to the concentration gradient and remove mass in directions with a component orthogonal to the concentration gradient. Since it distributes total mass, but does not distribute phases, an additional step is usually performed to distribute the phases among the directions so as to separate the phases along the interface (orthogonal to the concentration gradient).

A method originally proposed by Gunstensen et al (Gunstensen and Rothman, 1991) reassigns phase types by choosing grid directions closest to the concentration gradient, and preferentially filling them with the first phase (f_i^1 in the concentration gradient equation 3.48, and the remaining directions are then filled with the second phase in a two-phase system). This method has several drawbacks of which the most serious are lattice pinning, and relative complexity. An alternative proposed by Latva-Kokko and Rothman (Latva-Kokko and Rothman, 2005) allows a mixture of phases to be assigned to each direction and thus overcoming the lattice pinning of small bubbles associated with earlier methods and also greatly simplifying the numerical algorithm. If we define ρ_i^k as the mass density of species k in the direction i and ρ_k is the total mass density of a species k at a lattice site and abbreviate $N1_i^k \equiv N1(f_i^k)$ then the redistribution equation for two phases is given by

$$\begin{aligned} N2(f_i^1) &= \frac{\rho_1}{\rho_1+\rho_2}\left(N1_i^1 + N1_i^2\right) + \beta\frac{(\rho_1\rho_2)}{(\rho_1+\rho_2)^2}\Omega_i^1(f_i^1,\mathbf{0})\cos(\phi), \\ N2(f_i^2) &= \frac{\rho_2}{\rho_1+\rho_2}\left(N1_i^1 + N1_i^2\right) - \beta\frac{(\rho_1\rho_2)}{(\rho_1+\rho_2)^2}\Omega_i^2(f_i^2,\mathbf{0})\cos(\phi), \end{aligned} \tag{3.51}$$

where $\Omega_i^k(f_i^k,\mathbf{0})$ is the zero-velocity equilibrium distribution for species k and β is a parameter to gauge the completeness of the separation of phases at the interface (or interface thickness)

and can take value between 0 and 1. As β increases, the interface will become less diffuse. The angle ϕ is the angle between the local concentration gradient $\mathbf{F}$ and the grid direction c_i. This parameter can be tuned to reflect the characteristic distance of an interface between different chemically enriched domains in a melt. For discussions, refer to the next chapter on model details.

The combined operator $\mathbf{\Omega_2} \equiv N2[N1(\mathbf{f})]$ operates on the streaming mass $\mathbf{f}$ by applying $N1$ first, yielding $N1(\mathbf{f})$, then sequentially applying $N2$ to obtain the total $\mathbf{\Omega_2}(\mathbf{f})$.

These steps in $\mathbf{\Omega_2}$ operator are illustrated in the figure 3.11 showing the computation of the gradient, the $N1_i^k$ and $N2_i^k$ operators separately.

The cells of the lattice can be initialized with any combination of mass densities of the two phases, they need only obey that the sum of the two phases is equal to the nominal mass density $\rho*$ of the lattice. Therefore

$$\rho * (t = 0) = \rho_1(t = 0) + \rho_2(t = 0) = \sum_i \sum_k f_i^k. \tag{3.52}$$

In practice, the ratio of the phase 1 with respect to both phases defines a macroscopic enrichment parameter R which describes the bulk concentrations of phase 1 with respect to the total concentration of both species.

$$R \equiv \frac{\rho_1}{\rho*}. \tag{3.53}$$

It can be tuned to match that of the mixtures being simulated. This parameter will be discussed in the model details chapter.

An additional piece of algorithm is needed to deal with the mixture of the binary fluid and its effects on the viscosity parameters. Normally for a single component (phase, isochemical) flow, the viscosity is independent of the chemical composition at each point in the fluid. However, for multi-phase flows with different viscosities for each phase, the effective viscosity for the mixture maybe different at each point in the fluid due to differing concentrations of each phase at each point. Of course, other factors such as chemical reactions (polymerization, for example) can also alter the effective viscosity such that the viscosity as a function of space is not uniform. To deal with this problem during the collision process, two solutions are immediately apparent:

1) Use two collision operators and allow each fluid to reach its own equilibrium distribution in each cell using its own viscosity as defined in the single-phase case.

2) Use an interpolation scheme of any arbitrary order to approximate the local viscosity of the combined mixture and use one collision process to relax both phases and allowing the immiscible collision operator to discriminate and reassign phases.

Of course, if the two fluid components are not strongly mixing, or are not suspected of chemical reactions, solution (1) is quick and simple. For our purposes, we will have to adopt solution (2) and develop a simple linear interpolation scheme that assigns an effective viscosity at a cell based on parent phase viscosities weighed by the volume fraction of each phase.

First, we defined a parameter $\Phi = (\rho_1 - \rho_2)/(\rho_1 + \rho_2)$ where the ρ_k denotes the total volumetric fluid content of species k [$k = 1, 2$ in this case] in the cell under consideration. Then we define a linear interpolation scheme to compute the grid quantity which controls viscosity, ω_{avg}.

$$\omega_{avg} = 0.5 \times (1 + \Phi) \times \omega_1 + 0.5 \times (1 - \Phi) \times \omega_2, \tag{3.54}$$

so that the value of ω_{avg} smoothly and linearly approaches that of ω_k if the cell content is swayed toward more of species k. This approximation thus yields a new ω_{avg} which will be used in both the regular and the immiscible collision operators as the dimensionless relaxation frequency. A linear approximation is sufficient if the two fluids involved do not exhibit mixing reactions or interactions not hereby accounted for (chemical, electrical, magnetic, etc). In cases where mixing reactions or interactions occur, an exact scheme depending on the nature of the interaction is needed. However, in our case, we assume that any interactions between phases is far weaker than the surface tension component, and thus can be neglected.

The liquid state is thereby fully equipped for simulation of binary mixtures.

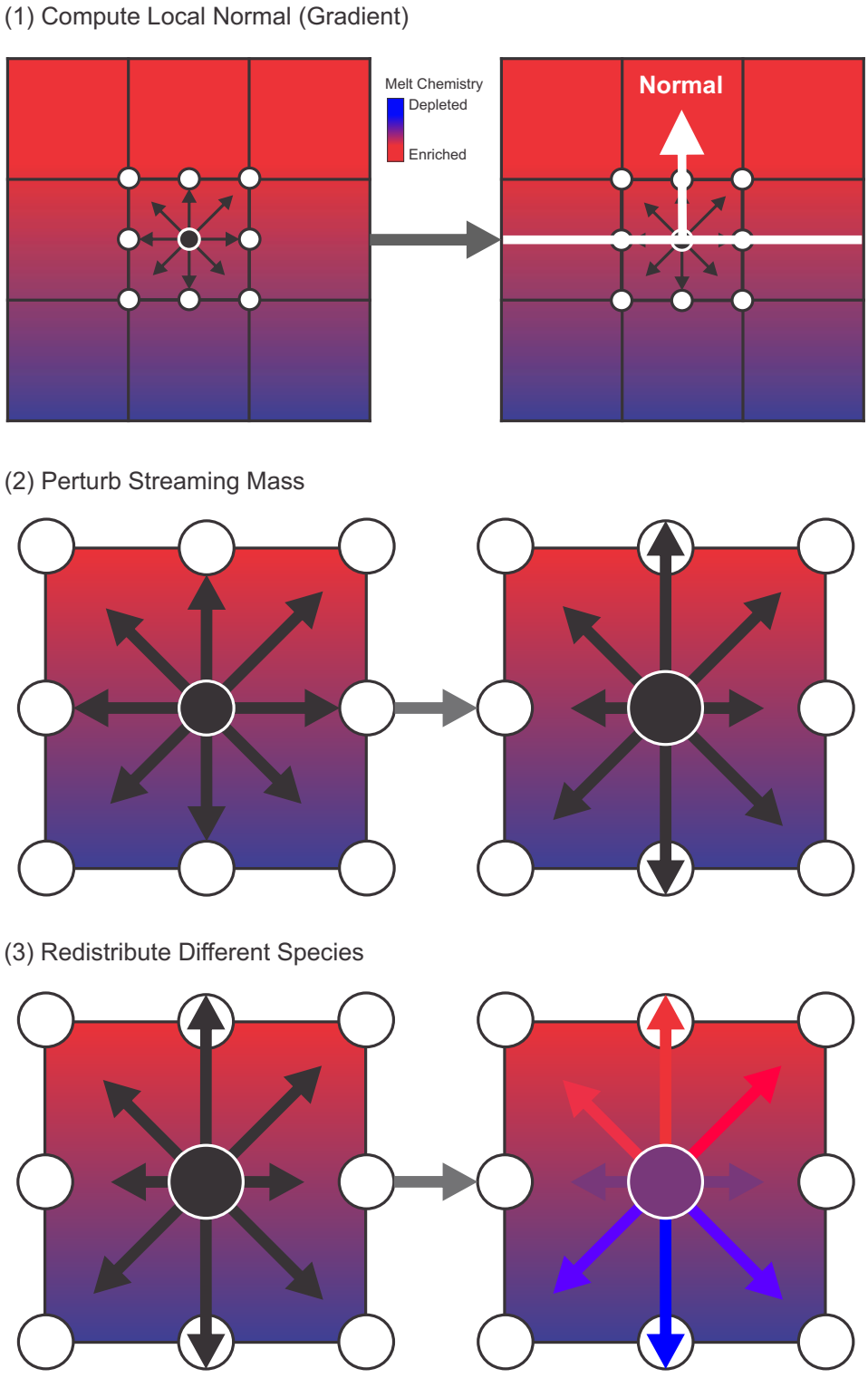


Figure 3.11 – $\Omega 2_i^k$ operator in 3 steps. Step 1 is the construction of the gradient vector (normal) at the site. Step 2 is perturbation of local populations. Step 3 is re-distributing different species preferentially to align with the gradient. In step 3, the coloured vectors illustrate the phase separation into like colours.

3.4 Stochastics

Stochastic modelling relies on random numbers (or more appropriately, pseudo-random numbers) from a random number generator to simulate a given macrostate by efficiently sampling underlying microstates in a fictitious manner without actually simulating the governing dynamics. It is a technique frequently used to simulate thermal behaviour such as phase transitions (i.e., Ising models). There are many different, well established techniques that uses stochastic modelling to explore the behaviours of given systems, such as Monte Carlo methods, Metropolis algorithms, Random Walk techniques and so on. We are interested in a stochastic model which can supplement the LBM in two distinct senses: the ability to focus on the microscopic since our LBM is macroscopic in scale, and the ability to simulate thermal behaviour (phase transitions) since our LBM is athermal.

While there are a large number of stochastic techniques for simulating a large range of topics, our needs are better satisfied if we construct an entire scheme based on some stochastic rules instead of using a pre-existing method since no one particular method fits our needs.

In our simulation, the LBM simulates the initial separation of the parent melt into plagioclase compatible and other mineral species via fractionation and surface tension. Coupled with this process is phase transition of plagioclase compatible portions of the melt into solid state plagioclase. This process must meet several requirements:

(1) rate of crystallization must exhibit a non-diffusively limited behaviour (linear rate) and be governed by the preferential enrichment of plagioclase in the melt,

(2) the individual crystals must have crystallographically controlled anisotropic growth rates,

(3) a scenario simulating the heterogeneous nucleation process and the subsequent epitaxial growth of new crystals on existing ones,

(4) some appropriate feedback into the liquid state (rejection of some latent heat of crystallization, surface tension changes in the vicinity of phase transition, etc).

Since the evidence of diffusively limited behaviours are lacking in the Rum plagioclase and we in fact observe the opposite (euhedral crystals, large crystals, very sparsely zoned plagioclase in thin sections), we know the textural pattern is not linked to diffusively limited aggregation.

Therefore, to avoid venturing into the diffusively limited regime of crystal growth, we need to simulate a melt environment that is likely both extremely well separated chemically and crystallized very slowly (at least for plagioclase).

Requirement (1) can be fulfilled by allowing either a long period of melt separation prior to crystallization such that the diffusivity of plagioclase in the melt does not effectively control the overall growth rate, or by choosing a reaction rate that is reasonably slow compared to the rate of diffusion in the melt. In practice, some other phase space considerations and simulation optimizations also dictate how this requirement will be fulfilled. This aspect will be discussed in the next chapter.

Requirement (2) attempts to simulate the preferential growth directions and elongation of structurally anisotropic crystals such as plagioclase. This requirement can be fulfilled by constructing a probability mass function which in every direction on the grid governs the probability of attachment (addition of new sites to an existing crystal). This is a natural extension of the typical isotropic attachment probabilities seen in LGM. This probability function modifies the existing probability of growth dictated by the concentration alone.

Requirement (3) stems from the need to allow new orientations of crystals to exist using the previous existing crystals as substrates from which to nucleate and grow. This allows us to change the direction of a growth front when the rate of crystallization of the existing crystal in the existing direction stalls. This requirement can be fulfilled in several ways, such as using some variety of anisotropic cluster-cluster dynamics (as in cluster-cluster aggregation, a more general form of diffusion limited aggregation) to simulate the composition of multiple crystal clusters. However, in a smoothly streaming LBM, the action of large discrete movements of clusters can be troublesome if the stochastic model is coupled to the LBM at the same resolution as the underlying LBM. The jittery movement can cause instabilities. A remedy of this situation frequently used in other aspects of fluid dynamics is to use sub-grids to locally refine in the vicinity of, or interpolate around, these larger objects to be moved and appropriately reduce the jittery, discrete movement behaviour. In our model, this would add additional complexity since multiple grids would have to be constructed, one for each site of local growth, and an additional parameter (set) would have to be introduced on the distribution size of clusters. In the spirit of the solution to requirement (2), we simply set two directions of anisotropy by extending the solution to requirement (2) and allow attachment coefficients at a growth site to

be varied depending on the underlying solid structure alignment.

Requirement (4) stems from the intrinsically athermal aspect of the LBM. In a thermal model, the effects of the latent heat of phase transition can be properly simulated as variations of the temperature locally, and other variables such as viscosity, surface tension, etc, which are sensitive to temperature, can be parameterized in the simulation as a function of temperature. Our model does not directly allow this, but thankfully, there is no evidence for strong thermal effects because the system is plutonic. This thermal aspect can be quite strong in some systems, where diffusive behaviour (of usually chemical potential or heat) is slow enough such that small thermal perturbations to a combination of viscosity, diffusivity, solid structure and surface tension can form a positive feedback to enhance preferential growth such as diffusively limited aggregation, thermally limited crystal growth, etc. At the most, we need a small perturbation to the important fluid state parameters as a function of pseudo-temperature (since there is no real temperature parameter) to simulate some of these effects. In our near-equilibrium growth of plagioclase, these effects are small, but included anyway. Fulfillment of requirement (4) entails making the LBM portion of the simulation crystallization "aware". A small local perturbation to the viscosity and surface tension parameters will be made in such a way as to reflect a drop in viscosity due to rejection of latent heat and an increase in surface tension between the crystal and the residual melt.

Some of these conditions can be described by equations. First requirement indicates that the growth rate shall be (locally) linear and proportional to the concentration (enrichment) of the melt near the growth front. If $\Delta\mathbf{r}/\Delta t$ is the rate of growth, and ρ_p is the local enriched plagioclase concentration, then we need to pick a function $f(\rho_p)$ such that

$$\frac{\Delta\mathbf{r}}{\Delta t} = f(\rho_p); \tag{3.55}$$

$$0 \leq f(\rho_p) \leq 1, \text{for } 0 \leq \rho_p \leq 1. \tag{3.56}$$

Additionally, we choose a function $f(\rho_p)$ such that it exhibits a sharp or steep cut off threshold of the growth rate with respect to the local concentration, so that for low concentration regions, there is effectively no plagioclase growth.

In reality, there are a large number of functions that satisfy the needs described above.

One merely has to choose one such function. It is in practice best to pick a function that satisfies some additional constraints not intrinsic to the model such as the ease with which the parameters can be tuned, the simplicity of the function and the computational load it presents to a given computer and so on. These criteria are generally satisfied if we choose a Gaussian distribution. Thus

$$f(\rho_p) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(\rho_p - b)^2}{2\sigma^2}}. \tag{3.57}$$

Since for various combinations of b, σ, the function $f(\rho_p) > 1$, this function can be divided by its maximum value so that the maximum rate of growth is 1. The maximum of the Gaussian function $f(\rho_p)$ occurs when $\rho_p = b$, thus the maximum is

$$max(f(\rho_p)) = \frac{1}{\sqrt{2\pi\sigma^2}}. \tag{3.58}$$

Therefore, the peak-normalized Gaussian function is

$$f(\rho_p) = e^{\frac{-(\rho_p - b)^2}{2\sigma^2}}. \tag{3.59}$$

In the above equation, σ^2 is the variance, and ρ_p (same as ρ_1) is the concentration of plagioclase compatible portion of the melt, and b is the mean. They are all available as parameters to be tuned for our use. In particular, b is a parameter in concentration allowing the centering of the Gaussian. Although our Gaussian function chosen above is normalized, it is not necessary as per our requirements for $f(\rho_p)$. For a few standard deviations away from the peak rate, $f[\rho_p \leq (b - n\sigma)]$ can be interpreted as the equilibrium concentration for which growth is so small that it is practically zero for some integer $n \geq 1$. For discussions on how b and σ will be chosen, refer to the later chapter on model details. Since these are parameters of the parent function, we can also designate the function $f(\rho_p)$ as $f(\rho_p, b, \sigma)$. These two notations will be used interchangeably when there exists a preference for one form over the other.

To achieve this rate of growth, the probability of phase transition $P1$ of a given lattice site will be set equal to this function. In this fashion, the average growth rate over a large growth front and many time steps should match our desired rate of crystallization.

Therefore,

$$P1(\mathbf{x}) = f(\rho_p) = e^{\frac{-(\rho-b)^2}{2\sigma^2}}. \tag{3.60}$$

When a cell makes a phase transition into solid, the plagioclase density fraction of that cell $\rho_p/\rho*$ becomes the density fraction of plagioclase to non-plagioclase content in that cell in the solid state. It should be mentioned that the overall rate of growth (maximally $\Delta\mathbf{r}/\Delta t \sim \Delta x/\Delta t$) can be scaled to observed rates of growth (~ 0.1 m / year, Emeleus et al. (1996)) if necessary, though this will increase the simulation time drastically to unreasonable values if the physical size of the grid spacing, $\Delta x' \sim 1$mm.

Of course, it is not entirely clear that the domains in the simulation are stoichiometrically plagioclase. In systems exhibiting observable unmixing coupled with subsequent ordering processes, nuclei are known to form both inside the domains, if the domains are sufficiently stoichiometrically similar to that of the final nucleated species; or along the interface of the domains, if some interface region exhibits corresponding stoichiometry. It is possible to extend the above idea to describe this situation as well as higher order mixtures (ternary, etc) where the final product maybe a combination of several constituents. If one were to replace the Gaussian function above with a multi-variate Gaussian function, whereby each of the coordinates specify the concentration of one species, and a value for b_k where b_k are stoichiometrically derived volume fractions of each species k in the final reactant (crystal), and a value σ_k in a similar fashion, one can retrieve a general description of growth rate for k phases as

$$P1_k(\mathbf{x}) = \mathbf{f}(\rho_k, \sigma_k, b_k). \tag{3.61}$$

In our requirement (2), we need a discretized attachment probability with dependence on direction c_i. This can be accomplished by setting up a probability mass function of attachment probabilities $\mathbf{p}2(i)$ in the form of a look up table (or vector) for each of the 8 directions in the D2Q9 lattice excluding the rest position (the zeroth direction is a rest direction only), summing over the neighbours to produce the total probability of attachment of type j structure due to anisotropy $P2(j)$. Thus

$$\mathbf{p}2(i) \equiv (p2(1), p2(2), p2(3), ..., p2(8)) = (p2_1, p2_2, p2_3, ..., p2_8). \tag{3.62}$$

Since it is a probability mass function, it must satisfy

$$\sum_{i=1}^{8} p2(i) = 1 \tag{3.63}$$

In practice, **p2** can be defined numerically with any real values assigned to any of its components so long as it satisfies the above condition of normalization. A discussion of how it will be chosen will be presented in the next chapter.

To obtain the total probability $P2(\mathbf{x})$ of attachment due to growth anisotropy, we sum over all nearest neighbours of the site (8 of them in D2Q9) and define a binary operator $B(\mathbf{x}+\mathbf{c}_i)$ where $B(\mathbf{x}+\mathbf{c}_i) = 1$ if $(\mathbf{x}+\mathbf{c}_i)$ is a solid, and $(\mathbf{x}+\mathbf{c}_i) = 0$ otherwise.

$$P2(\mathbf{x}) = \sum_{i=1}^{8} B(\mathbf{x}+\mathbf{c}_i) p2(i) \tag{3.64}$$

This final value $P2$ can then be multiplied by $P1$ to yield the total probability of phase transition at any given site.

The solution to requirement (3) is simply an extension of the above to 2 (or more) anisotropy types to simulate the change of direction upon heterogeneous nucleation and subsequent epitaxial growth.

Thus, for $j = 2$ degrees of anisotropy, the above equation then reads

$$\sum_{i=1}^{8} p2_i^j = 1, j = 1, 2 \tag{3.65}$$

A growing site will have up to 8 neighbours of either (of 2) types of plagioclase structures, the probability of attachment of both types sums to a maximum of 1. This is given as

$$\sum_{j=1}^{2} \sum_{i=1}^{8} p2_i^j = 1 \tag{3.66}$$

The total probability of structural type j for two different structural types can be written as

$$P2_j = \sum_{i=1}^{8} B_j(\mathbf{x}+\mathbf{c}_i) p2_i^j = 1 \tag{3.67}$$

where $B_j(\mathbf{x}+\mathbf{c}_i)$ is an extension of the binary operator presented above. Here $B_j(\mathbf{x}+\mathbf{c}_i) = 1$

if $(\mathbf{x} + \mathbf{c}_i)$ is a solid cell of structural type j and $B_j(\mathbf{x} + \mathbf{c}_i) = 0$ otherwise.

The total attachment probability independent of structure is then $P2 \equiv P2_1 + P2_2$. Thus, the total growth rate that a given site will make a phase transition due to the concentration ρ_p and the anisotropic growth condition is the product of the two $P1, P2$

$$P \equiv P1 \cdot P2. \tag{3.68}$$

When the cell makes a phase transition into a solid, it becomes the solid type with the highest attachment probability. Thus, if $P2_1 > P2_2$, the cell becomes a structural type 2. If the difference between $P2_1$ and $P2_2$ is smaller than a preset threshold a_1, so that effectively, they are equal, then the solid phase structure is randomly selected stochastically between the two types with equal probability. This parameter a_1 can be set fixed and equal to the smallest element of $p2_i^j$ if $min(p2_i^j)$ does not depend on j since this is the minimal amount by which the outcome of the structural type at the growth site is determined. This process is illustrated in figure 3.12.

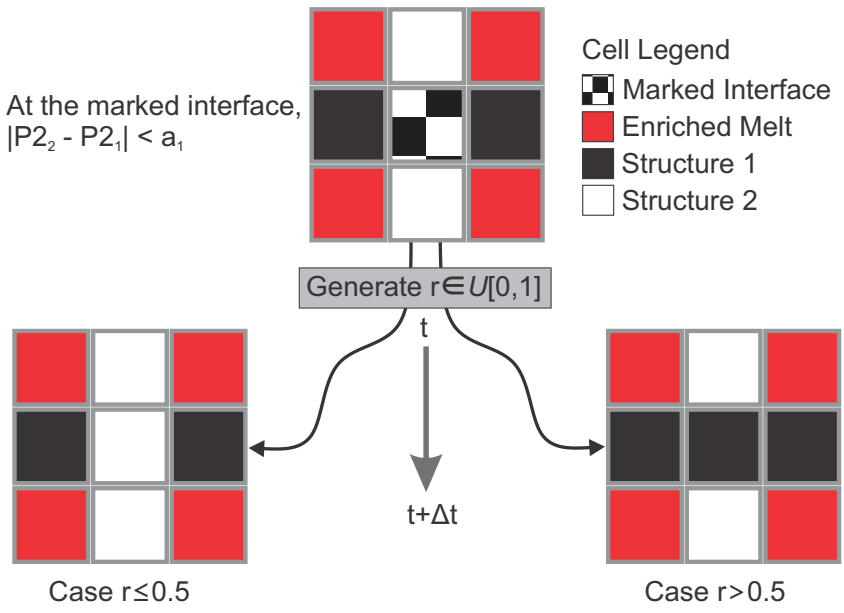


Figure 3.12 – Illustration of the treatment of a structurally under-determined interface site.

Of course, the growth site could lack a sufficient number of neighbours such as a nucleated site at $t = 0$, or a thin crystal growing into a region of rich saturations. In general, there are situations where the currently growing crystal has its fastest growing directions aligned

orthogonal to the current alignment of the enriched melt channel (ill-aligned). In this case, $P1$ will be high at the interface sites between the crystal and the enriched melt channel, but $P2$ is very low. Therefore, a threshold a_2 is set such that if $P1 - P2 > a_2$, then a new crystal nucleates at this interface autoepitaxially (when a substance crystallizes on an existing crystalline form of itself, but not exactly in the same structural alignment) with a solid structural type different from the parent crystal. This also allows richly saturated regions to be highly polycrystalline to resemble the condition of numerous heterogeneous nucleated crystals growing on each other. In regions where a channel changes orientation, or bifurcates, this allows us to simulate the effect of preferential growth of the most favourable crystal alignment. This process is illustrated in figure 3.13. This is a generalized form of the processes that result in twins in the sense that twins are a specific case of autoepitaxial growth when the crystals that participate in the epitaxy are related by a symmetry operation of the underlying crystal unit-cell (when unit-cells are all identical) or super lattice (when unit-cells are not identical, and the resulting structure is unit-cell modulated).

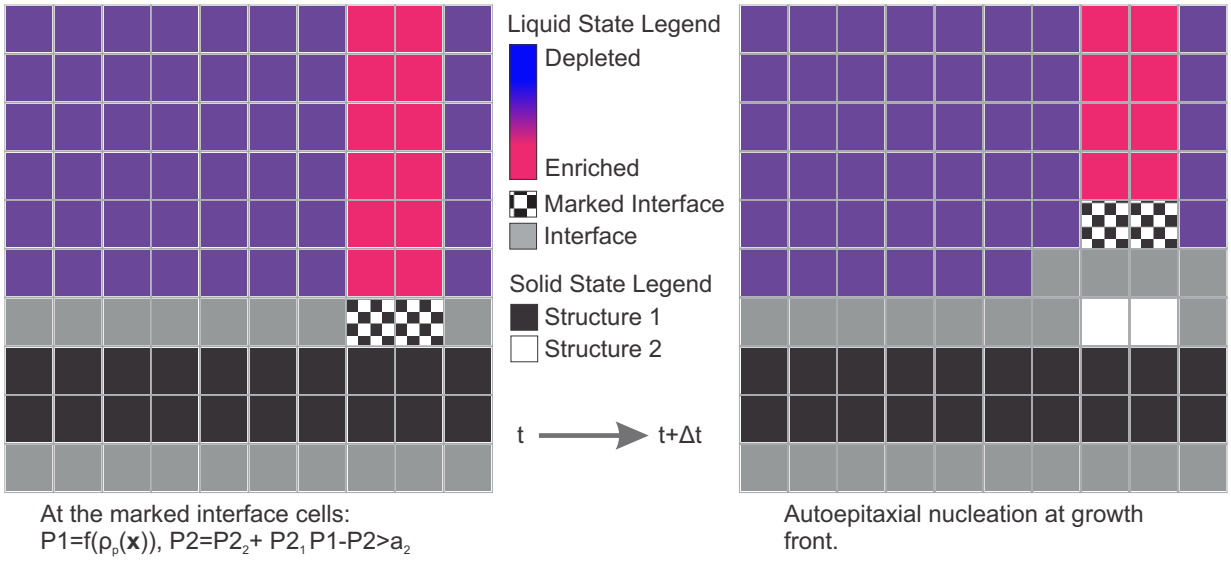


Figure 3.13 – Illustration of the autoepitaxial nucleation rule.

Now the process of crystallization at a growth site in a growth front can be divided into the following sequential steps:

(1) compute the probability of phase transition at the site, with equation 3.68,

(2) generate an uniform random number between 0 and 1, if that random number is less than the probability computed in (1), then go to (3), else no growth this time step,

(3) choose to attach the phase with the greatest attachment probability, if both phases are equally likely (to within a small tolerance a_1), randomly select a phase. If the autoepitaxial nucleation condition is satisfied, nucleate on existing crystal with a solid structural type different from the existing crystal.

A typical combination of above steps are illustrated in figure 3.14.

A growth front usually consists of a small surface of cells (a line of cells in 2D). This larger picture is illustrated in the figure below (see figure 3.15). The solid state is colour coded by white and grey, while interface cells under consideration for phase transition are patterned in hatched black and white. They are overlain on the liquid state which is coloured blue for plagioclase incompatible enrichment and red for compatible enrichment. Therefore, the solid state of plagioclase should overlay relatively red (or pinkish purple) regions. In one such region, lattice sites marked as the interface for phase transition has been selected by step (1) to undergo a phase transition in the current time step. The sites are then preferentially transitioned into most likely solid phase types following step (3), see figure captions for details. Step (2) is not illustrated. At the end of the growth stage (time step), these sites are solid with a definitive structural type.

Lastly, to allow feedback to the liquid state to satisfy requirement (4), we observe that the main effects of crystallization that affect the surrounding melt other than fractionation are a small local increase in temperature due to the latent heat of crystallization and an change in surface tension between plagioclase now in the crystal state and the surrounding melt due to the change in structure of the plagioclase enriched melt which modified the surface energy. It is not obvious at first if this change is positive or negative depending on the microscopic nature of the phase transition. If the thermal energy liberated during phase transition allows the dangling bonds at the interface (in the melt state) to better accommodate the now fixed surface bonds on the solid plagioclase surface, then the surface tension microscopically should decrease very near the interface. However, it is an unlikely scenario for a reasonable time after the phase transition, since thermal diffusion in intrusive melts occur much faster than growth

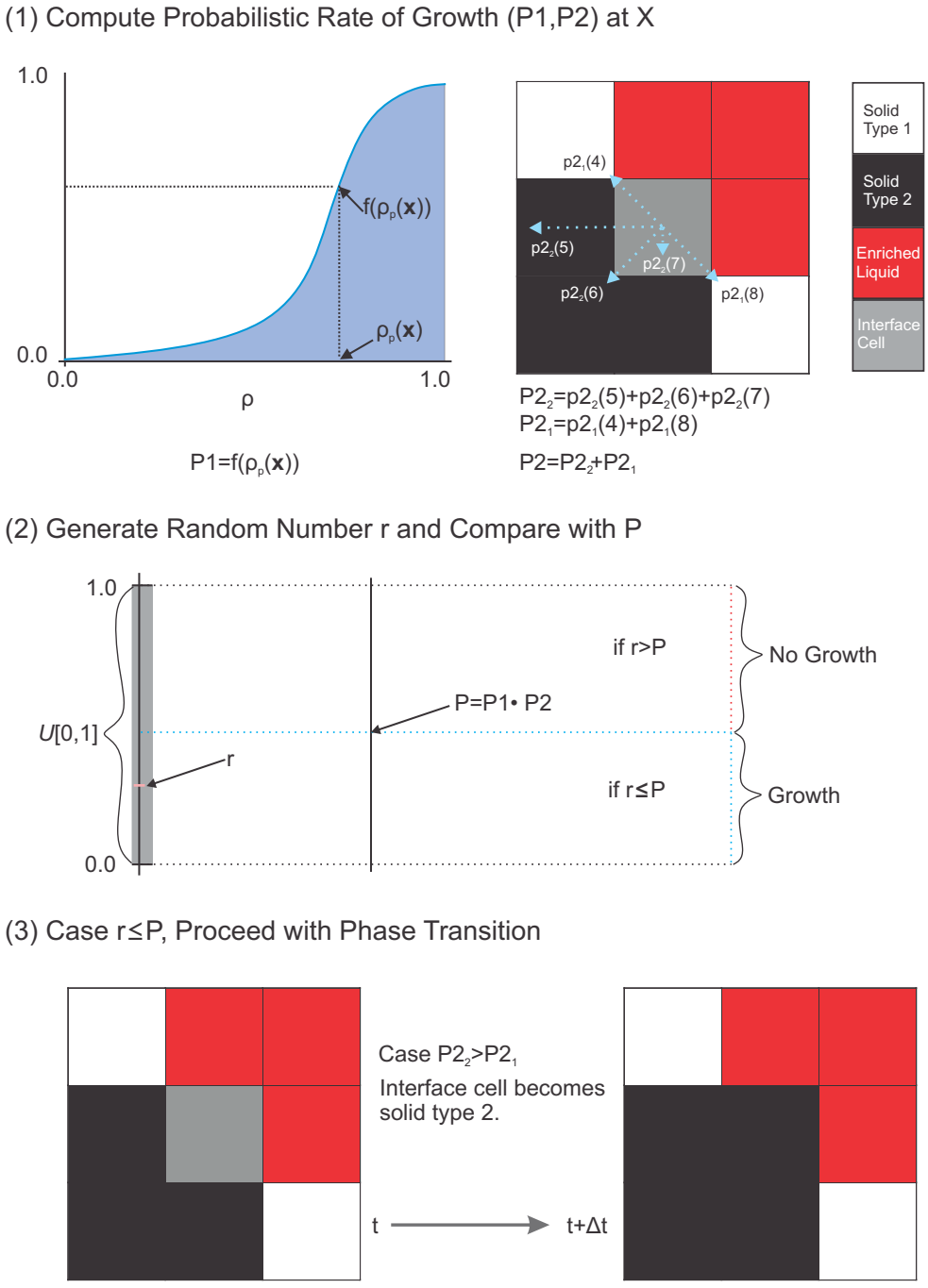


Figure 3.14 – Illustration of probabilistic growth process (1)-(3). See text for details.

rates. It stands to reason that the surface tension should increase in the vicinity of a plagioclase crystal-melt interface, due to the rigidity of the surface dangling bonds on the crystal surface lacking the deforming qualities of the melt surface dangling bonds.

The parameter β is more peculiar in the tuning process. A crystallizing front removes chemical components in the melt, leaving behind less compatible species in the crystal structure.

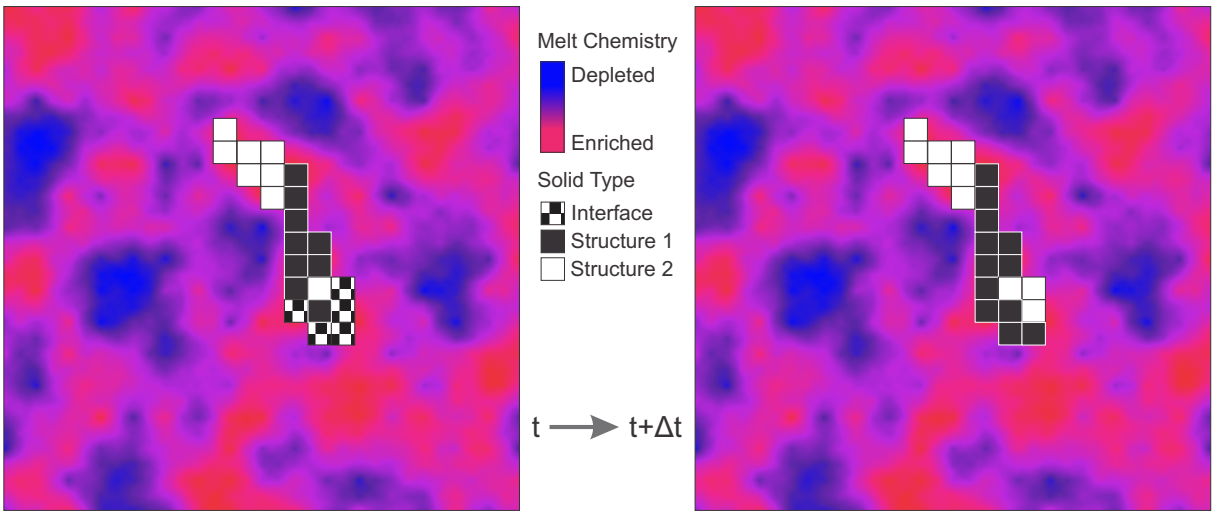


Figure 3.15 – Illustration of structural phase transition rule. Interface cells located in the vicinity of the highest concentration of the chemically enriched liquid state (more "red") are selected probabilistically to undergo phase transition in this time step. In a time step Δt, sites which grow laterally are favoured by the "white" structure, while sites which grow vertically are favoured by the "black" structure. Therefore, the interface cells transition into solid types which exhibit the highest attachment probability depending on direction of growth relative to neighbouring cells.

Therefore, the value of β should increase locally as we approach the crystallization front from the melt. This is inline with our expectations. However, it is not clear how much this value should, or could change. It depends on, along with other things, the scale of observation. There is no clear physically measurable value between microscopic to macroscopic scales. However, for the time being, we shall treat it the same as the other parameters, and allow a general adjustment.

The change induced by crystallization on the remaining fluid can be described as a function of the ratio of sites on the lattice which are solid versus sites which are still fluid in a given neighbourhood. The larger the neighbourhood, the longer the range of the intended effect and more macroscopic the behaviour. For small changes, we can treat the change as a small perturbation weighed by the above scheme.

Thus, if there are N_{total} number of neighbours in a neighbourhood defined around a site in the fluid state, with N_f and N_s denoting the number of sites in N_{total} which are respectively fluid and solid, then the viscosity ν and surface tension σ_{2D} can be modified as

$$\begin{aligned} \nu(\mathbf{x},t) &= \nu_0 + A' \frac{N_s}{N_{total}}, \\ \sigma_{2D}(\mathbf{x},t) &= \sigma_0 + B' \frac{N_s}{N_{total}}, \\ \beta(\mathbf{x},t) &= \beta_0 + C' \frac{N_s}{N_{total}}, \end{aligned} \tag{3.69}$$

where A', B', C' are some scalar parameters and σ_0 is the base value of σ_{2D}. Since $\nu \propto \tau$ and $\sigma \propto \alpha_k$ we can rewrite these two equations in a manner befitting coding. As the size of the neighbourhood $N_{total} \to \infty$, the fraction $Ns/N_{total} \to 1$. However, since the grid size is limited, and the microscopic neighbourhood size is much smaller than the macroscopic grid size, the limit as $N_{total} \to \infty$ does not accurately describe the behaviour of the fraction inside the equations. In reality, a neighbourhood size shall be chosen as a set of lattice sites which are within a certain distance from the site under consideration. This distance can be the Euclidean distance. However, since the neighbourhood is small, and evaluation of Euclidean distances on a per-site basis increases the number of arithmetic operations, it is simpler to set a square neighbourhood (sub-grid) which can be iterated, with a sub-grid size n such that $n^2 = N_{total}$, the total number of neighbouring cells in the neighbourhood. This neighbourhood is not Euclidean in distance by the adoption of the maximum norm. However, in reality, the choice of the neighbourhood shape is much less important than the choice of the neighbourhood size, and it is not entirely clear at this time what types of anisotropic structures occur in the melt surrounding the crystal-melt interface at critical temperatures, and what, if any, their alignments are to the crystal interface. Thus, despite the theoretical possibility of designing fascinating shapes and orientations of microscopic neighbourhoods of feedback interactions, the actual neighbourhood is assigned a simple shape with an isotropic orientation. Choosing a square neighbourhood satisfies the simplicity criterion and balances the computational cost and the requirement of an isotropic orientation. Therefore, since τ is used in its inverse form ω in the simulation code, the equations below will have modifications to ω

$$\begin{aligned} \omega(\mathbf{x},t) &= \omega_0 + A\frac{N_s}{N_{total}}, \\ \alpha_k(\mathbf{x},t) &= \alpha_{k,0} + B\frac{N_s}{N_{total}}. \end{aligned} \tag{3.70}$$

In practice, the neighbourhood size n (or neighbourhood subgrid size) will be much smaller than the size of the simulation grid. A, B, C are the final scalar parameters we can tune. In practice, the parameters A, B shall be chosen for a reasonable feedback effect while keeping the model stable (since it is a positive feedback in the context of our model). The value of C requires more experimentation. In fact, it is probably a good idea to not treat changes in β as a perturbation in response to the solid state and instead allow β to continuously depend on how close we are to the crystal surface. As we approach the surface of the crystal from the melt and pass the interface and into the crystal, we expect that the melt state has a separation length which becomes shorter at the interface and becomes effectively zero inside the crystal. Since there can be microscopic to macroscopic inclusions of residual melt material (solid or liquid phase) inside the growing crystal, it is not absolutely certain that β should approach 1 as we enter the crystal similar to an order parameter. β is not well defined microscopically in this sense. However, we rely on the fact that the plagioclase crystal is not structurally compatible with other mineral species and state that β may approach its maximal value (not necessarily 1) as we transition from the melt into the surface of the crystal. In other words, liquid-liquid domains defined by separation of the liquid state should have a separation that is much less complete than the solid-liquid (or solid-solid) domains, which should have a very sharp separation at the surface of the solid phase. In this sense, we modify the equation governing the change of the parameter β to introduce a continuously changing β (approximated as a linear equation) defined as

$$\beta(\mathbf{x},t) = \beta_0 + \left(\frac{1-\beta_0}{\frac{N_{total}-1}{N_{total}}}\right) C\frac{N_s}{N_{total}}, \tag{3.71}$$

where the factor $(N_{total}-1)/N_{total}$ adjusts for finite neighbourhood effects. The ratio N_s/N_{total} approaches 1 as $N_s \to N_{total}$. The parameter $0 \leq C \leq (N_{total}-1)/N_{total}$ adjusts the slope.

The entire β value as defined above allows a maximal range of β from β_0 to 1 depending on the neighbourhood solid content and parameter C. These parameters A, B, C and n will be discussed in the next chapter on model details.

Since a growing crystal poses surfaces to the liquid state, the surfaces becomes moving boundaries. In the limit of very slow velocity of the growth front where the rate of movement of a growing front is much less than the movement of the liquid surrounding it (i.e., the moving crystal face does not impart a noticeable force into the liquid), the boundaries between the crystal and the liquid state can be simulated by the same no-slip bounce-back boundary conditions above.

Chapter 4

Model

4.1 Phase Space Implementation and Parameters

A phase space of a corresponding system is the space spanned by its solutions. It is a numerical extension of the phase portraits of dynamical systems. In this nomenclature, a model which effectively explores the phase space by showing possible solutions (behaviours) to a dynamical system is called a phase space model. In computational modelling, discretization has to occur at some scaling level. From the topmost level downward, there are systems of differential equations (i.e., Navier-Stokes in hydrodynamics), that are solved using discrete solvers which approximate solutions to the system of equations. If one chooses to discretize at a lower macroscopic level, one has the discrete probabilistic description of particles in a system such as the Lattice Boltzmann kinetic equation which can be used instead. At an even lower macroscopic level, effectively approaching the microscopic level, one can discretize the probabilities into discrete levels and arrive at multi-particle simulations. At the lowest level, where one is discretizing a system into discrete particles, only one of which can (usually) be at a site at once, so that the continuum description of particles is represented by an ensemble of single particles, one arrives at the lattice gas models. For hydrodynamics and diffusion, this picture is well corroborated by theoretical physics whereby the governing equations transition from Navier-Stokes to Boltzmann to Liouville to Newton-Hamilton. For diffusion, for example, there is an intrinsic link from the microscopic dynamics (Brownian motion) to the macroscopic dynamics (diffusion) via the Einstein diffusion derivation. Regardless the level of discretization, numerical quantities are typically non-denominate numbers (dimensionless scalars), vectors (indexed sets

of scalars) and matrices (multiply indexed sets of scalars) that are represented as appropriate combinations of binaries, integers, real numbers, complex numbers, etc in any number of data structures. These non-denominate quantities are typically transformed into denominate quantities in the real world by appropriately scaling the numbers to obtain a correct correspondence of real world quantities and denominations (units). Generally, the scaling is chosen such that quantities which exhibit a non-infinite maximum such as density will be scaled so that it is normalized in the simulation (i.e., max density equals one). Other quantities which extend to infinity such as relaxation time will usually be inverted via scaling such that its value takes a range between zero and one and dimensionless. Solutions in this dimensionless phase space correspond to their counterparts in the real world via the scaling quantities. Therefore, the range of behaviours exhibited by the real system in the real world, if captured sufficiently by the model, should be represented by the range of behaviours in the phase space.

In general, one chooses a phase space model for several reasons. First and foremost, one is interested in reproducing the type of behaviour. For example, in hydrodynamics where one is interested in the change of flow behaviour from laminar to high vorticity and then turbulence, one can non-dimensionalize the system and change parameters such as driving force (pressure gradient usually, or flow speed) and/or viscosity (in fact, the ratio of the flow velocity and the kinematic viscosity multiplied by a characteristic length scale is the Reynolds number) and adjust the parameters to produce the desired behaviour then (if necessary) scale the parameter space (a space of all possible parameters) back to physical systems by using experimental data sets. Secondly, the real physical properties of the system being simulated are either unknown, incomplete, or unreliable. In this case, even if one was interested in simulating an exact scenario, there is no choice but to use dimensionless parameters until such data are available. It might be worthwhile to add that, for phase space simulations, one can always produce the same phase space as non-phase space physical simulations such as molecular dynamics and of course, the real world. In this spirit, the phrase "without loss of generality" is often attached to phase space explorations, numerical or otherwise.

As per reasons one and two discussed above, we resort to phase space simulations. We are foremost interested in reproducing in some aspect the stellate textures based on our proposed mechanisms. Secondly, there are no known complete set of physical properties of our system. This second difficulty digresses and diverges into many research topics on its own, including

experimental laboratory work, theoretical work, molecular dynamics and density functional simulations to investigate crucial properties such as surface free energy, bulk energy, viscosity as a function of temperature and composition, etc. Lastly, since we seek to only explain the formation of stellate plagioclase, not the exact physical parameters which form this texture, it is not entirely necessary for us to simulate outside of phase space.

Our fluid model allows us to simulate the interactions of two (miscible or immiscible) fluids. The mass densities $\mathbf{f}^k$ for species k are used to simulate plagioclase compatible chemicals (plagioclase molecules, or polymers with plagioclase-like chemistry) and non-plagioclase compatible chemicals. The non-compatible chemicals include, either fully liquid or partially liquid forms of minerals such as olivine, pyroxene, chromite, etc. If the non-compatible species have crystallized, then it is assumed that they are in a suspended state (where the solid particles are spaced apart with inter-particle melt), lubricated and mediated by their residual melt, so that they contribute to the liquid state by modifying the macroscopic viscosity. Since the simulation does not simulate nucleation, a site chosen will be set as the initial nuclei, usually in a plagioclase enriched region. The index k will take the form of $k = 1$ for plagioclase enriched species, and $k = 2$ otherwise. In this way, our binary mixture can simulate the effect of melt polymerization and unmixing as required by our hypothesis. Therefore, the vector mass densities $\mathbf{f}^1$ represents plagioclase chemical enriched composition in the simulation, and $\mathbf{f}^2$ represents the non-plagioclase portion.

In the phase space, the fluid model contains several parameters. These parameters can be divided into parameters intrinsic to a simulation method and parameters not intrinsic to a simulation method. Intrinsic parameters are parameters which are specific to a method of numerical simulation. Intrinsic parameters for the LBM as derived in the theory section includes viscosity (non-dimensional viscosity) which is actually set by τ the value of the single relaxation time. ω is the preferred parameter as the inverse of τ in typical simulations (see equation 3.23, since τ can range from the shortest time scale for which the simulation is stable (related to grid velocities and $\Delta x/\Delta t$)) to the largest time scale for which the simulated substance has extremely high viscosity approaching that of a solid. The stability range of τ can be found if we consider the time evolution of the kinetic equation (see equation 3.44). Simultaneously we

substitute the definition of ω and allow a vectorial description of F for f_i,

$$F(\mathbf{x}, t + \Delta t) = F(\mathbf{x}, t) - \omega \left[F(\mathbf{x}, t) - F^0(\mathbf{x}, t) \right]. \tag{4.1}$$

For a flow initially already uniform (spatial uniformity),

$$F(\mathbf{x}, t_0) = F(t_0) \tag{4.2}$$

should remain uniform at all later times, hence

$$F(t + \Delta t) = F(t) - \omega \left[F(t) - F^0 \right]. \tag{4.3}$$

Subtracting an equilibrium distribution function F^0 on both sides yields

$$F(t + \Delta t) - F^0 = F(t) - F^0 - \omega \left[F(t) - F^0 \right]. \tag{4.4}$$

Letting $\widetilde{F}(t) \equiv F(t) - F^0$ and substituting, we obtain

$$\widetilde{F}(t + \Delta t) = (1 - \omega)\widetilde{F}(t). \tag{4.5}$$

Here the heuristic interpretation of the equation is *the vectorial change in the next time step of the distribution functions at any site is equals to* $(1 - \omega)$ *times the vectorial change in the current time step.* Therefore, for the system to be stable in time, $|\widetilde{F}(t + \Delta t)| \leq |\widetilde{F}(t)|$. In other words, any magnitude changes must not amplify through time. Therefore, $|1 - \omega| < 1$ which implies

$$\begin{aligned} 0 < \omega < 2, \\ \frac{1}{\omega} = \frac{\tau}{\Delta t} > \frac{1}{2}. \end{aligned} \tag{4.6}$$

Similarly, by considering Δt as defined by grid speed $c^* = c\Delta t' \Delta x' = \Delta x / \Delta t$ where Δx is the size of a lattice cell (non-dimensioned, dimensioned quantities denoted with prime) so that

$\Delta x'^D$ is the volume of a cell in D dimensions, it is clear that

$$0 \leq \Delta t \leq \frac{\Delta x}{c}, \tag{4.7}$$

which is simply re-iterating our earlier statement that the fastest advection speed on the grid has to be slower than the grid speed (the speed of sound, or the fastest waves). This same constraint can also be arrived at if one chooses to perform the Fourier method of linear stability analysis by John von Neumann, (Wolf-Gladrow, 2005b). In fact, the advection speed needs to be much less than the speed of the fastest waves on the grid to retain stability through our use of the low-macroscopic-fluid-velocity, Taylor-expanded equilibrium distribution function. This is also known as the Courant-Friedrichs-Levy criterion.

Recall that the viscosity ν is defined in terms of ω and $\Delta t'$ in equation 3.44. We re-state this here

$$\nu = \frac{kT}{m}\left(\frac{1}{\omega} - \frac{1}{2}\right)\Delta t' = \frac{c^2}{3}\left(\frac{1}{\omega} - \frac{1}{2}\right)\Delta t' = \frac{c^2}{3}\left(\tau - \frac{\Delta t'}{2}\right). \tag{4.8}$$

Based on this, we can define a lattice viscosity ν^* by $\nu^* = \nu\frac{\Delta t'}{\Delta x'^2}$. Thus

$$\nu^* = \frac{1}{3}\left(\frac{\tau}{\Delta t'} - \frac{1}{2}\right). \tag{4.9}$$

Since viscosity depends on τ (or ω), viscosity ν and lattice viscosity ν^* can be in a range of

$$0 < \nu < \infty; \tag{4.10}$$

$$0 < \nu^* < \infty, \tag{4.11}$$

for a physically meaningful simulation.

However, in the simulation, for reasons of stability other than the above discussed non-linear stability (such as finite compressibility of LBM), ω usually takes a range somewhat narrower than the above limitations. The range of behaviours represented by ω continuously varies from thin fluid (water, light alcohols) for values of $\omega \sim 2$ to honey-like for $\omega < 1$.

In reality, the viscosities of the differently enriched regions of the melt may be a function of time, sometimes strongly so. For polymers, the longer the chain of polymers, the larger the

expected viscosity prior to the onset of phase transition. If these longer chained polymers are interspersed with shorter chained polymers, the overall macroscopic phase as a mixture has a lower viscosity. A well polymerized region in the melt is expected to exhibit a higher viscosity due to internal bonds which resist shear flow. In comparison, a region lacking in internal bonds or the range of internal bonds typically due to either environmental high temperatures or regional chemical nature should exhibit a lower viscosity. Therefore, it may be expected that the viscosities of the two phases in the simulation are not equal. However, we do not have knowledge of the actual composition of these domains during the spinodal nucleation process, since they are not necessarily the same chemically in stoichiometric proportions as that of the final solid phase product (as discussed earlier, see sub-section 3.2). In addition, the viscosities are expected to change as the melt undergoes spinodal decomposition and the regions become progressively more enriched and polymerized in their respective chemicals. To capture this dynamic behaviour, we can allow these parameters to change through the simulation after the onset of nucleation. Although a continuous change of these parameters as a function of the composition of the domains and simulation time would be more rigorous, the lack of any specific knowledge of the actual composition of these domains beyond that one of these types of domains yields plagioclase prevents us from doing so. We shall include some of this dynamic behaviour of viscosity as part of our viscosity parameter exploration.

The next parameter of interest is the interfacial surface tension of the immiscible aspect of the ILBM. It is also intrinsic to the liquid state simulation. The parameter α_k in the immiscible collision operator part 1 (see equation 3.49) is in general given for each species k. Due to the binary nature of our simulation, this parameter α_k for phases $k = 1, 2$ can be set so that $\alpha_1 = \alpha_2$. Subsequently a lattice surface tension proportional to this parameter will be defined as the lattice interfacial tension between the two species in a steady state. Following Reis and Phillips (2007) the surface tension corresponding to their immiscible collision operator $N1$ is

$$\sigma^* = \frac{5\rho(\alpha_1 + \alpha_2)}{6\omega_{avg}}, \tag{4.12}$$

where ω_{avg} is an average of the two inverse relaxation times ω_1, ω_2 associated with species

$k = 1, 2$. Since for us, $\alpha_1 = \alpha_2 \equiv \alpha$, this equation reduces to

$$\sigma^* = \frac{5\rho\alpha}{3\omega_{avg}}. \tag{4.13}$$

Of course, this parameter also has a D-dimensional physical equivalent: $\sigma_D = \sigma^* \frac{\Delta m' \Delta x'^{3-D}}{\Delta t^2}$ where if ρ_0 and ρ are three-dimensional real density and lattice density (obtained by scaling and non-dimensionalizing real density) respectively, then $\Delta m' = (\rho_D/\rho)\Delta x'^D$. D is the number of dimensions of the simulation and $D = 2$ for the D2Q9 lattice. Unless there exists a good reason, $\rho = 1$, and we shall adopt this unity density convention and use scalings if necessary. ρ_{2D} is the two-dimensional equivalent of ρ_0 obtained by assuming the vertical direction is homogeneous over the typical observable depth of stellate rays which is ~ 1 cm and multiplying ρ_0 by 1 cm to obtain ρ_{2D}.

Therefore the 2-dimensionalized surface tension is

$$\begin{aligned} \sigma_{2D} &= \sigma^* \frac{\Delta x'^3}{\Delta t'^2} \frac{\rho_{2D}}{\rho} \\ &= \frac{5\alpha}{3\omega_{avg}} \frac{\Delta x'^3}{\Delta t'^2} \rho_{2D}. \end{aligned} \tag{4.14}$$

Δt is usually chosen such that the compressibility of the underlying fluid in LBM simulations is minimized. This can be an important parameter if the underlying fluid is subjected to a large force, particularly a differential force which results in shear flow (LBM has reduced stability in shear flows). For our simulation, the largest forces present will be forces due to interfacial surface tension, this is usually (and for our cases) much smaller than any external force, such as gravity, and thus Δt can be chosen somewhat arbitrarily. Δx can be chosen by considering the smallest lengths the model needs to represent, $\Delta x'$. Then by our non-dimensionalization choices in equation 3.19, $\Delta x = \Delta x'/L$. We can choose $\Delta x = 1$ which implies $\Delta x' = L$. In other words, the value of L equals to the resolution we wish to attain in the simulation. Although simulation of unit cell sized lattice spacings would be desirable ($L \leq \mu$m), it will result in a massively sized lattice. For a macroscopic simulation on the order of about a meter (characteristic scale of the stellate textures), $N_{sim} = \text{m} \ / \mu \text{ m} = 10^6$, resulting in unreasonably

large memory and computational requirements. As will be discussed in the latter sections, the lattice sizes are limited to $N_{sim} \sim 10^3$ linear nodes. Therefore, the maximal resolution in the simulation represented by Δx at best would be about mm scale: $L \sim$ mm. This would mean that the linear grid length in dimensional quantities would be $N_{sim} \times L$ where N_{sim} is the number of lattice sites in any direction. Thus, for a square grid, the physical simulation area would be $N_{sim}^2 L^2$.

Lastly, the parameter R as defined in equation 3.53 for the ratio of one species to both species in the melt phase can be set to match the observed amount of plagioclase in the rocks with respect to the remaining non-plagioclase content. In this way, R can be chosen in a continuous range $0 \leq R \leq 1$. This is a parameter sensitive to lattice dimension scalings. For large observational scales over Dm of outcrops, the plagioclase concentration is essentially dictated by the composition of the entire stellate bearing layer. For observational scales from m-Dm, the composition of plagioclase varies extensively with spatial coordinates. Therefore, to simulate a stellate texture, we would expect the value of R to be somewhat larger than bulk averages, and to simulate the lack of stellate texture regions, we would expect the value of R to be either much larger than bulk averages (such that the layer content is essentially anorthositic), or much smaller (thus the plagioclase compatible chemically enriched regions becomes emulsions only and do not establish channels). To simulate only stellate textures, our theory limits the choice of R for meaningful runs for this texture to form instead of pegmatites, for example. The plagioclase chemical content (typically) has to be set so that, roughly, $R < 0.5$. In practice, we shall explore this parameter and observe a continuous range of behaviours as R is varied.

Despite the fact that our original goal of the simulation was to explore the phase space behaviour, and to reproduce in phase space the stellate patterns, we can use the above scalings and parameters to get a rough estimate for a few interesting constraints such as time between injections, time during injections prior to onset of large scale plagioclase crystallization, etc. These numbers will serve as corner stones for researches and researchers later.

The parameters in the stochastic portion of the simulations include the parameters b and σ in the gaussian function $f(\rho_p)$ (or equivalently $f(\rho_1)$) as defined in equation 3.60. Given that the parent function is a Gaussian, these parameters represent the mean and standard deviation respectively. Adjusting them allows us to shift and contour the function so as to produce realistic behaviour.

The parameter b can be adjusted so that the growth rate is maximized when the local concentration of plagioclase compatible chemicals ($\rho_p = \rho_1$) in the melt reaches a certain value. Past this value, there is no reason to suspect the rate should drop. Therefore, the mean of the Gaussian shall be centered at $\rho_p = \rho = 1$.

An illustration of the effect of b on the function $f(\rho_p)$ while fixing σ is given in figure 4.1.

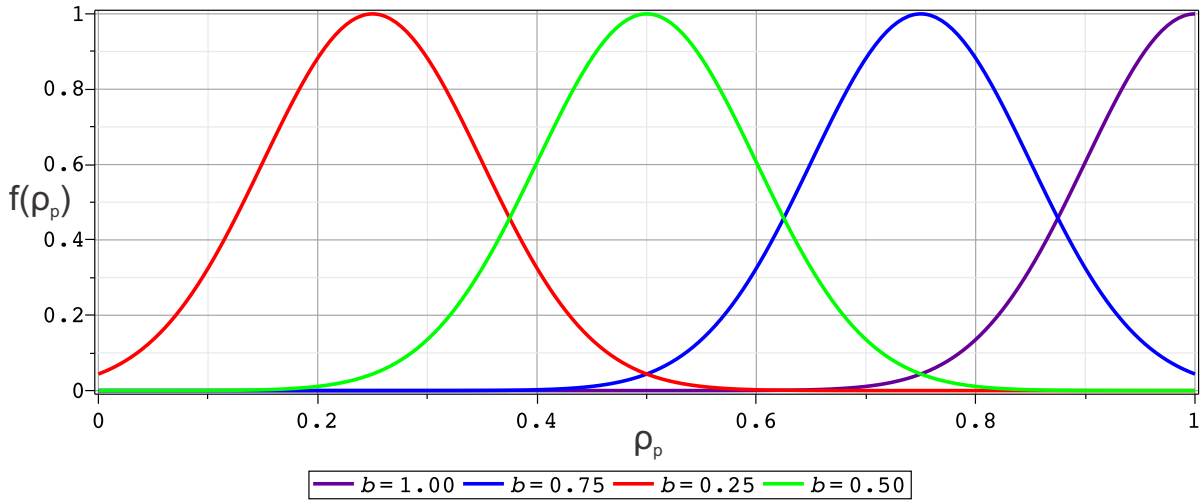


Figure 4.1 – Effect of adjusting b on the function $f(\rho 1)$. σ is fixed at 0.10.

Similarly, the standard deviation adjusts the sensitivity of growth rate on the concentration of plagioclase compatible chemicals. A few different σ values are plotted with b fixed in figure 4.2.

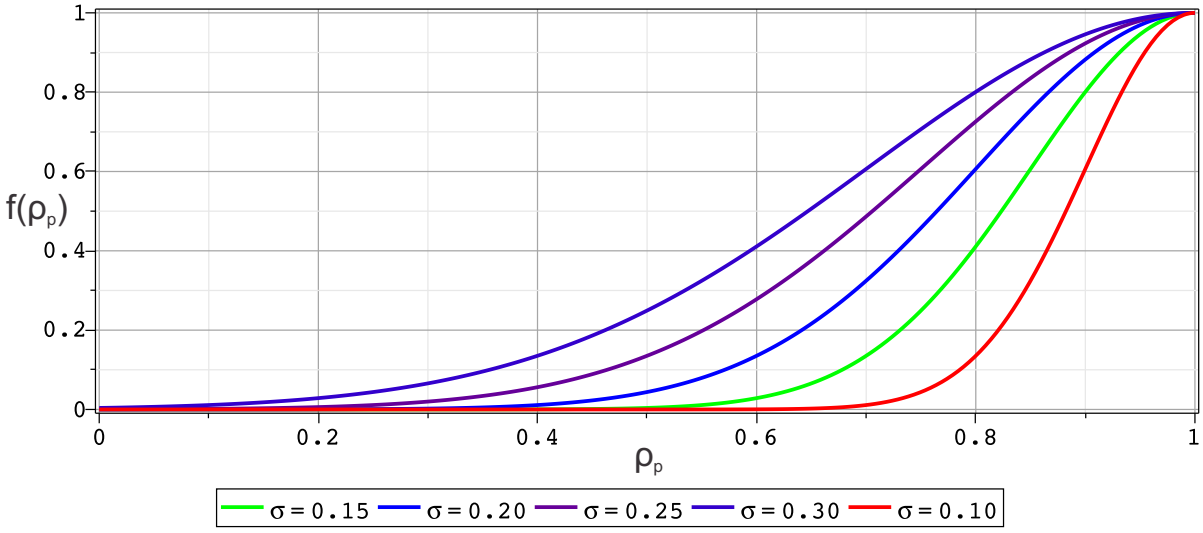


Figure 4.2 – Effect of adjusting σ on the function $f(\rho 1)$. b is fixed at 1.

The region over which the function $f(\rho_p)$ is steep is where small changes in concentration leads to large changes in growth rate. The steeper the slope from $\rho_p = 0$ to $\rho_p = 1$, the more sensitive the growth rate on the heterogeneous state of the melt. The sensitivity dependence of growth rate on concentration can be adjusted effectively via the parameter σ.

The three perturbation parameters A, B, C as defined in equations 3.70 and 3.71 can also be tuned as we wish.

For the parameter A, the collision frequency ω can be increased or decreased with respect to its nominal value ω_0. On the microscopic scale, phase transitions at a growing front rejects the heat of fusion into the surrounding melt, and to some degree, into the nearby crystal. Typically in melts, the thermal diffusivity is much higher than chemical diffusivity, and for a melt that has a significant amount of polymers, the chemical diffusivity should be even lower. Therefore, for a slowly forming crystal for which the growth rate is not diffusively limited, the local increase in temperature due to heat of fusion should be rather low. Therefore, the value of A shall be in a range $\omega \approx \omega_0$. The minimal value, of course, is 0.

Parameter B modifies surface tension α_i. The surface tension of a substance/phase is caused by the excess energy of dangling bonds (microscopically) on the surface of that substance/phase relative to the fully bonded state in the bulk sense. Therefore microscopically, if the chemical components of a solid and liquid phase are identical for a particular substance, and both are in contact with another phase (with a non-zero surface tension), then the surface tension between the solid phases should be greater than that between any other combinations, since the liquid state (melt) is both less structured and more deformable (accommodating) than a crystalline solid. Technically, this value can be scaled from the minimal value observed in the liquid-liquid surfaces, to the maximal value of solid-solid surfaces. However, although there are some liquid-liquid values established in certain melts, not even rough figures are available for the relevant solid-solid or solid-liquid surfaces. Therefore, this parameter shall be adjusted to its full range ($0 \leq B \leq 1$). However, the regime represented by $B \ll 1$ is probably more realistic in that the perturbation is kept small.

The last parameter C adjusts the strength of β which gauges the interfacial width (or effectively, the characteristic mixing length) of chemically distinct regions. The parent parameter β equals to 1 for nearly perfectly separated mixtures and 0 for completely miscible mixtures. Since we do not have a full structural understanding of the melt state near criticality (phase

transformations) yet, this parameter shall also be the subject of experimentation, allowing a full range of adjustments. Since the upper limit of β is 1 (actually, as the original authors who invented the collision operator employing the parameter β pointed out, β can actually be slightly greater than unity, somewhat abusing the slightly compressible nature of the LBGK, and so long as the negative populations which result from this are kept small, overall stability is maintained) and the lower limit is $\beta \approx 0$ but $\beta \neq 0$ for a separable mixture, and there are no stability issues related to the choice of β (so long as $0 \leq \beta \leq 1$), the choice of C shall be a range $0 \leq C \leq 1$.

Next, the neighbourhood size n needs to be chosen. In practice, the bigger this value is, the more likely a growing crystal will affect the melt surrounding it, and thus the more sensitive the melt to the presence of crystals. However, for this effect to be realistic, the value n must be microscopic in comparison to the overall simulation grid size, since the simulation is macroscopic in scale. Thus, for a simulation of linear grid size N_{sim}, the relationship $n \ll N_{sim}$ or $n \sim N_{sim}/100$ limits the upper meaningful size of n. In particular, as the width of the stellate rays simulated in our system are mm-cm (see above discussion on Δx), this neighbourhood should be smaller than that. If the width of the stellate rays are on the order of 10 grid spacings ($10\Delta x$), then the neighbourhood shall be taken at $n \sim 10\Delta x$. This satisfies the requirement that $n \ll N_{sim}$ if $N_{sim} \sim 10^3 \Delta x$ as discussed earlier. Therefore, our neighbourhood is restricted to

$$2 \leq n \leq \frac{N_{sim}}{100} \tag{4.15}$$

The two small threshold parameters a_1, a_2 mentioned in the stochastic section govern the likelihood a crystal will twin and the likelihood to nucleate epitaxially a different orientation.

The anisotropic attachment probability parameters $p2_i^j$ can be constructed for the D2Q9 lattice. The directions i are as defined by the D2Q9 grid to retain consistency. For $j = 1, 2$, a set of values can be presented in a table format. For a case of complete isotropic attachment probabilities, each direction is equally likely to grow. This case is illustrated in table 4.1. For an extremely strong anisotropic case where the crystals only grow in mutually perpendicular directions, the values can be chosen as in table 4.2. In this case, the entries are encoded by the directions $i = 1, 5$ for $j = 1$ and $i = 3, 7$ for $j = 2$. Since there are two directions per

structural type, the values in each of those directions for both are 1/2. In practice, these values are arbitrary and chosen for the desired strength of anisotropy. Several sets of values will be used to compare the strength of anisotropy vs. the patterns that result. The tables of values govern the microscopic growth phenomena, so it is expected that the macroscopic behaviour should resemble that of the microscopic behaviour. Thus, for the strongly anisotropic case, we expect that the resulting crystals would grow into thin laths, at mutually perpendicular angles.

Table 4.1 – Table of $p2_i^j$ for $j = 1, 2$ anisotropy types and $i = 1..8$ number of neighbours.

i,j	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8
j=1	1/8	1/8	1/8	1/8	1/8	1/8	1/8	1/8
j=2	1/8	1/8	1/8	1/8	1/8	1/8	1/8	1/8

Table 4.2 – Table of $p2_i^j$ for $j = 1, 2$ anisotropy types and $i = 1..8$ number of neighbours of which only 4 are utilized.

i,j	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8
j=1	1/2	0	0	0	1/2	0	0	0
j=2	0	0	1/2	0	0	0	1/2	0

For a more realistic case of anisotropy, we can pick values so that the dominant growth directions are not the only directions capable of attachment. An example case is given in table 4.3.

Table 4.3 – Table of $p2_i^j$ for $j = 1, 2$ anisotropy types.

i,j	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8
j=1	1/16	1/16	5/16	1/16	1/16	1/16	5/16	1/16
j=2	5/16	1/16	1/16	1/16	5/16	1/16	1/16	1/16

Lastly, the two parameters a_1, a_2 for the switching threshold and the growth rate below which heterogeneous (autoepitaxial) nucleation is dominant are set such that $0 \leq a_n \leq 1$. In practice, the real value of a_1 is set to $min(p2_i^j)$ when $min(p2_i^j)$ is independent on the choice of j. For a given site in a finite neighbourhood, the minimal change in the difference of the sum of probabilities of attachment of $j = 1, 2$ (denoted by $P2_1, P2_2$, as defined in equation 3.67) is $P2_1 - P2_2$. This difference between neighbours of different types is the smallest attachment probability in the probability table $p2_i$. For table 4.3, this would be 1/16. a_2 sets a threshold of growth rate, below which it is expected that autoepitaxial nucleation will occur to yield growth of new crystals on the existing surfaces of a crystal. Therefore, lower values of a_2 imply

progressively higher polycrystalline growth behaviour where the new growth fronts experience a higher proportion of nucleation and growth events rather than only growth. A good starting point for the value of a_2 can be deduced by considering the value of $f(\rho)$ at $\rho = b - m\sigma$, a pseudo-equilibrium growth rate established at m (an integer) standard deviations away from the peak value of $f(\rho)$. In general a_2 is chosen in an arbitrary way, to facilitate a desired balance between growth front nucleation events and growth only events. For specific cases, where the melt state is homogeneous enough, and the table $p2_i^j$ is known, a good estimate for the transition from heterogeneous nucleation driven behaviour to monocrystalline growth can be found by computing $P1 - min(p2_i^j) \equiv a_2^*$. Subsequently, values can then be chosen around this value to promote, or limit the nucleation events relative to growth. This estimate a_2^* becomes perfectly accurate in the limit when the melt becomes perfectly spatially homogeneous chemically. However, for heterogeneous states of melt of interest to us in realistic scenarios, this estimate is rather poor and thus some trials and errors are required to experimentally find, via direct simulation, a suitable value of a_2.

Since these parameters set lower thresholds on epitaxial growth probabilities in the solid state $P2_1, P2_2$ and $P = P1 \cdot P2$, and the upper maximal value for either parameter is obviously 1, these parameters must satisfy

$$\begin{aligned} min(p2_i^j) \leq a_1 \leq 1, \\ 0 \leq a_2 \leq 1. \end{aligned} \tag{4.16}$$

In addition, these are low growth rate thresholds, so the expectation for these parameters are such that they both must be much smaller than the maximal growth rate case. This is satisfied if we pick values so that

$$\begin{aligned} min(p2_i^j) \leq a_1 \leq 1, a_1 \ll 1; \\ 0 \leq a_2 \leq 1, a_2 \sim f(\rho = b - n\sigma) \ll 1, n \neq 0. \end{aligned} \tag{4.17}$$

In practice, a range of values shall be simulated to establish the behavioural differences as one sweeps these parameters. Parameter a_1 is not necessary to sweep, as it is established that this parameter is best set to $min(p2_i^j)$ since we will explicitly choose orthogonal anisotropies with entries for $j = 1, 2$ differ only by permutation, thus $min(p2_i^j)$ is not dependent on j.

To summarize the parameters, we conclude this section here with a list (see equations list 4.18) of all the parameters the model has to offer, divided into parameters which are intrinsic to the liquid state, solid state and common to both.

List of All Model Parameters

Liquid State Parameters

kinematic viscosity:

$$\nu = \frac{c^2}{3}\left(\frac{1}{\omega(t)} - \frac{1}{2}\right)\Delta t' \text{ controlled by } \omega$$

surface tension:

$$\sigma_{2D} = \frac{5\alpha}{3\omega_{avg}}\frac{\Delta x'^3}{\Delta t'^2}\rho_{2D} \text{ controlled by } \alpha$$

enrichment parameter:

$$R = \frac{\rho_1}{\rho}$$

mixing length parameter:

$$\beta_0 \rightarrow \text{adjusts interfacial width}$$

density:

$$\rho_{2D} = \rho\Delta m'/\Delta x'^2$$

Solid State Parameters

$$(b, \sigma) \rightarrow f(\rho_1, b, \sigma) \text{ adjusts growth rate function}$$

$$p2_i^j \rightarrow \text{sets anisotropy}$$

$$A, B \rightarrow \text{perturbation parameters of } \omega, \alpha$$

$$C \rightarrow \text{adjusts } \beta \text{ linearly}$$

$$n \rightarrow \text{neighbourhood size}$$

$$a_1 \rightarrow \text{structurally under-determined growth site threshold}$$

$$a_2 \rightarrow \text{autoepitaxial nucleation threshold}$$

Parameters Common to Both

cell dimension:

$$\Delta x' = \Delta x L$$

simulation time interval:

$$\Delta t' = (L/U)\Delta t$$

(4.18)

4.1.1 Coupling of Liquid to Solid

Having laid out the foundations of the liquid and solid states independently, we can now couple the two. The coupling of the liquid to solid state can be functionally broken into the pre-nucleation stage, where in the vicinity of plagioclase enriched concentration pockets there are no nuclei present. In this stage, the behaviour of the melt should be dominated by small scale domain coarsening as described by the spinodal decomposition (or spinodal nucleation) theory. Beyond this stage, the small pockets rich in plagioclase compatible chemicals generate stable nuclei and allow the process of crystallization to take over. During this stage of crystallization, the melt interacts with the growing crystal by allowing it to "bathe" in a mixture of chemically similar and dissimilar regions. Crystallization of plagioclase will proceed until its constituent chemicals in the melt are depleted.

Prior to the onset of nucleation of plagioclase, the melt can be simulated by the liquid state alone where the effects of spinodal decomposition take place. For complete rigour, a set of simulations should be run for the solidifying minerals prior to the onset of plagioclase crystallization (olivine, cpx, chromite, etc). However, we observe that the central effect of the earlier crystallizing minerals is to enrich the residual melt in latter-crystallizing mineral-compositions and create domains (or clusters) of mushy crystal aggregates. This effect can be adequately represented by just the spinodal decomposition in the simulation, since for intermediate stages of hydrodynamically controlled domain coarsening, the size of the domains increase linearly with time, similar to interface controlled crystallization. To simulate all phases crystallizing from the parent melt would be somewhat unfeasible, if only for the scaling constraints. Some of the minerals crystallized are much smaller than the plagioclase crystals found in the stellate rays. Chromite crystals for example, can be sub-milimeter, as well as some olivine crystals.

After the initial stage, a nuclei site will be chosen and crystallization can begin (satisfying the greatly simplified nucleation stage as per our model requirements). This is usually done in the center of the simulation grid to satisfy our hypothesis on the radial origin of the stellate texture and to maximize the effective area available for the simulation. Other schemes are possible such as computing the probability of nucleation per site by considering the local concentration of the plagioclase compatible portion of the melt, the imposed temperature of the melt plus various other details and using a stochastic routine similar to the phase transition routine above. This is not needed for our simulation as it is not thought to be crucial for the genesis of the

stellate textures due to large amount of existing crystal surfaces inside the melt that serve as heterogeneous nucleation sites.

In coupling the liquid simulations to the phase transition simulations for the growth stage, we need to consider the time scales of both of these processes independently and arrive at a proper ratio of time steps spent in the liquid state to the time steps spent in the phase transition portion. While there are no established relationships between time scales of plagioclase crystallization rates to the time scales of the melt unmixing/polymerization and fractionation rates, some insight can be gained on the coupling ratio if we consider some of the geological observations. For a large stellate texture, the rays of plagioclase are on the order of meters (if not more). There is no appreciable change in the width of these rays as a function of radial distance. Therefore, it is reasonable to expect that the channels established in the melt state did not change appreciably in the timespan needed to crystallize that individual ray. If this reasoning is correct, it can be inferred that after the onset of crystallization, the melt structure of the remaining liquid did not change very much. In other words, the rate at which the crystallization is taking place is faster, probably much faster than the rate at which the melt is undergoing decomposition and fractionation after the onset of crystallization. Specifically, this can be stated as a condition for interleaving the solid and liquid state simulations such that the solid occurs on a faster time scale than the liquid state. Therefore, a ratio of $1 : q$ of simulation time spent in solid phase to liquid phase will control the temporal coupling. Details on the determination of the numerical value of q will be given later.

This is in line with thermodynamic expectations (see figure 3.6) and experimental observations on polymers and melts. The growth of a new solid bulk phase should be favoured once the initial spinodal nucleation stage has passed. Therefore, the rate of crystallization should occur fast enough that the domains in the liquid state do not appreciably enlarge, but are allowed to slowly move or respond to the growth process. Since our liquid state simulates classic spinodal decomposition without coupling to nucleation (as it is not simulated) unlike the most current and general theory of spinodal nucleation and growth, the domains in our model will enlarge ad infinitum. To pause this process once the nucleation state has begun while allowing the growing crystal to perturb and influence the remaining melt and melt structure proves to be particularly tricky. Pausing the structural evolution of the melt can be done by ceasing the liquid state simulations, such that effectively, all the time steps past the onset of nucleation are spent in

phase transitions. However, this means the melt will not be able to react to the growth process, and many behaviours such as diffusive transport of plagioclase compatible chemicals toward the growing crystal will not be simulated. A somewhat better scheme is to allow the liquid to pause in structural evolution only. Observing that the fundamental role of the immiscible collision operator is to separate unlike phases, and the role of the regular collision operator is to allow hydrodynamic and diffusive mixing to occur (via collisions), we can interleave these two operators after the onset of the nucleation event in some pre-determined ratio to allow the melt to remain in stasis. In this stasis state, the melt is diffusively and hydrodynamically active and able to participate in the growth process. However, this scheme is a work around for the lack of a numerical model which adequately describes the spinodal nucleation process as one single algorithm. Another work around can be had by only running the liquid state simulations in a small radius surrounding the growing crystal, to adequately capture the effect of the growing crystal but not perturbing the remaining melt. This method is not used as it does not allow global forcing (gravity) and flow, both of which may prove to be latter interests. This interleaving condition can be described as

$$\begin{aligned} s_1 \Delta t &\rightarrow \Omega 1, \\ s_2 \Delta t &\rightarrow \Omega 2, \end{aligned} \tag{4.19}$$

where $\rightarrow$ means *spent in* and s_1, s_2 are smallest integer values such that s_1/s_2 gives a ratio of time steps spent colliding using $\Omega 1$ vs. $\Omega 2$. In practice, this can be implemented as $s_1 = 1$ and s_2 is taken as a multiple of s_1 to obtain an approximate solution, since the exact nature of the fraction s_1/s_2 maybe irrational or s_1 and s_2 maybe excessively large numerically. Choosing smaller values of s_1 and s_2 ensures that in a relatively short amount of time steps, the total change in the melt due to the interleaving condition is small. For large values of either s_1 and/or s_2, the switching of the operator may drastically change the melt state. In practice, this ratio will be determined by running the simulation after initial decomposition with trial sets of values of s_1, s_2 to determine an optimal set. These will not be parameters in the sense that they are tuneable. Only one set of optimal values is necessary.

This method has two drawbacks. Firstly, the value chosen is unique for each set of initial conditions. Changes in parameters including viscosities, and surface tension modifies the relative strengths of the immiscible and miscible collision operators rendering changes in the pair s_1, s_2. There is no systematic way of calculating s_1, s_2 given a set of initial conditions, and thus some trials and errors are always needed when certain parameters are changed. This adds great tedium to the exploration of results. Secondly, the values chosen are only smallest approximate integer pairs. Therefore, for simulation scenarios with a large time span from the onset of crystallization to the end of the simulation, the characteristic length scales in the melt state due to heterogeneities can be large enough in the sense of a gradual change from stellate-like textures to pseudo-granitic-like textures. In the opposite regime, where the miscible operator is run more often than needed, the melt becomes more homogeneous over time. In compensation, some trials and errors are also needed to establish an accurate set of s_1, s_2 such that the cumulative effects are deemed minimal. In practice, this is possible with some patience, although a different approach might be preferred.

In addressing the weaknesses with the simple interleaving condition described above, another method is used. In this new method, prior to the nucleation stage (start of crystallization) in a simulation run, an algorithm computes the standard deviation sum over the entire grid and remembers the result. After the start of crystallization at each time step the liquid state is run, the current two dimensional standard deviation sum over the entire grid is computed and compared to the value at the start of the simulation. If the current value is above the value at the onset of crystallization, the melt state is interpreted to be more separated than initially (at the start of crystallization) and the miscible collision operator is utilized to relax the liquid state. Conversely, if the current value is below that of the value calculated at the start of crystallization, the immiscible collision operator is run instead. In notation then, if σ_{std} denotes the standard deviation (not to be confused with any σs earlier, used as the standard deviation parameter in the Gaussian function $f(\rho_p)$) at site (i, j) where i, j are indices inside the two dimensional matrix of the simulation grid holding streaming mass vectors and $\rho_p(i, j), \rho_{np}(i, j)$ denotes the total melt mass fractions of plagioclase compatible and non-plagioclase compatible portions,

$$\begin{aligned}\sum_{i,j} 1 &= n \text{ for each site}(i,j) \text{ such that it is currently in the liquid state,} \\ \sigma_{std}^2 &= \frac{\sum_{i,j}[\rho_p(i,j) - \rho_{np}(i,j)]^2}{n} - \left\{\frac{\sum_{i,j}[\rho_p(i,j) - \rho_{np}(i,j)]}{n}\right\}^2.\end{aligned} \tag{4.20}$$

As computed by equation 4.20, the value of σ_{std} varies from 0 to 1 with 0 indicating perfect homogeneity and 1 reflecting perfect mixture separation. One can choose a set time interval prior to the onset of crystallization to effectively choose a characteristic separation in the liquid state as captured by one value of σ_{std}. This method overcomes the trial and error approach in finding the interleave coefficients s_1, s_2 and also allows us to rudimentarily characterize the liquid state numerically.

In general, the value σ_{std} need not be computed on a per site basis, but in fact can be coarse grained over a small neighbourhood. The advantage of this is the ability to compute the standard deviation with a characteristic spatial separation. This can be very useful in the future when combined with a complete algorithm to simulate the melt state based on spinodal nucleation and melts with compositional metastability. In our specific application, the value σ_{std} is computed per site. It should be mentioned that the standard deviation is only a function of the first and second moments in characterizing the heterogeneity of our system. Other moments can also be similarly used to either gain more insight into the melt state, or to better control the melt heterogeneities (when such algorithms as mentioned above becomes available). In a similar spirit, one can also employ Fourier transform techniques to characterize the heterogeneities in the system. The Fourier method would compute the n-dimensional transform of the n-dimensional melt state (n usually up to 3) and allow adjustment of heterogeneities based on desired wavelengths of separations and the respective spectral intensities. This is not employed here, although a remark should be made that the Fourier method is potentially a very fast (since the data is already in the form of n-dimensional matrices) and powerful method compared to the moments method in controlling and analyzing spatial heterogeneities.

This method of holding the current standard deviations at a desired threshold to interleave the two collision operators will ultimately be the method used in actual simulations.

4.2 Coding Illustration

As discussed in the previous chapter, important processes in the melt phase are simulated with the ILBM, the phase transition is simulated by the stochastics. A typical run scenario can be broken into a few different functional blocks. These blocks are organized and presented in a block diagram illustrating the simulation flow below in figure 4.3.

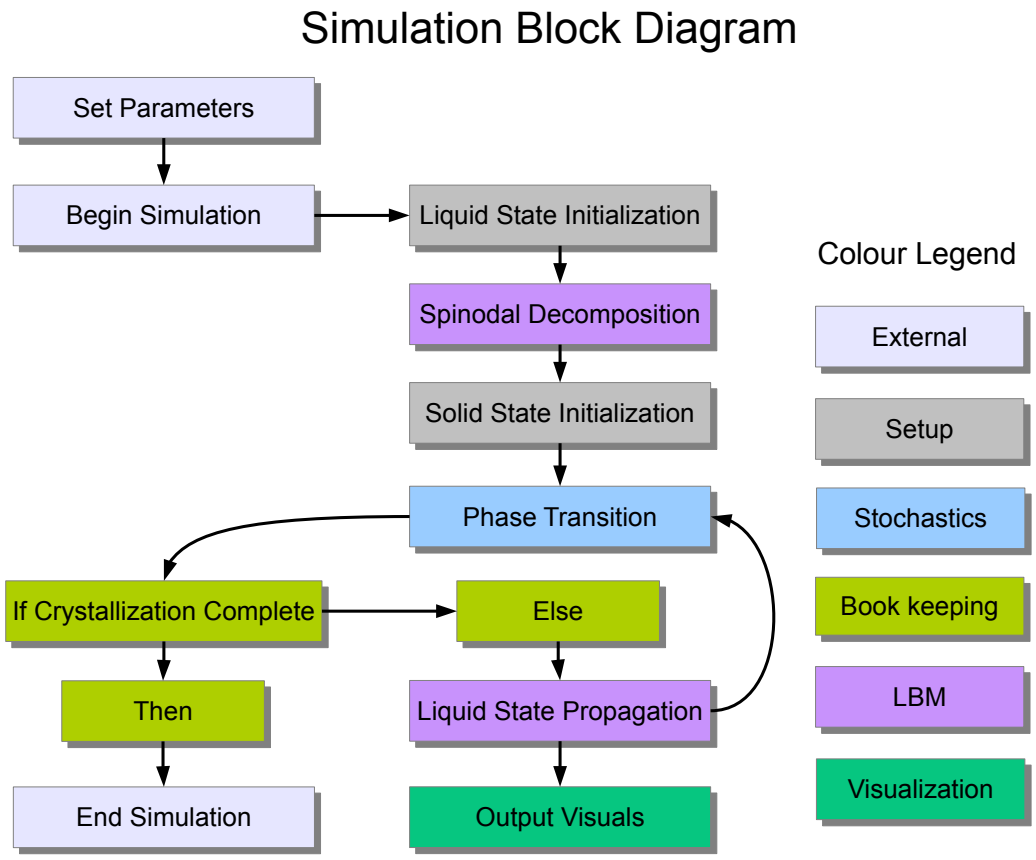


Figure 4.3 – Simulation block diagram showing a typical run. Blocks are colour coded to illustrate their relation to the simulation routines.

In this block diagram view of the simulation, the "Set Parameters" block occurs prior to the commencement of a simulation run. This is done by manually adjusting (entering) the parameters into the simulation code body.

Dividing the simulation into routines allows us to have a closer look. The main body of the simulation calls all lower routines. In the main body, the grid size is defined. This is done by allocating a multidimensional matrix of the size of the effective simulation area $N_{sim} + 2$. The additional 2 rows and columns of the simulation accounts for the outside boundaries. This grid is

duplicated once for streaming since during streaming step, neighbouring cells are altered. Since at each lattice site in the grid, 9 directions of distributions of mass are present in D2Q9, and we are interested in simulating a binary mixture of liquids, the total size of the multidimensional matrix is $[2 \times (N_{sim} + 2) \times (N_{sim} + 2) \times 9 \times 2]$. In this construct, the grid is square in shape. It should be noted that any geometry of layout is possible by choosing the two lengths of the simulation grid accordingly and constructing an appropriately shaped container boundary.

Then the various parameters, the coupling ratio, the time prior to nucleation $t_{liquid} = n_{liquid}\Delta t, n_{liquid} \in \mathbb{I}$ where $\mathbb{I}$ denotes the set of all integers, and the interval between successive outputs from the model is set. The random number generator is initialized based on a current time seed, to minimize the probability that simulations will yield the same sequence of pseudo-random numbers. Then the setup routine is called to initialize the liquid state by defining the outer walls by flagging them as a bounce back boundary and filling the container with the two melts in a perfectly homogeneous mixture spatially, with a small concentrated center and flagging their cell types as liquid. The liquid state then undergoes a period of only liquid-liquid decomposition for the specified time to simulate the unmixing portion of the combined spinodal nucleation process following the theory of spinodal nucleation and experimental works.

The simulation routines can also be classified into routines which cluster similar algorithms as defined by their usage. The collision operators comprise two of these routines: collision (binary immiscible) and collision (binary miscible). The streaming portion is defined as its own routine. These three routines comprise the liquid state simulations and are capable of miscible to immiscible behaviour, advection and flow, and purely diffusive behaviour. This portion of the routines is illustrated in figure 4.4.

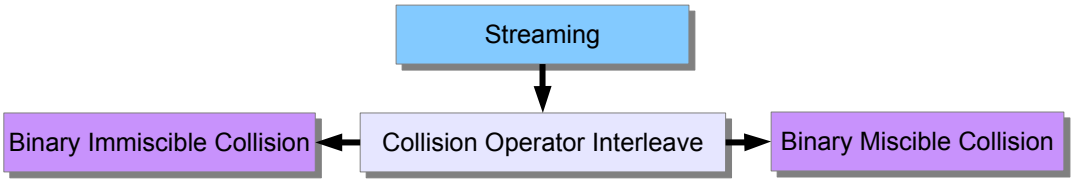


Figure 4.4 – Routines in the LBM portion.

During the pre-nucleation period, only the immiscible collision operator is used in the simulation routines. At the end of this period, the main body calls the setup routine, to initialize the solid state, seeding a nuclei in the center of the simulation area and marking its type as a solid nuclei, as well as marking its most adjacent cells (8 neighbours) as interface cells. Only

interface cells are allowed to make a phase transition. They are treated as liquid only, and collided-streamed as such until they make a phase transition. The main body then calls the phase transition routine, which uses the stochastic algorithms to mark cells that in this time step, has changed to a solid of either structural types. This is done for each site marked as an interface cell. At the end of this routine, the main body calls the book keeping routine which changes cell types. All cells marked as a solid in this time step will have their cell flags changed to solid. If the current time step is a multiple of the output interval, then an output routine is called to output the current configuration of the liquid and solid to a rasterized image with a time stamp indicating the time step out the output. Then the main body calls the collision routines. This is done respecting the interleaving interval for the collision routines such that every few time steps s_1, the immiscible collision operator is called $s_2 = 1$. The streaming routine then completes the liquid portion. The main body then repeats the events starting at the phase transition routine.

The book keeping routine which changes cell flags keeps track of the cell changes. If there are no cells left to be phase-changed in a given time step, the book keeping routine returns a binary variable $DONE = 1$ to inform the main body that phase transition for this melt has finished. By the definitions of simulation completion, if no more cells are capable of phase change, the simulation is considered complete. However, in a typical scenario for the stellate texture, a remaining site at the interface between existing plagioclase and a region depleted in plagioclase compatible chemicals have a vanishingly small, but non-zero probability for phase transition. In the sense of equilibrium as defined by a distance $b - m\sigma$, we are in equilibrium, and thus the simulation is considered finished regardless of the status of completion as defined by the lack of interface cells.

All the routines are coded in Fortran 95 from scratch. The compiler of choice is gFortran. During run time, compiler optimizations can be invoked to increase speed at the expense of some accuracy. The commonly used optimizations include unrolling loops, setting the optimization flag "-O3" for the most aggressive run time optimizations, "-mtune=corei7-avx" for Intel Core i-series machines with AVX support, and using the "-ffast-math" command line argument which reduces mathematical precision in a compiler selective fashion thereby increasing speed. The last optimization influences the outcome of the simulation in the most significant way for some scenarios. The immiscible portion of the ILBM uses small fluctuations in the liquid state to

separate phases. Thus, one can think of it as a positive feedback system where small fluctuations are exponentially amplified initially. Therefore, the use of reduced precision mathematical computation has the effect that numerical errors dominate the initial liquid state separation for a perfectly homogeneous mixture. To illustrate this, two sets of simulations are ran for the liquid state only. Both simulations begin with identical setups. The only difference is the use of "-ffast-math" in one case. One can readily see the difference between the simulations as a function of time in figure 4.5.

For this reason, "-ffast-math" is not used in simulation results presented here, but only for testing and debugging purposes where the use of this option gives a good trade off between simulation time and accuracy. This also serves to illustrate the sensitive nature of the ILBM algorithms. In practice, to avoid behaviours dominated by numerical errors, we intentionally initialize the initial fluid mass such that a small perturbation is added at each lattice site with a total magnitude much smaller than that of the total mass per site. For example, a fluid state which is initialized with $R = 0.35$ plagioclase chemical content over the entire grid has, at each site

$$\begin{aligned} \rho_p(\mathbf{x}, t = 0) &= \rho_1 = R + \epsilon \left[Rnd_0^1(\mathbf{x}) - 0.5 \right]; \\ \rho_{np}(\mathbf{x}, t = 0) &= \rho_2 = (1 - R) - \epsilon \left[Rnd_0^1(\mathbf{x}) - 0.5 \right], \end{aligned} \tag{4.21}$$

where $Rnd_0^1(x)$ is a uniform random number in the unit interval: $Rnd_0^1(x) \in U[0, 1]$ and subscript np denotes non-plagioclase. The ϵ factor ensures that the perturbation is small compared to total mass per site, and can be taken as $\epsilon = 0.01$. Each simulation run initializes the random number generator with a time based seed. Therefore, no two liquid states initialized are identical in a typical stellate simulation scenario. It should be mentioned that with this scheme, the use of "-ffast-math" compiler option still slightly alters the time evolution of the liquid phase relative to full precision arithmetics, although not as appreciably and is more prone to numerical instabilities when parameters are chosen near the boundaries of known instabilities.

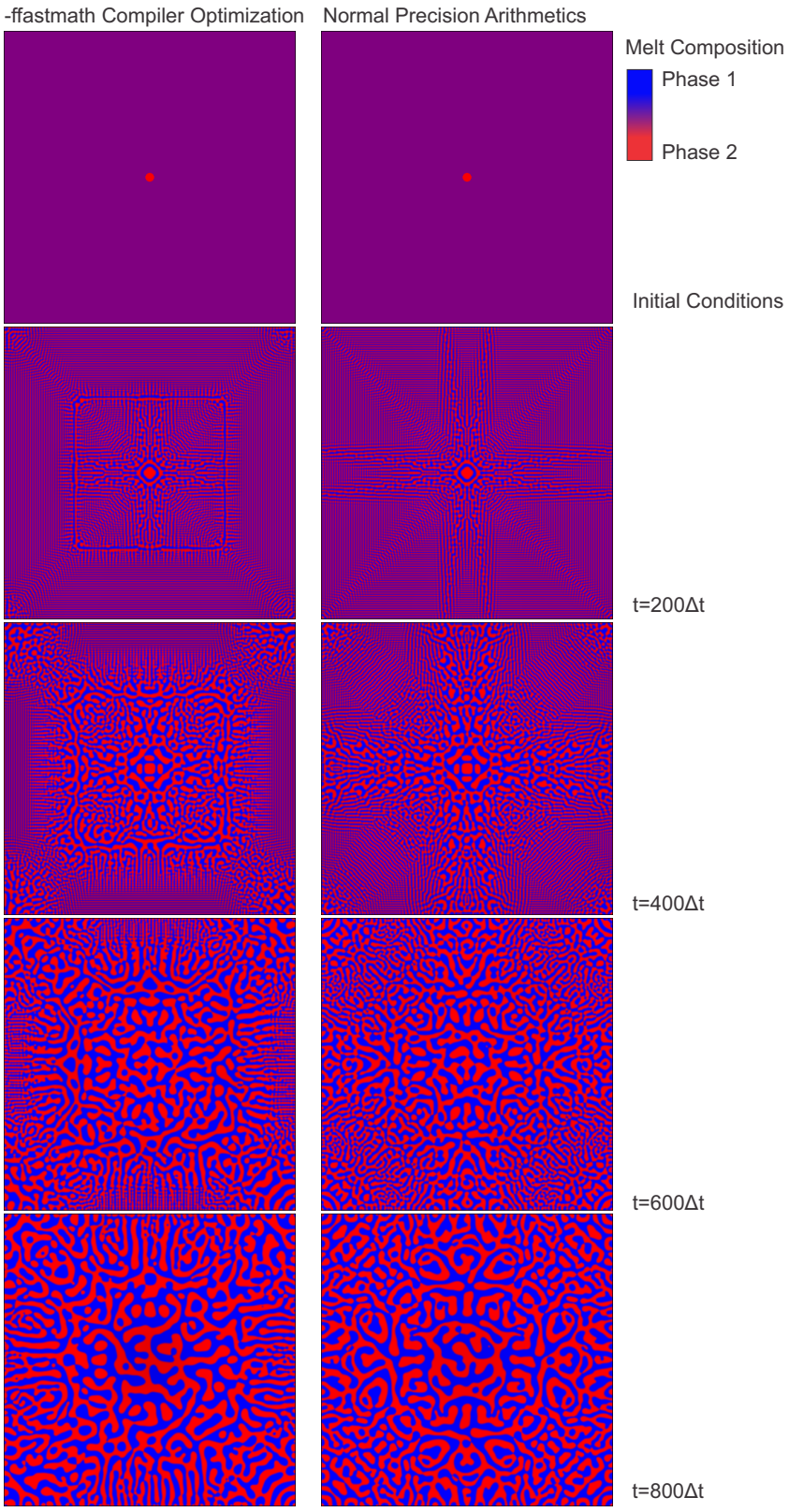


Figure 4.5 – Comparison of results as a function of arithmetic precision using "-ffast-math" as a compiler option. In both cases, we expect a central symmetry to gradually disappear with time. In comparison, one can clearly see the relative reduction in symmetry in the -ffast-math case. The normal precision case should be considered canonical, or reference.

4.3 Simulation Requirements

A common constraint for LBMs is the memory usage. This can be understood by considering the memory storage requirements for real numbers. A Fortran double precision real number is 8 bytes on non-Cray architectures. For the GNU compiler gFortran, this translates to 17 significant digits. Therefore, for a two dimensional simulation discussed above with dimensions of $[2 \times (N_{sim} + 2) \times (N_{sim} + 2) \times 9 \times 2]$, the memory requirement is $8 \times [2 \times (N_{sim} + 2) \times (N_{sim} + 2) \times 9 \times 2]$ bytes. For a realistic simulation size of $N_{sim} = 1001$, this calculates to 276.31 Megabytes (MB), which is nearly all of the memory used by an actual simulation of this size. About 10% more memory is used for book keeping, parameter arrays, eigenvectors, etc. Although several hundred MB on modern computers with several Gigabytes (GB) of physical Random Access Memory (RAM) is a trivial matter, one can easily deduce that if we double the linear dimensions of the simulation, we roughly quadruple the memory usage (memory usage $\propto N_{sim}^{D}$). For a three dimensional simulation of the same size, we would require about 270.64 GB of physical RAM. This is clearly impossible on modern consumer machines, but still within the limits of High Performance Computing (HPC) such as supercomputers or cluster networks. For memory usage more than 2 GB, a 64-bit operating system and executable binary is required due to limitations to address space under 32-bit operating systems.

Parallel computing is a LBM specialty. One could divide up the grid into sub-grids and pass each grid to a CPU core, or divide instructions from different parts of the simulation into different threads and allow the operating system to juggle the threads to different cores. Doing so requires more bookkeeping in the simulation. Since our focus here is exploratory, this extra baggage and time spent in coding is not needed. Therefore, our simulation only effectively makes use of a single thread.

Since floating point arithmetics are central to the LBM, a CPU with a strong floating point unit (FPU) greatly benefit the simulation speed. In addition, a strong branch predictor is also very helpful since the solid state portion of the code contains many conditional branches. If so desired, one can alleviate this situation by pre-generating a large array of random numbers, and use that sequence for comparisons at each time step during growth. In this manner, pre-fetches for functional calls past the statement which calls the random number generator can be made without waiting on the state of the random number being generated. Since the performance

of the code is largely constrained by the degree of optimization hard coded into the simulation code at the expense of readability and debug efficiency, it can be safely assumed that the simulation code on modern CPU architectures is not limited by execution resources inside the CPU. Although a large amount of floating point calculations are made, they are mostly of the simple type of data moves, multiplication and division. Complex functions such as the cosine, exponential and square root functions are used sparingly. As simulation areas become larger, it is expected that simulation time will eventually be constrained by memory read-write operations.

The liquid state easily dominate the amount of total simulation time spent by a factor of approximately $10 : 1$ for $N_{sim} \sim 1000$. This is due to several factors. The most important factor is the relatively large amount of memory used in the liquid state which excludes most of the liquid state information from the CPU's cache, and thus the performance is dominated by memory fetches as N_{sim} gets large, or equivalently, as the memory usage becomes much larger than the CPU's largest cache size (about several MB typically). The second factor is the extensive use of double precision floating point arithmetics in the liquid state. This ensures that the cumulative numerical errors are low. For typical scenarios, the liquid state results in about ≥ 8 decimal places of accuracy with no monotonic variations in the total grid mass (sum of all mass vectors per site over all sites) over time. For single precision arithmetics, a monotonic variation in the total grid mass can be observed, indicative of temporal accumulation of rounding errors. In the solid state, except for the parent simulation grid being passed from the liquid state to the solid state to compute the local mass density, no other need for double precision parameters exists. Lastly, the liquid state simulation is algorithmically more complex than the solid state simulation. Whereas the solid state simulation only comprises of a single routine which only performs reads over a small part of the grid, the liquid state simulation is at least two routines, both of which perform read-writes to all floating point values at all sites in the grid which are not yet solid. As the number of liquid cells dwindle into the late stages of the simulation, the time spent in the liquid state dramatically drops. Since the simulation run time in the liquid state is linearly proportional to the number of cells there are in the system, if we plot the total run time as a function of the total number of cells in the simulation (N_{sim}^2), we should observe a linear behaviour. This is illustrated in figure 4.6.

On modern architectures and operating systems, gFortran and G95 are both freely available

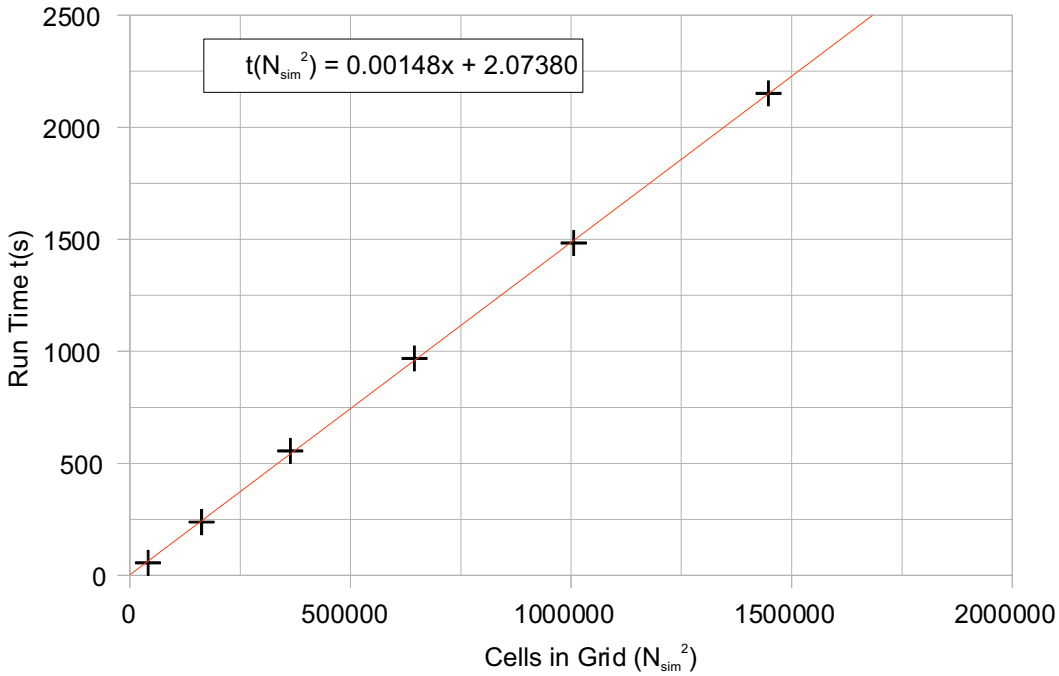


Figure 4.6 – Run times as a function of total number of cells in the simulation. The linear dimensions used are $203, 403, 603, 803, 1003, 1203$ cells respectively which are equivalent to N_{sim} sizes of $201, 401, 601, 801, 1001, 1201$. Simulation time is set at $1000\Delta t$ liquid state only. Compiler optimizations are "-O3, -funroll-loops". Compiler used was gFortran.

as cross-platform compilers. At the time of writing, only gFortran was able to produce 64 bit binaries on Linux and OS X platforms. The speeds of both compilers are comparable, although commercial compilers such as the Intel Fortran compiler are usually faster depending on the specificities of the code at hand.

Some typical run times are listed in table 4.4, results are similar for gFortran and G95, although gFortran seems to be a bit faster than G95 at the time of this writing. The time required to run the simulation is computed by calling the *CPU_TIME* command available in newer Fortran compilers. This command returns a real number set whose absolute magnitude has no meaning. However, a subsequent call to the same command returns a different real number, and the difference between this number and the earlier number gives the absolute difference in seconds of the time that elapsed between the two successive calls. This is similar to many other computational environment/languages' facilities such as Matlab's tick-tock timing. The final results are then converted to hours, minutes and seconds with the decimals truncated. Using this method, the results presented in figure 4.4 are highly reproducible. The differences between any two runs of the same compiled binary on the same architecture with the computer

not tasked by any other activities yields times that differ by less than a fraction of a second. For each architecture listed in table 4.4, the CPU specific optimization flags (arguments) are used to allow maximal adaptation of the compiled binary to the specific system.

Table 4.4 – Table of typical run times on different architectures for $N_{sim} = 1001$ as a function of different optimizations. Here only the liquid state is ran for $1000\Delta t$. To ensure simulation consistency, the stochastic solid state is precluded. Common compiler flags used are "-O3 -funroll-loops -ffast-math". Unique optimizations are indicated in independent columns.

Configuration and Run Time	With "-ffast-math"	Common Flags Only
Core i7 2640M 2.8Ghz, 1333Mhz RAM	17 minutes, 54 seconds	24 minutes, 43 seconds
Phenom II 3.6Ghz, 1600Mhz RAM	30 minutes, 8 seconds	34 minutes, 30 seconds
Core2Duo E8400 3.0 Ghz, 1066Mhz RAM	29 minutes, 49 seconds	32 minutes, 0 seconds
Xeon 5160 3.0Ghz, 533Mhz RAM	52 minutes, 40 seconds	56 minutes, 17 seconds

Several interesting things are illustrated in the results in table 4.4. First and foremost, the Xeon 5160 is severely hobbled by its slow memory subsystem, as it is a known common weak point of the Woodcrest series of Xeons (all Xeon 5xxxs) due to their exclusive use of fully-buffered memory (FBDIMM). The use of 4 independent channels of memory in Woodcrest based Xeons can not ameliorate the incredible latency in the memory subsystem. Since our simulation makes consecutive calls to the memory for one double precision numeric at a time, the memory strides are short, and the total necessary bandwidth needed is low, but is overall sensitive to the memory latency. The Phenom II CPU has a much more robust memory subsystem capable of much higher bandwidth and lower latencies than the Woodcrest Xeon based systems, and thus is much quicker in computations. However, compared to the Core2Duo E8400 which posses an older and slower memory subsystem, the execution core is not as adept at dealing with semi-optimized code. The Intel Core i7 based on the Sandy Bridge architecture has a fast memory subsystem with speculative prefetch (the memory controller speculates on what is needed next prior to actual computations by analyzing the patterns in previous related fetches in memory) and a very robust execution core with a strong FPU easily dominate the list of architectures considered.

As many instances of the code can be ran in tandem on a computer as there are cores available by taking advantage of the multi-core nature of modern CPUs, provided that the computer is not expected to perform other demanding tasks, and has the total memory required. For example, the above mentioned AMD Phenom II is a quad-core CPU, and four simultaneous (single-threaded) simulations can be ran at once to greatly increase the speed. The Intel Xeon

5160 is a dual-core CPU, but there are two physical CPUs in that particular computer, and thus four simultaneous simulations can be run at once. Running more copies of the simulation than there are physical cores (not virtual cores such as cores exposed by Intel HyperThreading on the Core i7 above, a form of simultaneous multi-threading) only results in slow downs as threads from each of the simulation runs are quickly swapped in an interleave fashion across the same physical core.

For purposes of generating 64 bit binaries to potentially make use of larger grid sizes and for the apparent speed advantage gFortran has over G95 in running our code, gFortran will be used to generate all binaries for results.

4.4 Output Methods

Due to the intrinsic visual nature of the stellate phenomenon and the corresponding visual nature of the simulation, a proper representation of the simulation output would be images. Therefore, during the simulation runs, static images are generated to capture the state of the system under simulation. These static images are two dimensional representations of our two dimensional simulation.

For the liquid state, at every site on the lattice $\mathbf{x}$ at time t, the concentration of the site for the two species are computed by summing the mass vectors of the respective species to produce $\rho_p(\mathbf{x}, t), \rho_{np}(\mathbf{x}, t)$.

$$\begin{aligned} \rho_p(\mathbf{x}, t) &= \sum_{i=1}^{9} f_i^1(\mathbf{x}, t), \\ \rho_{np}(\mathbf{x}, t) &= \sum_{i=1}^{9} f_i^2(\mathbf{x}, t). \end{aligned} \tag{4.22}$$

Then an 3-tuple integer I_{RGB} is constructed to represent RGB (red, blue, green) values in increments of 001 from 000 and to a maximum value per channel of 255.

$$\begin{aligned} R(\mathbf{x}, t) &= NINT(\rho_p(\mathbf{x}, t) \times 255); \\ G(\mathbf{x}, t) &= 0; \\ B(\mathbf{x}, t) &= NINT(\rho_{np}(\mathbf{x}, t) \times 255), \\ I_{RGB}(\mathbf{x}, t) &= (R, G, B) \end{aligned} \tag{4.23}$$

where NINT is the nearest integer function (otherwise known as rounding).

Repeating this procedure over the entire simulation lattice yields a two dimensional matrix whose values are 3-tuple integers. A file is created with an appropriate header for Portable Anymap Format (PNM) image format. Then the values of the matrix are row-wise sequentially written to this file and stored in a pre-specified directory with the title indicating the size and

time step of the simulation along with a proper extension ".pnm".

In this representation, portions of the liquid richer in plagioclase compatible chemicals are redder in colour while opposite compositions are progressively more blue in colour. Symmetrically mixed regions appear purple and non-symmetrically mixed regions appear tinted toward red or blue depending on the dominant chemicals.

For the solid state, a similar scheme is used. White is used to represent one plagioclase structural type while black is used to represent the other. However, since there are no "mixed" regions in the solid state, tinting is used to indicate concentration of plagioclase in the solid cell. A solid cell richer in plagioclase appears either more black or more white depending on structural type, while a cell with a heavy amount of impurities appears more grayish.

Other objects such as borders and the initial nuclei can have any colour attributed to them in representation. The only constraint is that they do not appear the same as their surrounding colours. Thus, the borders and a nuclei bathed in the liquid state might appear to be black on an outputted image $R = G = B = 000$, or green $R = B = 0, G = 255$. This brings out the different elements the image contains in great relief.

Other types of output routines are possible, based on this method of rasterization. The image format ".pnm" can be viewed with the open source viewers such as Gimp (`http://www.gimp.org/`), ImageMagick (`http://www.imagemagick.org/script/index.php`) and ImageJ (`http://rsb.info.nih.gov/ij/`), or any other viewers/editors capable of decoding Portable Network Graphics formats. ImageMagick combined with some quick Bash scripts allows batch-conversions/downsamplings/stitchings of all images to compressed formats, such as ".png", for storage and/or animation purposes. Both ".pnm" and ".png" formats are also readily accessible with video and image stack rendering programs such as ImageJ. By importing the folder of output images into ImageJ as a virtual stack, one can then save the stack as a movie or GIF animation, this animation (usually in ".avi" format with frame-wise PNG compression) can be subsequently encodeded by an encoder to reduce size, such as mencoder into an efficient video format by adding temporal compression in addition to frame-wise compression (available with the free application ffmpegX (`http://www.ffmpegx.com/`), for Mac OS X). A good video format is h.264, which results in roughly 2 orders of magnitude reduction in average file size from ".png" compressed ".avi" movies. Numerical computing environments such as Matlab are also able to read and render the images into movies provided one writes the appropriate code/script.

One principal benefit of this format is simplicity. Since there are no algorithms for encoding the image, and the image is written in ASCII format, it can be easily accomplished with a routine inside of Fortran. This incorporation of an output routine inside of Fortran greatly speeds up the output process by eliminating calls to third party output libraries or piping the results to an external capture/plot software. Another benefit is maximal control. One can control the levels of quantization of each cell (max values of R,G,B need not necessarily be 255) and yield very high dynamic range images by choosing R,G,B such that $max(R) = max(G) = max(B) > 2^n - 1, n \geq 8$ if necessary. Therefore, the quantization can be theoretically greater than 8 bits per channel, or greater than 24 bits total. Since modern computer monitors can seldomly display full 8 bits per colour channel, the availability of quantizations to represent real numbered densities is not a limiting factor in this representation of the simulation state. However, due to the lack of compression (lossless or otherwise) of the image data, this format is very wasteful of hard drive space. For a simulation of size $N_{sim} = 1001$, the resulting file is 12.1MB. Writing one file every 1000 time steps for a typical simulation run of the size specified results in several GB of storage space used. However, this is not a concern for modern computers, particularly since one has direct control over the output interval.

The output intervals are chosen manually such that when the number of time steps in the simulation is a multiple of some specified interval, the output routine is called to produce an image.

In higher dimensions, serial sectioning the simulation to produce a sequence of images is also possible. However the amount of disk space usage would become very high. Other techniques such as ray-tracing or OpenGL would be preferred to visualize the end result. In the simplest way, one could dynamically link the output from the simulation into a rendering engine, which then renders and displays the visualization on screen. For an example of how such techniques could be used to elegantly display simulation results, or to monitor the ongoing simulation in real time, see Thürey's excellent thesis on simulations of metal foam bubbles (Thürey, 2003).

In addition to image based outputs, other outputs are also possible such as position-data-tuples in the form of an ASCII file, which can be used to plot data for potential analysis. These methods are trivial and thus not discussed here.

Another type of output is used by the simulation to save/load desired liquid states to/from files. This output routine sequentially outputs the liquid state to a text file with a header

capturing some of the parameters such as the size of the grid, the amount of time elapsed, the viscosities, the parameter R and the value of σ_{std}. The corresponding loading routine can then read this formatted file to exactly (to within rounding errors) reproduce the conditions of a prior simulation. Lastly, another routine checks the save/load directory specified to find saves matching the parameters currently used. The operation of these routines can be automated in sequential runs. Typically, the save-checking routine tries to find a compatible save file, and when found, informs the main program body to call the corresponding routine to load the file. Subsequent simulations are run based on this initial condition in the file. If no saves matching the specified criteria are found, the liquid state simulation is run instead, and at the desired time prior to the onset of crystallization, the save routine is automatically called to save the current liquid state to a file. This allows the simulation to consecutively load the same file as initial conditions used for crystallization without actually re-simulating the entire liquid state, and it retains consistency in the more complicated parameter sweeps when the solid state is run together with the liquid state. Due to the stochastic nature of the solid state growth, saving the solid state along with the liquid state is less useful. This save/load method allows the simulation to quickly shortcut the liquid only time steps for every successive run with identical liquid parameters.

Chapter 5

Preliminary Results

In this chapter, we will explore the results generated by the model when individual parameters are varied. The purpose will be to illustrate the effects of various parameters and to provide a transition to realistic plagioclase stellate growth scenarios.

5.1 Liquid State Parameters

Several common parameters to all the simulation runs of this liquid-only section are set as follows unless otherwise stated in the subsections:

$$\begin{aligned} \text{Lattice Geometry} &\rightarrow \text{Square} \\ N_{sim} &= 1001 \\ \Delta t &= 1 \\ \Delta x &= 1 \end{aligned} \tag{5.1}$$

5.1.1 Viscosity Parameter ω

ω_i controls the lattice collision frequency, which sets the viscosity of chemical species i. In sweeping this parameter, two case scenarios will be considered: a synchronous sweep of ω_i, where $\omega_1 = \omega_2$, and a differential sweep of ω_i, where $\omega_1 \neq \omega_2$.

First we consider the synchronous sweep. Since the valid range of ω is from 0.5 to 2, we will cover most of that range by choosing a set of representative values $\omega \sim 0.4, \omega \sim 1.0, \omega \sim 2.0$. Values of ω very close to 2.0 yield a value of τ very close to 0.5, and the resulting viscosity is nearly 0. Since the simulation has finite precision (numerical nature of our simulation), values of ω too close to 2.0 present stability challenges. Therefore, the least viscous side of the parameter sweep shall have ω restricted to ~ 1.9. The exact value of ω that establishes the stability boundary in the simulation depends on our choice of the surface tension parameters α_i. For larger surface tension values, the least viscous value of ω is reduced somewhat. For our choice of surface tension parameter $\alpha_i = 0.01$ in the following runs, a numerically safe value of ω is about 1.9. This numerical value of α_i is chosen through simulation runs observing the relative stability and the rate at which the mixture coarsens. For values of ω such that $1.9 \leq \omega \leq 1.99$, instability randomly creeps in, yielding Not-a-Number (NaN) values for grid masses in random locations on the grid where division by zero (numerically) has occurred, invalidating the entire simulation run. No such restriction applies for the high viscosity end of ω. However, to stay well away from numerical roundoff problems, the lowest value of ω will be taken as 0.51. Therefore, the synchronous case of ω sweep will incorporate $\omega = 0.51, \omega = 1.0, \omega = 1.9$ as separate cases. Other values of ω are of course possible, but not necessary to illustrate the behaviour of the system as a function of changing ω. The simulation time is an arbitrary length of time deemed necessary to illustrate the differences of behaviours of the systems through time. Since ω controls viscosity, it is expected that lower viscosity values should coarsen more quickly. This synchronous sweep of ω is demonstrated in figure 5.1.

One can clearly see the differences presented by different ω values in figure 5.1. For small values of ω, the fluid mixture is expected to be viscous, and thus the phase separation should be slower. This leads to slow coarsening of the simulated mixture over time. For high numerical values of ω, the fluid behaves chaotically, and while the fluid mixture coarsens rather rapidly, the boundaries of the two mixtures are not necessarily smooth. There is also a distinct lack of smaller domains, either as separated islands, or islands inside of domains of other islands. In this case,

the chaotic fluid is able to relax fast enough such that the interface kinetics caused by the surface tension can not adequately bring all of the domain boundaries to pseudo-equilibrium shapes as exemplified by a lack of smooth curvilinear surfaces. For the mixture with a intermediate value of ω, the system coarsens at a medium rate. In particular, one can observe the fractal nature of the domains through time whereby the general patterns of the system undergoing mixing looks self-similar if one zooms in on the pictures of earlier time steps, or zooms out on the later time steps, to within limitations of the finite size of the grids used.

For the differential sweep, there is no particular reason to choose numerical values that are different from the procedure above for the synchronous sweep, except the pairings of the values of ω will be differential. Therefore, the numerical values of the various parameters in this section are still the same as above.

In the differential sweep as shown in figure 5.2, the contrast of viscosities is clearly seen when comparing the first column of results to the last column. In the first column, the greater viscosity of the red phase controlled by ω_1 is seen to form a background continuum inside of which blue islands reside. In the last column the exact opposite occurs while in the middle column, neither forms a continuum, and both can be observed with equal semblance. The higher viscosity species in the differential runs relaxes slower toward equilibrium, and thus the faster relaxing species form geometrical shapes with the least surface area possible, which in the case of two dimensions, are circles. For sufficiently contrasting viscosities of equal proportion binary fluid mixtures, it is then expected that the more viscous phase preserves the percolation structure state longer. The results of the first and last columns are equivalent statistically under a symmetric exchange of colour.

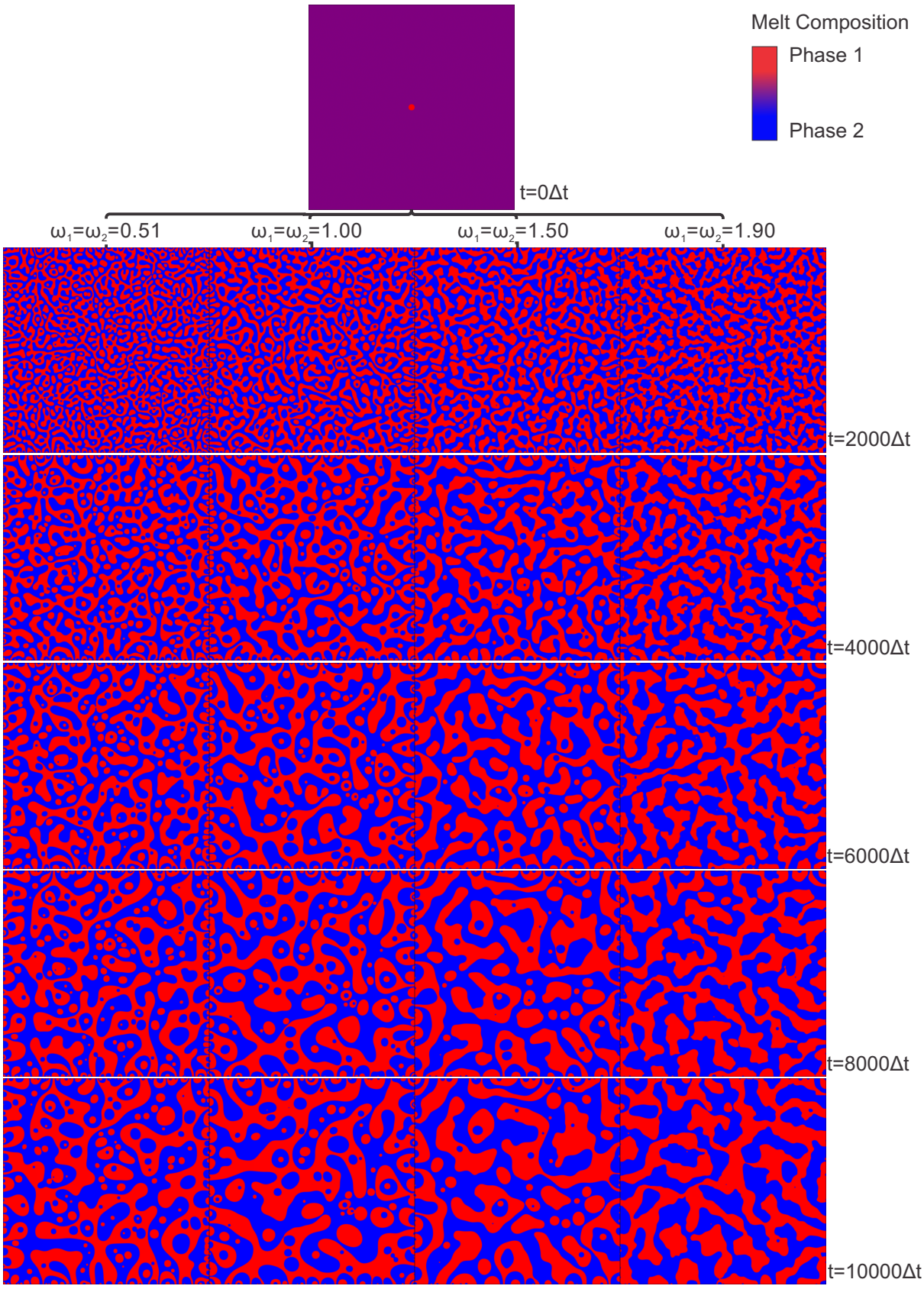


Figure 5.1 – Graphical representation of simulation results as a function of ω. Here $\omega_1 = \omega_2$, $R = 0.5$, $\alpha_1 = \alpha_2 = 0.01$, $\beta_0 = 1$. Other parameters not listed here do not affect the results.

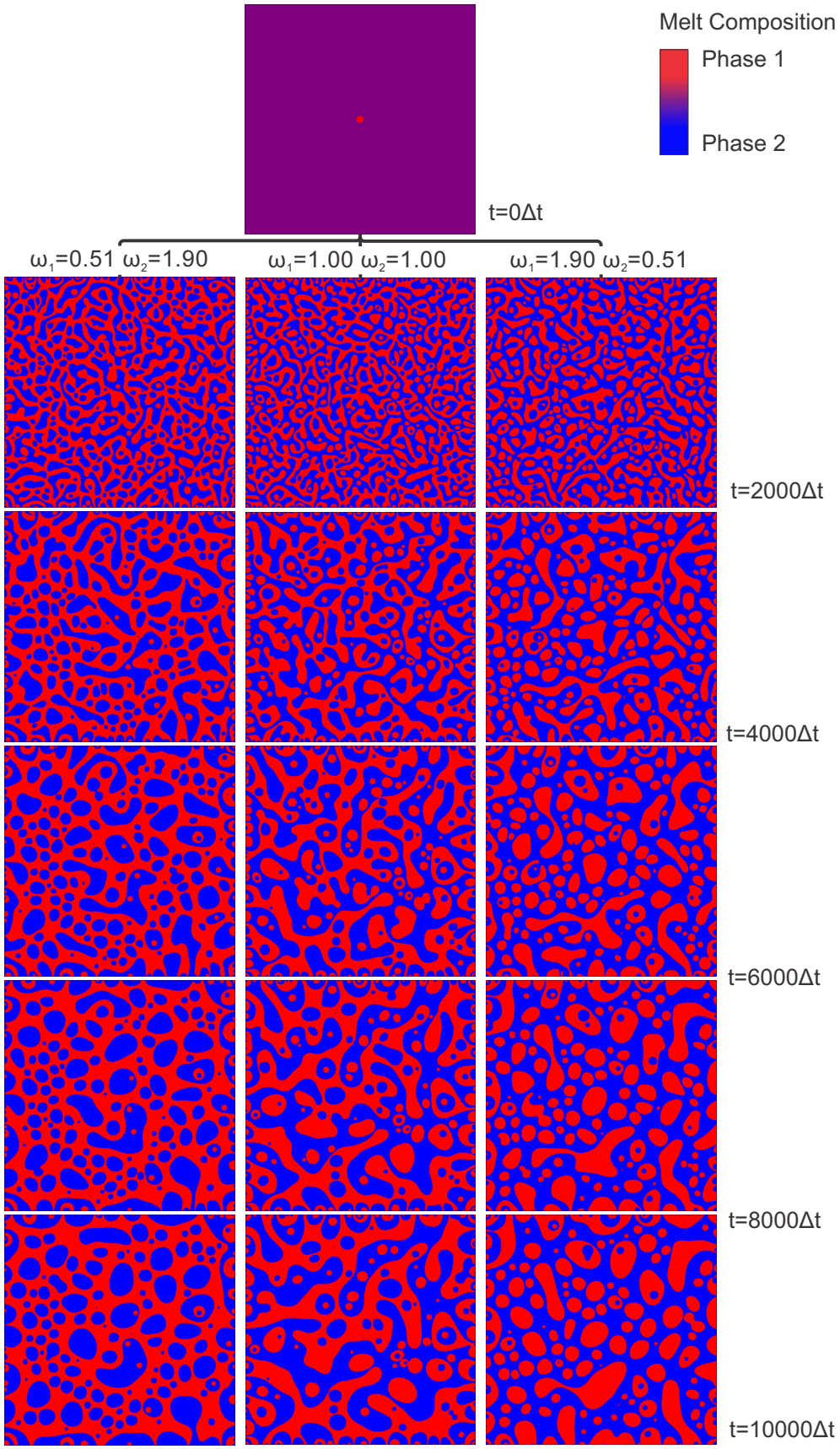


Figure 5.2 – Graphical representation of simulation results as a function of ω. Here $\omega_1 \neq \omega_2$, $R = 0.5$, $\alpha_1 = \alpha_2 = 0.01$, $\beta_0 = 1$. Other parameters not listed here do not affect the results.

5.1.2 Surface Tension Parameter α_i

The range of the surface tension parameter α_i is quite large, encompassing several orders of magnitude. There is no real (physical) lower limit to this parameter theoretically. However, due to the numerical nature of the simulation, a first order guess at the lower limit of the parameter α_i would be the constraint of fixed number of significant digits possible in the simulation. For 17 digits, the smallest possible number is obviously 1×10^{-16}. However, numbers this small are prone to numerical errors whenever the number is involved in computations. Therefore, a practical lower limit to this parameter for "double precision" variables in Fortran is at least one order of magnitude higher than the least significant digit. In general, one can consider the type of operations a number undergoes algorithmically to arrive at the lowest acceptable limit. For example, if a number is divided by another number once, whose upper limit is on the order of 10, then the lowest limit one might expect a parameter to be numerically valid would be the least significant digit multiplied by this divisor. Despite this consideration, the closer a number approaches the least significant digit, the more numerical errors contribute. For this reason, one might choose numbers in such a way as to minimize the contribution of numerical errors. Thus if we consider only numerical roundoff errors, it is clear that fluctuations to the principal number due to roundoff errors are on the order of half of the least significant digit, i.e., 50% in the lower limit of a representably non-zero number. If we choose the principal number one order of magnitude higher, this error decays to 5%, and so on. For a realistic simulation, one might choose a lower limit far enough from the least significant digit, while preserving a meaningful range of parameters. Thus, the lower limit to our surface tension parameter α_i can probably be safely set at 10^{-16} without worrying about unsurmountable numerical errors, although in practice, such low numbers require a long simulation time to generate meaningful patterns. Although there are no established higher limits to this surface tension parameter other than unity, an excessively high value of α_i combined with a high value of ω yields instabilities in the simulation, as mentioned earlier. A practical upper limit of α_i established by observing simulation stabilities for $\omega \sim 2$ is about $\alpha_i = 0.01$ which as we will see later on, is far more than we need. This is quite a sufficient range for our purposes. We shall explore this surface tension parameter from this upper limit to 10^{-4} covering three orders of magnitude in base ten logarithmic steps. The results of this sweep is shown in figure 5.3.

A differential sweep of α_i is not needed since the behaviour of the two fluids in un-mixing is

dominated by the fluid with the largest surface tension. This can be understood by considering the action of the immiscible collision operator. Each of the two phases in the fluid mixture is independently perturbed by an amount weighed by α_i away from the local interface. Therefore, even if one of the two phases exhibit a near-zero surface tension, the other phase is able to move away from this region via the perturbation step.

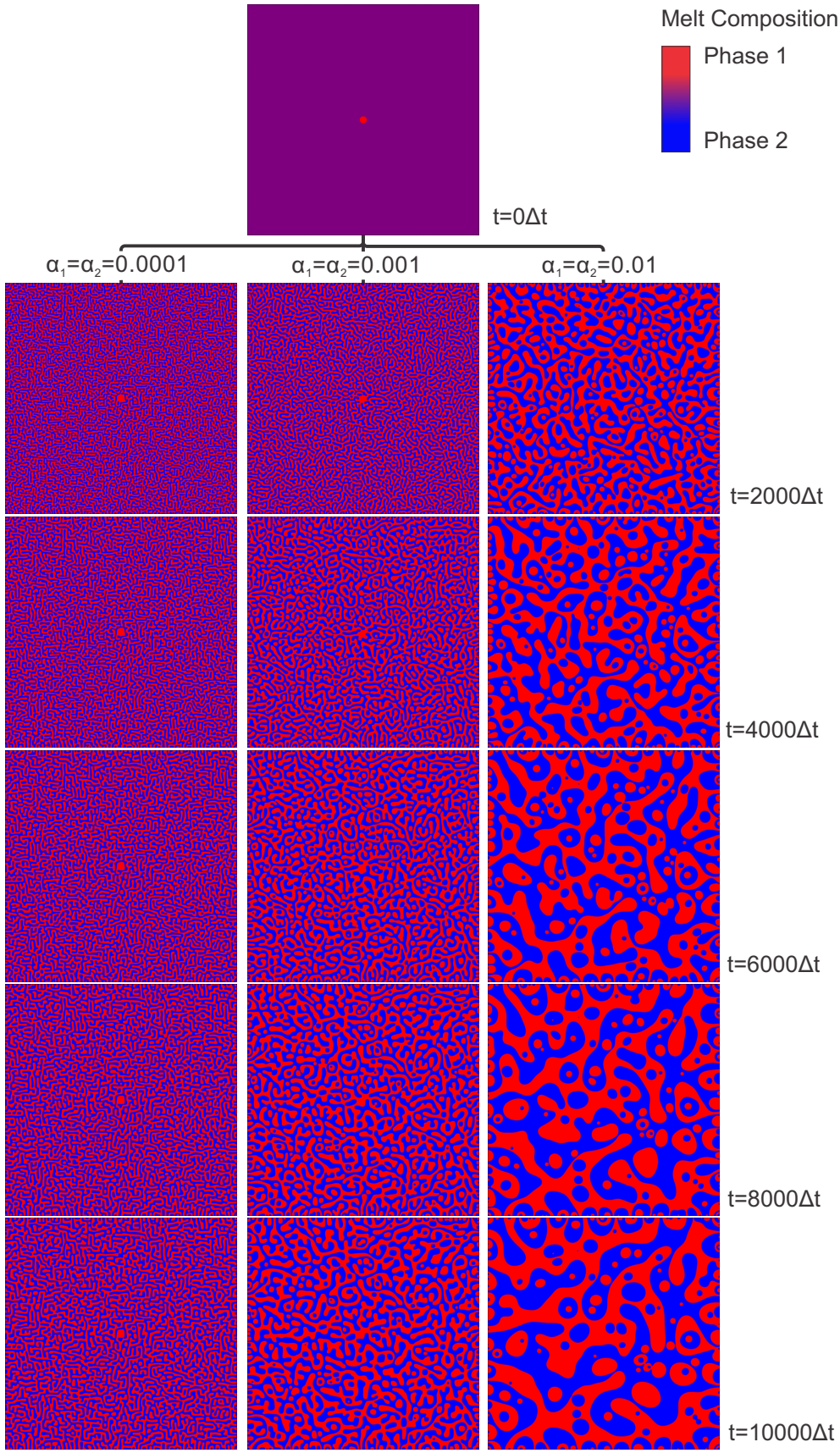


Figure 5.3 – Graphical representation of simulation results as a function of α_i. Here $\omega_1 = \omega_2 = 1.0$, $R = 0.5$, $\alpha_1 = \alpha_2$, $\beta_0 = 1$. Other parameters not listed here do not affect the results.

5.1.3 Global Plagioclase Enrichment Parameter R

It is expected that the proportions of fluids (volume fractions) participating in an un-mixing process directly control the patterns produced. In particular, for all other parameters fixed, a binary mixture has a threshold-type behaviour for transition from a continuous percolation-type pattern to an isolated emulsion (droplets) for any one of the two fluids in the mixture. In general, the higher amount of a particular phase in a binary mixture, the more likely it will produce a percolation type pattern while the lower amount past a threshold exhibits emulsions. For gradual transition from emulsions to percolation type patterns, see the results of parameter sweep of R in figure 5.4. In this figure, we can clearly see the enlargement of red droplets as R is increased to form a continuous percolation type structure and eventually, as $R > 0.5$, form a continuous background (not shown, but imagine substitution of blue for red and vice versa in the series of pictures in figure 5.4). The converse is true for blue droplets if we lower the value of R past 0.5. This is obvious, due to the concentration-symmetry of the two phases in a binary system. In higher phase mixtures, such simple symmetries can also be found if we consider a dominant phase and all other phases as a complementary (in the sense of a dual) phase.

As expected, the percolation structure appears to coarsen much more rapidly due to hydrodynamics compared to the droplet structures for $R < 0.5$, where only diffusion between domains is possible.

When there are other factors such as viscosity differences between the fluids in the mixture, the lower viscosity phase tends to develop a percolation type structure. It will be seen shortly that the parameter β_0 (and by extension, parameter C) also plays a big role in the macroscopic appearance of percolation structures.

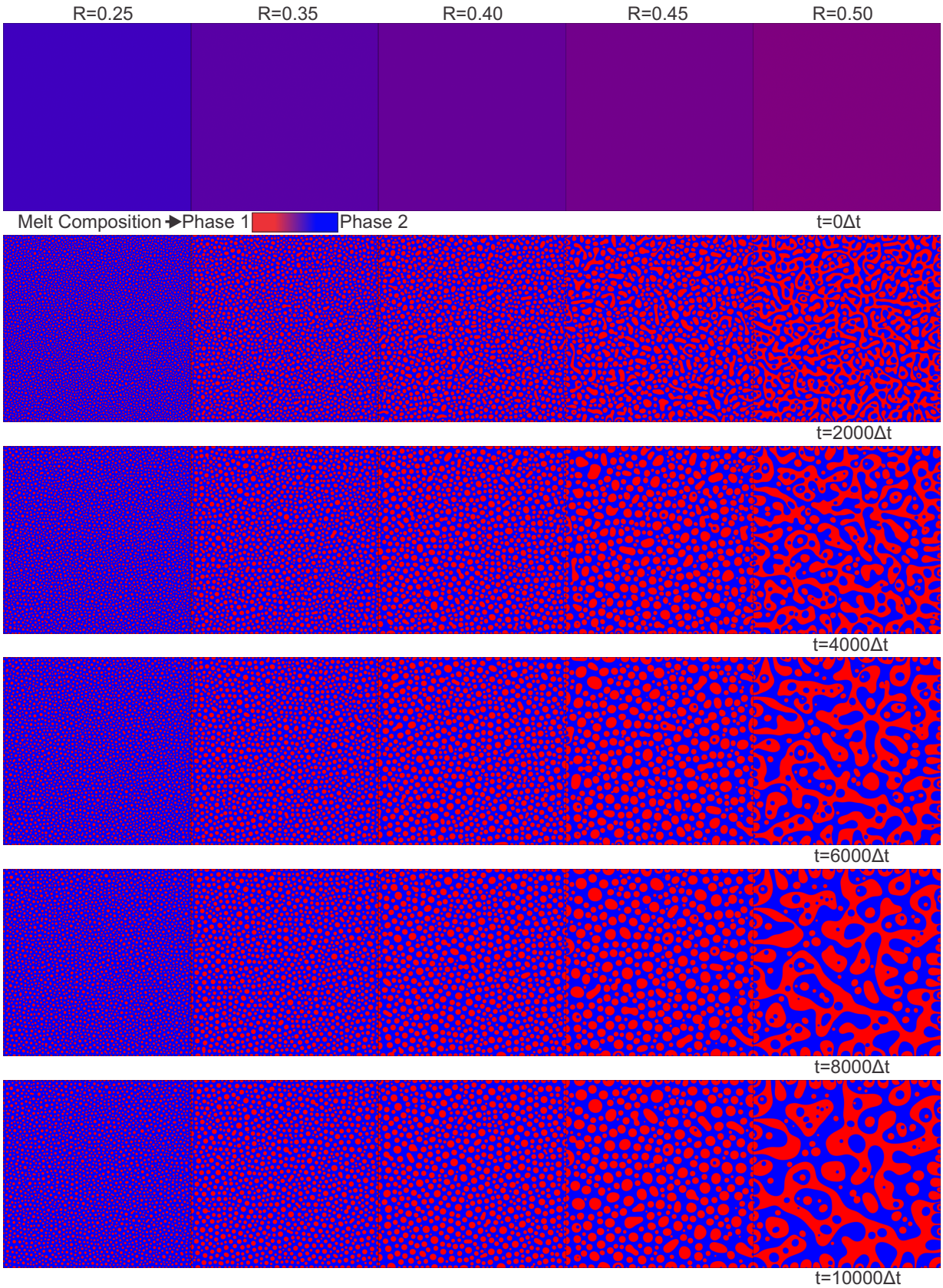


Figure 5.4 – Graphical representation of simulation results as a function of R. Here $\omega_1 = \omega_2 = 1.0$, $\alpha_1 = \alpha_2 = 0.01$, $\beta_0 = 1$. Other parameters not listed here do not affect the results.

5.1.4 Interfacial Region Mixing Length Parameter β_0

As alluded to above, this parameter pays a large role in addition to R in controlling the existence of percolation networks versus isolated droplets. This parameter effectively controls mixing length in the immiscible model although no direct scaling is available. A sweep of this parameter over its domain $0 \rightarrow 1$ will show the effects of this parameter on the result. This is presented in figure 5.5.

One can observe generally the interfacial fuzziness of low values of β_0 that transitions to sharp, nearly single-celled interfaces as we increase β_0 toward unity. This is an important parameter for controlling crystal growth behaviour in a macroscopic simulation. As one approaches the microscopic scale from the macroscopic scale, one expects that well defined interfaces between domains become replaced by average occupation probabilities of molecular constituents. Thus, at a sufficiently small microscopic scale, around that of the crystal unit cell macromolecular sizes, we expect the interfaces of these domains to be very fuzzy and that they can only be defined by temporal and spatial averages. This coincides with the idea of ramified droplets (or spinodal nucleation). Higher values of β_0 can be interpreted in this way to represent a more macroscopic description of the same system where we have zoomed out far enough such that the interfacial regions become small compared to the domain sizes. For a simulation with the lattice spacing set at ~ 1 mm, low values of β_0 yielding visibly fuzzy interface regions probably best suit realistic scenarios targeting the genesis of stellate textures.

Another interpretation of the results for β_0 comes from the way β_0 controls the tendency for the fluids to separate. For simple systems with short range interactions dominated by homophase interactions in the sense that heterophase interactions are purely repulsive such as in water and oil mixtures, a large β_0 is likely. For systems exhibiting strong heterophase interactions such as multi-saturated systems, it is logical that some of the phases tend to separate into distinct domains (into either independent or mixed phase domains) while other phases are mobile and exchanged from domain to domain with some probabilistic affinities or preferentially concentrate at domain interfaces. At the interface between a melt and a crystal with a chemistry containing a range of dissimilar ions, the melt might exhibit multiple temperature dependent values of β_0, one for each ion type. The β_0 considered here should then be considered as a weighted average of β_0 values for each ion the final crystal is composed of, at the temperature of consideration.

In general, in higher order mixtures, some of the phases can be seen as either enhancing the

separation of the dominant phases by establishing bonds for only homophase interactions, or seen as emulsifiers establishing bonds for heterophase interactions. Therefore, it is less likely that an average multi-phase melt mixture would exhibit extremely sharp interfaces (at every domain boundary) while the mixture is still held at high temperatures. For rapidly quenched mixtures, this need not be true if the observation is made after the quenching.

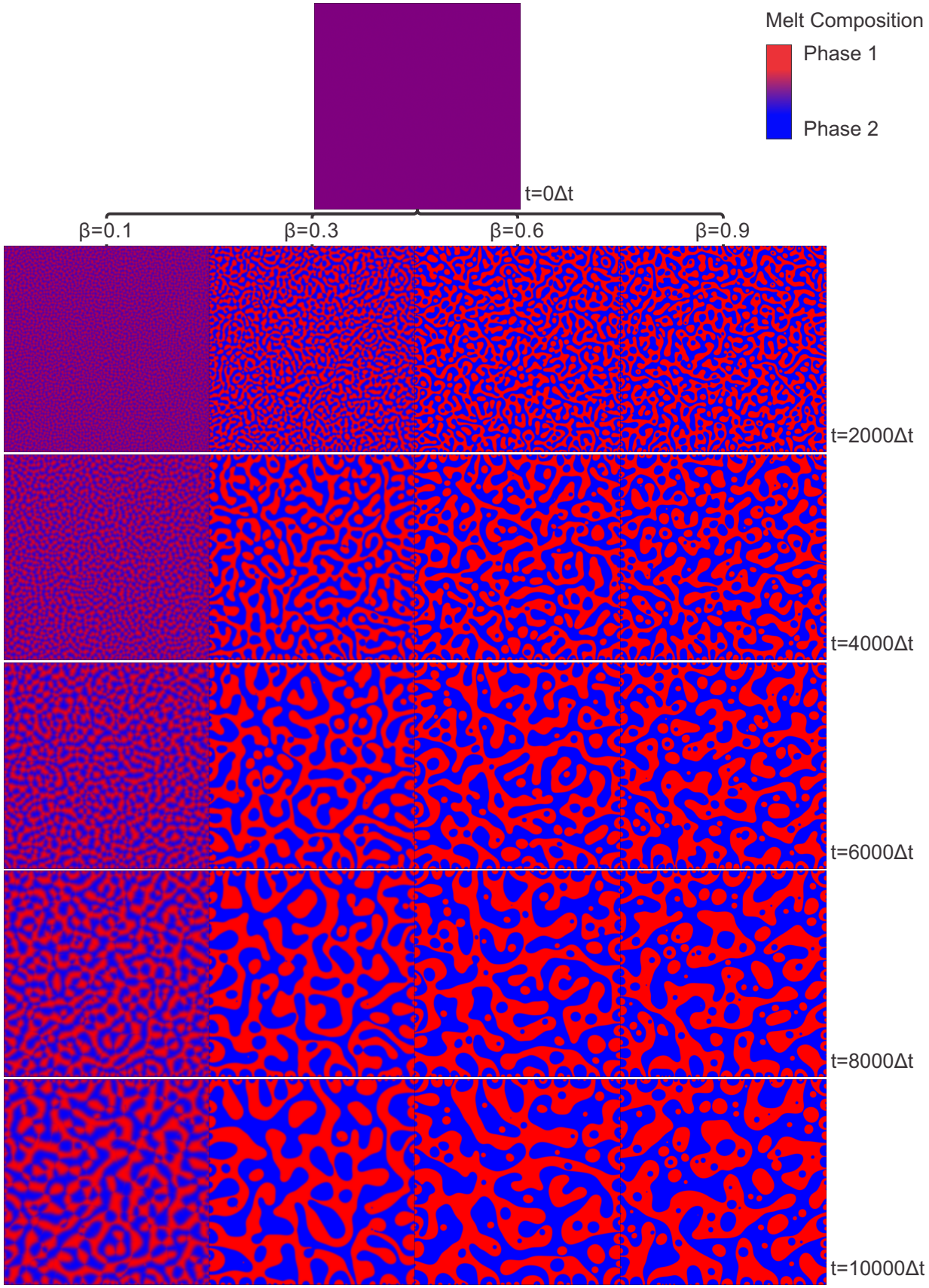


Figure 5.5 – Graphical representation of simulation results as a function of β_0. Here $\omega_1 = \omega_2 = 1.0$, $\alpha_1 = \alpha_2 = 0.01$, $R = 0.5$. Other parameters not listed here do not affect the results.

5.2 Solid State Parameters

To better illustrate the action of the parameters and guarantee reproducible results, the liquid state will be artificially generated. This follows the reductionist approach set out earlier in the liquid state parameters whereby only the effects of the specific parameter under consideration will be illustrated. The behaviours of coupled actions are not shown except for the perturbation parameters which specifically feed back into the liquid state and coupling parameters.

5.2.1 Gaussian Parameters b, σ

The parameters (b, σ) controls the growth rate as a function of the liquid state. The effect is probabilistic. However, for a large enough growth front, and sufficient number of time steps, one can observe the general trend posed by the adjustment of these parameters. In particular, σ controls the standard deviation of the Gaussian function, and thus, the sensitivity of the solid state to the liquid state. For $f[\rho_1 = (b - m\sigma)]$ where ρ_1 is the chemical component currently making the phase transition, m an integer such that $m > 1$, the growth rate is considered at equilibrium in the sense that the time scale needed to produce significant growth is much larger than the time scale of the processes in the melt state (such as unmixing, or other minerals crystallizing, etc).

b controls the center (or mean) of the Gaussian. Since a binary chemical mixture has two components, setting b at 1.0 implies that the growth rate of the chemical product is controlled by one chemical reactant. In this case, the single chemical reactant ρ_1 possess a chemical stoichiometry that is identical to the final product, and the difference being internal order in the sense that the final solid state exhibits strong long ranged order which is generally not present in the liquid state. In the liquid state, ρ_1 exhibits some short range internal order near the critical point, but no crystalline long range order. In the solid state ρ_1 should exhibit long range order as a result of the ordering process of the phase transition. When $b = 1.0$, the stoichiometry of the final crystalline product is $1.0\rho_1 : 0.0\rho_2$. Thus ρ_1 can be considered enriched relative to ρ_2. For stoichiometric ratios involving a mixture, when $b \neq 1.0$, mixed regions with the specified ratios of $b\rho_1 : (1 - b)\rho_2$ will be considered enriched instead. The value of m can be taken at a value such that $f(b - m\sigma) \ll 1$. Therefore, we expect that a growth front will appear to exponentially slow down as the melt phase become more depleted

and eventually becomes so slow that the hydrodynamics of the melt state will dominate the simulation behaviour. To show this behaviour, we can set up a liquid system with a pre-defined initial state with a concentration profile that changes spatially in a gradual way. To suppress the effects of the liquid state simulation, we hold the liquid constant by only running the solid state algorithms. The results of such a scenario is shown in figure 5.6. It can be seen the radial size of the solid state as a function of time depends on the value of σ where larger values of σ results in larger sizes as the solid state grows further into the less enriched regions of the liquid state.

Since b controls the center of the Gaussian, it controls the concentration with the maximal growth rate. The choice to have $b = 1.0$ is obvious if we assume ρ_1 is stoichiometrically identical to the final solid product. However, for other stoichiometric proportions of chemicals needed to make a specific mineral with $b \neq 1$, then the regions with the fastest growth will be away from the centers of the domains, where the proper mixture is reached. In this case, the value of b is less than unity. We can also sweep this parameter over its range to illustrate its effect. For this scenario, a linear gradient of ρ_1 to ρ_2 is constructed over the simulation area and a line of nuclei is seeded at the bottom of the simulation area. Growth then takes place upwards with the local growth rate dependent on $f(\rho_1)$. The results of this scenario are shown in figure 5.7. They are as expected.

Allowing $b = 0.5$ and sweeping σ from 0.10 to 0.20, we can observe the change in the solid state width as the composition of the local liquid state ventures away from the maximal growth rate mixture $0.5\rho_1 : 0.5\rho_2$. The results are shown in figure 5.8.

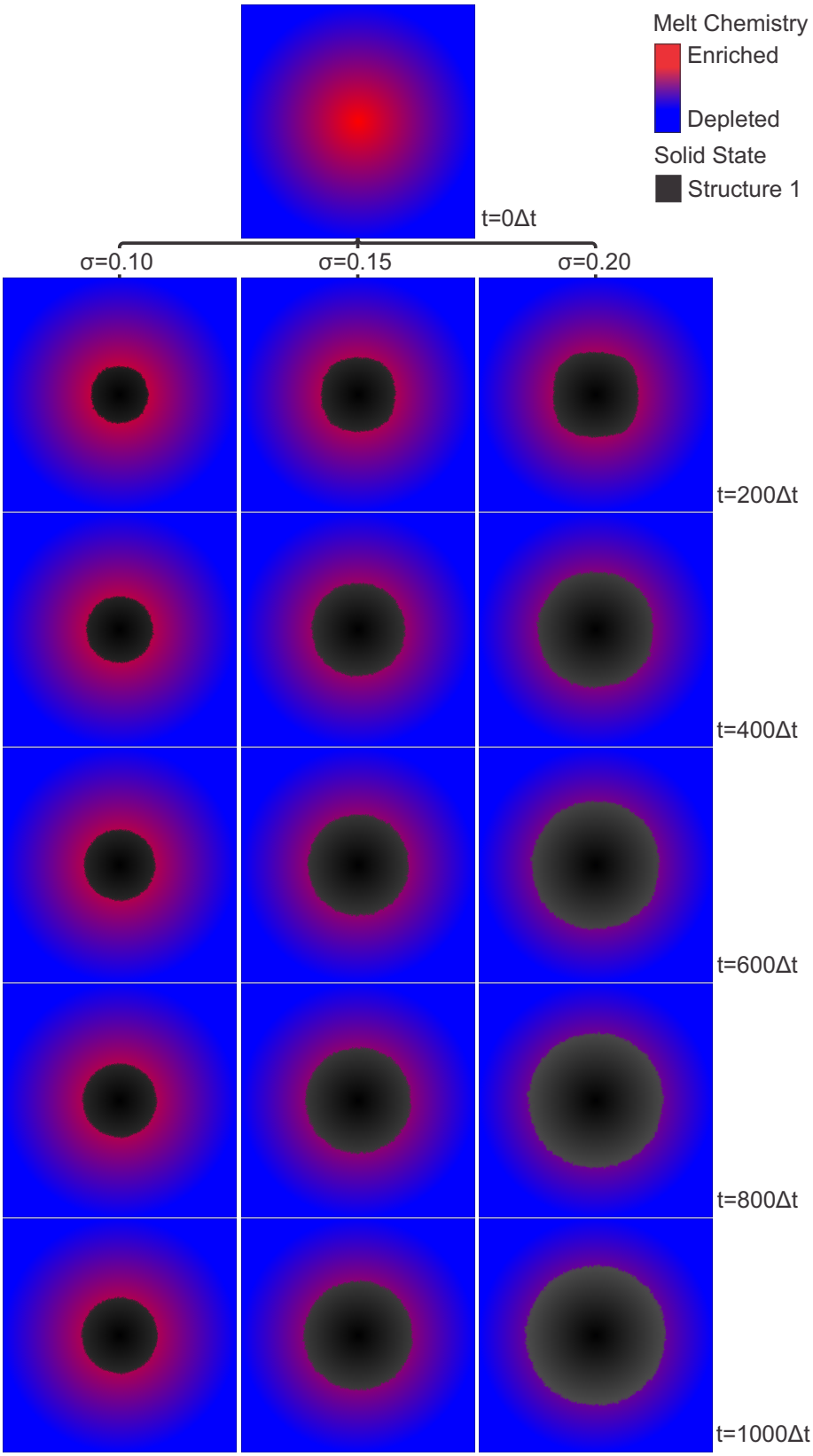


Figure 5.6 – Graphical representation of simulation results as a function of σ. Here $b = 1.0$. Enriched liquid in the legend is the same as ρ_1. Concentration in the liquid state decays radially from center where $R = 1.0$ to the edges of the simulation area where $R = 0.0$. Other parameters not listed here do not affect the results.

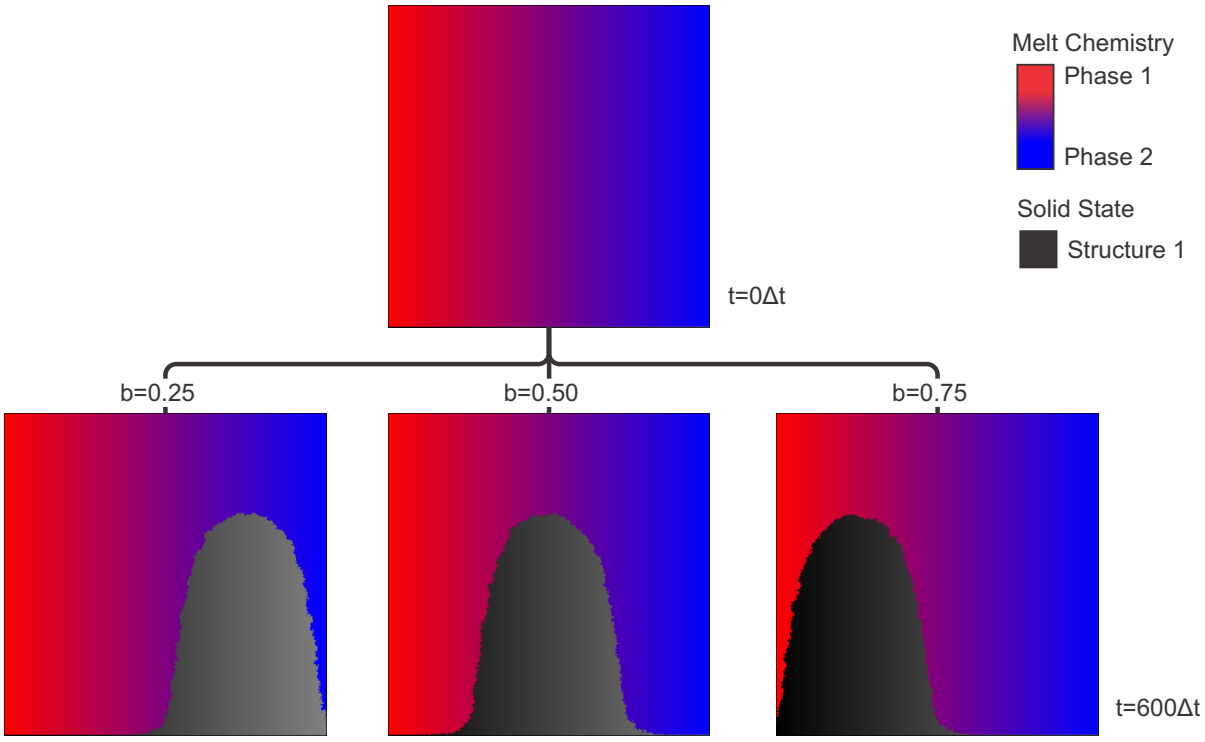


Figure 5.7 – Graphical representation of simulation results as a function of b. Here $\sigma = 0.10$. Concentration ratio of $\rho_1 : \rho_2$ varies from the left $(1 : 0)$ to the right $(0 : 1)$ equivalent to $R = 1.0$ to $R = 0.0$. $N_{sim} = 501$. Other parameters not listed here do not affect the results.

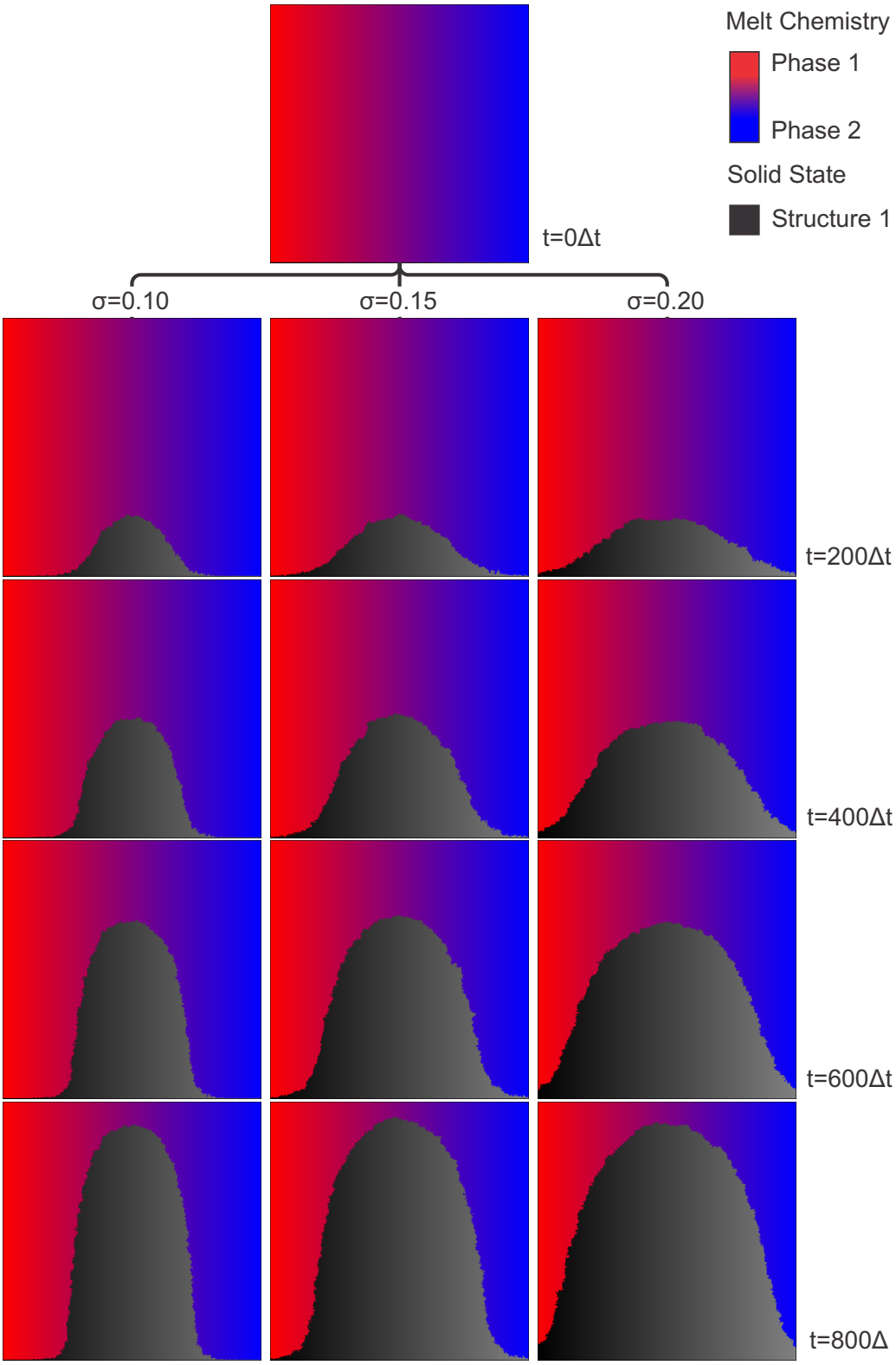


Figure 5.8 – Graphical representation of simulation results as a function of σ. Here $b = 0.5$. Concentration ratio of $\rho_1 : \rho_2$ varies from the left $(1 : 0)$ to the right $(0 : 1)$ equivalent to $R = 1.0$ to $R = 0.0$. $N_{sim} = 501$. Other parameters not listed here do not affect the results.

5.2.2 Anisotropy Parameter Table $p2$

The anisotropy parameter table controls the equilibrium shape of the individual crystals by assigning a probabilistic attachment coefficient to the surfaces of the crystal as a function of the natural lattice directions. In practice, any number of combinations of these values are possible, however due to symmetry arguments, only certain combinations are unique. The main effect of these parameters is to allow anisotropic growth to facilitate the occurrences of epitaxy. A completely isotropic parameter table can be given in table 4.1. For the purpose of exploring the effects of the parameter table on the equilibrium shape of the solid state, only one structural type is needed, and any rotation of that anisotropy table can be achieved by sequential permutations of the table $p2$. In this manner, we need only construct a set of values for specific equilibrium aspect ratios (defined by the ratio of the largest coefficient to the smallest coefficient). Therefore, table 4.1 can be referred to as a $1:1$ type with full 8 directions of neighbours. While it is sufficient to illustrate the anisotropic behaviour of the crystal as a function of the parameter tables without using diagonal neighbours, actual simulation runs show that the use of diagonal neighbours is crucial to obtain smooth, well defined epitaxy boundaries.

The 8 neighbours anisotropy parameter tables are computed by assigning the maximum attachment probability directions ($p2^{max}$) as the aspect ratio (Asr) multiplied by the minimum attachment probability directions ($p2^{min}$) and somewhat arbitrarily assigning an attachment probability between $p2^{max}$ and $p2^{min}$ for the diagonal directions $p2^{d}$. As constrained by geometry, diagonal coefficients represent the probability of attachment of a neighbour whose Euclidean distance is a factor $\sqrt{2}$ longer, and thus, by inverse weighting, the diagonal coefficients should be at least smaller by the same factor. A further geometric constraint can be sought, if we consider the equilibrium shape of the crystal to be a two dimensional object with a tendency to minimize surface area such that the surface of the object is always convex. This understanding forces us to choose a diagonal coefficient that is much closer to the minimal values and certainly much smaller than the maximal values for high aspect ratios. This situation is illustrated in figure 5.9. Through some experimentation, it is found that the diagonal values are best chosen close to, and somewhat smaller than the minimum attachment probabilities.

In the tables listed below, the diagonal values are chosen to be half that of the minimum probabilities, although any factor $n_d \leq 1$ seems to suffice. To restate the above mathematically for each structural type i, and combining the restriction stated in equation 3.65, we arrive at a

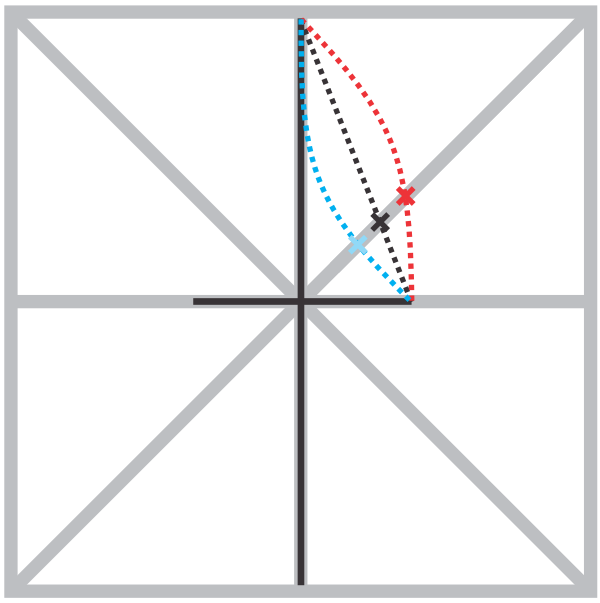


Figure 5.9 – Choice of diagonal coefficients relative to the maximal and minimal coefficients. Maximal and minimal coefficients represented by black bars vertically and horizontally respectively. The black dotted line is the linear line connecting from maximal to minimal values and represents the inflection line of surface curvature. Curves joining the maximal and minimal values below this dotted line would result in a concave surface while curves above the black dotted line results in a convex surface. The intersections of the diagonal coefficient with the black dotted line and a hypothetical convex surface (red dotted line) and a concave surface (blue dotted line) are shown as respectively coloured X's. Only coefficients resulting in convex surfaces are chosen, which limits the choice of $p2^d$. $N_{sim} = 501$.

set of equations that govern the coefficients of $p2$:

$$\begin{aligned} p2_i^{max} &= Asr \times p2_i^{min}; \\ p2_i^{d} &= n_d \times p2_i^{min}; \\ \sum_{i=1}^{8} p2_i^{j} &= 1, \end{aligned} \tag{5.2}$$

where n_d is a positive factor such that $n_d \leq 1$. For convenience, n_d is usually taken to be a simple fraction, such as $1/2$, and thus $n_d = 1/2$ unless otherwise specified for subsequent runs.

In addition, since the two structural types $i = 1, 2$ are two-fold rotationally invariant, we require that under a four-shift (eight neighbours divided by two folds of symmetry) permutation of index $j \rightarrow j \pm 4$ the value $p2_i^j = p2_i^{j\pm4}$, treating each set of entries $p2_i^j$ as a cyclic group

in index j. In fact, since the two structural types are usually chosen to be orthogonal to each other, a two-shift permutation brings $p2_i^{1,2}$ to its corresponding orthogonal set $p2_i^{2,1}$.

A few anisotropy parameter tables used are listed in tables 5.1, 5.2 and 5.3. To obtain higher aspect ratio anisotropy parameter tables, one can compute them in a similar fashion using equations 5.2 as defined above.

If a simulation scenario is setup such that the entire simulation area is initialized with enriched melt, and a nucleus is seeded centrally and epitaxy is forbidden, we can observe the effects of these anisotropy tables on the resulting shape of the crystal captured at some finite time after the start of crystallization. In this way, results presented with increasing aspect ratios can be seen below in figure 5.10.

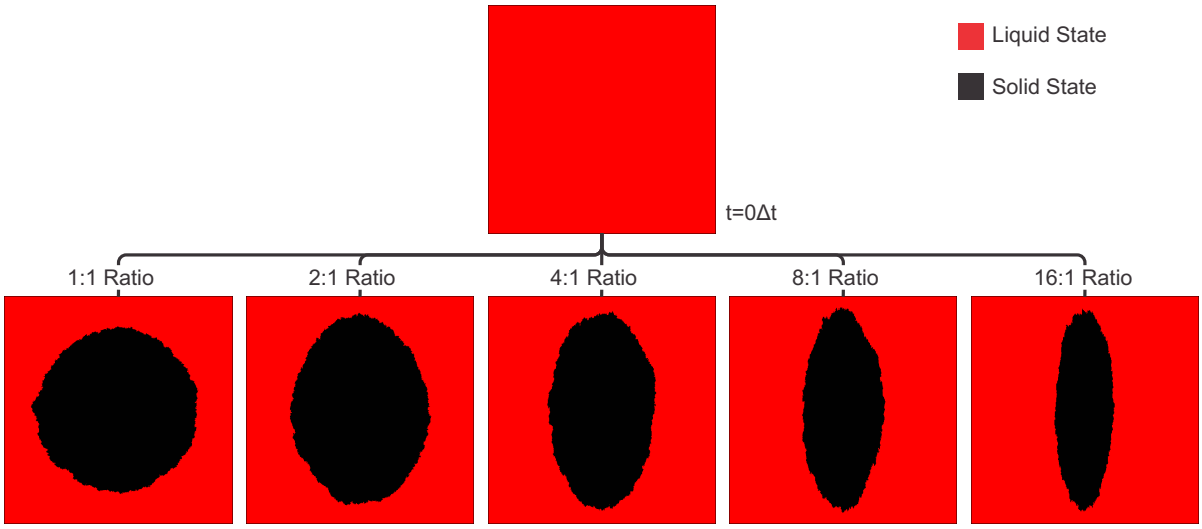


Figure 5.10 – Increasing aspect ratio and the effect on crystal morphology. The largest coefficient is aligned vertically. $R = 1$. Epitaxial twinning is not allowed. Other parameters not listed here do not affect the results. $N_{sim} = 501$.

Table 5.1 – $1:1$ aspect ratio table of $p2_i^j$ for $j = 1, 2$ anisotropy types and 1...8 number of neighbours.

i,j	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8
j=1	1/8	1/8	1/8	1/8	1/8	1/8	1/8	1/8
j=2	1/8	1/8	1/8	1/8	1/8	1/8	1/8	1/8

To observe how epitaxial interactions depend on the anisotropy, we can construct a test case scenario from a central nucleus such that the enriched liquid in narrow channels extend radially outward in the principal directions of the D2Q9 lattice. It can be observed that epitaxy is used

Table 5.2 – $2:1$ aspect ratio table of $p2_i^j$ for $j = 1, 2$ anisotropy types and 1...8 number of neighbours.

i,j	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8
j=1	1/8	1/16	2/8	1/16	1/8	1/16	2/8	1/16
j=2	2/8	1/16	1/8	1/16	2/8	1/16	1/8	1/16

Table 5.3 – $4:1$ aspect ratio table of $p2_i^j$ for $j = 1, 2$ anisotropy types and 1...8 number of neighbours.

i,j	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8
j=1	1/12	1/24	4/12	1/24	1/12	1/24	4/12	1/24
j=2	4/12	1/24	1/12	1/24	4/12	1/24	1/12	1/24

as a means to navigate these radial channels. For low anisotropy $2:1$, or no anisotropy $1:1$, it is observed that the structural composition of the solid state is orientation-independent. For higher degrees of anisotropy, it can be clearly seen that the growth of different structural types into different channels becomes orientation-dependent. This is illustrated in figure 5.11. The results below illustrate common results since single run results are stochastic.

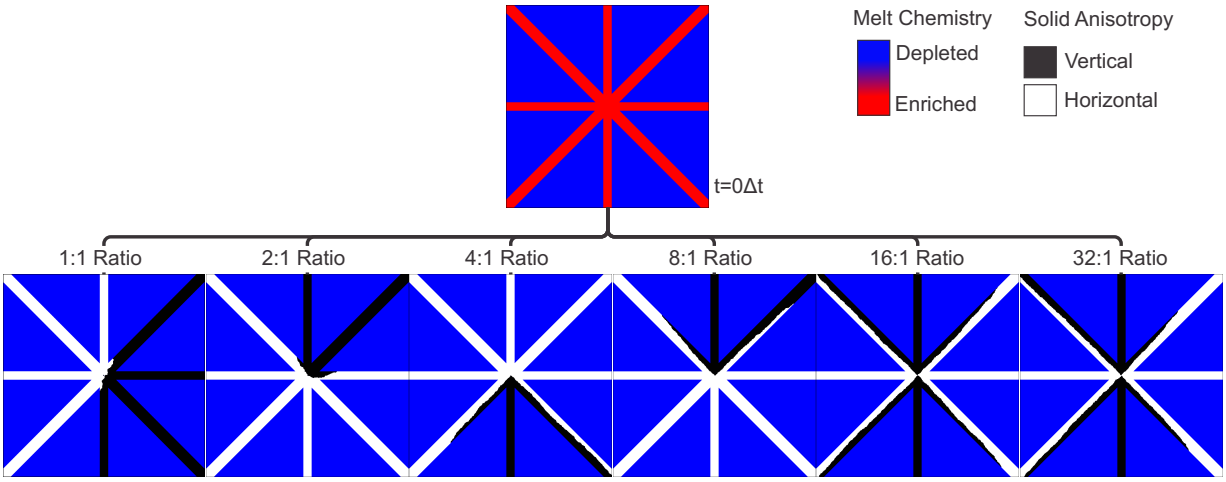


Figure 5.11 – Increasing aspect ratio and the effect on structural composition in radial channels. $R = 1$ in red regions, and $R = 0$ in blue regions in the liquid state. Threshold parameter a_2 chosen to reduce epitaxy $a_2 \sim 0.95$, a_1 set as $min(p2_i^j)$. The end result after growth has reached the edges of the simulation area is shown. Exact time elapsed per anisotropic ratio not important. Other parameters not listed here do not affect the results. $N_{sim} = 501$.

It is interesting to note the requirement of some minimum level of anisotropy, all else being equal for the solid state structure, for it to become orientation dependent. The use of higher degrees of anisotropy past about $4:1$ does not alter the structural composition of the vertical and horizontal channels. However, the composition of the diagonal channels appear to be more

mixed as the aspect ratio becomes higher, as well as the mixed nature of diagonal channels becoming more common for the same initial conditions. However, it should be noted that the structure in the sense of many repetitive runs with the same channel layout is more consistent for the higher aspect ratios. This observation is consistent with expectations.

5.2.3 Threshold Parameters a_1, a_2

As discussed previously, a_1 is routinely taken as $min(p2_i^j)$ and thus no sweeps of this parameter will be made. This parameter determines the threshold such that one structural type is dominant at a growth site over the other. It should be mentioned however, that the actual value used may be slightly lower than $min(p2_i^j)$ to overcome numerical roundoff errors, should they occur. Therefore, in practice, $a_1 = (1-\epsilon)min(p2_i^j)$ where ϵ is taken to be a small positive number. ϵ can be set at 0.01 since a_1 is defined as a single precision variable which has maximally 8 total digits of accuracy, of which up to 7 are decimals. This choice of ϵ gives us a large safety margin while not altering the outcome of the simulation in any way.

The threshold parameter a_2 gauges the difference in the rate of growth possible by the saturation ($P1$) of the liquid state to the rate of growth allowed by the interface attachment probabilities summed at the site of growth ($P2$) required to nucleate new crystals epitaxially. If the difference exceeds this threshold, epitaxial nucleation will occur probabilistically at the site of growth. Since the growth rate allowed by the liquid state depends (via the Gaussian function) on the local concentration of chemicals, the overall average growth rate of the simulation area depends on R, the ratio of chemical enrichment. This sets the upper bound of the threshold parameter a_2. The interface attachment probabilities are summed at the growth site, which varies up to a maximum of 1. Thus, the lower bound of the threshold parameter a_2 is 0. To observe only the effects of the threshold parameters, the parameters R and $p2_i^j$ will be fixed. For the first scenario, the entire grid is initialized with a homogeneous fluid with $R = 1$, and $32:1$ aspect ratio for the anisotropy tables $p2_i^j$. For changes in a_2 values in the neighbourhood of average values of $P1 - P2$ over the entire simulation grid, a rapid corresponding change of epitaxial frequency is expected as the average epitaxy behaviour transitions from epitaxy-favoured to epitaxy-forbidden. The exact threshold value is hard to establish numerically for general cases, but can be computed in the case of a simple, homogeneous fluid with a constant and fixed R value and $p2_i^j$. For $R = 1$, $P1 = 1$ and $p2_i^j$ corresponding to the $32:1$ aspect ratio anisotropy table, the value that corresponds to the cross-over in epitaxy behaviour can be given as $a_2^* = P1 - min(p2_i^j)$. Since $P1 = 1$, and the minimum value of $p2_i^j$ is $1/132$ which corresponds to the diagonal values of the table, the quantity $a_2^* = 1 - 1/132 = 131/132 \approx 0.9924$. a_i^* computed this way is accurate to within numerical limitations in the simulation for case scenarios simple enough such that this computation can be made. Therefore, in a sweep of

the parameter a_2, for values close to a_2^*, the qualitative behaviour of the texture generated is expected to change rapidly. Thus, the sweep of a_2 will be made non-linearly, with a smaller interval around a_2^* to properly capture this transition behaviour. The results of this sweep with the above mentioned scenario can be seen in figure 5.12.

In general, for the simulation runs with locally varying values of R (heterogeneous liquid states), this value a_2 is experimentally adjusted to produce the desired degree of epitaxy. For some scenarios, where $P1 \sim P2$, a very small value of a_2 maybe taken. In fact, a_2 may effectively be 0.0.

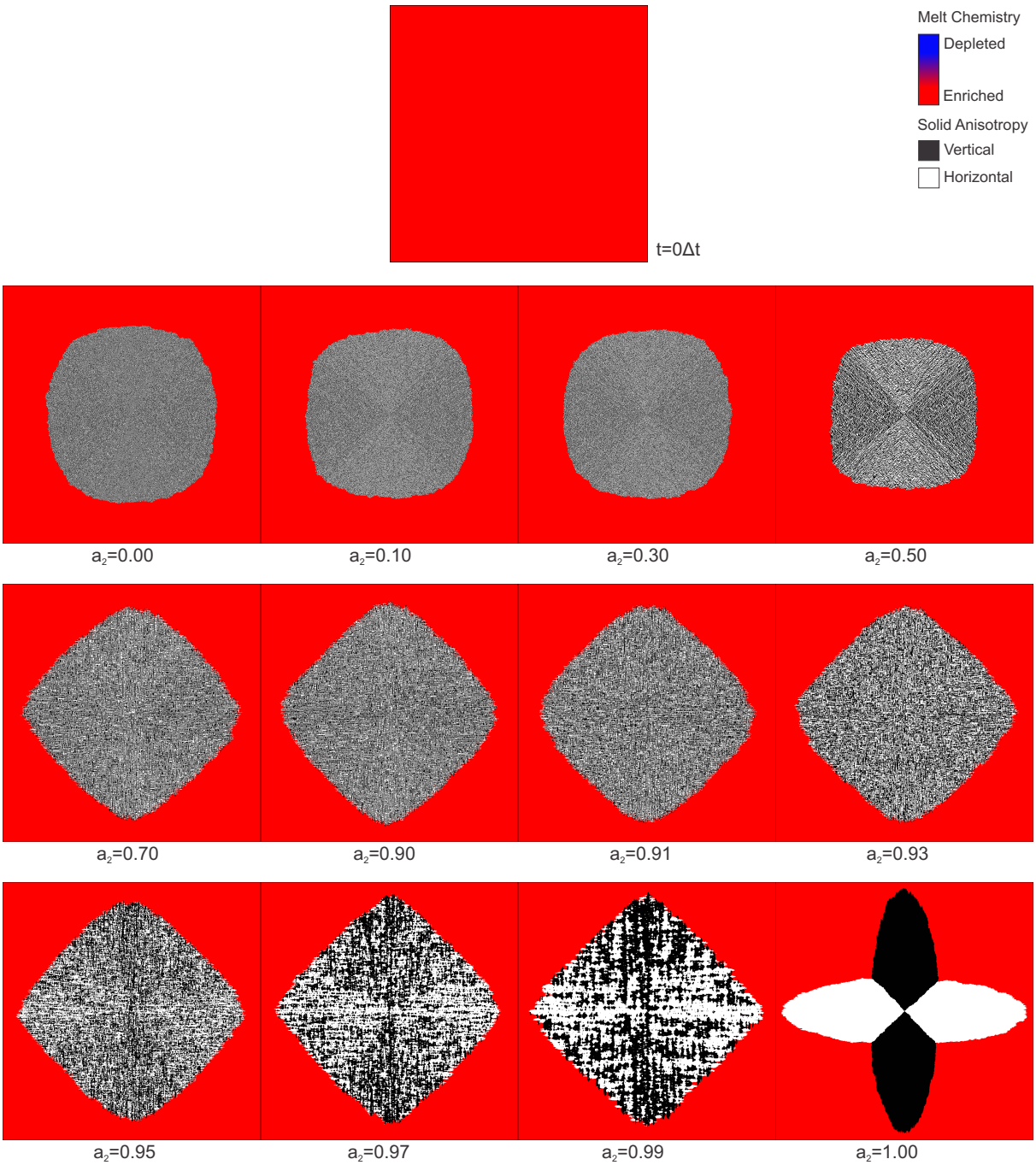


Figure 5.12 – Increasing a_2 and its effects on the resulting texture due to epitaxy frequency. $R = 1$ uniformly in the liquid state. $32:1$ aspect ratio anisotropy table. Other parameters not listed here do not affect the results. For a_2 smaller than, but arbitrarily close to a_2^*, the resulting pattern is similar to $a_2 = 0.99$. For a_2 greater than, but arbitrarily close to a_2^*, the pattern is similar to $a_2 = 1.00$. $N_{sim} = 501$.

5.2.4 Perturbation Parameters A, B, C

The perturbation parameter A allows a small local change of viscosity by modifying ω in response to the presence of the solid state. Similarly, B modifies the surface tension parameter α, and C modifies β, the extent of chemical separation. Due to the closed interval nature of ω and α, A, B must be chosen such that the net sum of ω and α after the perturbative modification are such that they are in the stable range as discussed earlier. The algorithm used to determine the strength of the local feedback depends on the ratio of solid and liquid neighbours in the neighbourhood as defined by the neighbourhood size n. Therefore, for a fixed size n, we can construct a scenario for which a significant amount of solid cells are present surrounding the liquid cells within a square (or max norm radial) of n lattice cells and allow the liquid state to run with immiscible collision operator as the only relaxation operator. A good choice of liquid-solid layout would be a modified checkerboard pattern, with the liquid separated by solid cells vertical and horizontal strips such that the number of solid neighbours per liquid cell is large with the lateral size of a domain of liquid cells around n.

As such, for a simulation area of $N_{sim} = 501$, and $n = 11$, an ideal size of liquid blocks such that all of the liquid cells have at least one solid neighbour would be 10 lattice cells. As we increase A, B monotonically, we expect the heterogeneities (as measured by their standard deviation method) at a fixed time inside the liquid domains to increase monotonically as well. Since the overall effect of A, B are perturbative, it is expected that the overall effect of increase in heterogeneities would be small. The result of such a sweep can be seen in figure 5.13.

As seen in figure 5.13, the resulting differences between the maximal and minimal case of A, B is difficult to discern visually. The σ_{std} values are best used in this case to characterize the differences, which in this particular scenario results in differences at the third decimal place. Based on the results, if we extrapolate from this maximal case toward the minimal case, it is expected that the perturbation effects are very small, certainly much smaller than the effects of any other parameter change of the same order of magnitude. Given that the solid state growth is stochastic, it may be beneficial to choose a set of non-zero A, B values to encourage the establishment of heterogeneities outside the solid state.

Parameter C is entirely different in its final implementation when compared to parameters A, B. As defined per equation 3.71, it is observed that, although the value of C can be chosen as perturbative in nature similar to parameters A, B, the microscopic considerations presented

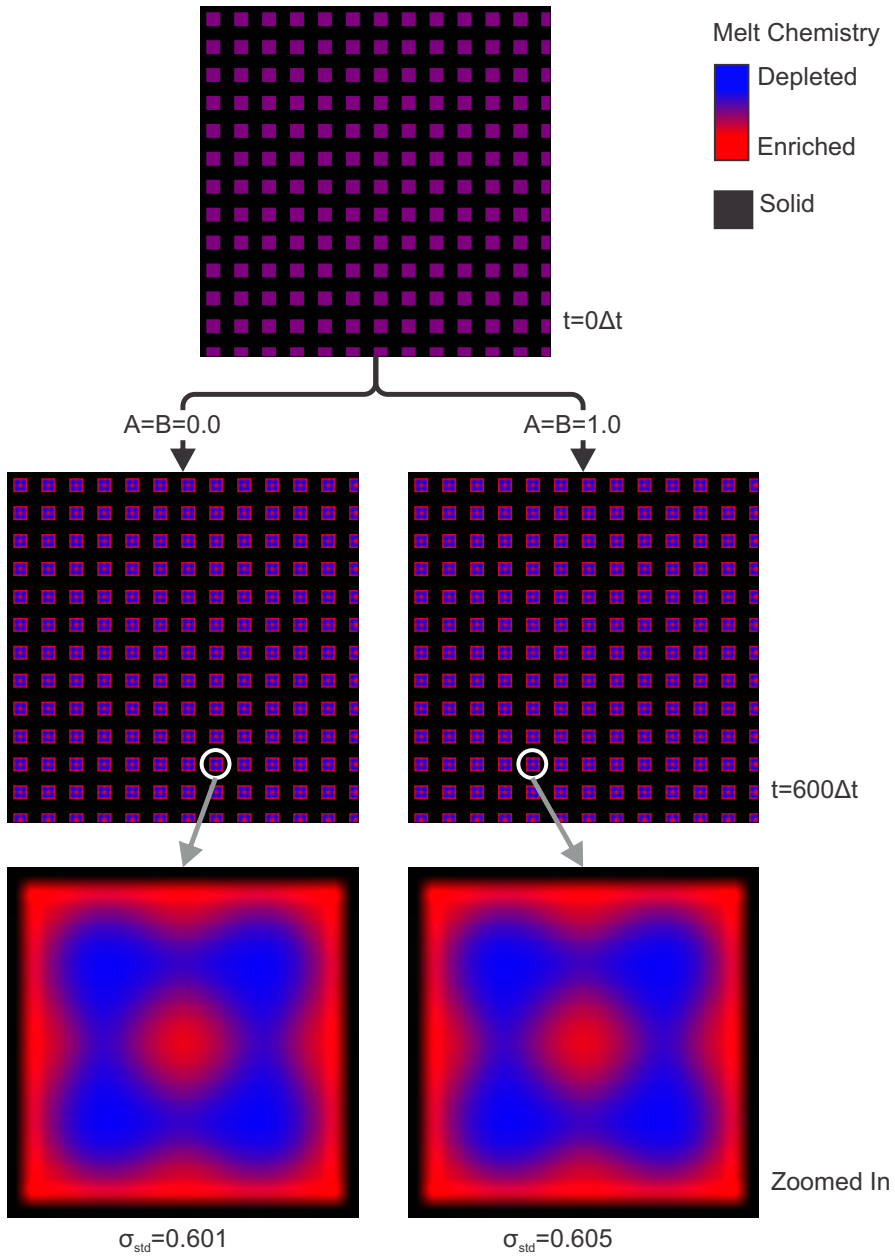


Figure 5.13 – Increasing perturbation parameters A, B and its effect on the liquid state. $N_{sim} = 501, n = 11, \omega_1 = \omega_2 = 1.0, \alpha_0 = 0.001, \beta_0 = 0.5, C = 1.0$. The solid state is forbidden to grow in this scenario.

earlier suggest that the solid state form of a substance with chemical stoichiometry of ρ_p is structurally incompatible with ρ_{np} which implies that β must increase to its maximal value as we approach the surface of a growing crystal from the melt when the growth is not diffusively limited. Depending on the scale of the model, β would likely increase as we approach the site of growth in the simulation, and since our system is not strongly undercooled, the nominal value of

β far away from the site of phase transition should approach its equilibrium value β_0. Therefore, $C = 0.0$ represents a scenario for which β does not change from β_0, or the observational scale is very macroscopic relative to the size of the crystal. As C is gradually increased, the value of β will begin to vary in an increasing fashion as we approach a more solid neighbourhood. Maximally, for a site surrounded by only solid neighbours and $C = 1.0$, $\beta = 1.0$ regardless of β_0. Complete separation of chemicals is expected at this site with a minimal mixing layer. The enriched chemicals are expected to "wet" the solid at the solid-liquid interface.

Based on this reasoning, a simulation scenario similar to that of the sweep of A, B can be set up but with a much wider grid spacing since the change in β_0 as seen in the sweep of β_0 in figure 5.5 shows that the liquid state is very sensitive to small changes in this parameter at low values of β_0. In fact, a scenario where the liquid state is in contact with a solid state and some of the liquid state are within the neighbourhood n of the solid state should show a change in the tendency of separation of chemicals in the liquid state as a function of the distance away from the solid-liquid interface. Such a scenario is constructed with 1/4 of the cells at the bottom of the simulation area as solid, and the remaining cells initialized as a $50\% - 50\%$ mixture of ρ_p and ρ_{np}. The neighbourhood size n is set at 21 to exaggerate the transition region. N_{sim} is the same as before at 501. A, B are both set to 0.0. The results of this scenario sweeping C from 0.0 to 1.0 is shown in 5.14.

As expected, the liquid state forms a "wetting layer" at the solid-liquid interface for $C = 1.0$. The separation of ρ_p (in red) and ρ_{np} (in blue) becomes more pronounced near the interface for $C \sim 1$.

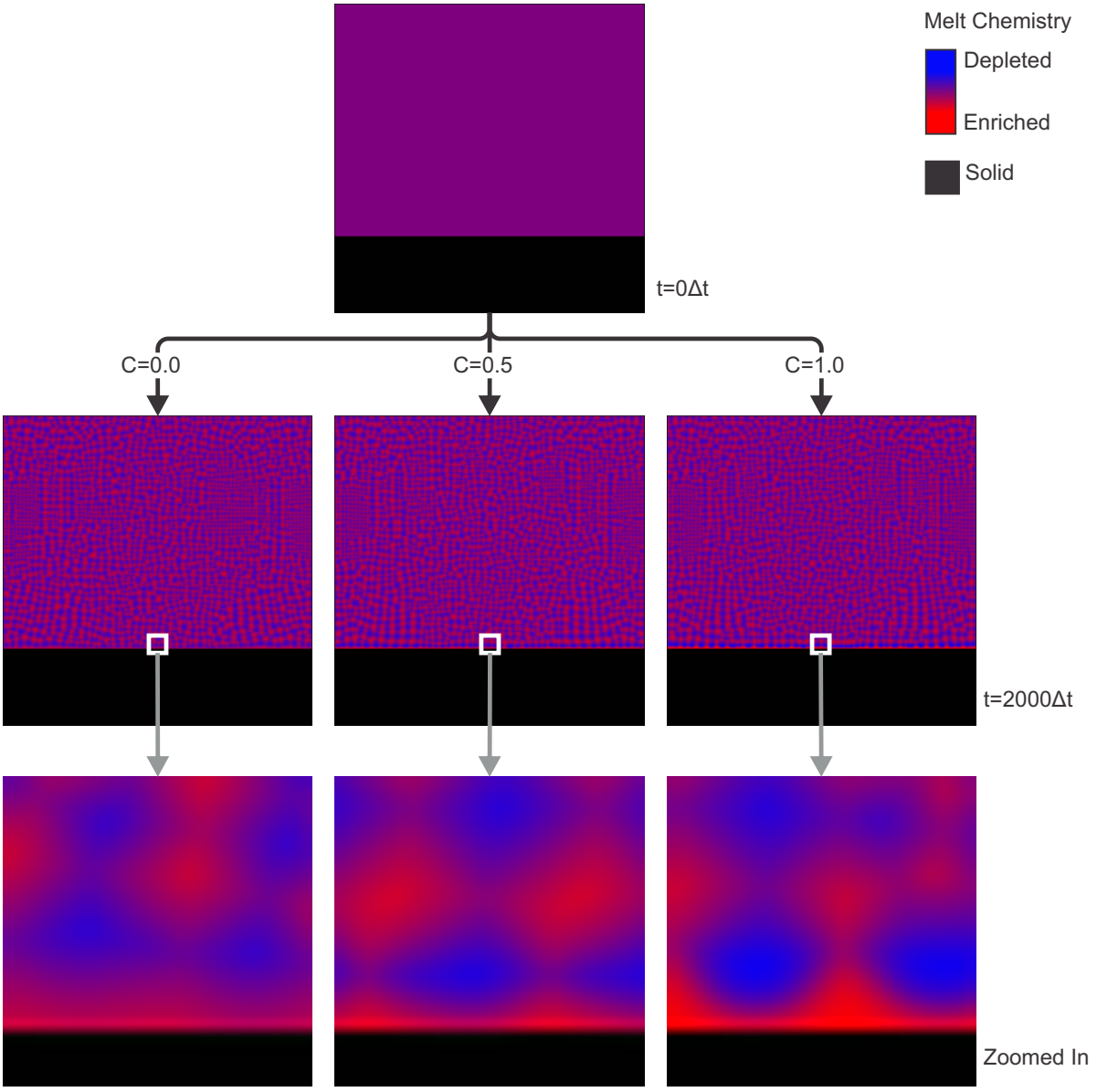


Figure 5.14 – Increasing parameter C and its effect on the liquid state. $N_{sim} = 501, n = 11, \omega_1 = \omega_2 = 1.0, \alpha_0 = 0.001, \beta_0 = 0.15, A = B = 0.0$. The solid state is forbidden to grow in this scenario.

5.2.5 Neighbourhood Size Parameter n

The neighbourhood size parameter n controls the effective size over which the parameters A, B, C apply by setting a sub-grid area of maximally $n \times n$ over which the cells (sites) are considered neighbours to the current cell (site). Using the same scenario as before for the sweep of the parameter C, we can observe the effects of changing n on the thickness of the layer of well separated ρ_p and ρ_{np} at the solid-liquid interface. Such a scenario is represented by the results shown in figure 5.15. At $n = 5$, the discernible interface is dominated by ρ_p, similar to a wetting layer. This thickness is roughly 2 lattice units wide. At $n = 11$, the thickness is roughly 6 lattice units wide. Due to the formation of the wetting layer which has the same chemical composition as the solid phase, a layer of rejected ρ_{np} is seen in blue but with β values approaching β_0. At the highest value of n, $n = 21$, the interface layer becomes very thick at ~ 18 lattice units wide and the composition of this layer includes both ρ_p and ρ_{np}. For the $n = 21$ case, the domains/channels in the bulk liquid enriched in ρ_p can clearly be seen to continuously merge with the thin, wetting layer of ρ_p immediately touching the solid phase. The ρ_{np} is passively rejected away from the interface in this same process.

While it is not certain which of these cases is more realistic, it is clear that for a large simulation area N_{sim}, we can reasonably set a large n value while still satisfying $n \ll N_{sim}$. Since the value of n is fixed for a fixed N_{sim} during realistic scenarios, it is expected that for smaller sizes of N_{sim}, the interface layer will be small due to the restriction that $n \ll N_{sim}$. For larger simulation grids with $N_{sim} > 1000$, n can be set at > 10, yielding solid-liquid interfaces similar to the cases of $n = 11, 21$ shown in figure 5.15. Therefore, the case $n = 11, 21$ are probably more realistic for simulations with scaling of a lattice unit such that $\Delta x' \sim 1$ mm or finer while choosing smaller values of n significantly decreases simulation time.

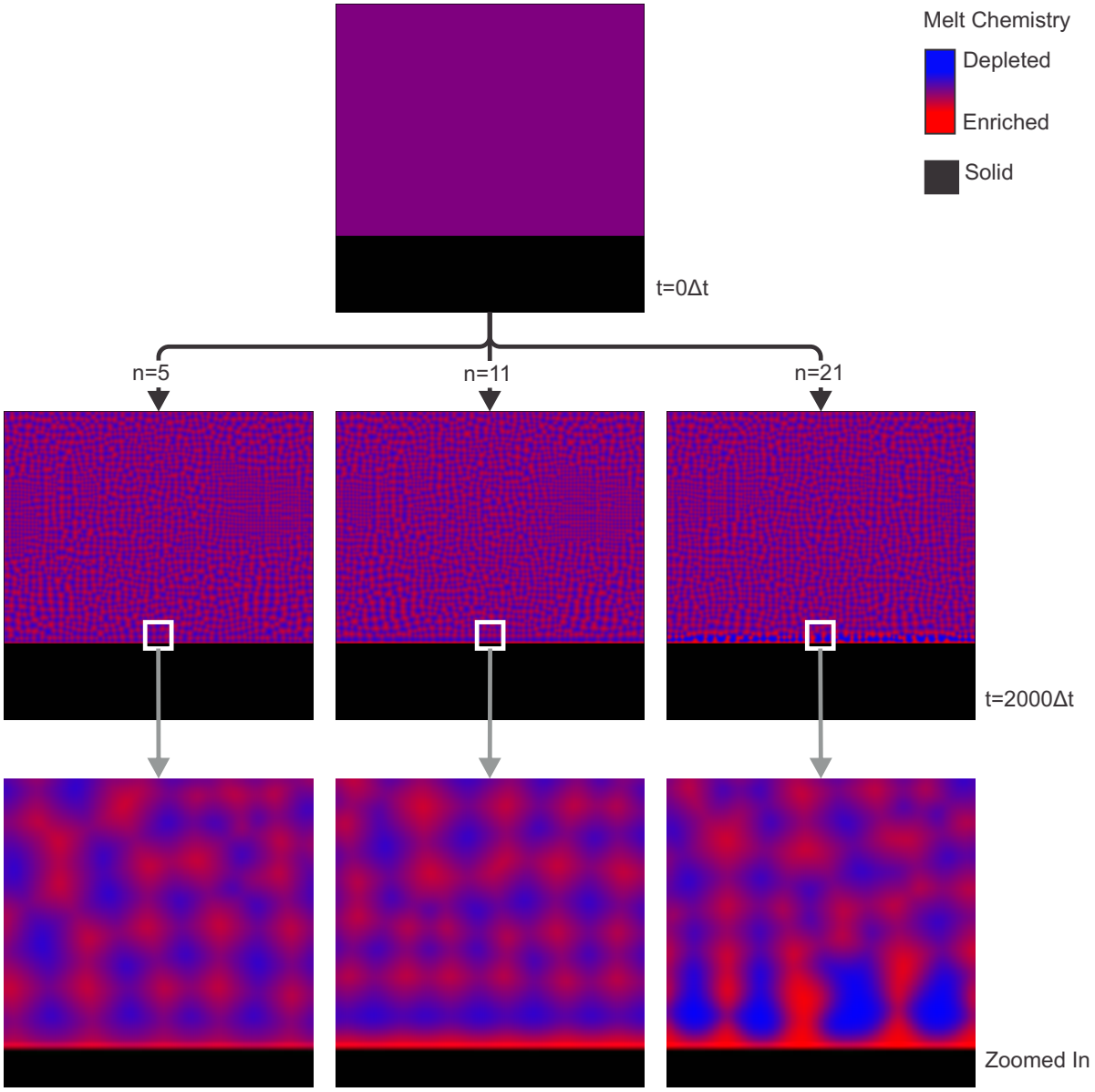


Figure 5.15 – Increasing neighbourhood size parameter n and its effect on the liquid state. $N_{sim} = 501, n = 11, \omega_1 = \omega_2 = 1.0, \alpha_0 = 0.001, \beta_0 = 0.15, A = B = 0.0, C = 1.0$. The solid state is forbidden to grow in this scenario.

5.3 Liquid-Solid Coupling Parameters

5.3.1 Solid State to Liquid State Parameter q

The coupling parameter q allows the liquid state to run once for each q number of time steps spent in the solid state since the time step size $\Delta t_s, \Delta t_l$ for the solid and liquid state are not necessarily equal in size. This value can be adjusted to properly match the time scales associated with the solid state and the liquid state. If the value of q is unity, the time scales in the solid state exactly matches that of the liquid state. For values of q larger than unity, the time scales in the solid state is much shorter than the liquid state. The restriction of a maximum of 1 layer of solid can be grown per solid time step sets an absolute rate of growth in the solid state to be $\Delta r/\Delta t_s$ where r is the radial size of a growing solid. The average characteristic wavelength of domain separation $\langle\lambda\rangle$ in the liquid state increases linearly as a function of time initially such that $\Delta\langle\lambda\rangle/\Delta t_1 \propto \alpha$ where α is (still) the surface tension parameter. For large times, $\Delta\langle\lambda\rangle/\Delta t_l$ tapers asymptotically to 0 as the domains reach their maximal size within the simulation area. The relationship $\Delta\langle\lambda\rangle/\Delta t_1 \propto \alpha$ appears to be true so long as the value of $\langle\lambda\rangle$ stays small compared to N_{sim}. q in this context is used to scale Δt_l by $\Delta t_l = q \cdot \Delta t_s$. If we let the diameter of a growing crystal cluster to be $2\Delta r/\Delta t_s$ and fix the initial time for which the melt is allowed to develop hetereogeneities, then we can compare this rate to $\Delta\langle\lambda\rangle/\Delta t_1$ and arrive at a few cases:

$$\text{when } \begin{cases} 2\frac{\Delta r}{\Delta t_s} \ll \frac{\Delta\langle\lambda\rangle}{\Delta t_l}, \text{ case 1} \\ 2\frac{\Delta r}{\Delta t_s} \sim \frac{\Delta\langle\lambda\rangle}{\Delta t_l}, \text{ case 2} \\ 2\frac{\Delta r}{\Delta t_s} \gg \frac{\Delta\langle\lambda\rangle}{\Delta t_l}. \text{ case 3} \end{cases} \tag{5.3}$$

In case 1, the solid state grows much slower than the rate of increase of separation of chemicals in the liquid state. In this case, we expect the overall texture to be controlled by the liquid behaviours only. In case 2, the solid state grows with a rate roughly equivalent to that of the separation rate in the liquid state. In this case, we expect the overall texture to be controlled by the liquid state, but with a drastic change in the size of solid crystals as crystallization time becomes large. In case 3, we expect the liquid state not to further develop significant hetereogeneity after the onset of crystallization. In this case, we expect the overall texture to be controlled largely by the morphology of the solid state. Case 3 is obviously more

realistic at Rum, since the stellate rays do not widen or change behaviour appreciably over its length. Case 1 and 2 can probably be produced in natural systems but are more likely to occur in laboratory settings. With case 3 chosen, the value of q can then be approximately determined. To find out what $\Delta r/\Delta t_s$ and $\Delta \langle\lambda\rangle / \Delta t_l$ are, we consider the averages of both quantities for a few runs. A scenario can be setup in the liquid state without solid to observe the coarsening rate as measured by $\Delta \langle\lambda\rangle / \Delta t_l$. This is accomplished by 2D Fast Fourier Transform (FFT) on the resulting image output at sequential time steps. The dominant mode of λ at each time step can then be plotted as a function of time, and a few different values of A can be chosen to determine the sensitivity of $\Delta\lambda/\Delta t_l$ with respect to A. An illustration on how this method is used to obtain $\langle\lambda\rangle$ is presented in figure 5.16. The results of this are presented in figure 5.17.

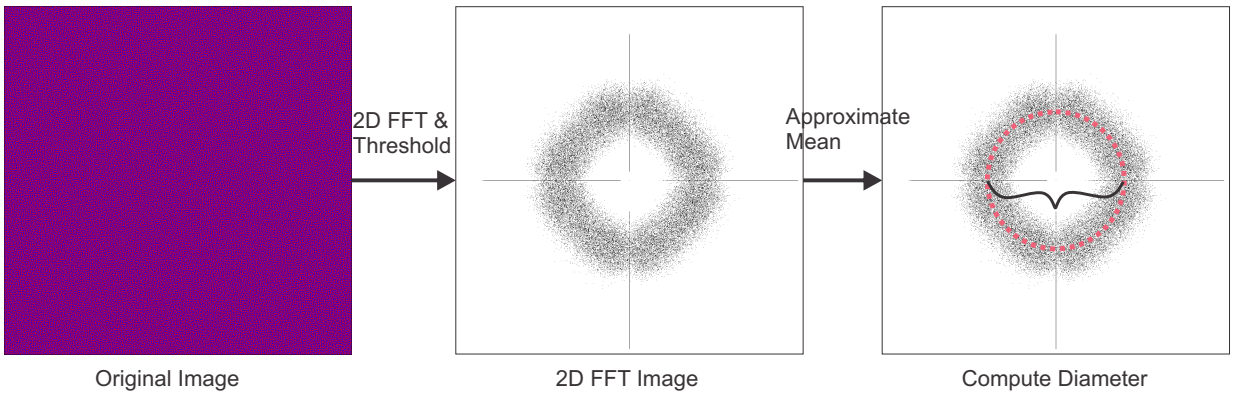


Figure 5.16 – Illustration of the steps utilized to obtain $\langle\lambda\rangle$ values. In the FFT image, the center of the image is the infinite (with respect to grid size) wavelength DC component and the edges of the image are shortest wavelength AC component (2 pixels).

From results presented in figure 5.17, it is seen that the peak value of $\Delta \langle\lambda\rangle / \Delta t_l$ is roughly about 0.033 and that the average rate of coarsening in the liquid state depends on the parameter α, although not linearly, at least for $\langle\lambda\rangle \ll N_{sim}$. To estimate the values of $\Delta r/\Delta t_s$, we need to consider the product $P1 \cdot P2$. Since $P1$ is controlled by the liquid state ρ_p via the Gaussian function $f(\rho_p, b, \sigma)$, we can choose a range of reasonable values of σ and fix $b = 1.0$ and $R = 0.5$ as before to estimate $P1$. As seen earlier, σ is suitably chosen with a small value compared to unity around $0.15 \rightarrow 0.20$. This allows us to estimate the value of $P1$ by computing the average value of $f(\rho_p, b\sigma)$ over this interval as:

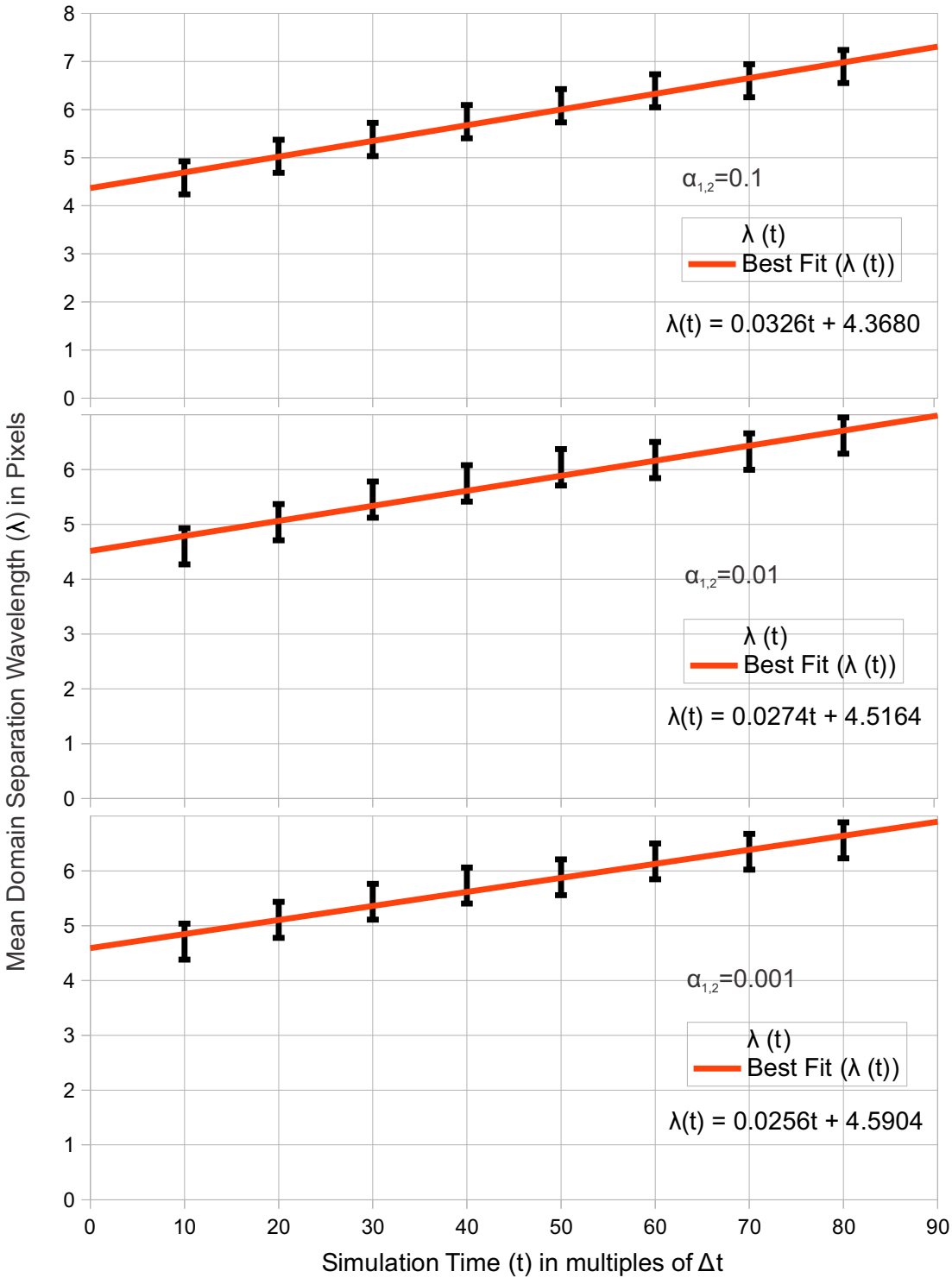


Figure 5.17 – FFT results of $\langle\lambda\rangle$ versus Δt_l for several values of α_i. The other parameters are fixed at $\omega_{1,2} = 1.0$, $\beta_0 = 1.0$, $R = 0.5$. Other parameters not listed here do not affect the results.

$$\begin{aligned}
\langle P1 \rangle &= \frac{\int_{\sigma=0.15}^{\sigma=0.20} f(\rho_p, b, \sigma) d\sigma}{\int_{\sigma=0.15}^{\sigma=0.20} d\sigma} \\
&= \frac{\int_{\sigma=0.15}^{\sigma=0.20} f(0.5, 1.0, \sigma) d\sigma}{\int_{\sigma=0.15}^{\sigma=0.20} d\sigma} \\
&\approx 0.110.
\end{aligned} \tag{5.4}$$

For an estimate of $P2$, we assume that a typical growth front has a geometry such that it is nearly flat at the site of growth where half of all available neighbours are liquid, thus $\langle P2 \rangle = 1/2 \sum_i p2_i$. We expect that for an average growth front, the contribution to $p2_i^j$ from $j = 2$ structural types becomes equal, and thus:

$$\begin{aligned}
\langle P2 \rangle &= \frac{1}{2} \sum_{i=1}^{i=8} \sum_{j=1}^{j=2} \frac{1}{2} p2_i^j \\
&= \frac{1}{2} \sum_{i=1}^{i=8} \left(\frac{1}{2} p2_i^1 + \frac{1}{2} p2_i^2 \right) \\
&= \frac{1}{2} \left(\frac{1}{2} + \frac{1}{2} \right) \\
&= \frac{1}{2}.
\end{aligned} \tag{5.5}$$

Thus, an average value of $\Delta r / \Delta t_s = \langle P1 \rangle \cdot \langle P2 \rangle = 0.110 \cdot 1/2 \approx 0.055$.

The requirement as per case 3 requires $\Delta r / \Delta t_s \gg \Delta \langle \lambda \rangle / (q \cdot \Delta t_s)$ which based on the above findings puts $q > 10$. Since most simulation runs at typical simulation sizes of about 1000 requires $\geq 10000 \Delta t$, it is important to pick q such that $q \ll 10000$. Taking $q \sim 10$ satisfies all criterion thus far.

Having established a rough range of q that is suitable, we shall conduct a parameter sweep to observe how the simulation behaves with respect to q. For this parameter sweep, both the solid and liquid states are allowed to fully interact. A, B, C are chosen at $0.5, 0.5, 1.0$ to allow feedback between the solid and the liquid state. The liquid state is run to develop heterogeneities and saved as initial conditions. This ensures the initial states are always the same between successive runs. The solid state is then run with a central nucleation event. The total time steps spent

for each q is fixed at $12000\Delta t$. The standard deviation method is used to keep the liquid state as constant as possible while still permitting local flow and feedback from the solid state. It is expected that at low values of q, the feedback from the solid state to the liquid state is much stronger, and thus the solid state should be able to grow faster as compared to higher values of q. At very high values of q, it is expected that the solid state has little impact on the liquid state, and the liquid state dictates the overall layout of the solid state. The results of this parameter sweep is shown in figure 5.18. The results agree with expectations. For high values of q it is seen that the solid readily follows the contours of the liquid state, and exhibits a lower overall growth rate due to limitations of the liquid state, while for lower values of q, the solid state exhibits a cross-hatched pattern characteristic of a combination of vertical and horizontal anisotropies. For values of $q < 10$, it is seen that the standard deviation method is not able to keep the liquid state at a characteristic heterogeneity. The prediction that the liquid state is able to react much faster than the growth of the solid state for $\Delta t_s \sim \Delta t_l$ from above is justified for $q < 10$. For $q = 1$, it can clearly be seen that a large number of isolated growth fronts have immediate access to large margins of relatively enriched melt ρ_p. For $q = 5$, this is still true. Starting from $q = 10$ upwards, there are a large number of growth fronts whose solid state lateral size is equal to the size of the liquid channel of relatively enriched melt surrounding it. This is equivalent to the formation of a thick wetting layer for lower values of q when $q < 10$ and a thin to non-existant wetting layer for $q > 20$ by controlling the feedback coupling between the solid state and liquid state.

If there are external flows present, then the flow velocities will need to be taken into account as well when scaling q, in addition to the other characteristic velocities discussed above. However, the presence of global flows are not being considered here.

Based on the above results, it seems appropriate to pick q in the interval $10 \leq q \leq 15$ when $R = 0.5$. In realistic scenarios, the value of R will probably be such that $R < 0.5$. In this case, picking q in the lower range of the interval seems appropriate $q \sim 10$. It is observed that for small changes in q, and certainly within the range $10 \leq q \leq 15$, no qualitative changes are expected in the solution. The most obvious quantitative change in the solution comes from the average size of the crystallized area.

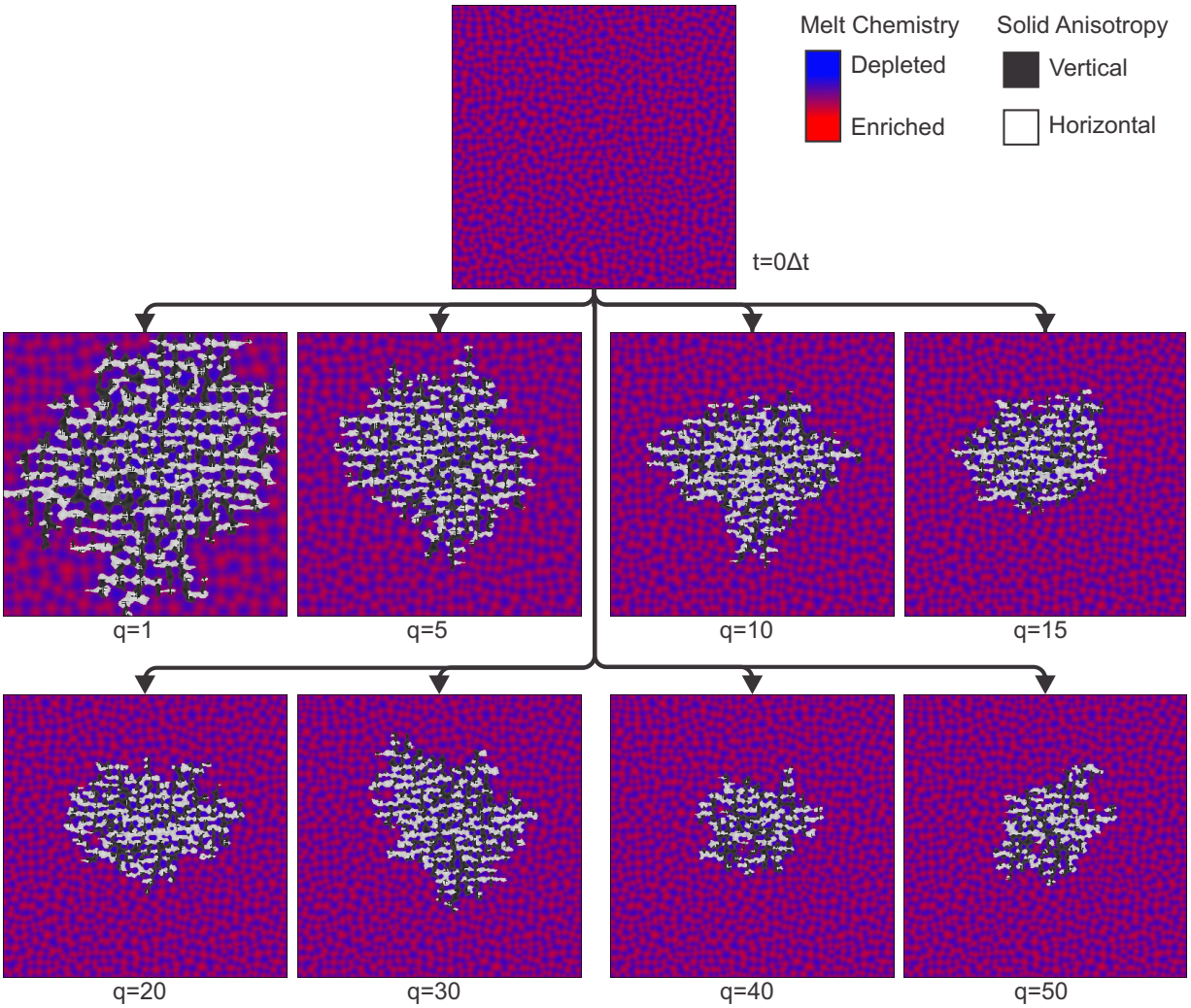


Figure 5.18 – Results of solid state texture as a function of parameter q. The other parameters are fixed at $\omega_{1,2} = 1.0$, $\alpha_{1,2} = 0.001$, $\beta_0 = 1.0$, $R = 0.5$, $a_2 = 0.1$. Other parameters not listed here do not affect the results.

Chapter 6

Stellate Results

In this chapter, we will explore a few special remaining parameters and some interactions of certain parameters. After these parameters are explored, all the parameters for which physical equivalents exist are transcribed. For the remaining parameters some estimation/deduction is used instead. The maximal domain of the complete set of parameters are then simulated, analyzed and discussed.

6.1 Towards Stellate Growth Scenarios

In constructing a realistic growth scenario, we need to consider all available parameters and their appropriate ranges. By the independent parameter sweeps above, some of the values have obvious choices even if the underlying physical value is not known. Parameters such as viscosity, surface tension and density can be easily set to desired ranges provided one has chosen the appropriate $\Delta x', \Delta t'$ and determined the characteristic velocity scale U. Other parameters such as $p2, R, b, \sigma$ plays a big role in the kinetics of the simulation but are not directly known or possible to establish currently. Other parameters are heuristic such as A, B, C, a_2, q. First and foremost, we shall explore a parameter pertinent to the simulation of an actual growth scenario, that was not considered earlier. This parameter is the amount of elapsed time $\Sigma\Delta t_{liquid}$ in the liquid state prior to the nucleation and crystallization (coupled liquid and solid state). Another parameter not explored earlier is the central enriched pocket size described by its radius: Δr_{enr} containing the nuclei. The parameter Δr_{enr} should be chosen such that $\Delta r_{enr} \ll N_{sim}$, for obvious reasons. The first parameter $\Sigma\Delta t_{liquid}$ is an important parameter. It directly controls

the extent of heterogeneity, when all other parameters which can control heterogeneity such as viscosity, surface tension are fixed. The second parameter Δr_{enr} is much less significant than $\Sigma\Delta t_{liquid}$, since it only needs to be large enough such that $L\Delta r_{enr} \sim$ the radius observed/desired for stellate central enrichment. The existence of this parameter lends a central tendency and symmetry to the results. These parameters are added to a summary of parameters used thus far in a list of equations 6.1.

List of Common Parameters

$$
\begin{aligned}
N_{sim} &\rightarrow \text{as constrained by run time and computational resources, around 1001} \\
&\textbf{Liquid State} \\
\Sigma\Delta t_{liquid} &\rightarrow \text{a range of values will be utilized} \\
\Delta r_{enr} &\rightarrow \text{a range of values such that } \ll N_{sim} \\
\omega &\rightarrow \text{as long as stable and reasonable} \\
\alpha &\rightarrow \text{as long as stable and reasonable} \\
R &\rightarrow \text{reasonable, probably } < 0.5 \\
&\textbf{Solid State} \\
(b, \sigma) &\rightarrow \text{as long as reasonable, for example } (1.0, 0.15) \\
p2 &\rightarrow \text{high degrees of anisotropy is preferred, } 32:1 \\
(A, B) &\rightarrow \text{as long as stable, for example } (0.25, 0.25) \text{ if } \omega < 1.55 \\
C &\rightarrow \text{can reasonably maximize this value to unity} \\
n &\rightarrow \text{chosen such that } \ll N_{sim}, \text{ for example } n = 11 \text{ for } N_{sim} \sim 1001 \\
a_1 &\rightarrow \text{chosen such that } = min(p2_i^j) \\
a_2 &\rightarrow \text{chosen a reasonable value to allow desired strength of epitaxy} \\
&\textbf{Coupling} \\
q &\rightarrow \text{chosen inclusively between 10 to 15}
\end{aligned}
\tag{6.1}
$$

For definitions of parameters not discussed here, see list of parameters 4.18.

The choice of $\Sigma\Delta t_{liquid}$ directly changes the measurable quantity σ_{std} as defined in equation 4.20. This choice is not arbitrary. As seen earlier, the characteristic separation distance in the liquid state can be measured by $\langle\lambda\rangle$ via the Fourier method. As $\langle\lambda\rangle$ becomes large, it is possible that either $\langle\lambda\rangle \times L$ where L is the characteristic spatial scaling becomes too large, thus the simulation becomes physically unrealistic, or $\langle\lambda\rangle \sim N_{sim}$ so that the domains are too large for the simulation grid size. $\langle\lambda\rangle \times L$ is an interesting constraint, and certainly one that is harder to constrain using limits. Experimental studies, as pointed out earlier, show that sharp separations in the melt phase of roughly basaltic compositions without the presence of crystals is at best mm sized, if the melt is held absolutely stable with no external agitation. In the presence of external agitations that are not mixing in nature (standing waves comes to mind), the liquid-liquid separation tendencies are tremendously enhanced, from apparently metastable behaviour to unstable. Without knowing the exact external influences or self-excited external influences on the target melt at Rum, it is hard to establish what a reasonable estimate on the upper limit on $\langle\lambda\rangle$ might be in the melt, prior to the onset of plagioclase crystallization. One approximate method, is to invoke the criterion that after the initial nucleation induction period, when the liquid-liquid separation effects are thought to be the determining factor in $\langle\lambda\rangle$, $\langle\lambda\rangle$ does not change significantly. In this view, one can study the crystal size distributions of Rum, and crystal cluster (such as olivine clumps, or clinopyroxene clumps) distributions to get an idea of the average value of $\langle\lambda\rangle$. This is best done at a macroscopic scale. Based on the macroscopic observations earlier, it might be expected that a good average value of $\langle\lambda\rangle \times L$ is around $\sim$ cm, by recalling that the width of stellates are typically $\sim$ cm, and the size of clinopyroxene nodules are roughly also $\sim$ cm. Therefore, if we choose L to be $\sim$ 1 mm, then it implies an approximate value of $\langle\lambda\rangle$ at $10\Delta x$. In observations, it is seen that the characteristic sizes of clinopyroxene nodules are somewhat larger than the characteristic widths of the plagioclase rays, and the characteristic sizes of olivine clumps are yet larger than that of clinopyroxene nodules. However, since no observations point to the primary participation of either olivine or clinopyroxene in the structural formation of stellates, constraints should be made with strong consideration of plagioclase only. Due to the binary nature of this simulation, it is not possible to simultaneously capture the behaviour of olivine and clinopyroxene in addition to plagioclase, but rather only the behaviour of plagioclase, and a chemical component that is an average of all the other mineral species in the parent melt. With this in mind, picking a maximal value of

$\langle\lambda\rangle$ around mm-cm scale seems appropriate which then dictates the choice of $\Sigma\Delta t_{liquid}$.

For the first semi-realistic growth scenario, we sweep $\Sigma\Delta t_{liquid}$ to determine the coupled interactions between this parameter and the solid state. The following values are chosen as listed in the parameters list 6.2. Where possible, the values in each field are set in the middle of their respective ranges as listed earlier. The results of this particular sweep are presented in figure 6.1.

Common Parameters List for Sweep of $\Sigma\Delta t_{liquid}$

$$
\begin{aligned}
N_{sim} &= 1001 \\
&\textbf{Liquid State} \\
\Sigma\Delta t_{liquid} &= 0, 1000, 2000, 3000 \\
\Delta r_{enr} &= 0.03 N_{sim} \\
\omega &= 1.25 \\
\alpha &= 0.001 \\
R &= 0.5 \\
&\textbf{Solid State} \\
(b, \sigma) &= (1.0, 0.15) \\
p2 &\rightarrow 32:1 \text{ tables} \\
(A, B) &= (0.45, 0.45) \\
C &= 1 \\
n &= 11 \\
a_1 &= min(p2_i^j) \\
a_2 &= 0.1 \\
&\textbf{Coupling} \\
q &= 12
\end{aligned}
\tag{6.2}
$$

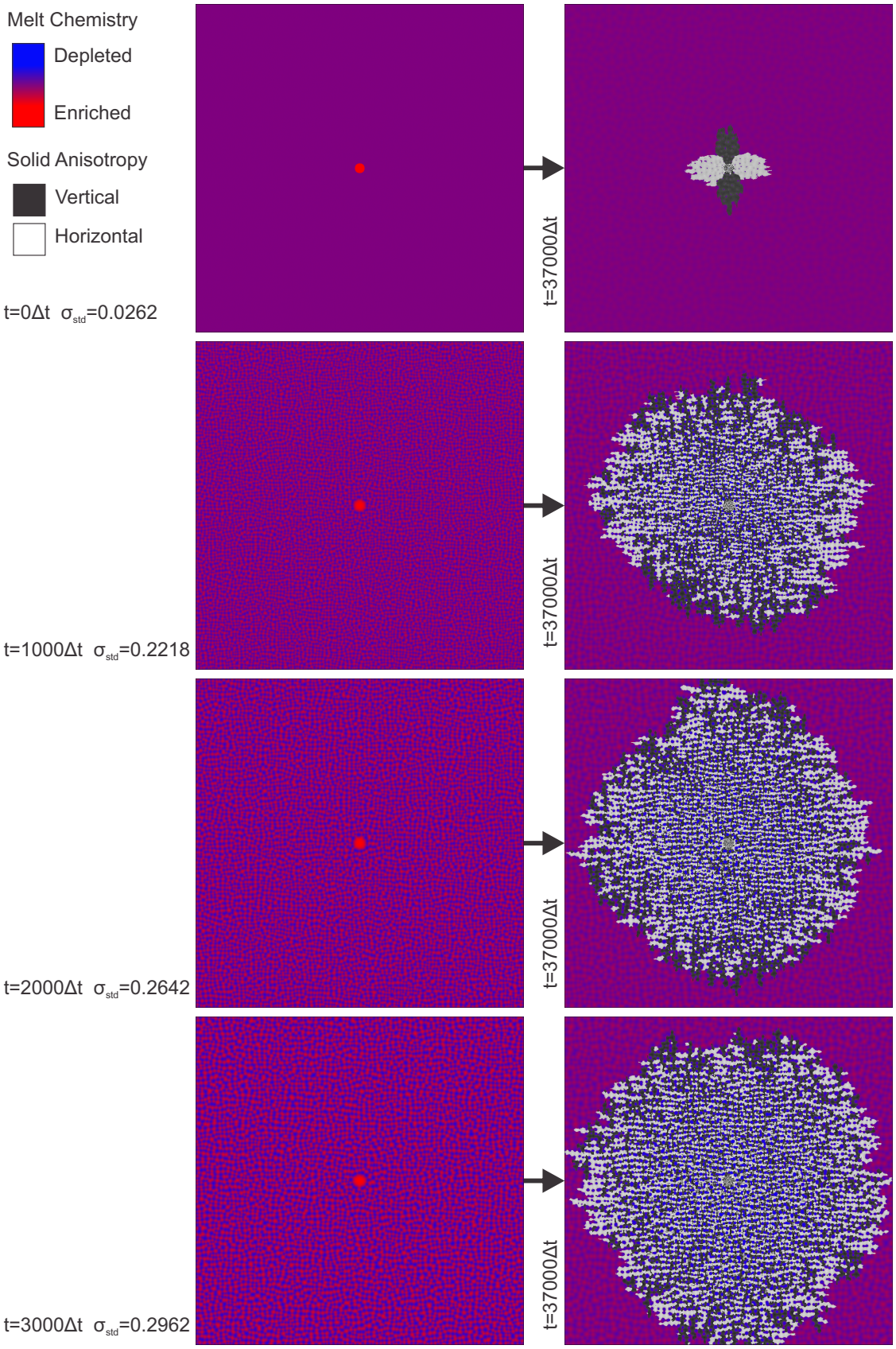


Figure 6.1 – Results of solid state texture as a function of parameter $\Sigma\Delta t_{liquid}$. An initially homogeneous state is used in the liquid state, with a small enrichment in the center as described by Δr_{enr}.

For higher values of $\Sigma\Delta t_{liquid}$, the heterogeneities are higher in the liquid state as exemplified by the increasing values of σ_{std}. For a fixed set of (b, σ) values, this translates to faster average growth as a function of σ_{std}. This results in the progressive larger texture patches for the same duration of simulation after the onset of crystallization when the initial values of σ_{std} are higher. For $\Sigma\Delta t_{liquid} = 0$, the liquid state only develops hetereogeneities after the onset of crystallization due to some rejection of solute at the surfaces of the growth and the partial-effectiveness nature of holding the liquid state at the same σ_{std} value (see sub-section 6.2 for results and discussions pretaining to σ_{std} and collision operator interleave artifacts). For higher heterogeneities in concentrations, there exist small spots where the local concentration of plagioclase-compatible chemicals are sufficiently rich to allow a high frequency of epitaxy similar to the initial enriched center. For $\Sigma\Delta t_{liquid} = 0$, the heterogeneities are insufficient to induce epitaxy at the growth front, and a coarse, nearly mono-mineralic appearance is observed. In addition for $\Sigma\Delta t_{liquid} = 0$, the growth front is jagged and exhibits a high surface area (perimeter length in 2D) to internal volume (surface area) ratio, portraying a classic diffusively limited growth scenario outside of the enriched center. At the rims of the solid state for $\Sigma\Delta t_{liquid} \neq 0$, the tendency for the crystal to grow epitaxially is lost due to sufficiently high concentrations of ρ_p. This is due to the very gradual coarsening of the liquid state, resulting from the approximate method used to interleave the two collision operators from the onset of crystallization. Since the surface tension parameter α as well as viscosity parameter ω both change the quantity $\Delta\sigma_{std}/\Delta t$, therefore, changing either of those parameters at fixed $\Sigma\Delta t_{liquid}$ values change σ_{std}. Thus, the results of changing α and ω will not be independently varied since the resulting changes in σ_{std} are similar to what is observed via directly changing $\Sigma\Delta t_{liquid}$.

The next parameters of interest are in the solid state. Since σ directly controls the sensitivity of bulk plagioclase growth to the local concentration, this parameter should reasonably determine the overall stellate growth rate and tendency to branch at liquid enrichment junctions. For low values of σ, the growth should be more preferentially bounded inside only the most enriched locations, while the opposite is true for higher values of σ. This solid selectivity of liquid enrichment as controlled by σ for very high values of σ should become low enough such that, for large time steps, nearly the entire grid is crystallized into solid plagioclase with a high volumetric fraction of inclusions. The parameters used are listed in table 6.3. The results of changing the parameter σ for a typical growth scenario is shown in figure 6.2.

Common Parameters List for the Sweep of σ

$$
\begin{aligned}
N_{sim} &= 1001 \\
&\textbf{Liquid State} \\
\Sigma\Delta t_{liquid} &= 2000 \\
\Delta r_{enr} &= 0.03 N_{sim} \\
\omega &= 1.25 \\
\alpha &= 0.001 \\
R &= 0.5 \\
&\textbf{Solid State} \\
(b, \sigma) &= (1.0, 0.10), (1.0, 0.15), (1.0, 0.20) \\
p2 &\rightarrow 32:1 \text{ tables} \\
(A, B) &= (0.45, 0.45) \\
C &= 1 \\
n &= 11 \\
a_1 &= min(p2_i^j) \\
a_2 &= 0.1 \\
&\textbf{Coupling} \\
q &= 12
\end{aligned}
\tag{6.3}
$$

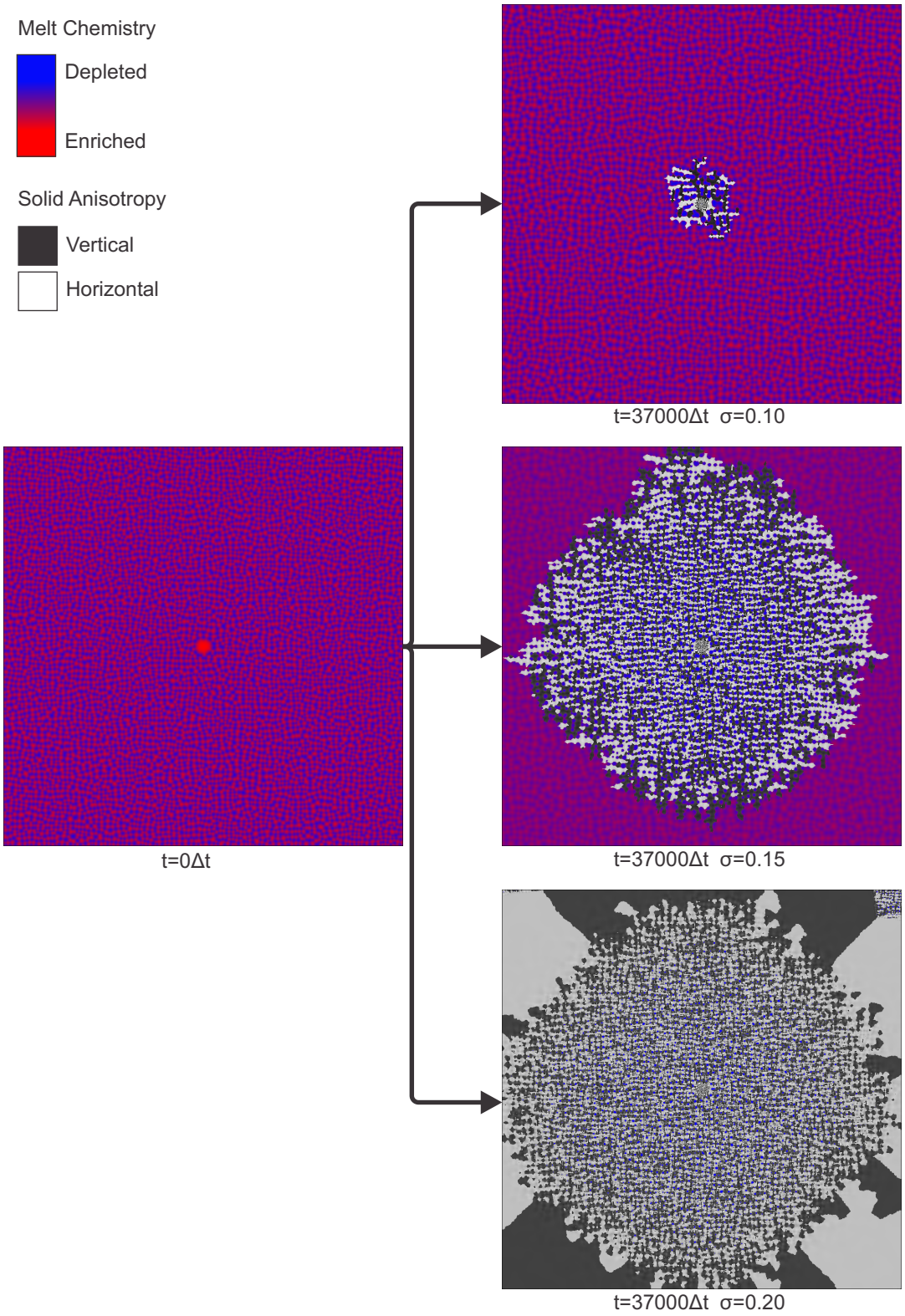


Figure 6.2 – Results of solid state texture as a function of parameter σ. An initially homogeneous state is used in the liquid state, with a small enrichment in the center as described by Δr_{enr}. This homogeneous state is allowed to evolve for $2000\Delta t$, and the resulting hetereogeneous liquid state is used for each value of σ. All the parameters used in this sweep are listed in list 6.3.

As can be seen in figure 6.1, the overall size of the plagioclase stellate in a fixed amount of time is a strong function of the parameter σ. Changing this parameter while fixing the parameter R allows us to set the enrichment selectivity of the solid state. For $\sigma = 0.20$, the probability of phase transition is sufficiently high such that the entire domain is fully solidified at large times. Part of this behaviour is due to the the attempt to keep the liquid state at constant heterogeneity after the start of crystallization, which is not entirely successful. In the next section we shall explore this aspect more thoroughly.

6.2 Realistic Stellate Growth Scenarios

Some of the physically known parameters can be transcribed into lattice parameters. If we set a value of $L = 1$ mm, then $\Delta x' = \Delta x L = 1L = L$. As was mentioned earlier $U \equiv c = \sqrt{3kT/m}$. While we could find an average mass for basaltic magmas, instead we use the equivalent form of $U = \sqrt{3\Re T/M}$ where M is the molar mass, a value we can easily calculate for a typical basalt composition and $\Re$ is the ideal gas constant. Suppose the average basalt composition we are interested in contains 50% SiO_2, 15% Al_2O_3, 10% FeO, 10% MgO, 10% CaO, 5% combined $(Na + K)_2O$, then

$$\begin{aligned} M &= 0.50(28.086 + 2 \cdot 15.999) + 0.15(2 \cdot 26.982 + 3 \cdot 15.999) + 0.10(55.845 + 15.999) \\ &\quad +0.10(24.305 + 15.999) + 0.10(40.078 + 15.999) + 0.05(0.5 \cdot 22.990 + 0.5 \cdot 39.098 + 15.999), \\ &\approx 64.511 \simeq 65 \text{ grams/mole.} \end{aligned} \tag{6.4}$$

It is worth noting here that small changes $\sim \pm(5-10)\%$ in composition of the heaviest oxides only result in small changes $\sim \pm 2$ grams/mole in M, thus this value is considered characteristic of a basalt melt. Since the value of U changes with both T and M, an effective range of U can be obtained by considering the largest variations in T and M. For basalt compositions, an upper bound in molar mass deviations about the mean value calculated in equation 6.4, ΔM, is around ± 2 grams/mole, whereas the changes in T can be large from $> 1200^\circ$ C to $\sim 600^\circ$ C. A quick calculation shows $\Delta M \partial U/\partial M \ll \Delta T \partial U/\partial T$. This results in overall changes in U dominated by temperature T and not composition M, for the range of both quantities considered. Using the value of M (in Si units) calculated earlier in equation 6.4, and a realistic range of $T = [900, 1200]^\circ$ C (where $[a, b]$ denotes the closed, real valued interval from a to b) deduced from known liquidus temperatures of basalt, at various compositions (including water content) and pressures, U can be calculated,

$$U = c = \sqrt{\frac{3\Re T}{M}} \approx \sqrt{\frac{3 \cdot 8.3145 \cdot ([900, 1200] + 273.15)}{65/1000}} \simeq [671, 752] \text{ m/s.} \tag{6.5}$$

Thus,

$$\Delta t' = \Delta t \cdot \frac{L}{U} = 1 \cdot \frac{0.001}{[671, 752]} \simeq [1.3, 1.5] \times 10^{-6} \text{ s}. \tag{6.6}$$

However, choosing such a small Δt results in small values of α as will be seen later.

The kinematic viscosity can be calculated from known dynamic viscosities (η) of basalt magmas. Taking a value of η prior to crystallization (since the presence of crystals drastically increase the viscosity) puts $\eta \approx [50, 500]$ Pa $\cdot$ s (taking common viscosity values of basalts). Thus if ρ_0 of basalt magma is taken to be $[2800, 3000]$ kg/m^3, $\nu = \eta/\rho_0 = [0.017, 0.18]$ m^2/s. Thus,

$$\langle \omega \rangle = \left[\frac{3\nu}{c^2 \Delta t'} + \frac{1}{2} \right]^{-1} = [0.77, 1.76] \tag{6.7}$$

Lastly, a known surface tension σ allows us to calculate the parameter α, if we first transform the implied surface in 3-dimensions to 2-dimensions. Assuming spherical geometry, in 3-dimensions, a sphere has area $4\pi r_c^2$ whereas in 2-dimensions, the "surface"-circumference has "area"-length $2\pi r_c$. Therefore, a measurement taken in 3D can be converted to 2D via

$$\sigma_{2D} = \sigma_{3D} \frac{4\pi r_c^2}{2\pi r_c}, \tag{6.8}$$

where r_c is a characteristic radius of interest. For our purposes, this value r_c can be approximated by $\langle \lambda \rangle$, the wavelength of the domain separation as calculated via the Fourier method, or by considering the average size of crystal grains in a thin section of Rum rocks. This method may seem less exact than desired. However, as seen earlier in the Fourier analysis of coarsening, the rate of coarsening is proportional to the surface tension, but not linearly. Indeed, the rate of coarsening changes slowly with the parameter α, and thus a small change in the surface tension due to the choice of r_c should not affect the overall dynamics. Thus via heuristics, r_c can be taken at 5 to 20 times that of L. Using this range, the only known values of surface tension (via measurement and theoretical calculation) in the previously indicated (see Chapter 3.2) range of $(5 - 10)$mN/m can be transcribed into 2D values as

$$\sigma_{2D} = [0.005, 0.010] \frac{4\pi [0.005, 0.020]^2}{2\pi [0.005, 0.020]} = [0.63, 5.03] \text{ mN}. \tag{6.9}$$

Similarly, the density can be transformed using $r_c = 10L$ as

$$\rho_{2D} = \rho_0 \times 0.01 = [2800, 3000] \times 0.01 \text{ kg/m}^2. \tag{6.10}$$

Then,

$$\begin{aligned} \alpha &= \left[\frac{10}{6\omega_{avg}\sigma_{2D}}\frac{\Delta x'^3}{\Delta t'^2}\rho_{2D}\right]^{-1} \\ &\simeq [3.0 \times 10^{-9}, 1.6 \times 10^{-8}], \end{aligned} \tag{6.11}$$

which are small values, but still usable if we use double precision variables. These numbers can be increased by scaling Δt. For example, a choice of $\Delta t = 10$ yields an upper bound of $\alpha = 1.6 \times 10^{-6}$, while $\Delta t = 100$ results in $\alpha = 1.8 \times 10^{-4}$. However, choosing a value of $\Delta t > 100$ results in ω values close to the limits of stability.

Parameter R is not known precisely over the distance of a meter (typical for stellates), but from bulk rock analyses, this parameter can be set at 0.17 and higher in the vicinity of a stellate (see modal analysis of Labonté in sub-section 2.3), at least in three dimensions. In two dimensions, to achieve the same number of connections, the value of R needs to be higher. Taking this value around 0.35 seems reasonable, as it is more than 0.17 and certainly within the limits of the composition range of gabbro in 2D (or thin-sections). a_2 will be picked to set twinning frequency so that it qualitatively corresponds to observations. $p2$ will be set at $32:1$ assuming ellipsoidal geometry. Δr_{enr} is set at about $3\%N_{sim}$ based on field observations. Parameters $b, \sigma, \Sigma\Delta t_{liquid}, A, B, C, n, q$ are not easily inferred but are chosen based on respective parameter sweeps. b will be set at 1 so that the domains of red correspond in chemistry exactly to the final plagioclase solid. $\Sigma\Delta t_{liquid}$ will be set at $2000\Delta t$. A, B for now will be set at $0, 0$ and C at 1. n will be set at 11. β_0 will be set at 0.15.

The whole set of parameter values are then established and tabulated in list 6.12.

List of Established Parameter Values

Range or Value Chosen

$$
\begin{aligned}
L &= 1 \text{ mm} \\
\rho_0 &= [2800, 3000] \text{ kg/m}^3 \\
M &= 64.511 \text{ grams/mole} \\
T &= [900, 1200]^\circ \text{ C} \\
\eta &= [50, 500] \text{ Pa} \cdot \text{s} \\
\sigma_{3D} &= [5, 10] \text{ mN/m} \\
r_c &= [5, 20] \times L \text{ mm}
\end{aligned}
$$

Resulting Parameter Range

$$
\begin{aligned}
\Delta x' &= 0.001 \text{ m} \\
\Delta t' &= [1.3, 1.5] \times 10^{-6} \text{ s} \\
\rho_{2D} &= [280, 300] \text{kg/m}^2 \\
\omega &= [0.77, 1.76] \\
\alpha &= [3.0 \times 10^{-9}, 1.6 \times 10^{-8}]
\end{aligned}
$$

Inferred and Unknown Parameters

$$
\begin{aligned}
\Sigma \Delta t_{liquid} &= 2000 \Delta t \\
\Delta r_{enr} &= 3\% N_{sim} \\
R &= 0.35 \\
a_2 &= 0.1 \\
\beta_0 &= 0.15 \\
p2 &\rightarrow 32:1 \\
(b, \sigma) &= (1.0, 0.15) \\
(A, B, C) &= (0, 0, 1.0) \\
q &= 12 \\
n &= 11
\end{aligned}
$$

(6.12)

Since there are two extremes for each parameter known, two simulations of realistic scenarios will be initially carried out. The extremes of the parameter ranges established above can be categorized by their relative effect on the melt. The lowest values of ω, α reduces the amount of melt hetereogeneity per unit time. Therefore, for the same amount of $\Sigma\Delta t_{liquid}$, the concentration of $\rho_1 = \rho_p$ is less where $\rho_1 > \rho_2$, and thus the overall growth is slower. When the same value of a_2 is fixed, this results in less twins via lessened tendency to twin. For the highest values of ω, α, the opposite holds true. The results are given in figures 6.3 and 6.4. As can be seen in figure 6.3, at early times ($\leq 80000\Delta t$), the growth is clearly faster in the case where ω, α are larger.

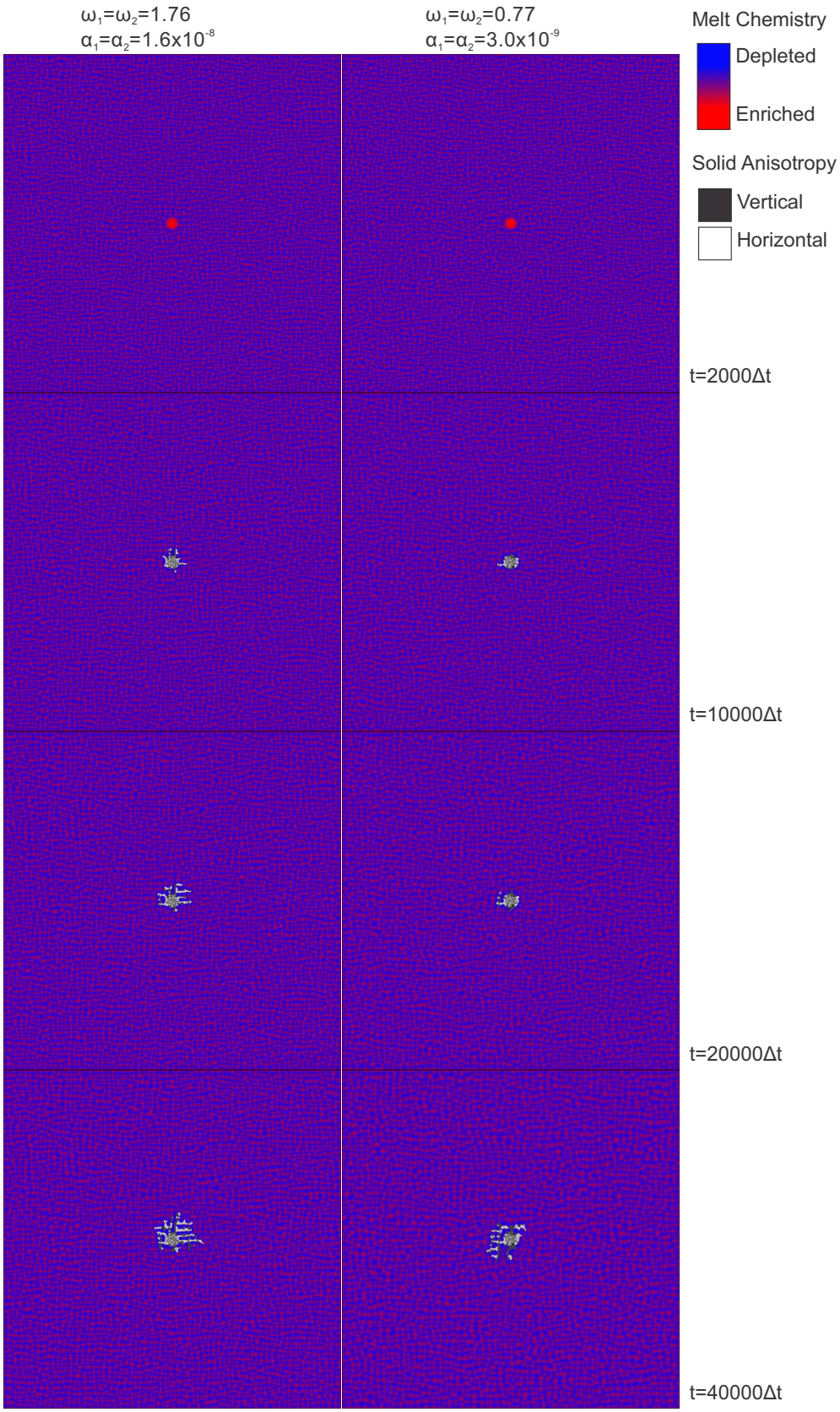


Figure 6.3 – Results of two extremes of physically realistic parameter sets in comparison. All the parameters used in this sweep are listed in list 6.12. Figure continued in figure 6.4. Initial conditions of the liquid at $t = 0$ not shown.

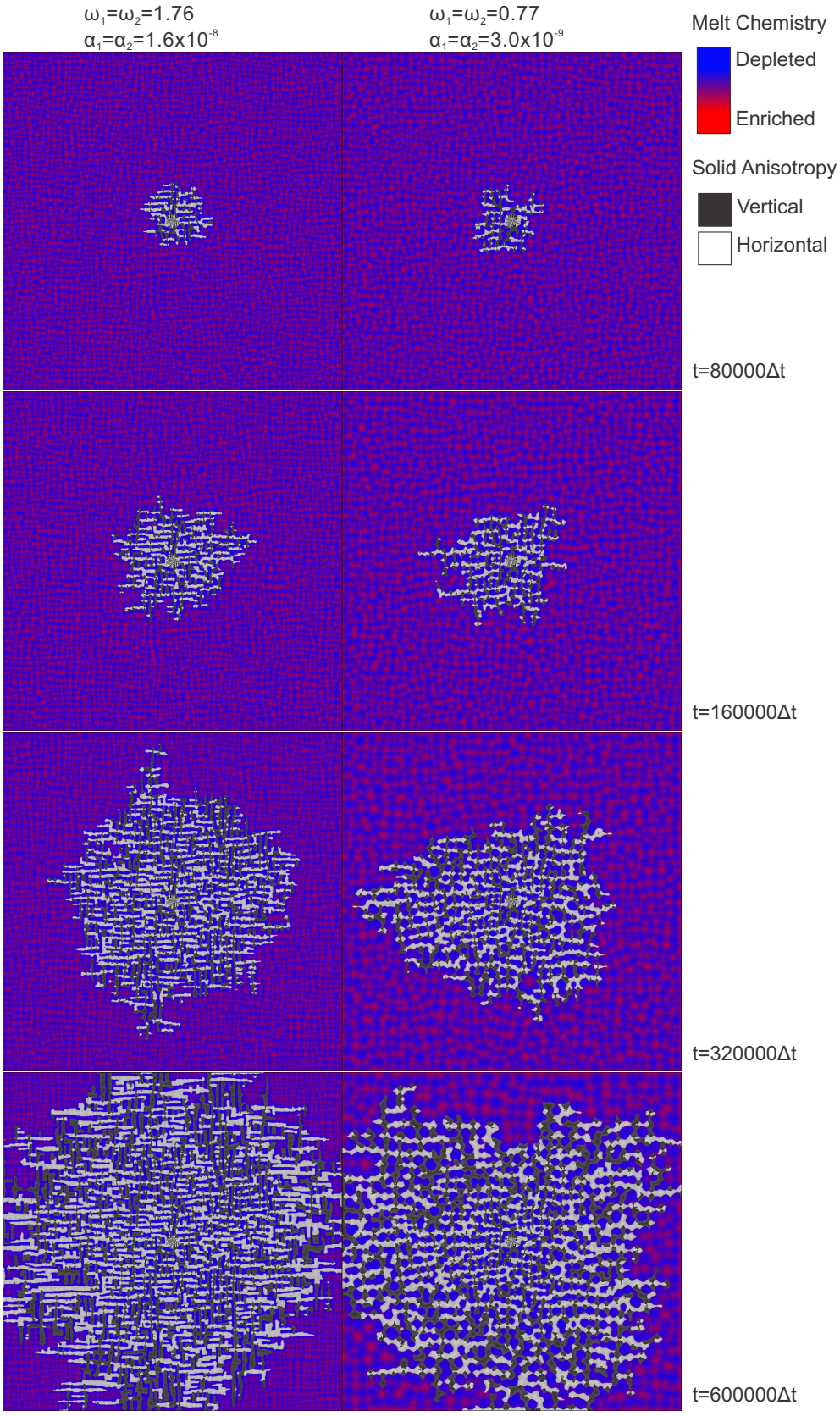


Figure 6.4 – Results of two extremes of physically realistic parameter sets in comparison. All the parameters used in this sweep are listed in list 6.12. Figure continued from figure 6.3. Initial conditions of the liquid at $t = 0$ not shown.

However, for larger times, it is not immediately obvious why the liquid state established larger average domain sizes for the case where ω, α are smaller. One hypothesis is that since the value of β_0 is low, the higher viscosity of the melt and low surface tension prevented very quick separation in the liquid state. Thus, the liquid state becomes more connected like that in a percolation structure seen earlier when $R = 0.5$. This connectivity allows hydrodynamic coarsening of the liquid state that is not possible when the value of ω, α is raised. However, a quick set of simulations with only the liquid state proves that the coarsening rate is indeed proportional to α and inversely proportional to ω for the two extremes of those values considered, so that the hypothesis is incorrect. The other hypothesis is that the attempt to keep the liquid state at a constant heterogeneity is not working properly for higher viscosity situations. This is a reasonable hypothesis since the coarsening behaviour of the liquid state is much less sensitive to ω than α; however, the mixing behaviour of the liquid state is much more sensitive to ω than α when the two collision operators are interleaved. During immiscible collisions, the phases are separated at a similar rate for both extremes of ω, α. Since the value of α is small in both cases, the coarsening rate of the two extremes are only slightly different (about $\sim 0.01\Delta\sigma_{std}/20000\Delta t$ between the two extreme values of α). The difference in ω results in large differences in viscosity (over 2 magnitudes in Pa $\cdot$ s), which translates into a higher frequency of Δt spent in the mixing collision operator $\mathbf{\Omega}_1$ compared to the opposite case of ω which results in a lower apparent liquid viscosity when the simulation time becomes much larger than the collision operator interleave time.

Another interesting observation possibly correlated to this is that the higher valued ω, α case possesses a stellate structure which is clearly more solid equilibrium shape-controlled. It is a peculiar behaviour for the liquid state at large Δt to contain domain orientations matching the simulation grid geometry. By keeping the liquid state constant, the dominant behaviour is not the coarsening of the liquid state but other, secondary or higher order behaviours. The lower valued case of ω, α does not appear to exhibit these simulation artifacts, since it does coarsen through time despite an attempt to keep a constant heterogeneity in the liquid state. To verify this hypothesis, a corresponding set of simulations was run with the same parameter values, but with the interleaving condition of the collision operators turned off. The results are shown in figure 6.5. With the interleaving condition turned off, the liquid state at large times coarsen and the peculiar behaviours observed earlier in the liquid state are diminished. The

total amount of time required for the stellate to achieve radius $\simeq N_{sim}$ is reduced when the interleave condition is disabled by a factor of $\simeq 3$.

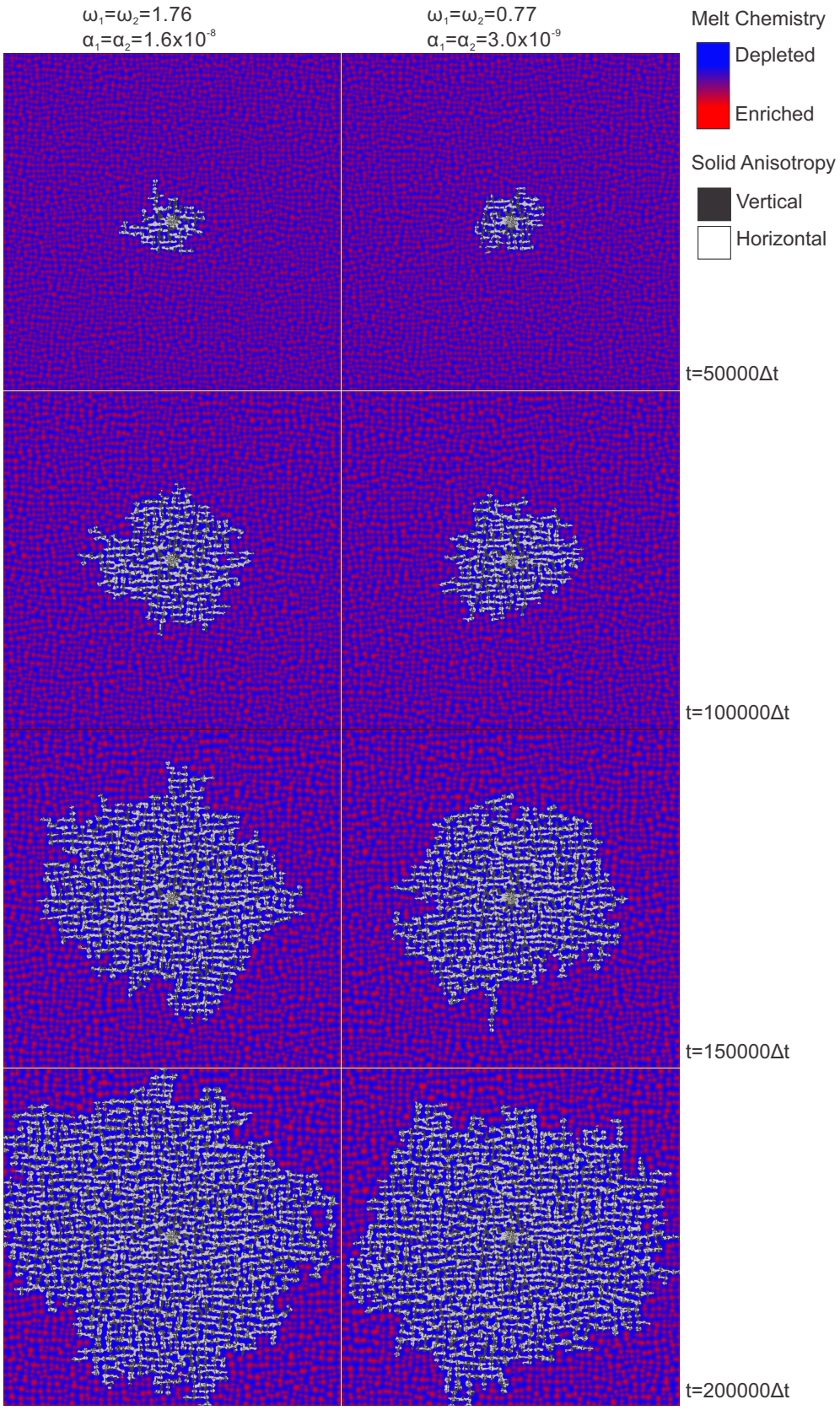


Figure 6.5 – Results of two extremes of physically realistic parameter sets in comparison. All the parameters used in this sweep are listed in list 6.12. Collision operator interleave disabled. Initial conditions of the liquid at $t = 0$ and $\Delta t = 2000$ not shown.

It is clear from the simulations, the geometry of the plagioclase rays that form the stellate is constrained by the number of degrees of freedom resulting from epitaxy. Reasonably then, in a magmatic system with parameters similar to what is used here would have a greater variety of angles between connected plagioclase crystals. This is due to the extra dimensionality in physical systems giving rise to more angles of epitaxial attachment, and composition dependent unit cell geometries. The most common angles of epitaxial attachment should correspond to the surface-to-surface combinations with the least amount of interfacial energy. It is highly plausible that most of the relationships between individual crystals in an array is a twin operation of some type, since naturally occurring twins have the least interfacial energy between the individual crystals. For plagioclase, the dependence of unit cell parameters combined with a large variety of twin operations aids in the growth of plagioclase stellate structure through the navigation of liquid heterogeneities. This means that a rock bearing a sufficiently high concentration of plagioclase crystallized, comparable to two-dimensional simulation values $R \sim 0.35$, should exhibit networks of plagioclase, not necessarily radial in symmetry, and analogous to what is seen in the liquid state with a percolation structure (probably what Philpotts et al. (1999) observed). For the formation of a radial stellate, a strong preference to twin on a given surface with a specific angle between the crystals at the expense of all other surfaces should yield a stellate structure with rays branching from each other at angles equal to the twin angle. For Carlsbad twins in plagioclase, this means the obtuse angles should be maximally $\sim 120^\circ$ when orientation permits (see figure 2.37 and related discussions). Indeed, there are numerous occurrences of angular branches that seem to fit this observation (see figure 2.7 for a typical radial stellate structure).

The concentration of the plagioclase along each ray decays from the center of the ray outwards, largely dictated by the heterogeneities of the liquid state. In a magmatic environment, this should dictate a spatial measure of the characteristic heterogeneity in the liquid state at the onset of plagioclase crystallization. If the exact mechanism causing the heterogeneity is clearly established with a known rate, then this could yield a length of time prior to the onset of plagioclase crystallization. For a layered igneous intrusion, this means the time between injections, since presumably, the nucleation induction period is nil in a layered magmatic system, since there are plethora of surfaces in the form of crystallites/crystals from previous layers, earlier crystallizing minerals, country rocks, etc. The central enrichment is needed to generate radial

symmetry. However, since the formation of the central enrichment is due to rejected solute and convective transport, it is presumably possible that some portions of stellate rays originate from convectively transported rejected solute as a form of genesis of the melt heterogeneities rather than spinodal nucleation and growth of other minerals in a given layer of the intrusion. Nevertheless, this is less likely as a general occurrence over the Rum igneous complex where stellates are found since there are numerous instances of patchy stellates with a distinct lack of radial symmetry supporting the existence of a percolation structure of the melt due to heterogeneities prior to the onset of plagioclase crystallization.

Based on known physical data on the Rum intrusions, we can now establish some constraints on important unknown parameters. The rate of growth of the solid in the simulation is $\sim N_{sim}/200000\Delta t_s$ or about 1 m per $200000\Delta t_s$. The rate of growth for the Rum rock plagioclase is not known directly, but suppose that the whole rock solidification rate established by Emeleus et al. (1996) is applicable (with great reservation) for the stellate rays, then the physical growth rate is $\sim [0.1, 1.0]$ m $/(365\times24\times60\times60)$ s $\approx [3.2\times10^{-9}, 3.2\times10^{-8}]$ m/s. The simulation growth rate is ~ 9.5 orders of magnitude higher than physical growth rate. This is a direct consequence of simulating macroscopic systems with microscopic kinetics/dynamics. To properly simulate the solid growth rate, we would need a rate factor in the form of a constant multiplying the $P1$ rate term derived from the Gaussian function. However, this is not currently possible with modern computers and the algorithms used due to the ~ 9.5 orders of magnitude increase in total simulation run time. Crystallization is an intrinsically slow process compared to other processes that occur in the liquid state, thus this is an anticipated compromise since we wish to simulate macroscopically based on microscopic kinetics/dynamics.

An interesting constraint we can infer from the results is the ratio of the time required to let the melt develop heterogeneities to the time required to grow a stellate pattern to a given diameter. The amount of time required to allow the liquid state to reach a desired amount of heterogeneity ($\langle\lambda\rangle \sim 10L$) is $\sim 10^3\Delta t_l$ while the amount of time required to grow the stellate to 1 m ($1001L$ diameter is $\sim 10^5\Delta t_s$. If hypothetically the time scale is about the same in the liquid as in the solid state so that $\Delta t_s \simeq \Delta t_l$, then the time required to grow the stellate (as measured by diameter) and the time required to increase $\langle\lambda\rangle$ by the same amount is about the same order of magnitude. However, it is seen that the solid state is simulated at ~ 9.5 orders of magnitudes faster than physical observations, hence, the proper ratio between the time scales in the liquid

state and the time scales in the solid state is $\sim 10^{9.5}$. If the processes that occur in the liquid state that contribute to the formation of the heterogeneities is purely classical nucleation and growth with a negligible nucleation induction period (so that the plagioclase grows to $1001L$ size and the non-plagioclase domains in the liquid state to $10L$ at a similar rate to the plagioclase growth), then this ratio reduces to about $\geq 10^2$ ($\sim 1001L/10L \approx 10^2$). This sets an estimate of the upper bound for the amount of time required before the commencement of mass plagioclase growth at about ~ 1 year and the full crystallization of plagioclase, assuming radial growth of a 1 meter stellate at ~ 10 years. No reasonable lower bound can be deduced the same way, since the minimum time required based on hetereogeneities in the melt is none, if a combination of effects of previous layers and existing melt heterogeneities prior to injection is present. In this case, a minimum time before start of plagioclase crystallization is best estimated from thermodynamic constraints by considering the inferred rate of cooling and the magma injection temperature. Using cooling rate estimates from Emeleus et al. (1996), $(1-10)$°C / year, and assuming an injection temperature of $\simeq 1300$°C, the melt would have required minimally $(5-50)$ years of cooling prior to the onset of plagioclase crystallization. Compared to this estimate based on the cooling rate, the time required to develop heterogeneities is trivial, and the time of crystallization of plagioclase is comparable. More details of this thermal history can be understood if we correlate the composition variation in plagioclase with supposed cooling rates. If the data from Labonté (2004) can be generalized for at least a few top layers of stellate bearing gabbroic layers, then the tight clustering of compositional data in the Ab-An system in the interval $\mathrm{An}(80-75)$ with less frequent amounts of $\sim\mathrm{An}(50-75)$ suggests that a large amount of the plagioclase in the stellate layers were crystallized while the temperatures in the melt were not changing significantly (likely from about 1300°C to 1275°C at about 10 kbar), and lesser time spent crystallizing lower An content plagioclase (perhaps due to re-injection events either intra-layer or new layers).

In addition, for the $\Sigma\Delta t_{liquid}$ chosen, we can calculate the distribution of plagioclase along the stellates by looking at the histogram of the concentrations of $\rho_1 = \rho_p$. Although some of this information is known in a different form via the measure σ_{std}, a histogram gives much more direct information about the solid and liquid states without assumptions about their respective distribution shapes. Corresponding to the results shown above with the upper bound set of ω, α with $\Sigma\Delta t_{liquid} = 2000\Delta t$, the histograms are given as a sub-figure in figure 6.6. It can be

clearly seen that the most probable value of ρ_1, ρ_2 are $R, (1-R)$ as expected.

As the crystallization continues, the growing solid state preferentially incorporates ρ_1 leaving the liquid residual enriched in the rejected ρ_2, so we expect this is reflected at large Δt. For simulation time total of $200000\Delta t$, the histograms for the solid state are shown as a sub-figure in figure 6.6 while for the residual liquid, it is in figure 6.6. In the solid state histogram figure, the distribution of ρ_1 represents the actual amount of plagioclase along stellates where as ρ_2 are incorporated residual liquid in the stellate, corresponding to inclusions of olivine, clinopyroxenes, etc. In the residual liquid state histogram, the opposite is true, although no phase transition algorithms are used to simulate phase transitions of non-plagioclase chemicals, and therefore the residual liquid should be considered an amalgamation of all non-plagioclase liquid and solid chemicals. Since the liquid residual histograms represent a time in the simulation such that the solid state radius roughly equalling the simulation grid size, some enriched liquid domains remain in the liquid state. Therefore, the final residual liquid histograms represent a mostly crystallized state (about 78.5% crystallized). The solid state composition is more enriched than the initial heterogeneities, due to a combination of rejection of residual and further liquid state heterogeneity development. It is obvious that the solid state composition strongly depends on the value of σ, b as seen before during the respective parameter sweeps. By adjusting σ, b and to a lesser extent R, we can produce a range of plagioclase proportions in the solid state to match experimental data. This is beyond the scope of this thesis as it involves the complete statistical description of stellate patterns with respect to their modal compositions.

Based on the patterns observed in the realistic scenarios, it certainly seems possible that the genesis of plagioclase stellates depends on an interplay of melt heterogeneities and repeated epitaxial growth at the growth front combined with relative tranquility of the liquid state as hypothesized in figure 3.2. Since the growth rate is much slower than possible flow velocities for the liquid state, any large-scaled (on the order of the size of a stellate pattern) flow in such thin igneous layers would result in shear and destroy the ability to form stellate patterns, unless the large-scaled flow velocities are much smaller compared to crystal growth rates (as it is seen in our simulation results, even small changes in the melt state can drastically alter the morphology of the stellate structures, see the discussion and results of sub-section 6.2). This does not, however, eliminate local flow, and obviously any flow prior to the onset of plagioclase stellate growth. It is very possible that the local enrichments in plagioclase where the stellate patterns are found are

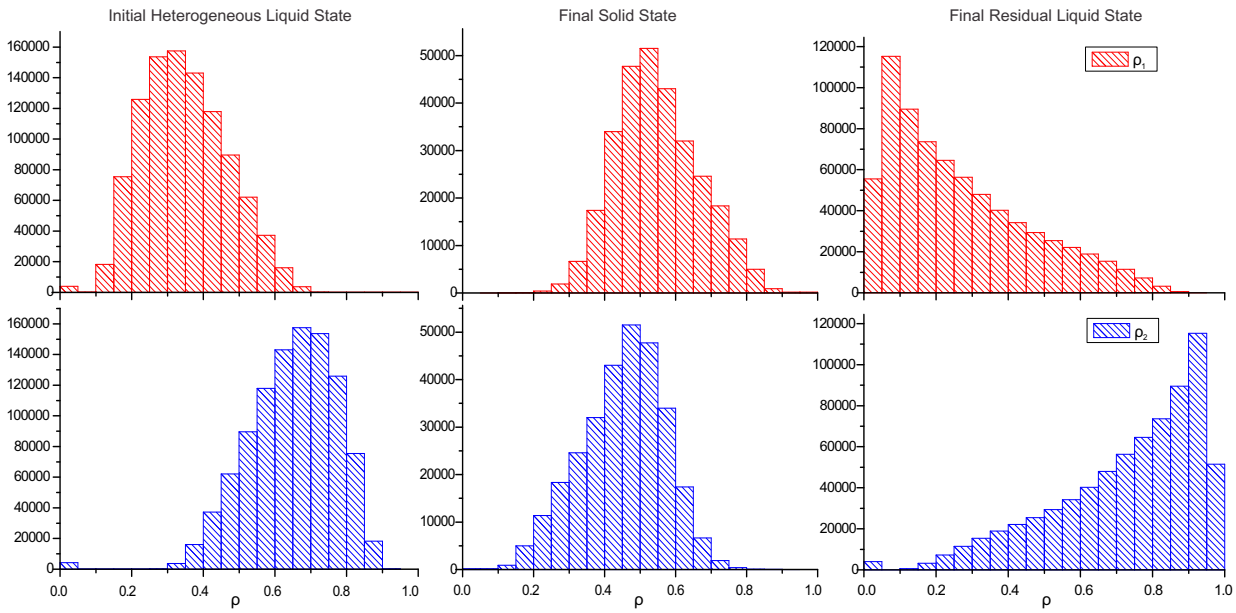


Figure 6.6 – Histograms of concentrations of ρ_1 and ρ_2 in the initial heterogeneous liquid ($t = 2000\Delta t$), and subsequent solid and liquid states ($t = 200000\Delta t$) for the upper bound set of ω, α.

caused by a combination of intrinsic processes that occur in the melt in each layer due to phase transitions (spinodal nucleation and/or subsequent crystallization) and extrinsic processes that occur between layers due to infiltration/convection/injection of residual melt as hypothesized in figure 3.1. The exact physical mechanisms responsible for seemingly pushing the extrinsic source of plagioclase enrichment into the top of each igneous layer in the context of layered igneous intrusions has the usual suspects - residual melt buoyancy/convection, injection pressure related viscous fingering/upward melt movement (as first pointed out by Labonté (2004)) and perhaps to a lesser extent, some magma source related fluctuations in silica content. By extending our model into three dimensions, coupling with some forces including gravity, some of these additional mechanisms can be incorporated and simulated. However, memory requirements will curtail such efforts for the near future.

6.3 Analysis of Results

While the idea of comparing the synthetic stellates to the naturally occurring stellates is the most logical, it is unfortunately, not very practical. This process would involve at the very least another field session to Rum for sampling. The original samples were taken under heavy preservation constraints. As such, the samples were mostly for compositional analysis and not for meter-scaled pattern analysis. The specific analysis efforts are also quite laborious; a complete characterization of a natural stellate pattern can easily constitute a thesis. In such a characterization, one would have to study, at least in two dimensions, some principal measures such as the distribution of twins along the branches, the density of twins at the ray junctions and the plagioclase content along the rays. From these principal measures, one can construct for example, the correlation between twin densities and types of twins and between the plagioclase concentration and the radial distance (for radial systems). Imaging techniques that results in a three-dimensional, tomographic image would also be ideal, although this is probably beyond current technological capabilities. Instead of comparisons of the synthetic and natural stellates, this thesis shall focus on the analysis of our synthetic stellates, due to the above mentioned difficulties.

For the first analysis, we can generate a histogram of a large-time stellate pattern, categorizing the liquid and solid cells. Large-time is some minimum number of simulation time steps such that the solid structure typically grows large enough compared to the simulation grid dimensions, as determined for realistic growth parameters. In particular, we can show that the frequency of both structural types is nearly equal, for sufficiently large patterns. Such a histogram is presented in figure 6.7.

For the next analysis, we can compute the solid-liquid correlation as a function of radial distance. If the lattice cells are binned radially over all angles for each fixed increment to the radius, then it is possible to compute the sum over the cells inside each bin such that, a solid of any type is considered a one, and a liquid is considered a zero. Then we can produce a plot of the radial correlation function for a typical stellate run, using realistic parameters. Such a plot can be seen in figure 6.8.

However, such a plot does not yield much information about the structure. It simply states that the solid is roughly about 400 Δx big in radius, taking the half height at the steep slope

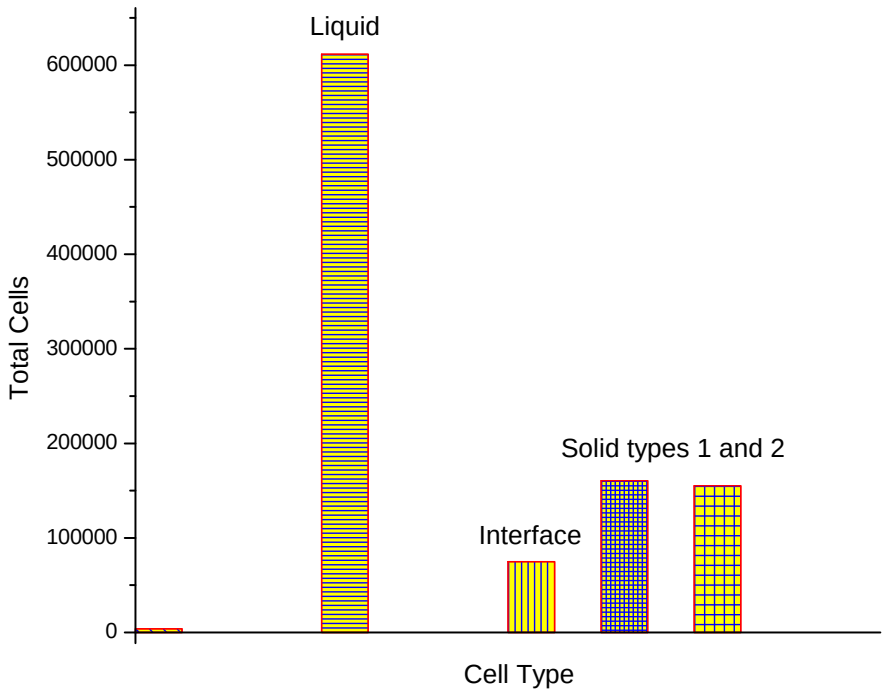


Figure 6.7 – Histogram of cell types of a synthetic stellate at large times.

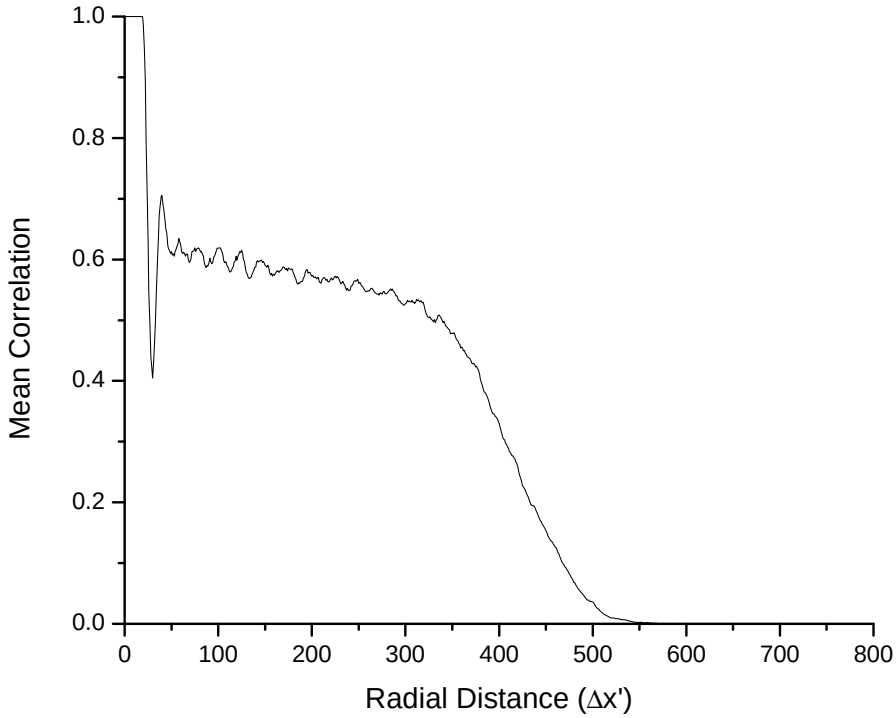


Figure 6.8 – Radial correlation function of a synthetic stellate for large times, realistic stellate parameters used, total simulation time $\sim 200000\Delta t$.

of the mean correlation function. At the centre, where the initial rich liquid pocket is found, the correlation is at its maximum value - 1.

Since the stellate structure seems to display angle dependence, coinciding with the principal growth directions of the anisotropic solid, it is possible to analyze the large-time, large-radius angle dependence of the solid structure. A correlation can be computed by binning the cells into single angular degree increment slices around the grid centre. In this case, the bins are kept to the same size by only counting cells that are within an Euclidean circle defined with the radius equal to the distance from the centre of the grid to the nearest edge of the grid. The actual correlation is computed by summing over each slice (each degree) a count that is one if the underlying cell is a solid (of any type) and zero otherwise. Fortran's **atan2** function combined with integer rounding function **anint** facilitate single-pass binning of the data for each simulation run. Since the anisotropy is measured by Asr, the anisotropic aspect ratio, it is possible to sweep a few Asr values, covering a range of solid anisotropies, and compare the overall correlation function. The correlation result from a single synthetic stellate is rather noisy. Therefore to reduce the noise, multiple simulations results (~ 25) are taken per Asr value. The mean correlation results of this analysis is given in figure 6.9.

These average correlation functions seem to suggest that for large-radius and large times, the distribution of the solid structure is angularly independent. This is not unexpected, since the average angular structure depends on the liquid heterogeneities. Therefore, for different initializations of the liquid state, the solid distribution should differ on the scale of the heterogeneities, which is much smaller than the large-radius chosen. Therefore, when the averaged large-radius results are taken, with the radius much larger than the characteristic (mean) spacing of the heterogeneities in the liquid state, the structure should appear to be isotropic.

At a much smaller, "local"-scale, the structure is certainly expected to be anisotropic, seen as nearly orthogonal stretches of stellate rays in the synthetic stellates. This is better captured by looking at small-radius, large-time averaged angular correlations. This can be accomplished by restricting the radius to a small neighbourhood around a given solid cell and computing the average angular correlation confined to this neighbourhood. This small-radius size can be varied to observe the angular correlation as a function of max neighbourhood size. This is done for the same Asr values used earlier. The results of $Asr = 1$, $Asr = 8$, $Asr = 20$ and $Asr = 32$ are shown respectively in figures 6.10, 6.11, 6.12 and 6.13.

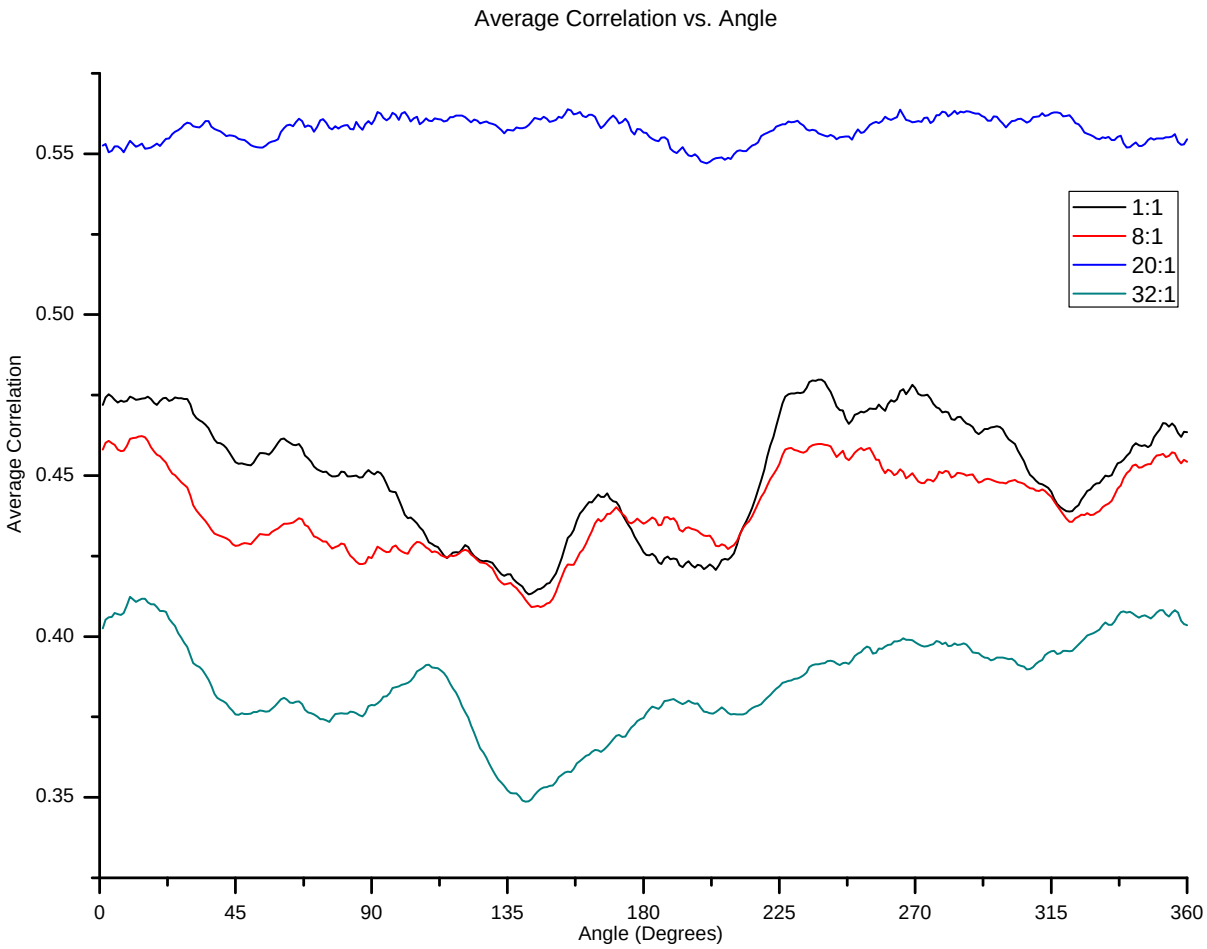


Figure 6.9 – Large-radius, large-time, average angular correlation of synthetic stellates as a function of varying amounts of solid anisotropy (Asr=1,8,20,32).

At neighbourhood radius equals to 1, the only peaks present corresponds to the angles where nearest neighbours can be found, given the lattice geometry (multiples of 45°). Hence, this is considered a spatial quantization error when a circular neighbourhood is projected onto the underlying square lattice. As the neighbourhood size becomes progressively larger, this quantization error subsides, and the approximation to a circle becomes correspondingly better. This is a common trend across all Asr values. For $Asr = 1$, it is clearly seen that there is no angle-dependence of the solid structure at any radii aside from quantization errors. However, for $Asr \geq 8$, the development of a large-wavelength (far larger than the quantization error wavelength) sinusoidal-like variation in the correlation function from neighbourhood radius of ~ 10 and up is unmistakable. In fact, for progressively higher Asr values, the principal growth directions of the solid becomes the only distinguishing feature on these correlation figures once

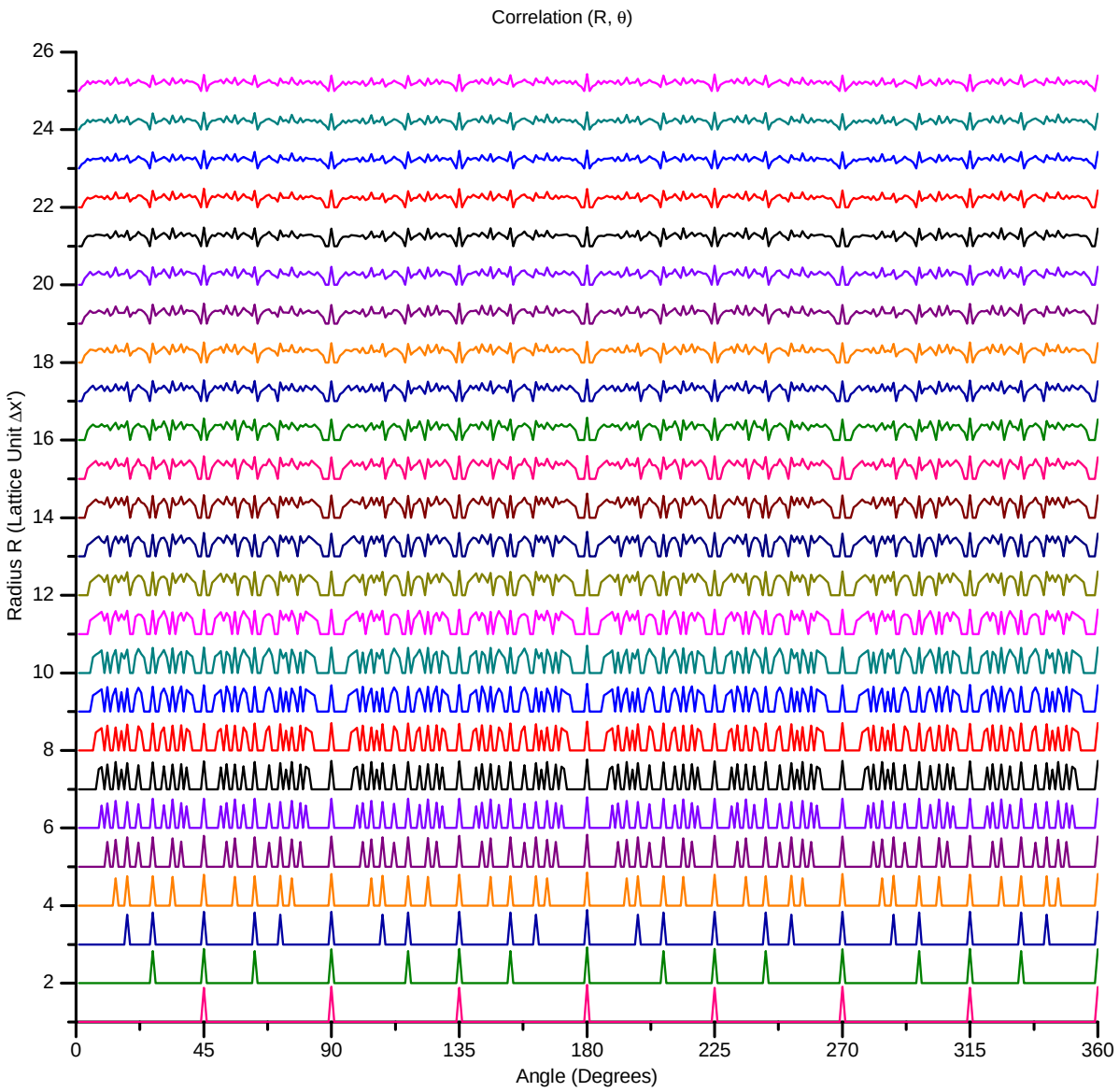


Figure 6.10 – Small-radius, large-time, average angular correlation of synthetic stellates, $Asr = 1$.

the radius is large enough, since the quantization error of the lattice is seen to be attenuated for neighbourhood sizes larger than ~ 10.

Therefore, if the information for large-radius and small-radius correlations are combined, we can interpret that the solid structure is isotropic regardless of solid anisotropies if the observation is on a scale much larger than the individual crystal size in the synthetic stellates. However, at intermediate scales, when the characteristic radii is much smaller than the entire stellate system (or lattice), but about an order of magnitude bigger than the smallest quantization

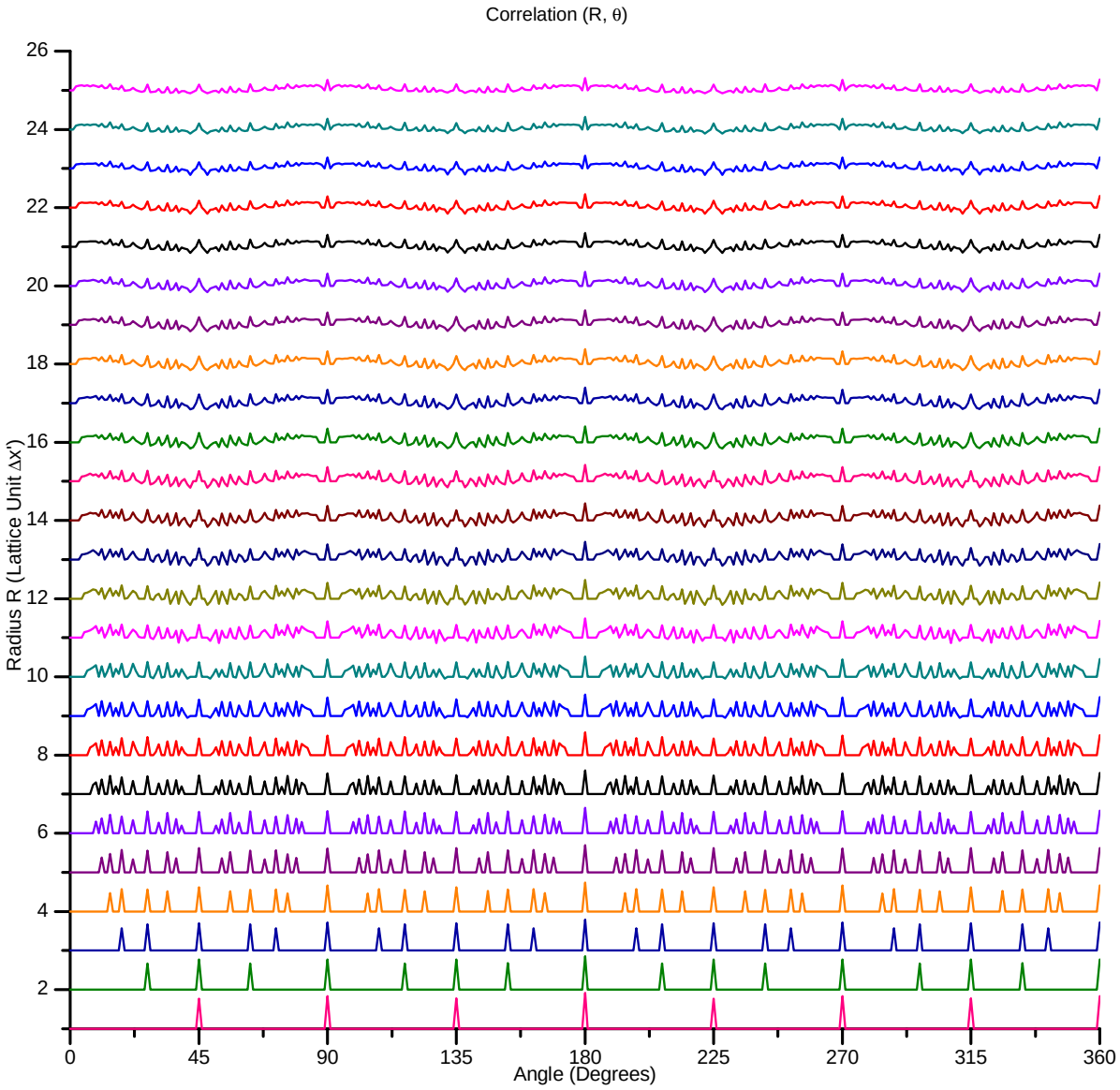


Figure 6.11 – Small-radius, large-time, average angular correlation of synthetic stellates, $Asr = 8$.

of the lattice, the synthetic solid structure is angularly dependant. If this observation can be extended to natural stellate systems, we can expect that on the scale of the melt heterogeneities, the plagioclase crystals should exhibit abundant epitaxy (twinning, for example), and at large scales, perhaps at the scale of a meter-sized stellate pattern, the radial pattern itself is isotropic and not coincident with the solid anisotropies. A corollary of this can be stated as: if plagioclase (or any other crystal, for that matter) can exhibit epitaxy with a characteristic wavelength spanning a few scales of observation, and the melt can exhibit a few corresponding scales of

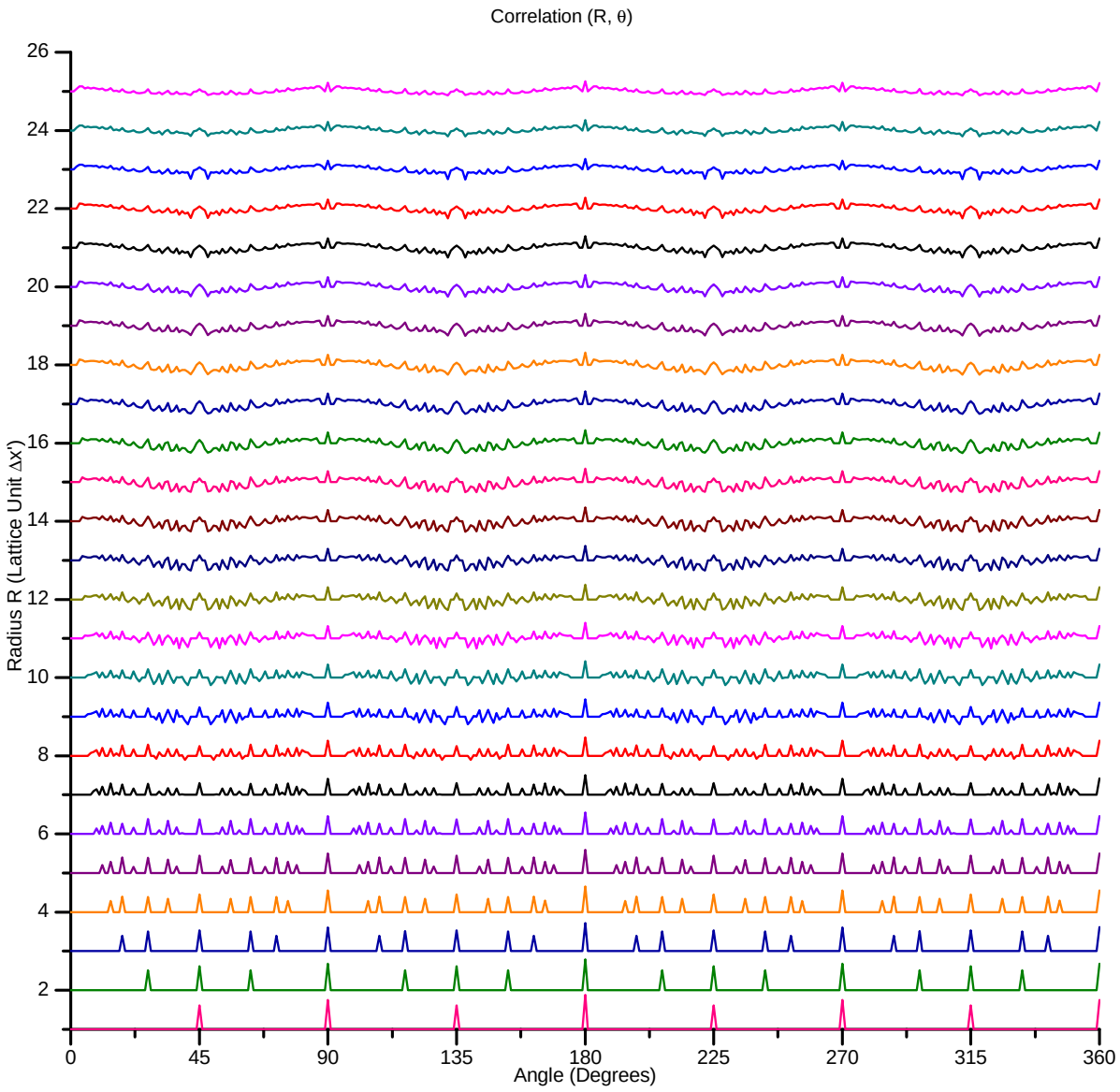


Figure 6.12 – Small-radius, large-time, average angular correlation of synthetic stellates, $Asr = 20$.

spatial-chemical heterogeneities, then it is possible to obtain a stellate-like pattern on different scales, from the meter-or-more-scaled patterns at Rum to cm-scaled or less. Of course, whether or not the pattern is self-similar over these different scales will depend strongly on the strength of the coupling of plagioclase growth kinetics to the changing thermodynamic variables for each spatial scale. For example, for higher crystallization rates and finer melt heterogeneities, the plagioclase may crystallize very finely and exhibit finer epitaxy intervals, the resulting pattern will still be stellate-like if the crystal habit for both the higher growth rate and finer melt

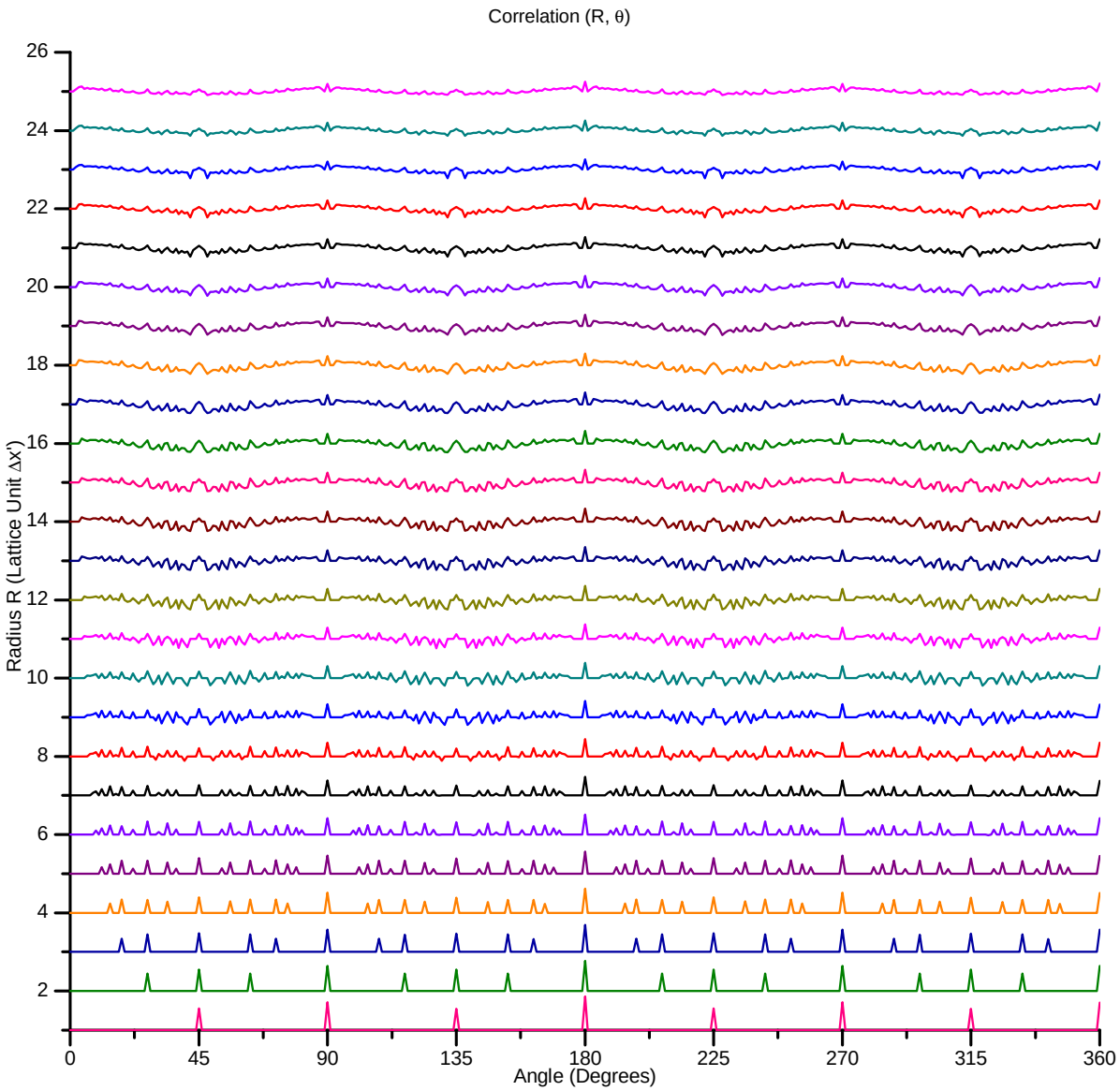


Figure 6.13 – Small-radius, large-time, average angular correlation of synthetic stellates, $Asr = 32$.

heterogeneity does not change appreciably. When the crystal habit is changed drastically, such as strongly driven systems in which, the plagioclase becomes skeletal to lath-like, the overall pattern will likely be incomparable to stellates and much smaller in scale. In systems where the crystallization takes place extremely slowly, the epitaxy of plagioclase may be disfavoured, so that a stellate-like pattern is not developed, regardless of scale. These considerations coincide with our initial hypothesis that Rum exhibits a crystallization environment for plagioclase that is certainly neither too slow, nor too fast to allow for proper wavelengths of epitaxy to develop,

navigating the melt heterogeneities.

A detailed examination of the solid structure for the range of Asr values considered is possible if only one of the two structural types (e.g., horizontal) is plotted. The results of $Asr = 1$, $Asr = 8$, $Asr = 20$ and $Asr = 32$ are shown respectively in figures 6.14, 6.15, 6.16 and 6.17.

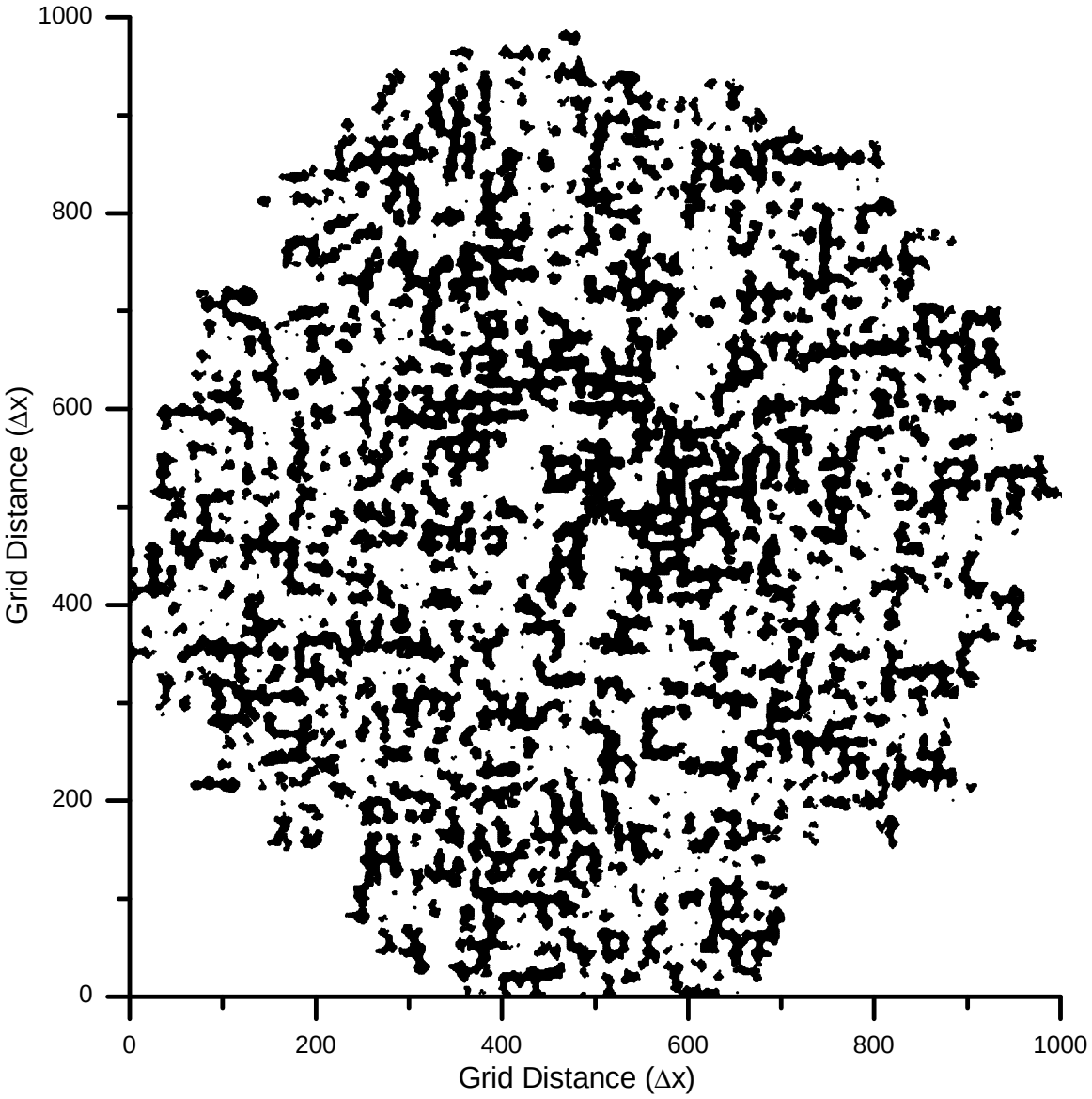


Figure 6.14 – Typical plotted image of the horizontal solid structure for $Asr = 1$.

When $Asr = 1$, the solid structure does not seem to exhibit preferential alignments, nor does it appear to exhibit long-range continuity. When $Asr \geq 8$, it becomes clear that the long-range alignment is horizontal and that there are long chains of continuous, elongated plagioclase

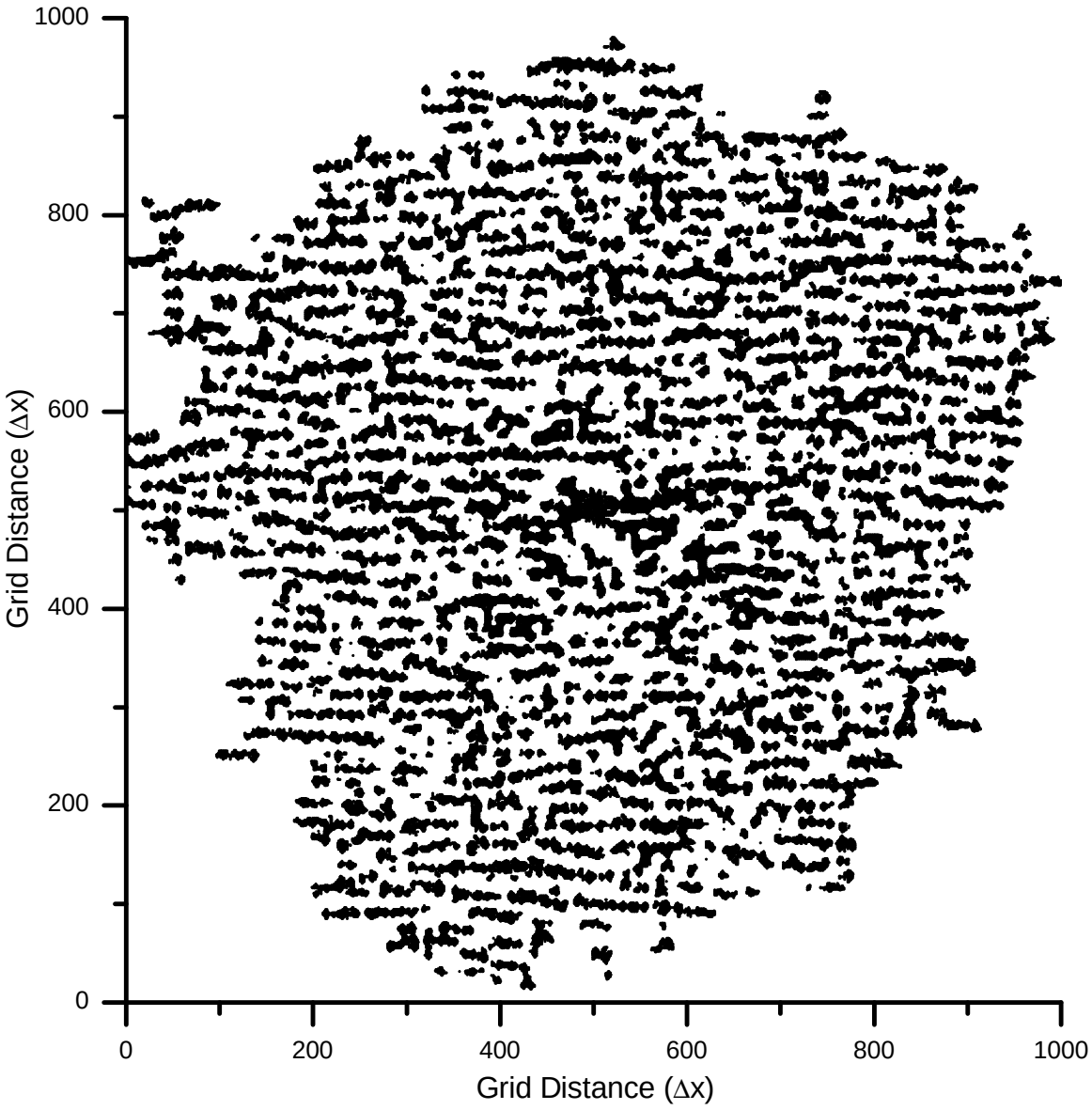


Figure 6.15 – Typical plotted image of the horizontal solid structure for $Asr = 8$.

that seems to be continuous for $\sim$ 10cm. As this value is increased further, the most obvious observation is that the elongated plagioclase seems to be continuous over a much larger distance of $\sim$ 20cm or more. In the case when $Asr = 32$, individual plagioclase crystals can be seen spanning nearly half ($\sim$ 50cm) of the entire grid. If this trend holds for natural stellate systems, then the ability for a single plagioclase crystal to span a ray is increased dramatically as the underlying crystal becomes more anisotropic in directional growth rates. Since the junctions of stellate rays are epitaxially related with other stellate rays, which in our case are orthogonal, the behaviour of long stretches of plagioclase in natural stellates that seem to stem from an

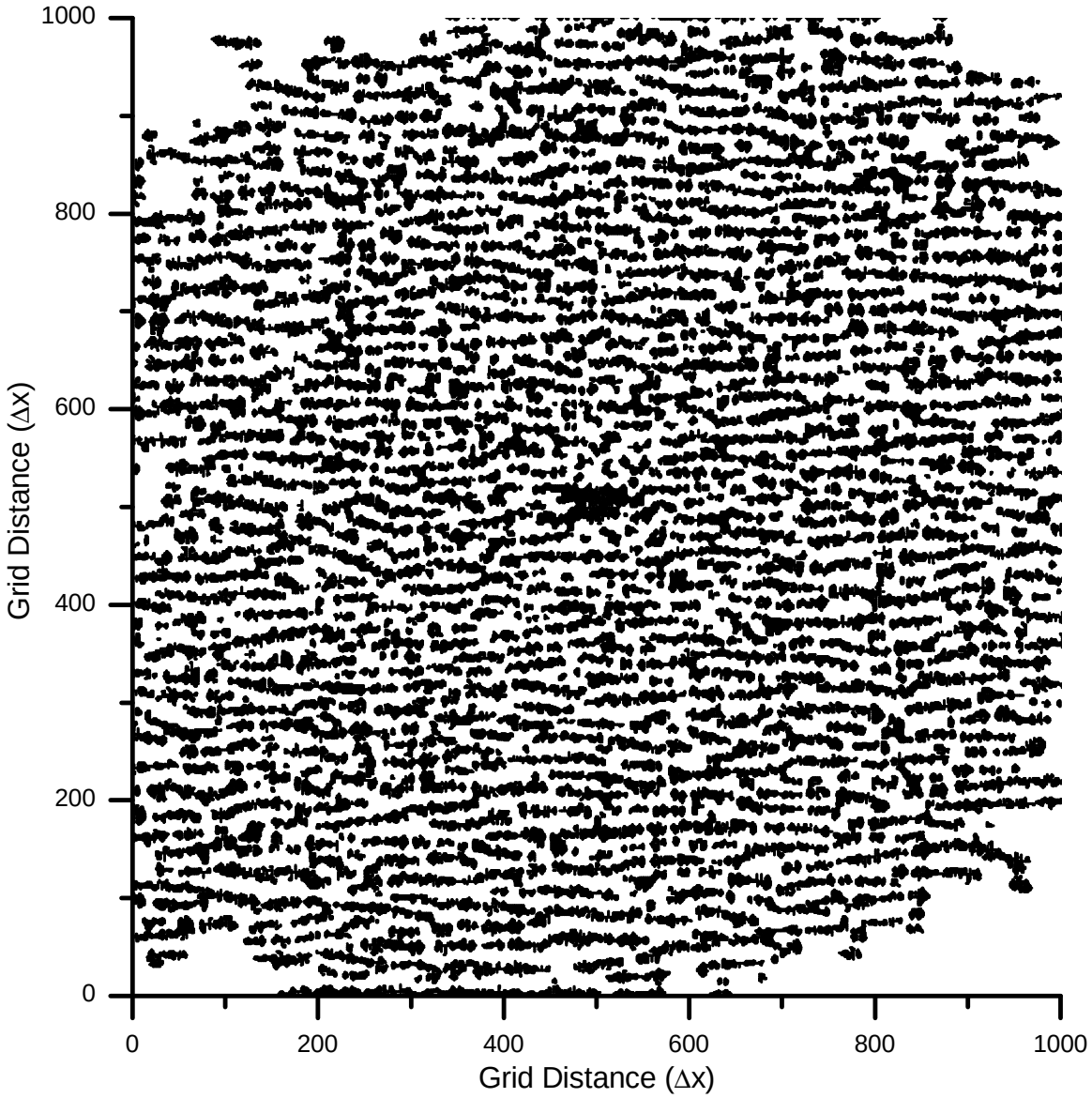


Figure 6.16 – Typical plotted image of the horizontal solid structure for $Asr = 20$.

existing junction appears to be captured adequately by the model. An additional observation on the solid structure is that for $Asr = 20$, there appears to be an increase in the growth rate of the solid structure, while for all other values of Asr, the growth rate is retarded. Since the epitaxy threshold a_2 is held constant for varying values of Asr, the faster growth rate can be interpreted to be an ideal amount of solid anisotropy for a given value of a_2, R and β_0, such that the gain in growth rate due to epitaxial realignment far outweighs the cost to undergo epitaxy. Therefore, at an optimal value of Asr, the solid structure exhibits a maximal growth rate, when all other parameters are fixed. This should translate to the natural stellates if the energy to

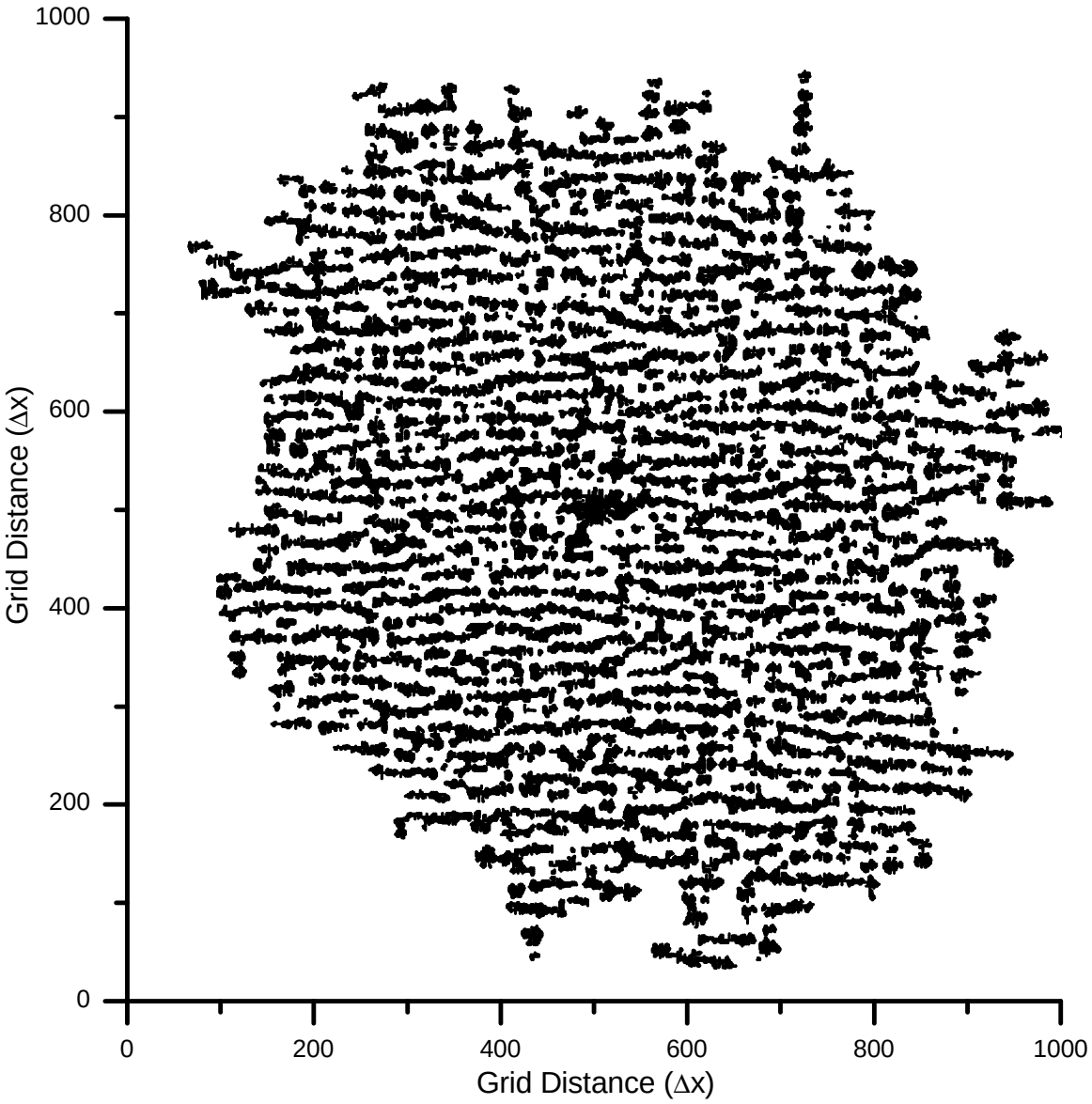


Figure 6.17 – Typical plotted image of the horizontal solid structure for $Asr = 32$.

undergo epitaxy for plagioclase is minimal, but non-zero, and the liquid state is sufficiently heterogeneous.

The epitaxy behaviour in the solid structure is very dependent on some crucial parameters of the simulation, most importantly, the mixing length parameter β_0 and the melt plagioclase enrichment parameter R. Since epitaxy is encouraged when the liquid state concentration is high enough relative to the surface attachment kinetics, it can be expected that a lower value of β_0 would discourage epitaxy, and vice versa. Similarly, when the value of R is higher, it can be expected that on average, the frequency of epitaxy is increased. The results of both sweeps are

probably best analyzed by using 2D-FFTs. Since the 2D-FFT results are isotropic with respect to the principal directions of the grid, the horizontal and vertical FFT spectra are averaged to produce a single, one-dimensional figure that is easier to comprehend. A set of results capturing the effect of changing β_0 on the solid epitaxy FFT is shown in figure 6.18.

There are two obvious features in figure 6.18. The first is the development of a frequency peak centred around $\sim$ 45Hz which translates to a wavelength of $\approx 22\Delta x' = 22$mm (22 dimensionalized lattice distance units). Since the spacing between the synthetic stellate-solids is about equal to the stellate widths, it is safe to assume that this frequency characterizes the alternation of stellate ray and spacings. This frequency peak is enhanced when the value of β_0 is around 0.15 and diminished for all other values. Since the kinetic barrier to epitaxy is controlled through a threshold value a_2, and the chemical concentration is weighed by the Gaussian function with parameters σ, b, the exact value of β_0 that contributes to the maximal development of this spatial frequency can be adjusted by compensating the parameters a_2, σ, b. However, the value of a_2 is usually chosen to be small, so that the kinetic barrier to epitaxy is small. Hence, for fixed values of a_2 and β_0, the sensitivity of plagioclase growth to the concentration of the melt dictates the epitaxy frequency. The second is the relative lack of high frequency content as β_0 is lowered. Obviously this is due to the relatively poor separation in the melt state with a relatively low saturation of plagioclase-compatible chemicals in domains. The slow growth rate in this case discourages epitaxy. To illustrate these observations better, some typical solid state images are plotted showing the change in the overall structure for each value of β_0 in figures 6.19,6.20,6.21,6.22,6.23 and 6.24.

If the chemical heterogeneities in the melt played a major role as hypothesized, then the stellate patterns should be reflective of the underlying strength of separation. If the thermal aspects of the natural system can be expected to be uniform over an appreciable portion of the sampled area, then the density of epitaxy in the said area should be positively correlated with the extent of chemical separation in the melt. Further more, if the composition of the layers do not change appreciably over the CS, then the intra-layer processes generating melt heterogeneities must have been at its greatest where the stellate patterns are seen in the CS, if the stellate patterns truly do not exist outside of the sampled area (i.e., not due to lack of weathering, etc.). The model presented here is unable to substantiate the relative contribution of inter-layer to intra-layer effects, and thus, the presence or lack of stellate patterns over the

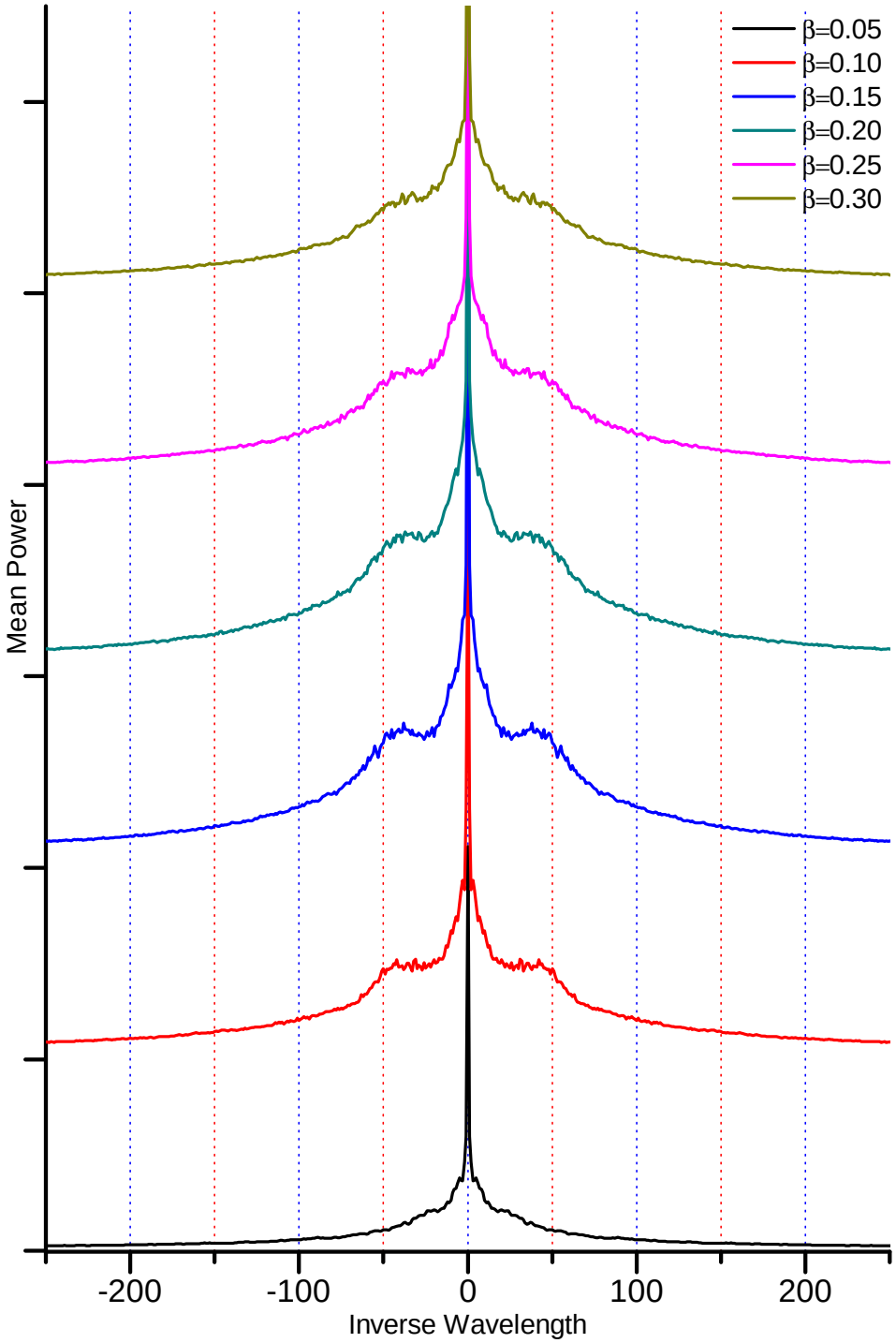


Figure 6.18 – 2D-FFT transforms averaged along principal dimensions down to 1D of the results from sweeping values of β_0, total simulation time =200000Δt per β_0, all other parameters taken from the realistic stellate parameters.

CS could also be contributed by weaker inter-layer interactions.

Changes in parameter R should allow for more or less epitaxy. If R is increased, it becomes

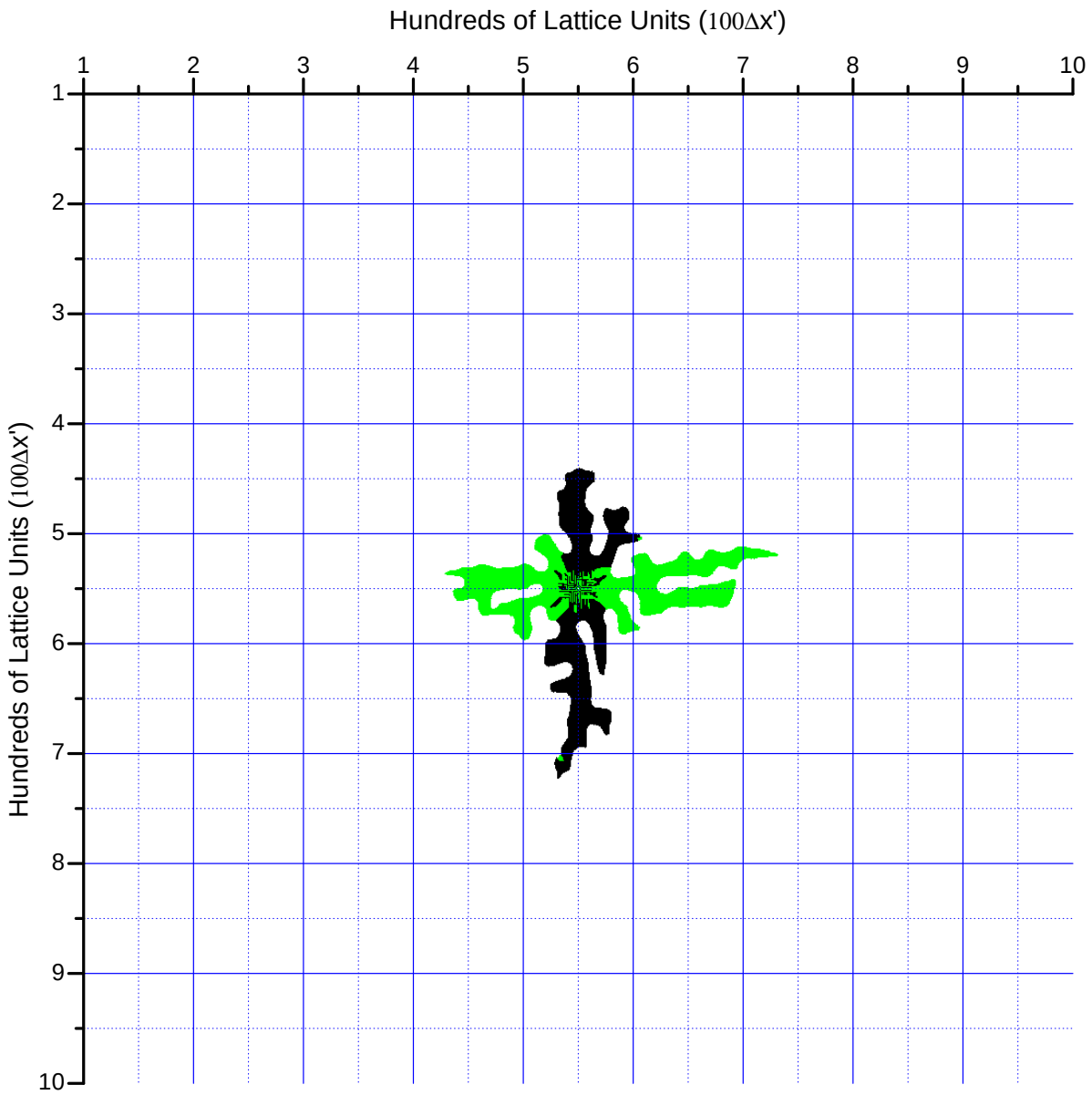


Figure 6.19 – Plotted image of the synthetic stellate solid. $\beta_0 = 0.05$, total simulation time $=200000\Delta t$, all other parameters taken from the realistic stellate parameters.

more probable to find growth sites such that the growth becomes surface attachment limited, when all other parameters are fixed. Therefore, it is expected that the FFT spectra of various values of R will reflect the increase in higher frequency content as R is increased. The results of this sweep is shown in figure 6.25.

Indeed, the higher frequency content is more abundant for higher values of R. It can also be seen that the position of the high frequency peak (the "shoulder" in the FFT spectra) not only develops for $R \geq 0.35$ but also slightly moves out, towards higher frequencies. This is reflecting

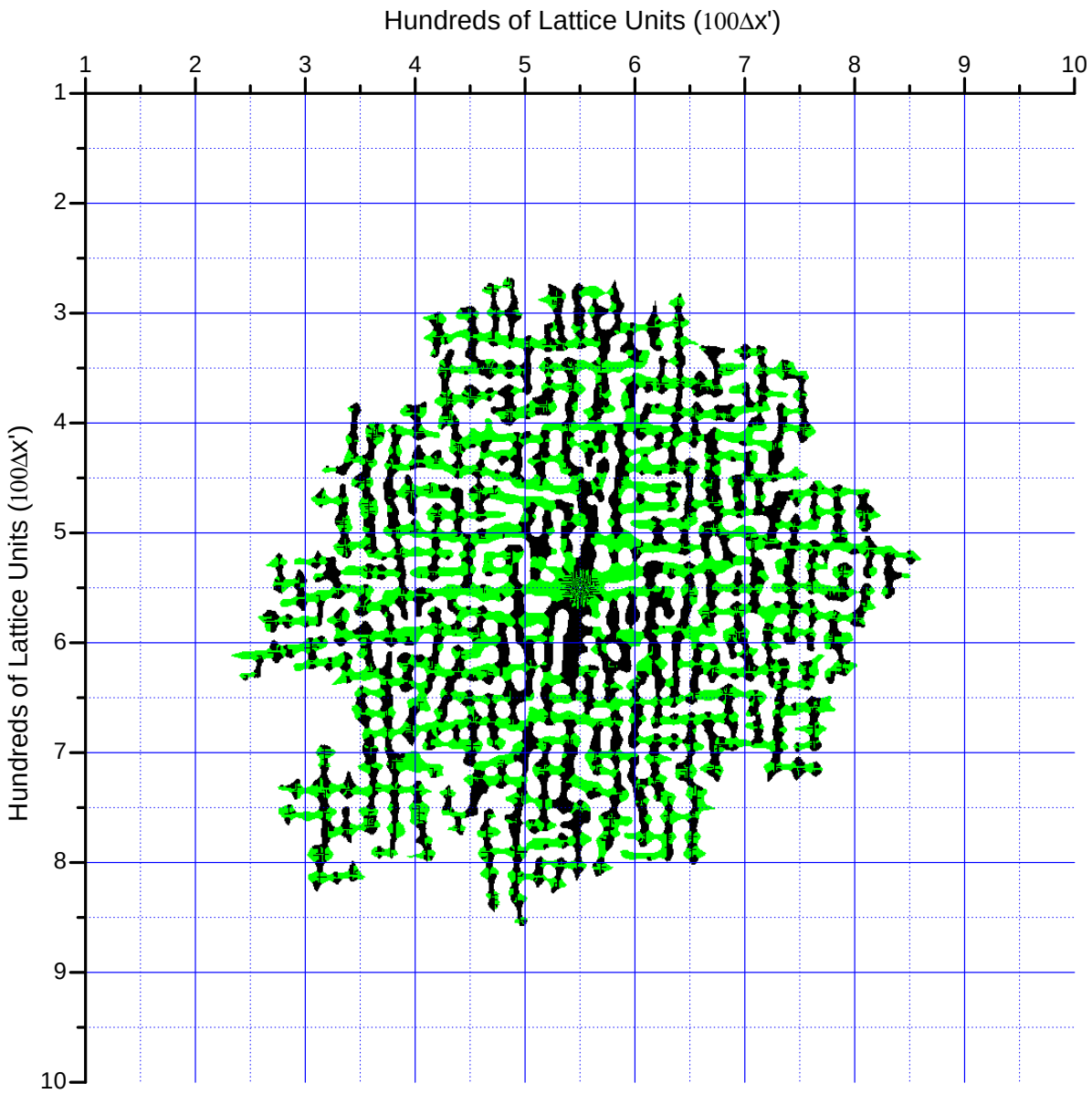


Figure 6.20 – Plotted image of the synthetic stellate solid. $\beta_0 = 0.10$, total simulation time $=200000\Delta t$, all other parameters taken from the realistic stellate parameters.

the average increase of the frequency of epitaxy as R is increased. Again, the plotted images of the solid-only structure are shown in figures 6.26,6.27,6.28,6.29,6.30 and 6.31.

From these images, it can be seen that a higher value of R past 0.35 results in such rapid growth that the solid completely fills the available space on the lattice. It can also be seen that the growth is highly sensitive to the parameter R, due to the selection of parameters (σ, b). If the natural stellate systems are similar to the synthetic ones in this regard, then the type of pattern observed should be reflective of the volume fraction of the plagioclase compatible chemicals in the

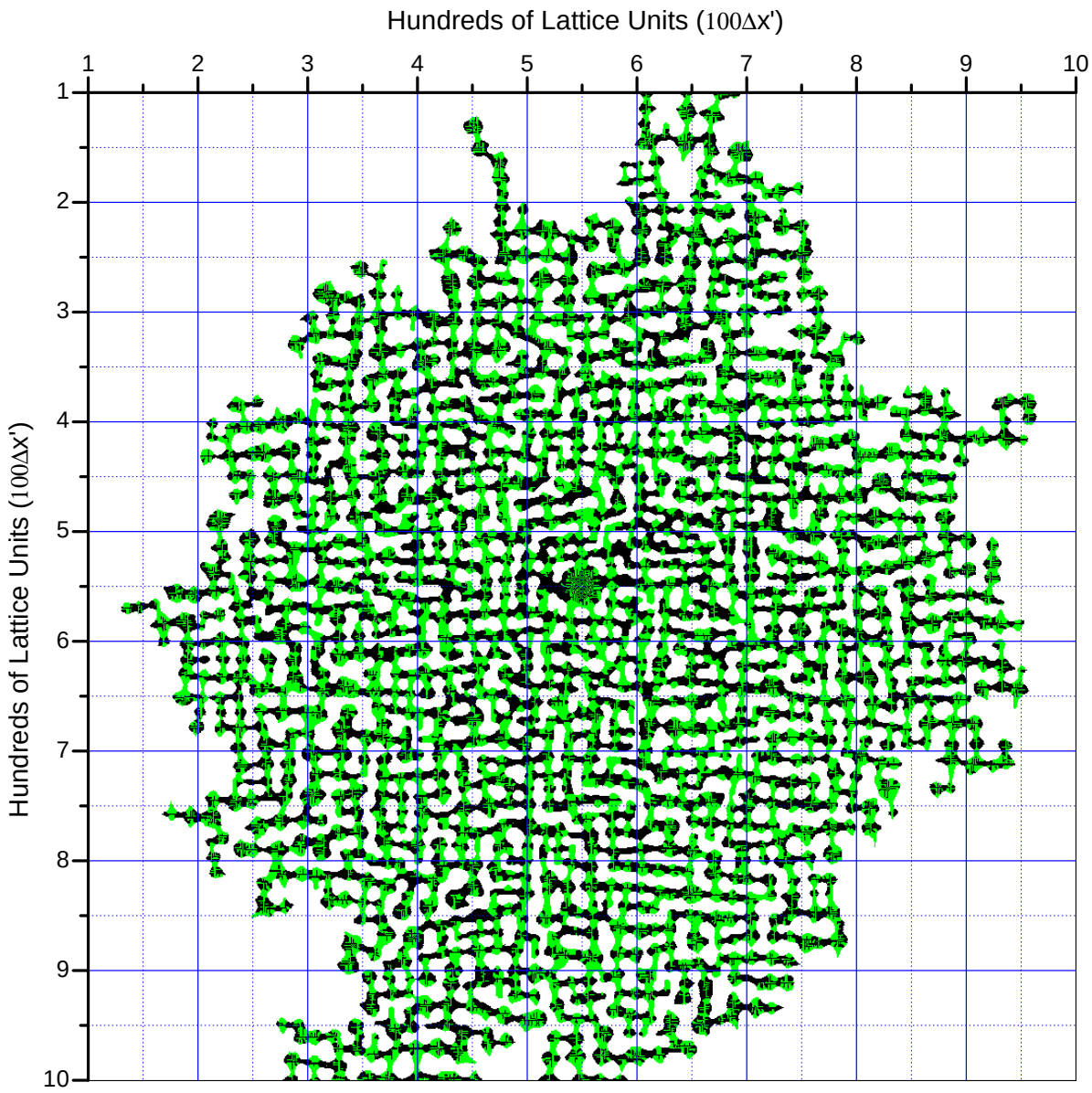


Figure 6.21 – Plotted image of the synthetic stellate solid. $\beta_0 = 0.15$, total simulation time $=200000\Delta t$, all other parameters taken from the realistic stellate parameters.

melt. Where the stellates are very interconnected and massive, without clearly distinguishable centres, the volume fraction of the chemicals is likely very high. This variation can be due to the unevenness of the inter-layer interactions or factors enhancing intra-layer interactions. The characteristic spatial separation between the synthetic rays of plagioclase is on the order of cm, which is the same as in the natural stellates. Therefore, the spatial separation of natural stellate rays, as well as their long, stretching continuity and junction-epitaxy relationship seems to be adequately similar to the synthetic stellate rays.

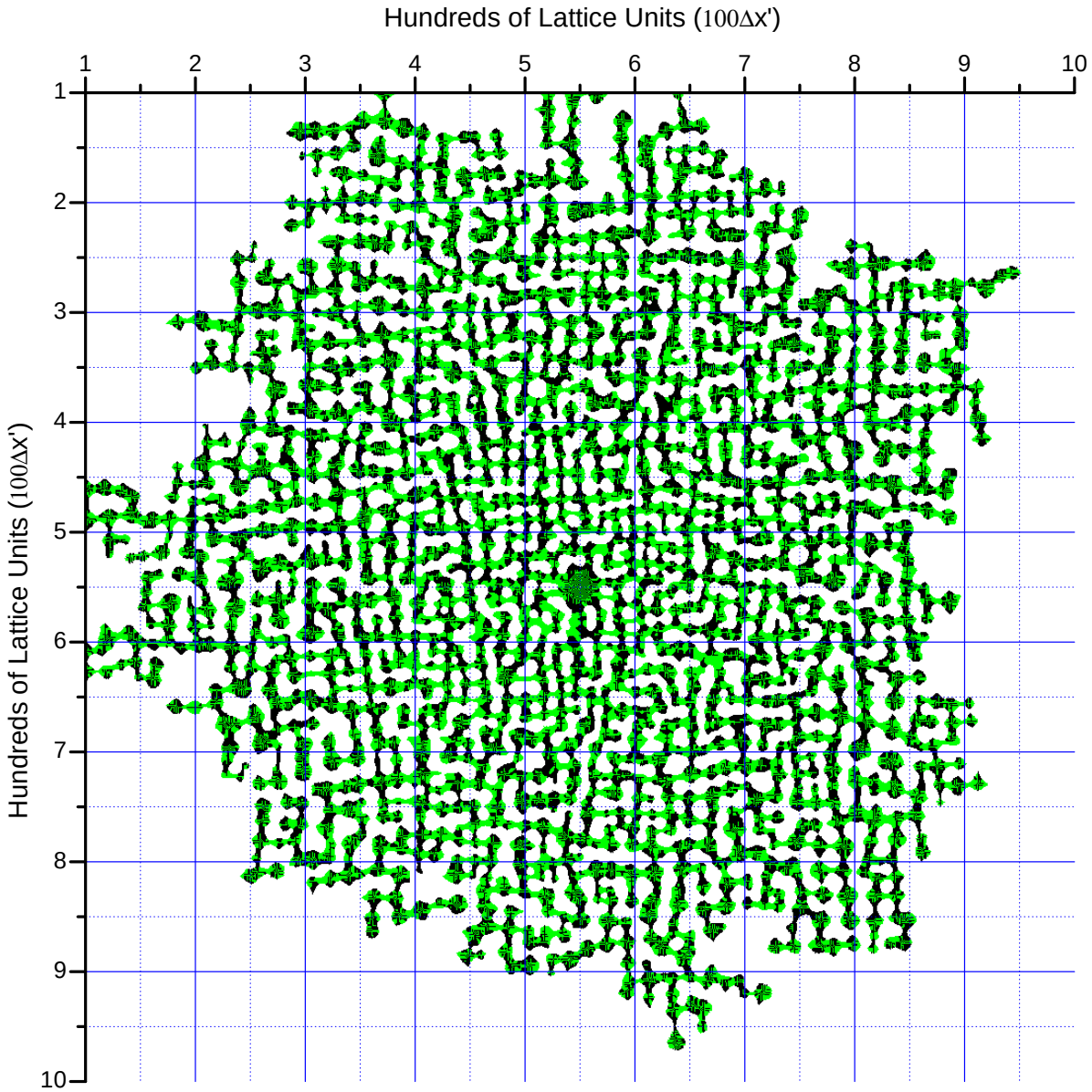


Figure 6.22 – Plotted image of the synthetic stellate solid. $\beta_0 = 0.20$, total simulation time $=200000\Delta t$, all other parameters taken from the realistic stellate parameters.

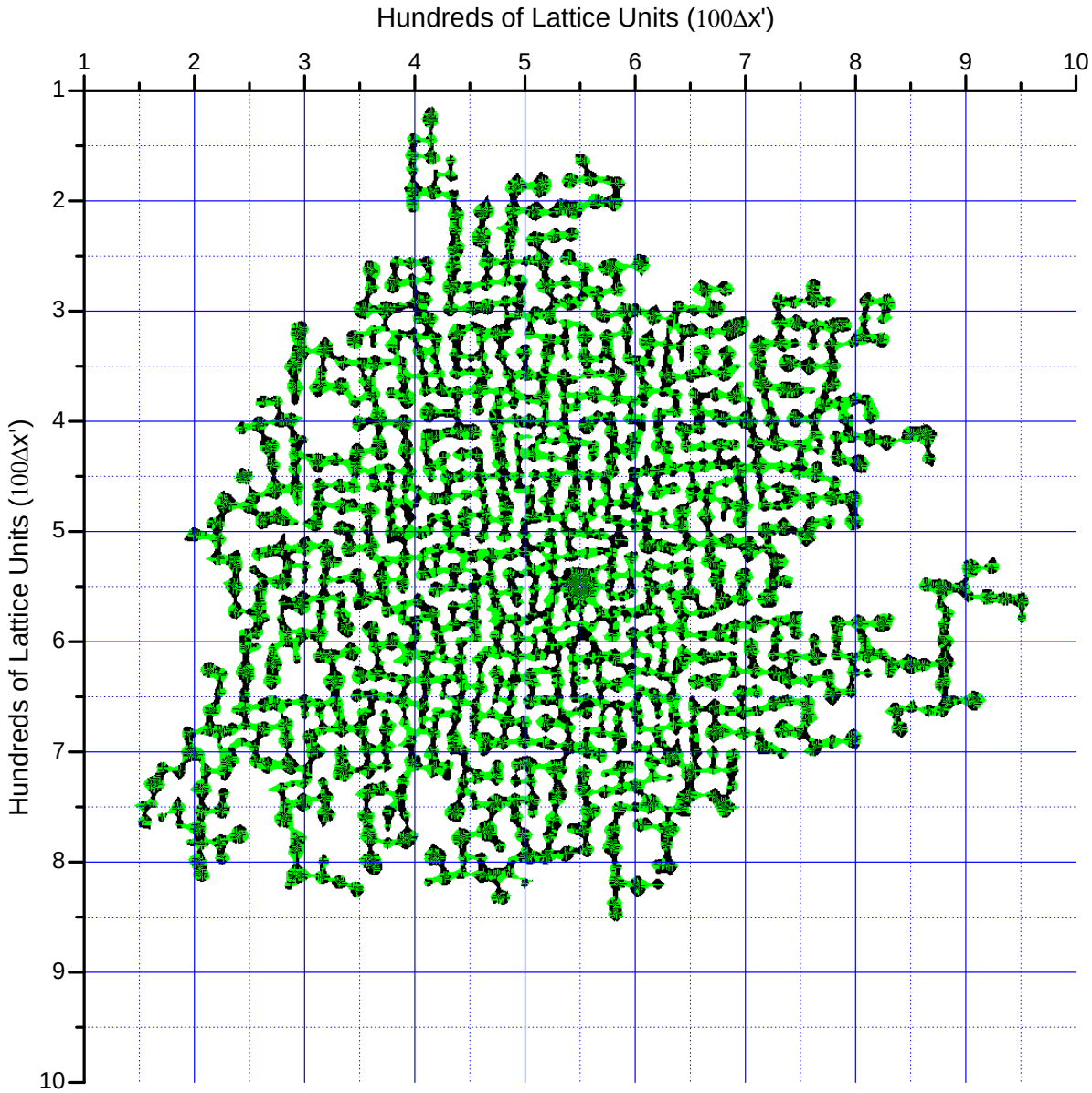


Figure 6.23 – Plotted image of the synthetic stellate solid. $\beta_0 = 0.25$, total simulation time $=200000\Delta t$, all other parameters taken from the realistic stellate parameters.

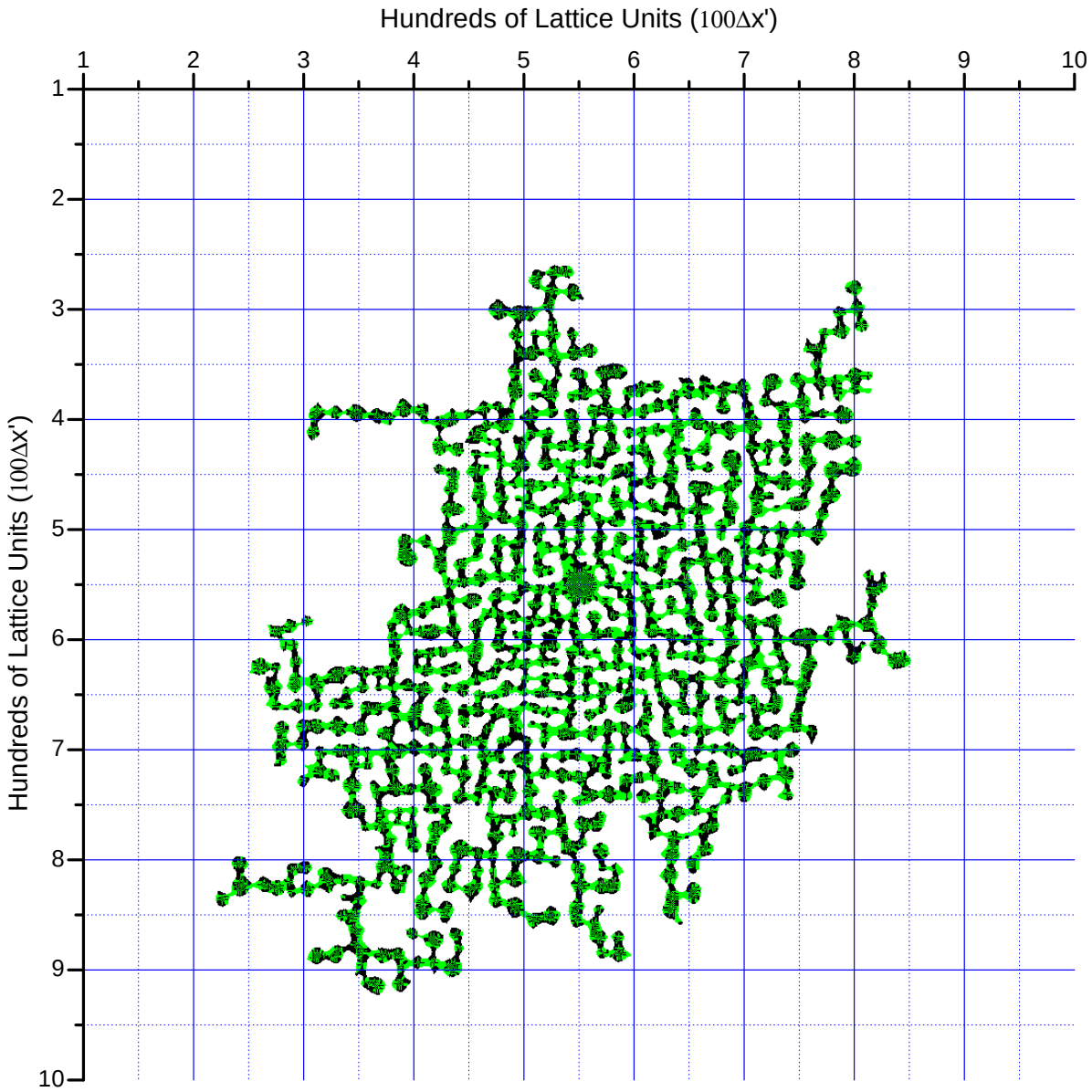


Figure 6.24 – Plotted image of the synthetic stellate solid. $\beta_0 = 0.30$, total simulation time $=200000\Delta t$, all other parameters taken from the realistic stellate parameters.

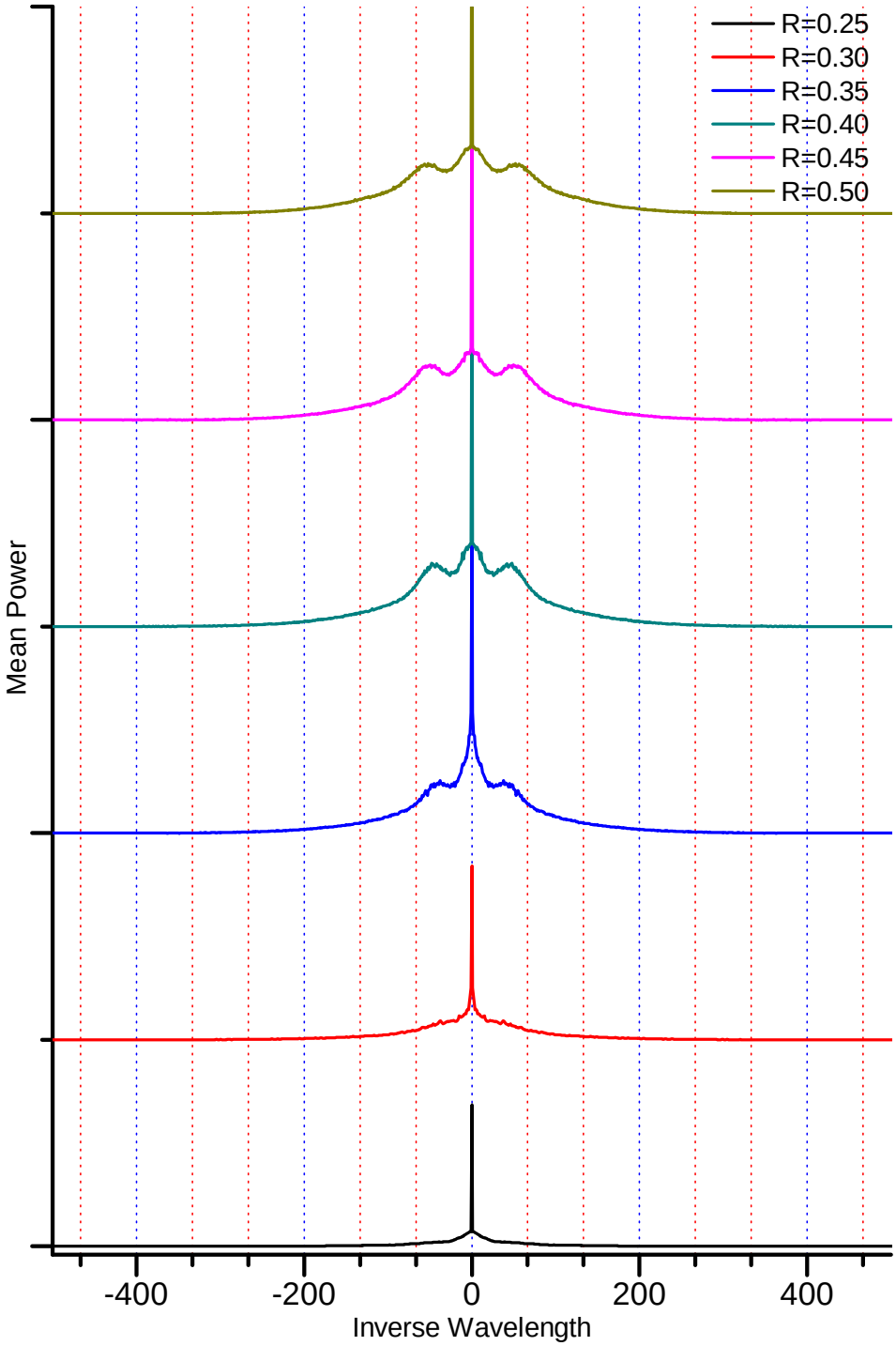


Figure 6.25 – 2D-FFT transforms averaged along principal dimensions down to 1D of the results from sweeping values of R, total simulation time $=200000\Delta t$ per R, all other parameters taken from the realistic stellate parameters.

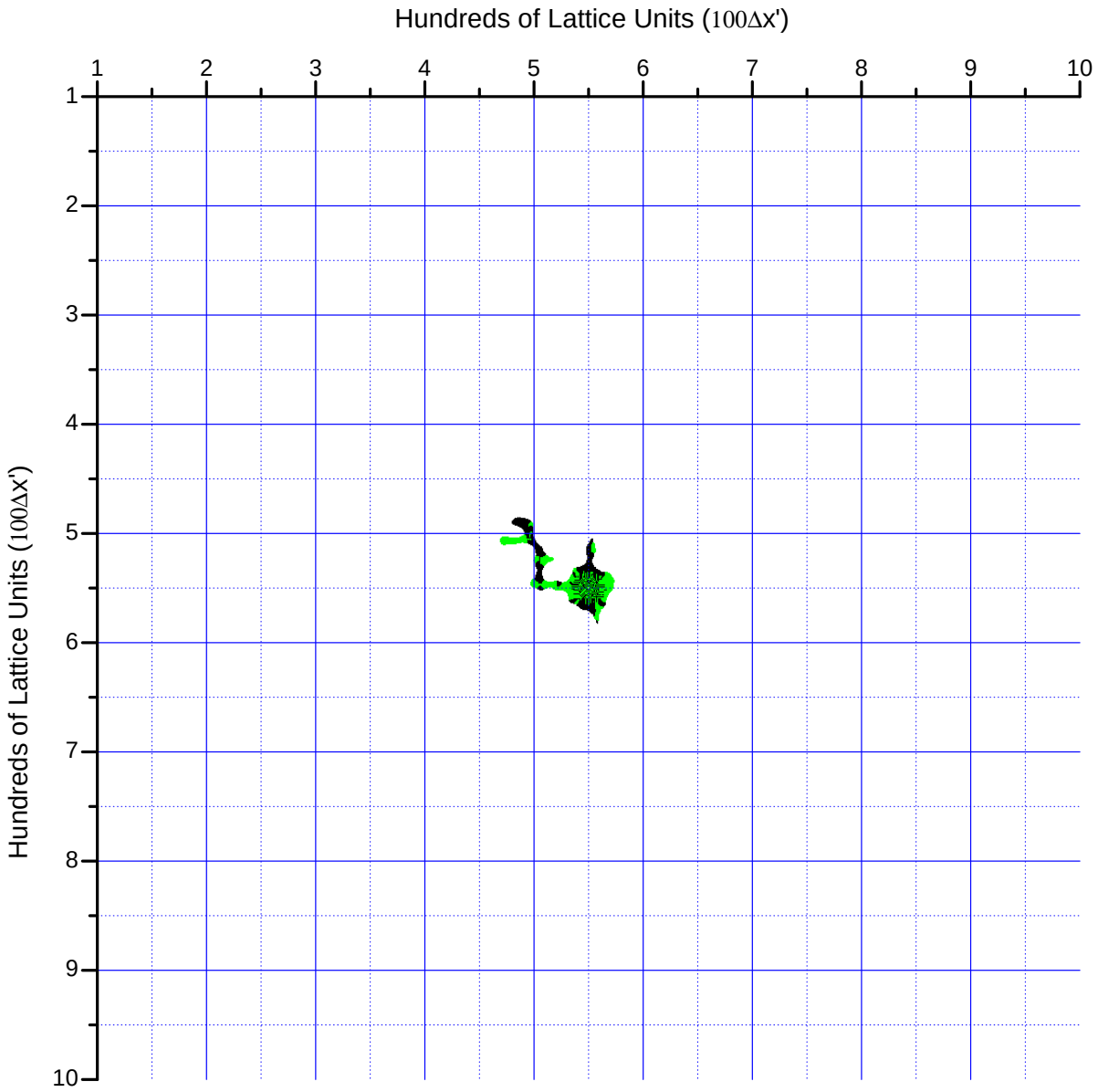


Figure 6.26 – Plotted image of the synthetic stellate solid. $R = 0.25$, total simulation time $=200000\Delta t$, all other parameters taken from the realistic stellate parameters.

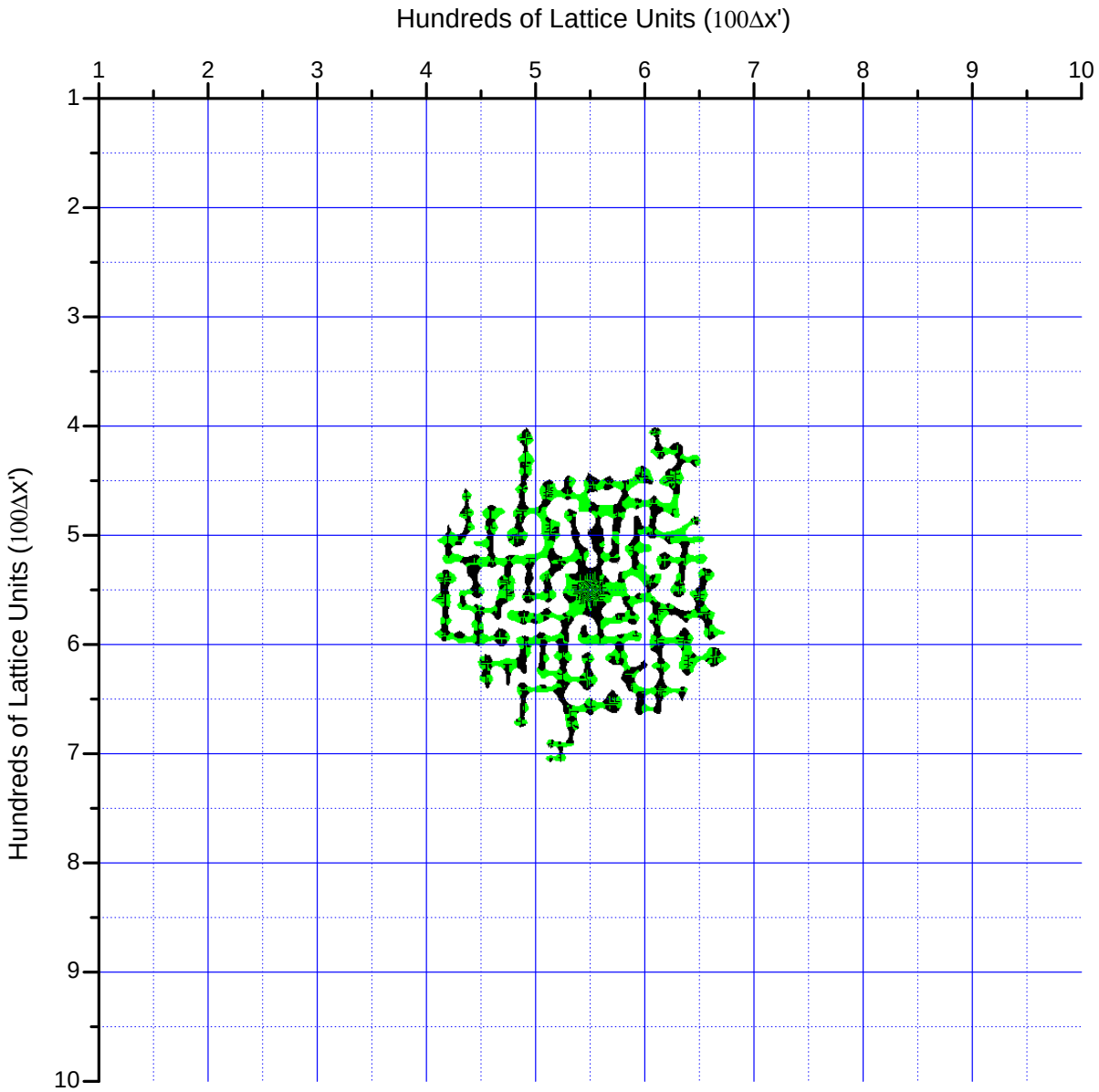


Figure 6.27 – Plotted image of the synthetic stellate solid. $R = 0.30$, total simulation time $=200000\Delta t$, all other parameters taken from the realistic stellate parameters.

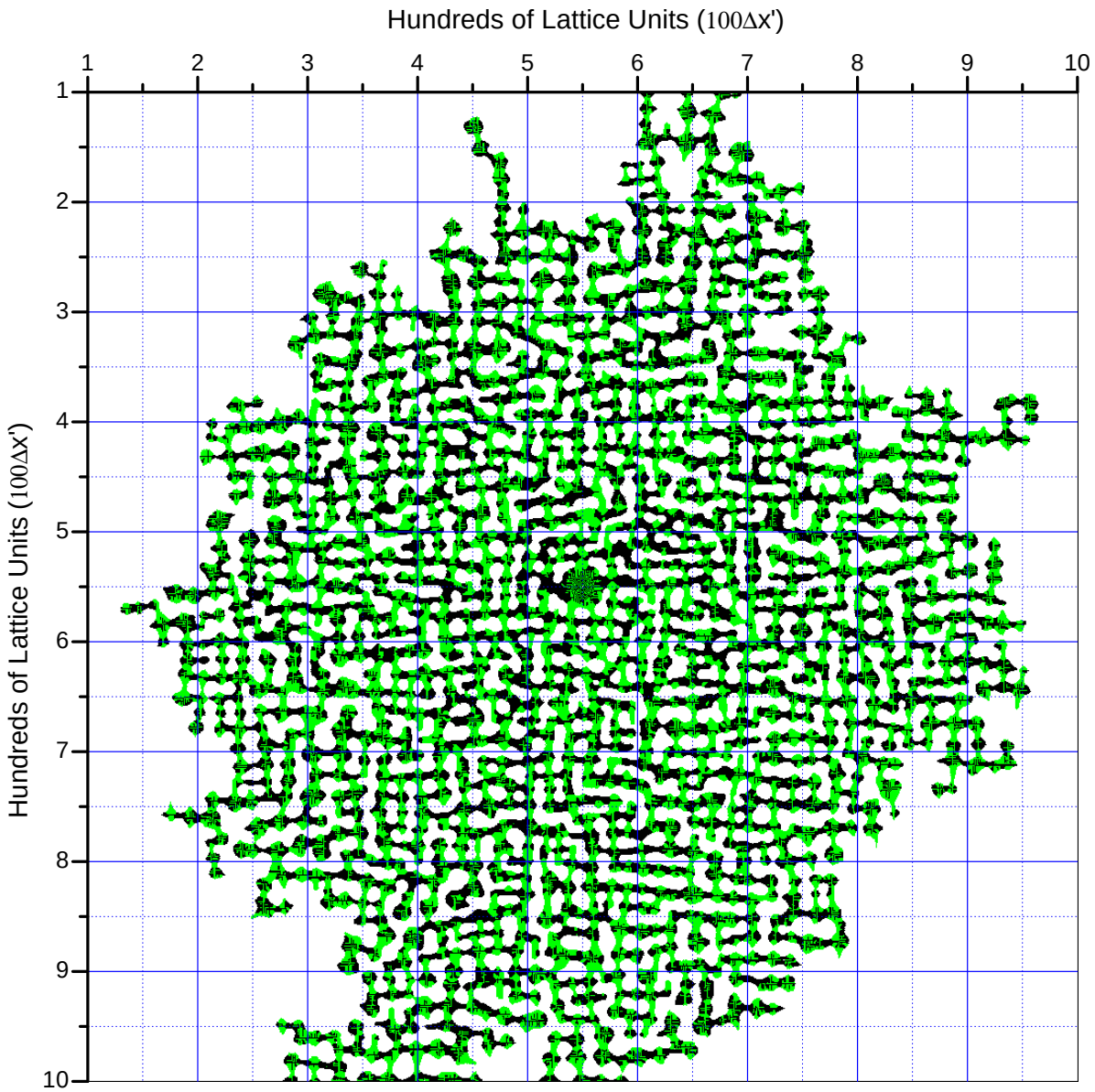


Figure 6.28 – Plotted image of the synthetic stellate solid. $R = 0.35$, total simulation time $=200000\Delta t$, all other parameters taken from the realistic stellate parameters.

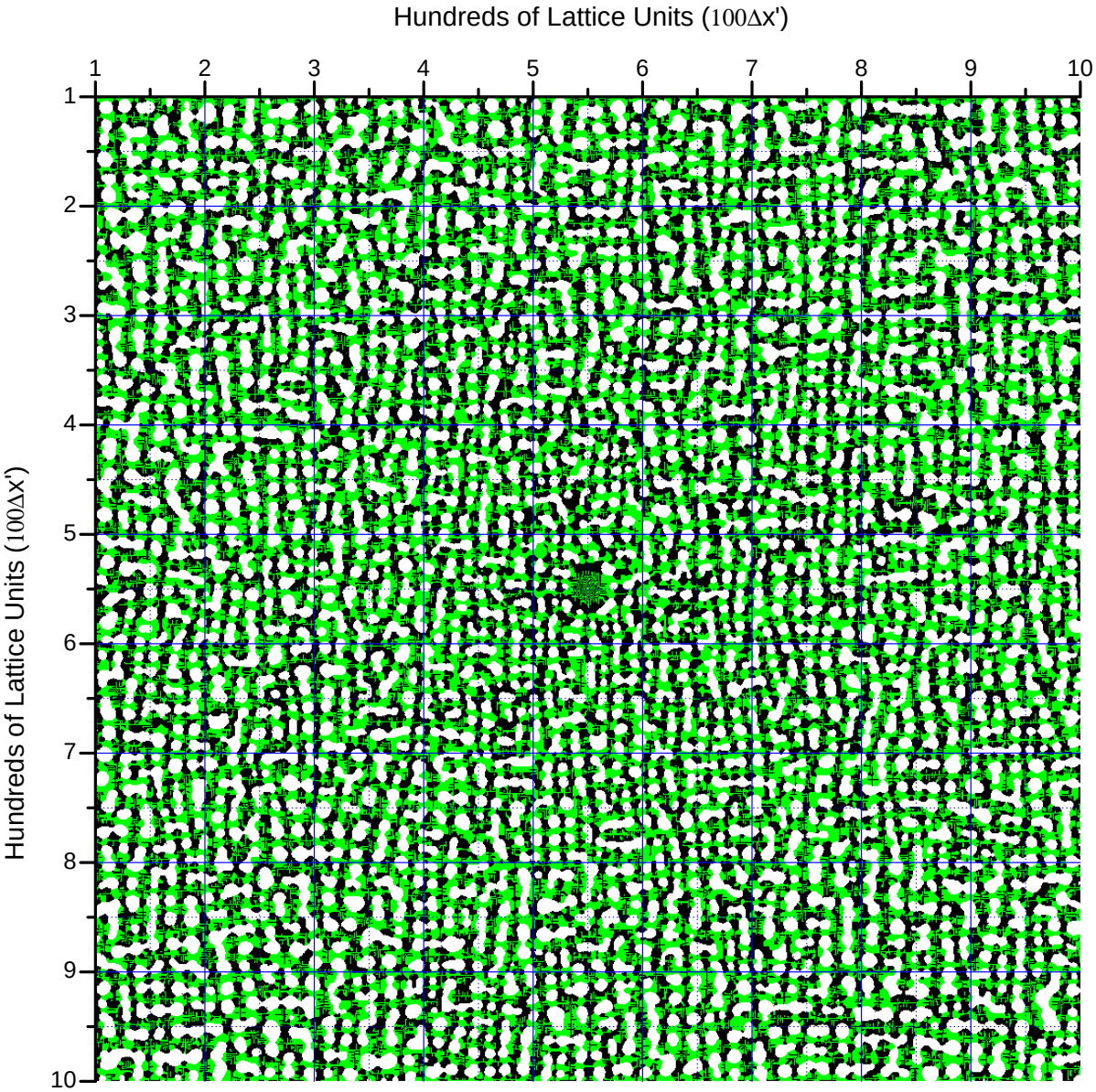


Figure 6.29 – Plotted image of the synthetic stellate solid. $R = 0.40$, total simulation time $=200000\Delta t$, all other parameters taken from the realistic stellate parameters.

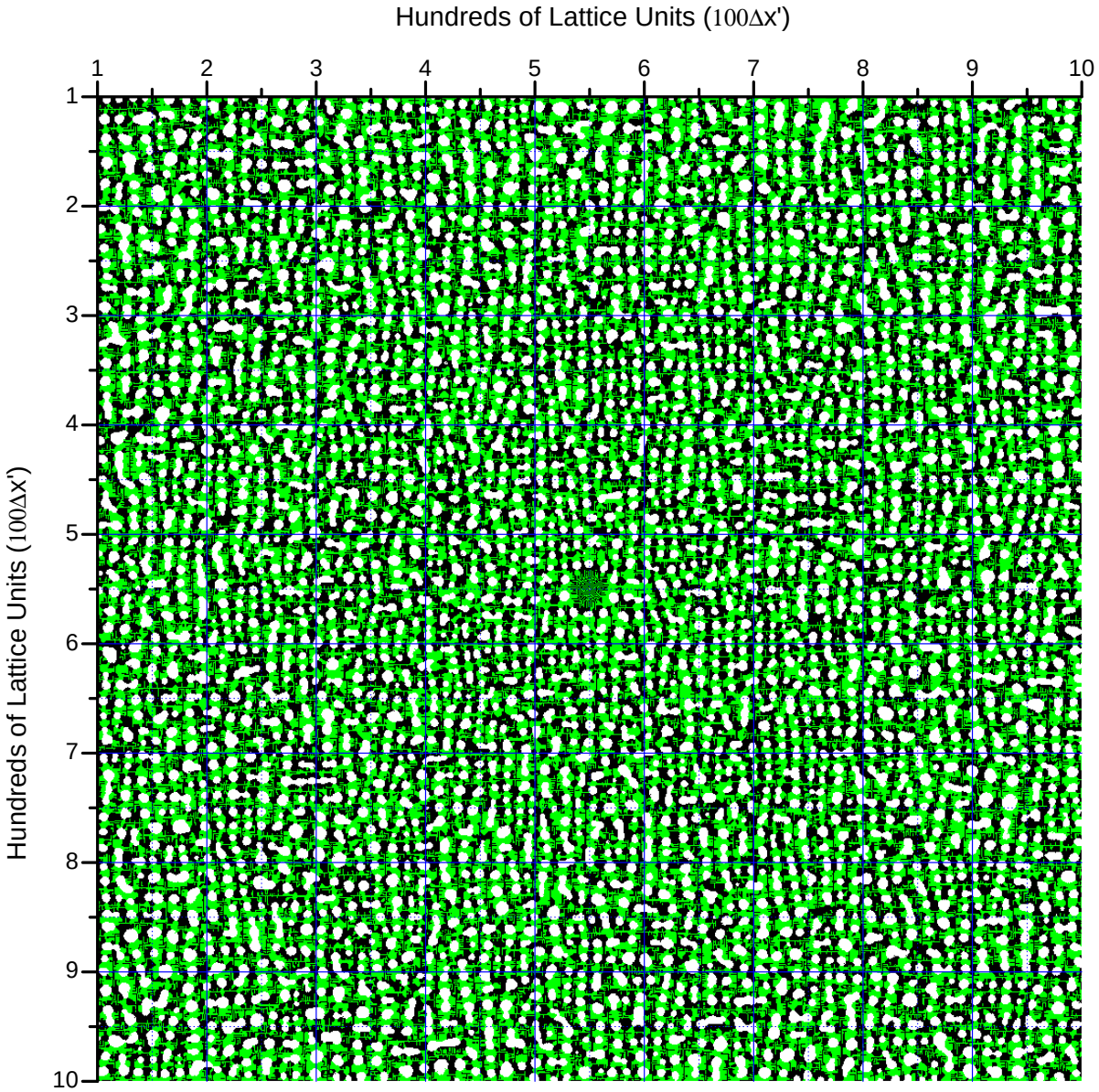


Figure 6.30 – Plotted image of the synthetic stellate solid. $R = 0.45$, total simulation time $=200000\Delta t$, all other parameters taken from the realistic stellate parameters.

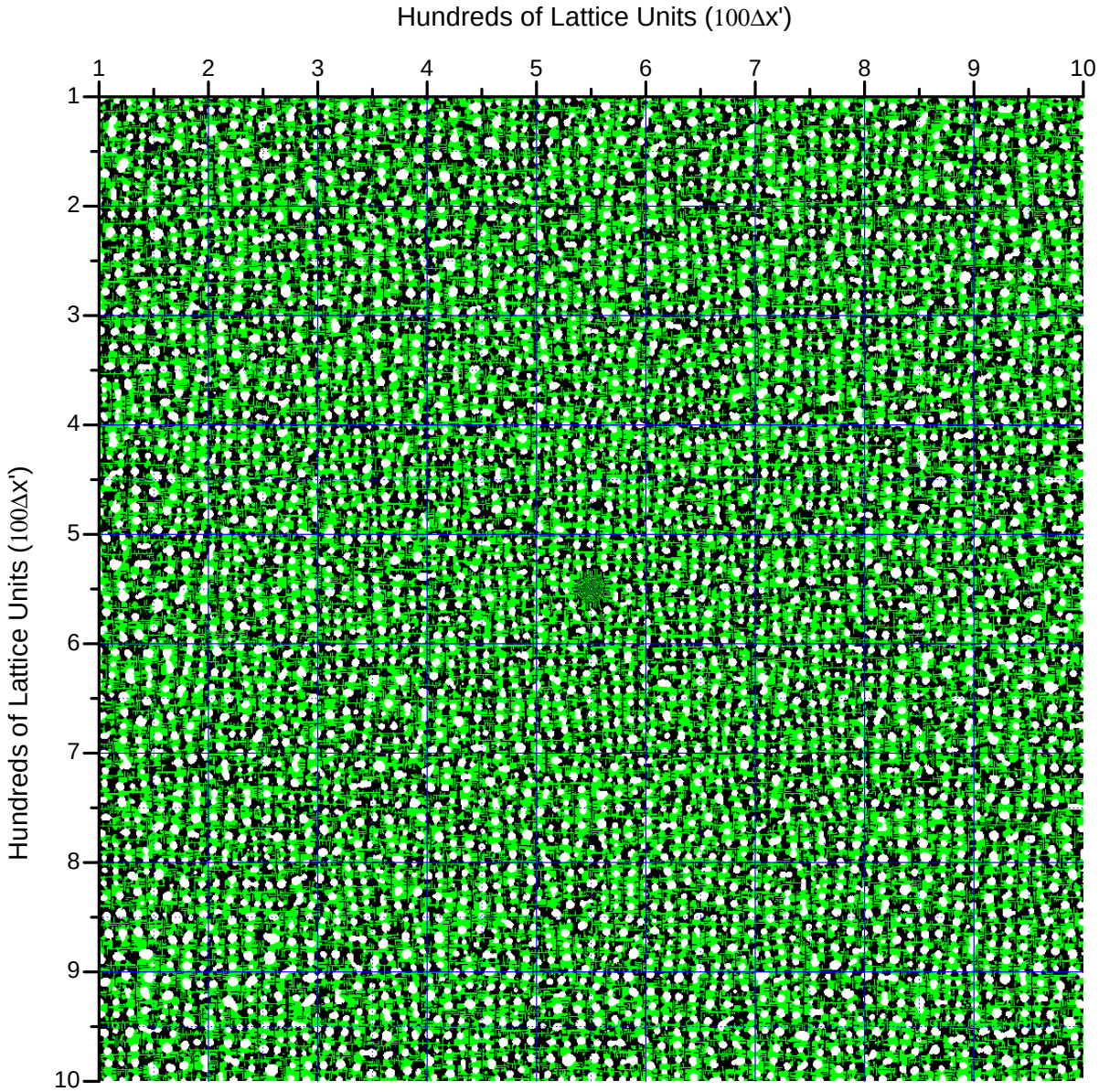


Figure 6.31 – Plotted image of the synthetic stellate solid. $R = 0.50$, total simulation time $=200000\Delta t$, all other parameters taken from the realistic stellate parameters.

Chapter 7

Conclusion

As was first pointed out by Donaldson et al. (1973), the existence of the plagioclase stellate patterns was a perplexing sight to behold. Following the footsteps of Fowler (e.g., Fowler (2000), Fowler et al. (2005)) and Labonté (2004), this thesis proposes a hypothesis for the formation of the stellate patterns based on the important geological, macroscopic and microscopic observations. The proposed mechanism of the genesis of intra-layer heterogeneities is a combination of unmixing/nucleation, growth and subsequent fractionation. The proposed mechanism responsible for the inter-layer enrichments (heterogeneities) is due to hydrodynamic interactions (such as convection). This hypothesis then leads us to explore the modern theories on various aspects of phase transitions. Having examined various constraints and balancing both observations, characteristics of the plagioclase stellate patterns and computational requirements, a numerical model is proposed to test various aspects of the hypothesis, along with the necessary theoretical developments. This model is then dissected, discussed at length and various performance and implementation details are presented. The results are then explored. First, by varying the parameters independently to observe the range of behaviours offered by our numerical model. Second, we proceed in a similar fashion but with some combined parameter variations. Third, the parameters without known ranges are explored. Lastly, the known parameters are transcribed into model equivalents combined with the most optimal values of parameters without known ranges to yield results similar to macroscopic and microscopic observations of real plagioclase stellates.

As seen in the results chapters, the most crucial aspects involved in the formation of the plagioclase stellates are: the proportion of chemicals in the melt that will eventually crystallize

into plagioclase, the presence of a mm-cm scaled heterogeneity in the melt, undisturbed period (thermally and hydrodynamically) for the crystallization of plagioclase, a low symmetry unit-cell resulting in many non-identical epitaxial configurations and a rate of plagioclase growth favouring frequent epitaxy, but low enough such that there are no microscopically observable evidences of diffusion limited behaviours. The synthetic stellates are analyzed in various ways and the results are discussed. A visual examination of the synthetic stellates and the natural stellates would suggest that the model produces a simplified version of the natural stellates. Some important aspects are well captured by the model, including the characteristic spatial separation of stellate rays and the long, stretching crystals that link to others via epitaxy. Given that the model uses only two principal directions of anisotropy, it can be expected that the texture will become more radial with added directions of anisotropy. For some parameter ranges, the texture becomes coarser or finer and more or less radial to completely patchy, illustrating that the model is capable of capturing a range of observed stellate behaviours. Of course, the model does not capture the processes that occur in the depth direction, since the model is two dimensional. With the availability of the third dimension, additional inter-plagioclase crystal linkages can be established, and the effect of convective channels, percolating through the melt-mush layer can be explored. It is possible that some of the radial symmetry is due to some form of convective cell, forming inside the stellate-bearing layers prior to and possibly coincident with the formation of the plagioclase stellates.

In the spirit of future researches and researchers on the subject of Rum plagioclase stellates, various aspects of the hypothesis and the numerical model can be refined and tuned pending the arrival of new data. Due to the difficulty of access to the Isle of Rum (particularly from Canada) and the limitations of scientific sampling, important details such as the three dimensional distribution of the plagioclase stellates and the distribution of plagioclase stellates per layer as a function of igneous layers are very poorly understood. This knowledge would allow one to construct a better hypothesis and/or allow one to constrain the contribution of inter-layer to intra-layer heterogeneities. Ongoing experimental and theoretical research into the structure of glass-forming melts will eventually give us a better liquid model of the genesis of intra-layer heterogeneities. When this new liquid model is combined with a thermodynamic model for the crystallization sequence (such as MELTS `http://melts.ofm-research.org/`), a very realistic model can be developed that can simulate both thermodynamic, kinetic and hydrodynamic

aspects to allow both forward and reverse modelling of nearly all igneous textures.

The model developed is a general liquid-liquid, liquid-solid phase transformation model with anisotropic solid unit-cells and epitaxy. It is capable of modelling a large variety of phenomena with underlying fundamental behaviours in fluid dynamics, diffusion (both uphill and downhill) and phase transformations. Some potential specific applications include multi-phase fluid flows in various situations, including porous media, and domain coarsening dynamics in liquid and/or solid systems (e.g., material sciences, metallurgy and polymer sciences).

As applied to the Rum stellates, the limitations of the 2D implementation prevent the quantification of the relative contribution of inter-layer to intra-layer melt heterogeneities. With the development of more memory-efficient fluid dynamic models and/or advances in performance computing, it is possible to eventually explore this topic on a large scale ($\sim$m). The lack of a complete theory of phase transformation as applied to geological systems also prevents us from accurately gauging the relative contribution of unmixing to nucleation in the formation of the melt heterogeneities. A lack of extensive sampling of Rum prevents a thorough comparison of the natural stellate patterns to the synthetic ones. With better access and a strategic sampling strategy, it is possible to eventually make such a comparison.

References

Bhatnagar, P., P., G. E., and Krook, M. K. (1954). A model for collision processes in gases. i. small amplitude processes in charged and neutral one-component systems. *Physical Review*, 94(3):511–525.

Binder, K. (1998). Spinodal decomposition in confined geometry. *Journal Non-Equilibrium Thermodynamics*, 23:1–44.

Binder, K. and Fratzl, P. (2013). *Materials Science and Technology, Reprint.* Wiley-VCH.

Binns, P. E., McQuillin, R., and Kenolty, N. (1974). The geology of the Sea of the Hebrides. Technical Report 73/14, Institute of Geological Science Report.

Bray, A. J. (2003). Coarsening dynamics of phase-separating systems. *Philosophical Transactions of the Royal Society of London, A*, 361:781–792.

Butcher, A. R., Young, L. M., and Faithfull, J. W. (1985). Finger structures in the Rhum Complex. *Geological Magazine*, 122:491–502.

Cahn, J. W. (1969). On spinodal decomposition. *Acta Metallurgica*, 9:795–801.

De Yoreo, J. (2012). Non-classical pathways of mineralization: pre-nucleation clusters and oriented attachment. In *22nd Goldschmidt Geochemistry Conference*, Montréal, Canada.

Demichelis, R. (2012). Early stages of amorphous calcium phosphate nucleation: a computational study. In *22nd Goldschmidt Geochemistry Conference*, Montréal, Canada.

Dietrich, S. (1988). *Phase Transformations and Critical Phenomena, Volume 12: Domb, C., Lebowitz, S. L. (Eds.).* New York: Academic Press.

Donaldson, C. H., Drever, H. I., and Johnston, R. (1973). Crystallization of poikilo-macrospherulitic feldspar in a Rhum peridotite. *Nature Physical Science*, 243:69–70.

Emeleus, C. H. (1994). Solid geology map of Rum (2nd edition). Edinburgh: Scottish Natural Heritage.

Emeleus, C. H. (1996). Geology of Rum and the adjoining islands. Memoirs of the British Geological Survey.

Emeleus, C. H., Cheadle, M. J., Hunter, R. H., Upton, B. G. J., and Wadworth, W. J. (1996). *The Rum Layered Suite*. Elsevier Science.

Emeleus, C. H., Wadsworth, W. J., and Smith, N. J. (1985). The early igneous and tectonic history of the Rhum Tertiary Volcanic Centre. *Geological Magazine*, 122:451–457.

Forster, R. W. (1980). *A geochemical and petrological study of the Tertiary minor intrusions of Rhum, northwest Scotland.* PhD thesis, University of Durham.

Fowler, A. D. (2000). Rum field notes, observations and subsequent analytical results. Private communications, 2000-2012.

Fowler, A. D., Labonté, F., Zhang, S., Cheadle, M. J., and Emeleus, C. H. (2005). Branching plagioclase and laminated olivines within gabbros of the central layered series, isle of rum, scotland - evidence of open system dynamics and far from equilibrium mineral growth, unpublished manuscript.

Gale, J. (2012). Simulation of non-classical crystallization of carbonate minerals. In *22nd Goldschmidt Geochemistry Conference*, Montréal, Canada.

Getson, J. (2006). Effect of plagioclase crystallization on liquid and magma viscosity in the anorthite-diopside-forsterite-quartz system. Master's thesis, University of Missouri-Columbia.

Grunau, D., Chen, S., and Eggert, K. (1993). A lattice Boltzmann model for multiphase fluid flows. *Physics of Fluids, A*, 5:2557–2562.

Gunstensen, A. and Rothman, D. (1991). Lattice Boltzmann model of immiscible fluids. *Physical Review, A*, 43:4320–4327.

Hammel, J. J. (1967). Direct measurements of homogeneous nucleation rates in glass-forming systems. *Journal of Chemical Physics*, 46:2234–2244.

Heermann, D. W. and Klein, W. (1983). Percolation and droplets in a medium-range three-dimensional ising model. *Physical Review B*, 27(3):1732–1735.

Holness, M. B., Stripp, G., Humphreys, M. C. S., Veksler, I. V., Nielsen, T. F. D., and Tegner, C. (2011). Silicate liquid immiscibility within the crystal mush: late stage magmatic microstructures in the Skaergaard intrusion, east Greenland. *Journal of Petrology*, 52(1):175–222.

Holness, M. B., Tegner, C., Nielsen, T. F. N., Stripp, G. R., and Morse, S. A. (2007). A textural record of solidification and cooling in the Skaergaard intrusion, east Greenland. *Journal of Petrology*, 48:2359–2377.

Jones, R. A. L., Norton, L. J., Kramer, E. J., Bates, F. S., and Wiltzius, P. (1991). Surface-directed spinodal decomposition. *Physical Review Letters*, 66(10):1326–1329.

Klein, W. and Unger, C. (1983). Pseudospinodals, spinodals, and nucleation. *Physical Review B*, 28:445–448.

Kostorz, G., editor (2001). *Phase Transformations in Materials*. Wiley-VCH.

Krausch, G. (1995). Surface induced self assembly in thin polymer films. *Materials Science and Engineering: R: Reports*, 14(1):v–vi, 1–94.

Labonté, F. (2004). Rayons radiés de plagioclase et olivine aligné à l'intérieur de péridotite feldspathique, Série Centrale, île de Rum, Écosse: Évidence d'écoulement magmatique. Master's thesis, University of Ottawa.

Lasaga, A. C. (1998). *Kinetic theory in the Earth sciences*. Princeton Series in Geochemistry.

Latva-Kokko, M. and Rothman, D. (2005). Diffusion properties of gradient-based lattice Boltzmann models of immiscible fluids. *Physical Review, E*, 71(44):056702.

McClurg, J. E. (1982). *Petrology and evolution of the northern part of the Rhum ultrabasic complex*. PhD thesis, University of Edinburgh.

Mitra, M. K. and Muthukumar, M. (2010). Theory of spinodal decomposition assisted crystallization in binary mixtures. *The Journal of Chemical Physics*, 132:1–6.

Morse, P. M. (1969). *Thermal Physics, Second Edition.* The Benjamin/Cummings Publishing Company, Inc.

Morse, S. A. (1980). *Basalts and Phase Diagrams, An introduction to the Quantitative Use of Phase Diagrams in Igneous Petrology.* Springer-Verlag.

Morse, S. A., Owen, B. E., and Butcher, A. R. (1987). Origin of finger structures in the Rhum Complex: phase equilibrium and heat effects. *Geological Magazine*, 124:205–210.

Mussett, A. E., Dagley, P., and Skelhorn, R. R. (1996). *Time and duration of activity in the British Tertiary Igneous Province. In: Morton, A. C. and Parson, L. M. (eds.).* Special Publications of Geological Society of London.

Parsons, R. A., Nimmo, F., Hustoft, J. W., Holtzman, B. K., and Kohlstedt, D. L. (2008). An experimental and numerical study of surface tension-driven melt flow. *Earth and Planetary Science Letters*, 267:548–557.

Philpotts, A. R., Brustman, C. M., Shi, J., Carlson, W. D., and Denison, C. (1999). Plagioclase-chain networks in slowly cooled basaltic magma. *American Mineralogist*, 84:1819–1829.

Puri, S. and Frisch, H. L. (1997). Surface-directed spinodal decomposition: modelling and numerical simulations. *Journal of Physics: Condensed Matter*, 9(10):2109.

Reis, T. and Phillips, T. N. (2007). Lattice Boltzmann model for simulating immiscible two-phase flows. *Journal of Physics A: Mathematical and Theoretical*, 40:4033–4053.

Renner, R. and Palacz, Z. A. (1987). Basaltic replenishment of the Rhum magma chamber: evidence from unit 14. *Journal of Geological Society*, 144:961–970.

Schmelzer, J. W. P., Fokin, V. M., S., A. A., Zanotto, E. D., and Gutzow, I. (2010). How do crystals form and grow in glass-forming liquids: Ostwald's rule of stages and beyond. *International Journal of Applied glass Science*, 1(1):16–26.

Smith, J. V. and Brown, W. L. (1988). *Feldspar Minerals Volume 1 (2nd revised and extended edition).* Springer-Verlag.

Smith, N. J. (1985). The age and structural setting of limestones and basalts on the Main Ring Fault in southeast Rhum. *Geological Magazine*, 122:439–445.

Sparks, R. S. J. and Huppert, H. E. (1984). Density changes during the fractional crystallization of basaltic magmas: fluid dynamic implications. *Contributions to Mineralogy and Petrology*, 85:300–309.

Sterling, J. D. and Chen, S. (1996). Stability analysis of lattice Boltzmann methods. *Journal of Computational Physics*, 123:196–206.

Thompson, R. N. and Gibson, S. A. (1991). Subcontinental mantle plumes, hotspots and pre-existing thinspots. *Journal of Geological Society, London*, 148:973–978.

Thürey, N. (2003). *A single-phase free-surface Lattice Boltzmann Method*. PhD thesis, Friedrich-Alexander-Universität Erlangen-Nürnberg, Institut Für Informatik (Mathematische Maschinen und Datenverarbeitung).

Unger, C. and Klein, W. (1984). Nucleation theory near the classical spinodal. *Physical Review B*, 29:2698–2708.

Upton, B. G. J. (1988). *History of Tertiary igneous activity in the North Atlantic borderlands.* Early Tertiary volcanism and the opening of the North Atlantic. Geological Society of London, Special Publications.

Veksler, I. V., Dorfman, A. M., Borisov, A. A., R., W., and Dingwell, D. B. (2007). Liquid immiscibility and the evolution of basaltic magma. *Journal of Petrology*, 48(11):2187–2210.

Veksler, I. V., Káhn, J., Franz, G., and Dingwell, D. B. (2010). Interfacial tension between immiscible liquids in the system K_2O-FeO-Fe_2O_3-Al_2O_3-SiO_2 and implications for the kinetics of silicate melt unmixing. *American Mineralogist*, 95:1679–1685.

Volker, J. A. (1983). *The geology of the Trallval area, Rhum. Inner Hebrides.* PhD thesis, University of Edinburgh.

Volker, J. A. and Upton, B. G. J. (1990). The structure and petrogenesis of the Trallval and Ruinsival areas of the Rhum ultrabasic complex. *Transactions of the Royal Society of Edinburgh: Earth and Environmental Science*, 81:69–88.

Welander, P. (1954). On the temperature jump in a rarefied gas. *Arkiv für Fysik*, 7(44):507–553.

Wolf-Gladrow, D. A. (2005a). *Lattice-Gas cellular automata and lattice Boltzmann models - an introduction*. Springer.

Wolf-Gladrow, D. A. (2005b). *Lattice-Gas cellular automata and lattice Boltzmann models - an introduction*. Springer.

Wolf-Gladrow, D. A. (2005c). *Lattice-Gas cellular automata and lattice Boltzmann models - an introduction*. Springer.

Xu, H. F., Buseck, P. R., and Carpenter, M. A. (1997). Twinning in synthetic anorthite: A transmission electron microscopy investivation. *American Mineralogist*, 82:125–130.

Appendix A

Diffusion and Model Verification

The transport of a given substance driven by a difference in concentration is called diffusion. Diffusion is a constituent of transport phenomenon along with convection, advection and radiation as some of the most common examples. Since the individual particle behaviour is non-deterministic and the particle number is exceedingly large, usually a statistical treatment is necessary to convey governing dynamics of the evolution in both time and space of the overall ensemble. This appendix will derive the governing diffusion equation in two ways, and solve the equation with boundary conditions and finally compare the analytical solution to the numerical solution.

A.1 Classical Derivation

As with other types of transport phenomena, the continuum formulation can be sought as a mathematical connection between fluxes and forces. In three dimensions, an elementary volume in cartesian coordinates spanned by (dx, dy, dz) is embedded in a given concentration of matter. The amount of change in the total matter inside this elementary volume is given by the difference of the flux entering and leaving this unit volume in all three dimensions.

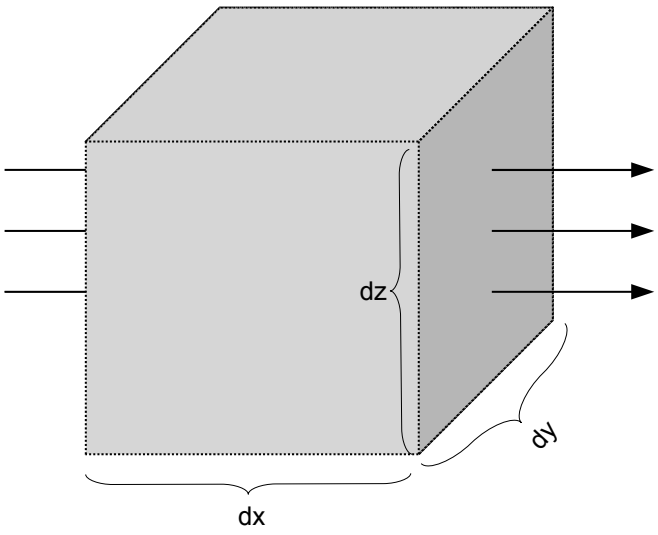


Figure A.1 – Illustration of elementary volume and flux vectors

Therefore the total change $dM = -\nabla \cdot \vec{J} dt dx dy dz$ where $\vec{J}$ is the vector flux through an elementary volume, dM is the net change in matter in the unit volume and dt is an infinitesimal length of time. If $C(\vec{P}, t)$ describe the concentration at a point $\vec{P}$ and time t, then the change in C, or dC is related to dM by $dC = \dfrac{dM}{dxdydz}$. Therefore, combining the above equations we arrive at

$$\frac{\partial C}{\partial t} = -\nabla \cdot \vec{J}, \tag{A.1}$$

which is the well known continuity equation. The flux $\vec{J}$ should be proportional to the concentration gradient, and thus

$$\vec{J} = -\mathbf{D}\nabla C, \tag{A.2}$$

where $\mathbf{D}$ is the 3×3 diffusion coefficient tensor. By combining equations A.1 and A.2, we arrive at the diffusion equation

$$\frac{\partial C}{\partial t} = \nabla \cdot \mathbf{D}\nabla C. \tag{A.3}$$

In general, when the diffusion coefficient is spatially variant, $\mathbf{D}$ is not commutable with ∇ and thus must be kept inside the dot product. When $\mathbf{D}$ is spatially homogeneous, the equation further simplifies to

$$\frac{\partial C}{\partial t} = \mathbf{D}\nabla \cdot \nabla C. \tag{A.4}$$

In the special case where $\mathbf{D}$ is also isotropic, $\mathbf{D}$ becomes a diagonal matrix with all diagonal elements equal, i.e., $D_{xx} = D_{yy} = D_{zz}$.

A.2 Microscopic Perspective

A microscopic derivation of the diffusion equation based on a more atomistic consideration gives better insight into the probabilistic nature at the atomic scale and the diffuse behaviour which arises from it. In his 1905 paper on Brownian motion, Einstein treated the diffusion phenomenon with a general and microscopic perspective. Following his reasoning closely, this approach considers the introduction of the probability that a particle, initially at a position x at time t will jump to $x + dx_o$ at time $t + t_o$ due to the impact by some neighbouring particles whose energy is distributed according to e^{-E/k_BT}, where E is the kinetic energy of a neighbouring particle, k_B is the Boltzmann constant and T is the absolute temperature of the system. This probability defines a probability distribution $\Phi(x, x_o, t_o)$ where x_o is some distance, t_o is the time interval. It is obvious that in any time interval t_o, the particle always moves from $x \to x + x_o$, even if $x_o = 0$.

$$\int_M \Phi(x, x_o, t_o) dx_o = 1, \tag{A.5}$$

where M denotes the collection of all possible jumps. Therefore, if the concentration at a given position is described by $C(x, t)$, then, at a later time $t + t_o$, the probability density is increased by the contribution of probability from points $x - x_o$ for all x_o. Adding all the contributions from all possible jumps nearby yields the Einstein Master Equation

$$C(x, t + t_o) = \int_M C(x - x_o, t)\Phi(x - x_o, x_o, t_o) dx_o. \tag{A.6}$$

The probability density originally at x has a probability $[1 - \Phi(x, 0, t_o)dx_o]$ to move away from x and thus always moves in that small time step t_o and has zero contribution to the total density at x.

Under the sensible assumption that it is possible to choose t_o such that the distance jumped x_o during that time span is sufficient small, the concentration term can be expanded via Taylor series as

$$C(x, t + t_o) = C(x, t) + t_o \frac{\partial C}{\partial t} + O(t_o^2) + ..., \tag{A.7}$$

$$C(x - x_o, t) = C(x, t) - x_o \nabla C + \frac{{x_o}^2}{2} \nabla \cdot (\nabla C) + O(x_o^3) + ... \tag{A.8}$$

$O(t_o^n)$ and $O(x_o^n)$ in equations A.7, A.8 denotes n-th order expansion terms. Inserting equations A.7 and A.8 into equation A.6 gives,

$$\begin{aligned} C(x,t) + t_o \frac{\partial C(x,t)}{\partial t} + ... = \quad & C(x,t) \quad \int_M \Phi(x - x_o, x_o, t_o) dx_o \\ -\nabla \quad & C(x,t) \quad \int_M x_o \Phi(x - x_o, x_o, t_o) dx_o \\ +\frac{1}{2} \nabla \cdot \nabla \quad & C(x,t) \quad \int_M {x_o}^2 \Phi(x - x_o, x_o, t_o) dx_o + ... \end{aligned} \tag{A.9}$$

Since by the definition of probability A.5, the jumping particles must be somewhere after time t_o,

$$C(x,t) \int_M \Phi(x, x_o, t_o) dx_o = C(x,t). \tag{A.10}$$

Taking the limit as $t_o \to 0$ ensures $O(t_o^2) = 0$.

$$\lim_{t_o \to 0} \frac{1}{t_o} \int_M x_o \Phi(x - x_o, x_o, t_o) dx_o = \frac{\langle x_o \rangle}{t_o} = \langle v(x_o) \rangle, \tag{A.11}$$

$$\lim_{t_o \to 0} \frac{1}{2t_o} \int_M {x_o}^2 \Phi(x - x_o, x_o, t_o) dx_o = \frac{\langle {x_o}^2 \rangle}{2t_o} = \mathbf{D}, \tag{A.12}$$

where $\langle\rangle$ has been used to denote a statistical average. $\langle v(x_o) \rangle$ and $\mathbf{D}$ denotes the average velocity and the diffusion coefficient respectively. Combining A.10, A.11, A.12 into A.10 allows it to be rewritten as

$$\frac{\partial C(x,t)}{\partial t} = -\langle v(x_o) \rangle \nabla C(x,t) + \nabla \cdot \mathbf{D} \nabla C(x,t). \tag{A.13}$$

Notice that equation A.13 is a more general form (advection-diffusion) of the diffusion only equation A.4. If $\langle v(x_0) \rangle = 0$, then the behaviour of $C(x,t)$ is diffusive only.

A.3 Solutions to the Diffusion Equation

Solutions to this differential equation requires an accompanying set of initial/boundary conditions. If the boundary conditions at $x = l$ specify some fixed concentration value, such as $C(x,t) = 0, (x = l, t > 0)$, then the boundary condition is known as a Dirichlet boundary condition, named after the mathematician Johann Peter Gustav Lejeune Dirichlet. If instead, the boundary conditions are given by specifying the flux through them, such as $\frac{\partial C}{\partial x}, (x = l, t > 0)$, then the boundary conditions are known as Neumann boundary conditions, named after John von Neumann. In general, multiple approaches are possible to obtain a particular analytical solution to a diffusion problem. The choice usually depends on boundary conditions and initial conditions.

Consider a two dimensional system in cartesian coordinates with domain $x \in [-L/2, L/2], y \in [-L/2, L/2]$ and spatially homogeneous D,

$$\frac{\partial C(x,y,t)}{\partial t} = D\left\{\frac{\partial^2 C(x,y,t)}{\partial x^2} + \frac{\partial^2 C(x,y,t)}{\partial y^2}\right\}, \tag{A.14}$$

$$\frac{\partial C(x,y,t)}{\partial n} = 0, \forall t, \text{at the boundaries}, \tag{A.15}$$

$$C(x,y,0) = \delta(x)\delta(y), \tag{A.16}$$

where A.15 is the Neumann boundary condition specifying the flux through the normal component $\frac{\partial}{\partial n}C(x,y,t)$ of the boundaries is zero, i.e, the boundaries are reflective or the domain is physically closed to outside world. A.16 is the initial condition describing that the concentration starts as a pulse with $\int C(x,y,0) = 1$.

This system is solved using Fourier's method, see subsection A.3.1. The solution is stated

here for now:

$$\begin{aligned} C(x,y,t) =& \frac{1}{L^2} \\ &+ \sum_{n=1}^{\infty} \frac{2}{L^2} e^{-\frac{4D}{L^2}n^2\pi^2 t} cos(\frac{2n\pi}{L}x) \\ &+ \sum_{m=1}^{\infty} \frac{2}{L^2} e^{-\frac{4D}{L^2}m^2\pi^2 t} cos(\frac{2m\pi}{L}y) \\ &+ \sum_{n=1}^{\infty}\sum_{m=1}^{\infty} \frac{4}{L^2} e^{-\frac{4D}{L^2}(n^2+m^2)\pi^2 t} cos(\frac{2n\pi}{L}x) cos(\frac{2m\pi}{L}y). \end{aligned} \tag{A.17}$$

A.3.1 Analytical Solution to the Diffusion Equation

One particular method involves exploiting the linear property of the diffusion equation. Each linear differential equation naturally defines a linear functional (or linear operator, or one-form, etc) $\mathfrak{L}$ such that $\mathfrak{L}$ satisfies the following properties:

$$\mathfrak{L}(\alpha_1 u_1 + \alpha_2 u_2) = \alpha_1 \mathfrak{L}(u_1) + \alpha_2 \mathfrak{L}(u_2), \tag{A.18}$$

where α_1, α_2 are some constants and u_1, u_2 are some solutions to the differential equation.

If $\mathfrak{L}(u_i) = f_i$ where u_i are solutions to the linear differential equation $\mathfrak{L}$ and f_i may be non zero for the non-homogeneous case, then $v \equiv \sum_{i=1}^{n} \alpha_i u_i$ satisfies the differential equation $\mathfrak{L}(v) = \sum_{i=1}^{n} \alpha_i f_i$. This is the so-called superposition principle.

This property allows new solution to be built from a linear combination of simpler solutions and is central to both Green's method and Fourier's method.

To solve the problem using Fourier's method, the solution needs to be written as a separable product of functions,

$$C = T(t)X(x)Y(y). \tag{A.19}$$

Here C replaces $C(x,y,t)$ for convenience, and $T(t)$, $X(x)$, $Y(y)$ are assumed functions of t, x and y only.

Substituting this hypothetical solution into the original equation yields:

$$\frac{\partial}{\partial t}T(t)X(x)Y(y) = D\left\{\frac{\partial^2}{\partial x^2}T(t)X(x)Y(y) + \frac{\partial^2}{\partial y^2}T(t)X(x)Y(y)\right\}. \tag{A.20}$$

Rearranging both sides yields:

$$\frac{1}{T(t)}\frac{\partial T(t)}{\partial t} = D\left\{\frac{1}{X(x)}\frac{\partial^2 X(x)}{\partial x^2} + \frac{1}{Y(y)}\frac{\partial^2 Y(y)}{\partial y^2}\right\}. \tag{A.21}$$

The left hand side involve only the variable t and the left hand side the variables x, y. This is the separation of variables technique in PDE analysis. Since either side is independent of the variables of the other, taking a derivative of the equation with respect to t implies the existence of a constant λ (separation constant) such that

$$\frac{1}{T(t)}\frac{\partial T(t)}{\partial t} = D\left\{\frac{1}{X(x)}\frac{\partial^2 X(x)}{\partial x^2} + \frac{1}{Y(y)}\frac{\partial^2 Y(y)}{\partial y^2}\right\} = -\lambda. \tag{A.22}$$

Therefore,

$$\frac{1}{T(t)}\frac{\partial T(t)}{\partial t} = -\lambda, \tag{A.23}$$

$$\frac{1}{X(x)}\frac{\partial^2 X(x)}{\partial x^2} + \frac{1}{Y(y)}\frac{\partial^2 Y(y)}{\partial y^2} = \frac{-\lambda}{D}. \tag{A.24}$$

Now the solutions for A.23 is simply just an exponential function of time,

$$T(t) = Ae^{-\lambda t}. \tag{A.25}$$

Solutions in the spatial dimensions can be further separated since x and y are still yet independent. Thus,

$$\frac{1}{X(x)}\frac{\partial^2 X(x)}{\partial x^2} = -\alpha^2, \tag{A.26}$$

$$\frac{1}{Y(y)}\frac{\partial^2 Y(y)}{\partial y^2} = -\beta^2, \tag{A.27}$$

$$\alpha^2 + \beta^2 = \frac{\lambda}{D}. \tag{A.28}$$

Solutions to differential equations A.26 and A.27 can be given as

$$X(x) = A_1 sin(\alpha x) + A_2 cos(\alpha x), \tag{A.29}$$

$$Y(y) = B_1 sin(\beta y) + B_2 cos(\beta y). \tag{A.30}$$

However, since the solution must be symmetric about $(0,0)$, the asymmetric component in both solutions must be zero for all (x,y), i.e, $A_1 = B_1 = 0$. The final general form of a particular solution to the original diffusion equation A.14 is now:

$$C(x,y,t) = \mathfrak{B} e^{-\lambda t} cos(\alpha x) cos(\beta y). \tag{A.31}$$

Here $\mathfrak{B}$ denotes an unknown constant which will be determined later.

Applying the boundary conditions A.15 to this solution yields $\alpha = \frac{2n\pi}{L}$ and $\beta = \frac{2m\pi}{L}$ and thus $\lambda = \frac{4D}{L^2}\left(n^2 + m^2\right)\pi^2$.

Applying the superposition principle and summing over both n and m integer indices produces

$$C(x,y,t) = \sum_{n=0}^{\infty}\sum_{m=0}^{\infty} \mathfrak{B}_{nm} e^{-\lambda_{nm} t} cos(\alpha_n x) cos(\beta_m y). \tag{A.32}$$

$\mathfrak{B}_{nm}$ are some coefficients typically referred to as Fourier coefficients. They can be evaluated by using the initial condition A.16 and some work.

$$\delta(x)\delta(y) = \sum_{n=0}^{\infty}\sum_{m=0}^{\infty} \mathfrak{B}_{nm} cos(\alpha_n x) cos(\beta_m y), \tag{A.33}$$

multiply both sides by an orthogonal set of cosine functions with integer indices i and j then integrating over both x and y yields

$$\begin{aligned} &\int\int \delta(x)\delta(y) cos(\frac{2i\pi}{L}x) cos(\frac{2j\pi}{L}y) dxdy \\ &= \sum_{n=0}^{\infty}\sum_{m=0}^{\infty} \mathfrak{B}_{nm} \int\int cos(\frac{2n\pi}{L}x) cos(\frac{2m\pi}{L}y) cos(\frac{2i\pi}{L}x) cos(\frac{2j\pi}{L}y) dxdy, \end{aligned} \tag{A.34}$$

where

$$\int\int \delta(x)\delta(y)cos(\frac{2n\pi}{L}x)cos(\frac{2m\pi}{L}y)dxdy = 1, \tag{A.35}$$

regardless of n or m.

Hence,

$$\sum_{n=0}^{\infty}\sum_{m=0}^{\infty}\mathfrak{B}_{nm}\int\int cos(\frac{2n\pi}{L}x)cos(\frac{2m\pi}{L}y)cos(\frac{2i\pi}{L}x)cos(\frac{2j\pi}{L}y)dxdy = 1. \tag{A.36}$$

If $i = 0$ and $j = 0$, the integral becomes trivial and the equation becomes

$$\mathfrak{B}_{oo}L^2 = 1. \tag{A.37}$$

If $i \neq 0$ and $j = 0$ the integral becomes

$$\sum_{n=0}^{\infty}\mathfrak{B}_{no}L\int cos(\frac{2n\pi}{L}x)cos(\frac{2i\pi}{L}x)dx = 1. \tag{A.38}$$

If $i \neq n$, the integral yields only zero by orthogonality of sinusoidal functions. If $i = n$ then the integral yields $L/2$ and so

$$\mathfrak{B}_{io}\frac{L^2}{2} = 1. \tag{A.39}$$

Similarly, for $i = 0$ and $j \neq 0$, the equation evaluates to

$$\mathfrak{B}_{oj}\frac{L^2}{2} = 1. \tag{A.40}$$

If $i \neq 0$ and $j \neq 0$, A.36 evaluates to

$$\mathfrak{B}_{ij}\frac{L^2}{4} = 1. \tag{A.41}$$

So, the Fourier coefficient is summarized as

$$\mathfrak{B}_{\mathfrak{nm}} = \begin{cases} \frac{1}{L^2} & \text{if } n = m = 0; \\ \frac{2}{L^2} & \text{if } n = 0 \text{ or } m = 0; \\ \frac{4}{L^2} & \text{if } n \neq 0, m \neq 0. \end{cases} \tag{A.42}$$

Putting the constants back into A.32 yields the entire formal solution:

$$\begin{aligned} C(x,y,t) =& \frac{1}{L^2} \\ &+ \sum_{n=1}^{\infty} \frac{2}{L^2} e^{-\frac{4D}{L^2} n^2 \pi^2 t} cos(\frac{2n\pi}{L} x) \\ &+ \sum_{m=1}^{\infty} \frac{2}{L^2} e^{-\frac{4D}{L^2} m^2 \pi^2 t} cos(\frac{2m\pi}{L} y) \\ &+ \sum_{n=1}^{\infty} \sum_{m=1}^{\infty} \frac{4}{L^2} e^{-\frac{4D}{L^2}(n^2+m^2)\pi^2 t} cos(\frac{2n\pi}{L} x) cos(\frac{2m\pi}{L} y) \end{aligned} \tag{A.43}$$

A.4 Comparison of Simulation with Analytical Solution

The derived solutions now serve to verify the diffusion aspect of the simulation in our Lattice Boltzmann model. To test the diffusion aspect of the model, the collision operator is truncated to only first order (thus to conserve mass, and neglect conservation of momentum). The diffusion coefficient D is then related to the parameter ω via (e.g. Wolf-Gladrow (2005c))

$$D = \frac{1}{2n}\left(\frac{2}{\omega} - 1\right) \tag{A.44}$$

where n is the number of lattice dimensions in DnQm notation.

Time is the only variable allowed to vary to observe the diffusion behaviour quantitatively and compare the difference between our simulation and the analytical solution. The output from our data is a n-tuple representing the vector $[x, y, C(x, y)]$. The corresponding output from the analytical solution is also the same vector structure, over the same domain size. The initial condition is $C(49, 49) = 1$ for the simulation and for the analytical solution $C(x, y) = \delta(x - 49)\delta(y - 49)$.

Firstly, the data from the experimental output is sampled not at every site, but rather every 1/3rd to reduce the data samples displayed to avoid data-point crowding. To illustrate this, consider the following figure showing the relative pattern of the sampling overlaid on a analytical solution in topographical view.

Now starting with $t = 100\Delta t$, the analytical and numerical solutions overlain on each other are given in figures A.3, A.4, A.5, A.6, A.7 and A.8.

It is worth mentioning that quantitatively, the total difference between the numerical results and the analytical results are on the order of the smallest representable floating point number in Fortran. For double precision numbers in Fortran, this is the last digit of 17 digits. This translates to about 1 part in 10^{16} total numerical error. A picture is worth 1000 words. Therefore, the set of figures (A.3,A.4, A.5, A.6, A.7, A.8) concludes this appendix.

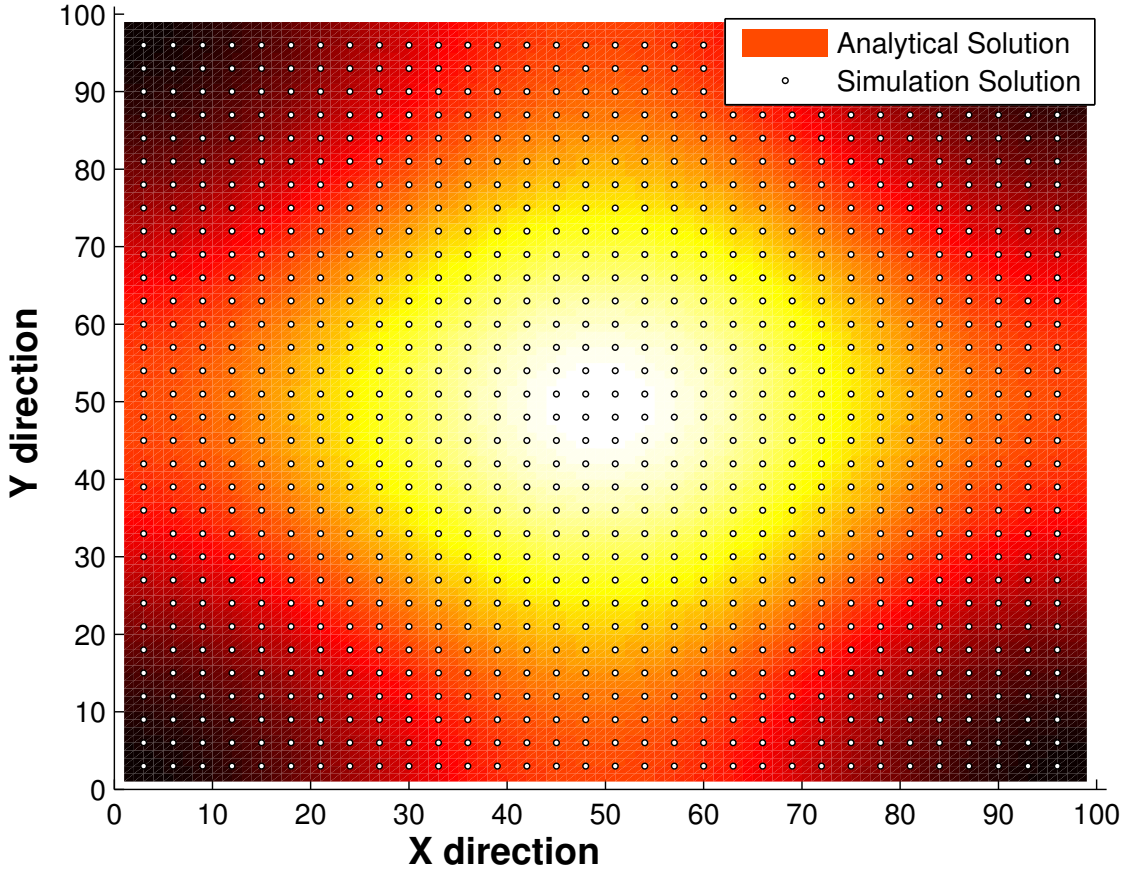


Figure A.2 – Illustration of data sampling pattern over domain.

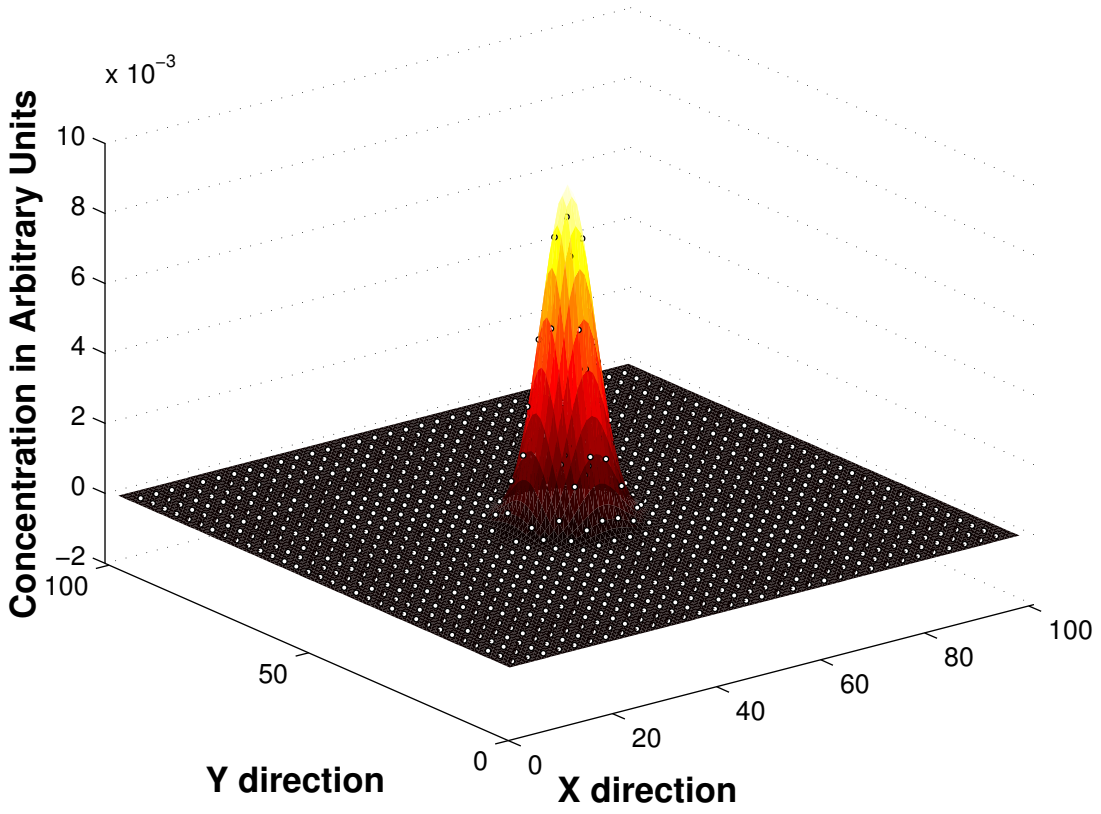


Figure A.3 – Data matching at $t = 100\Delta t$

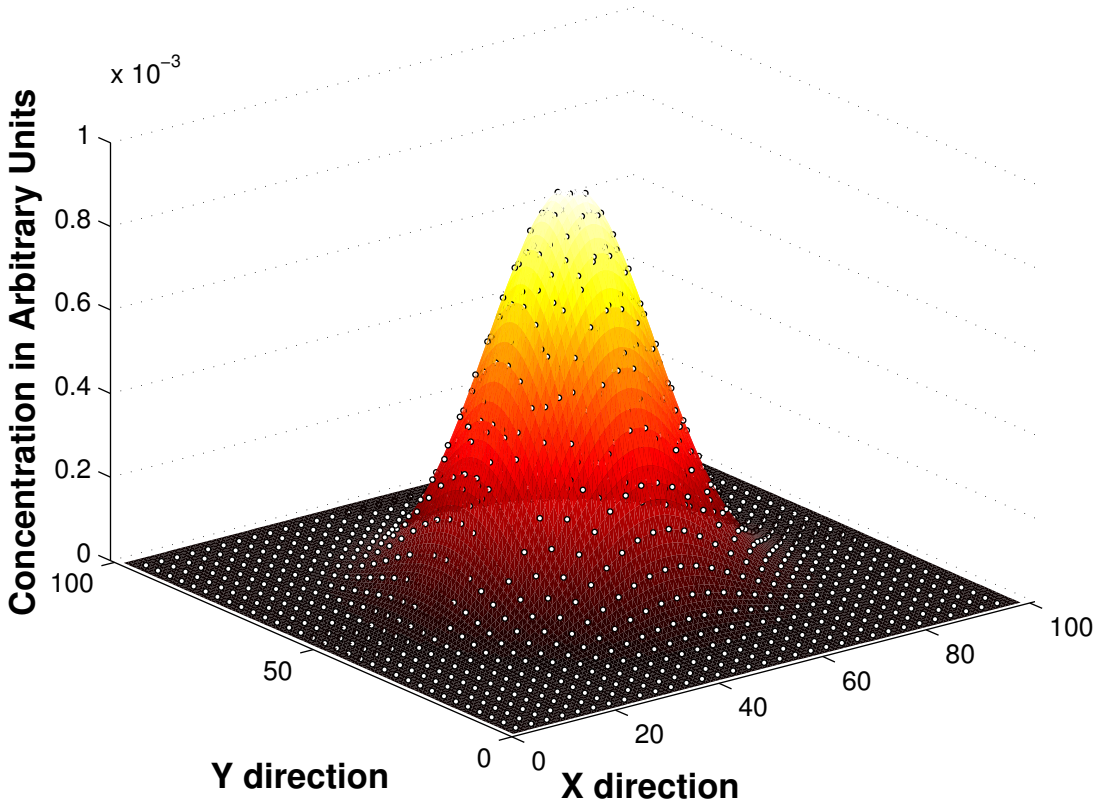


Figure A.4 – Data matching at $t = 1000\Delta t$

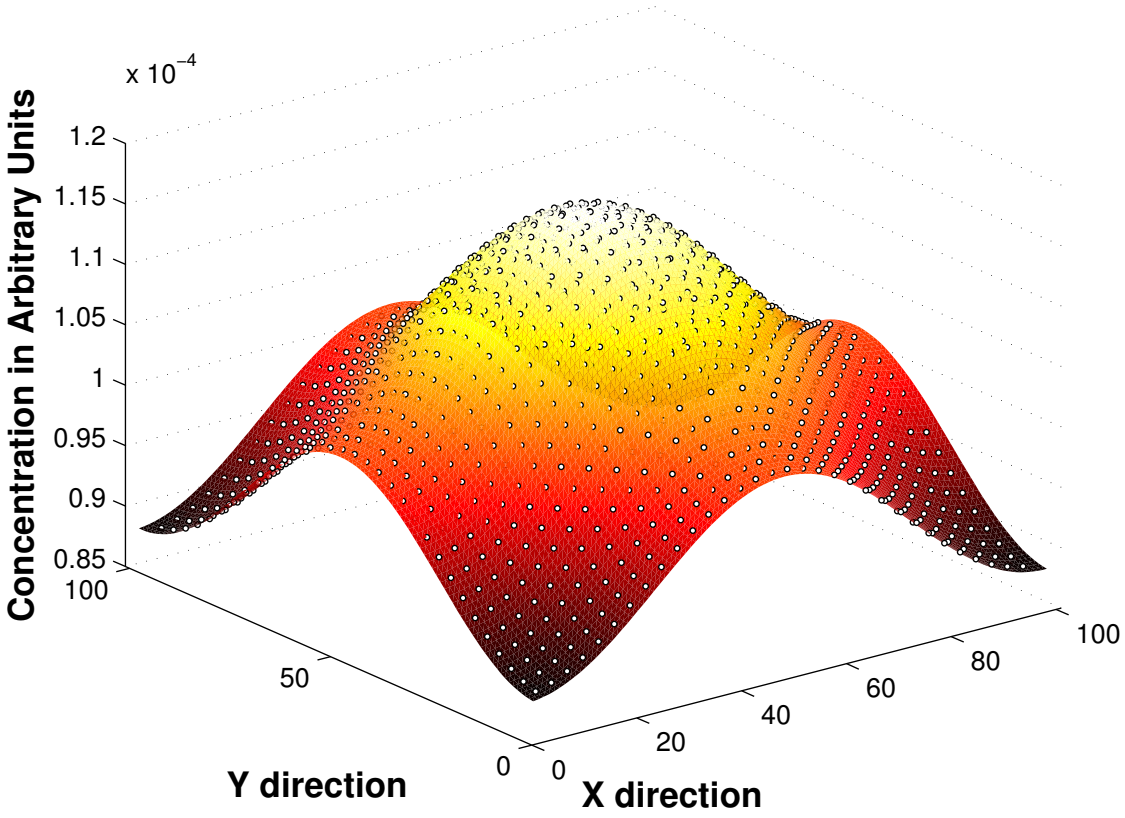


Figure A.5 – Data matching at $t = 10000\Delta t$

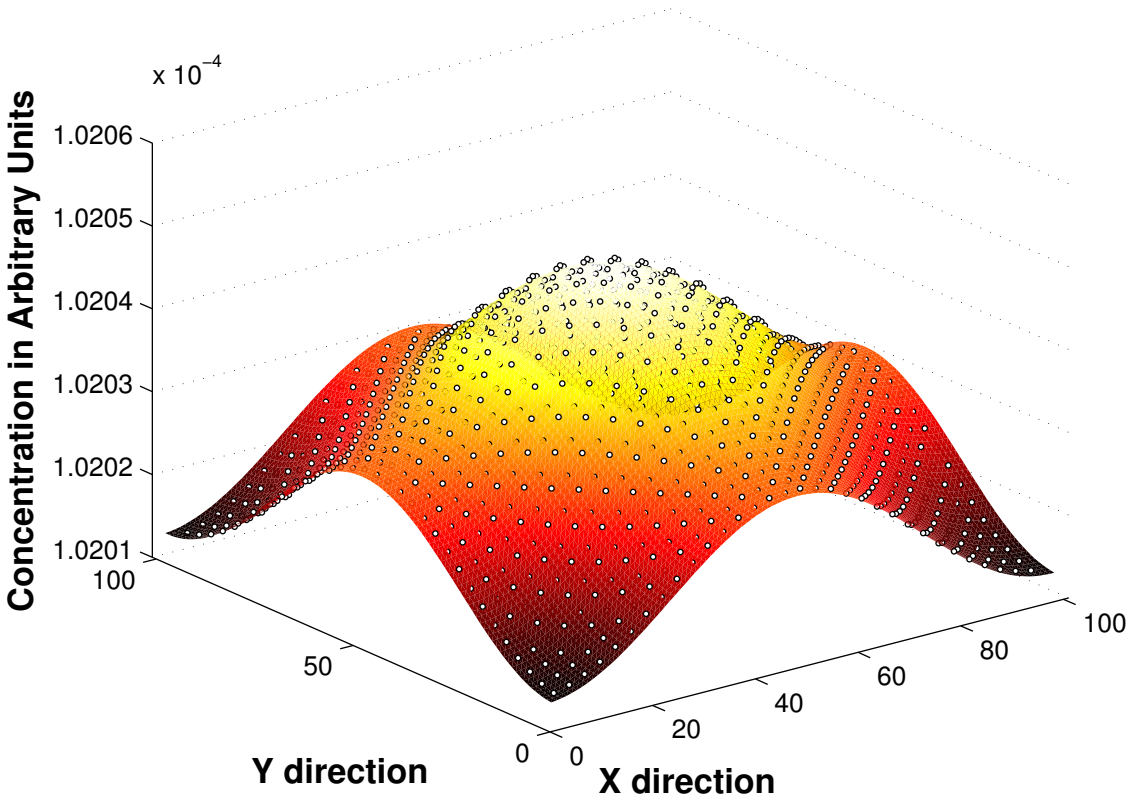


Figure A.6 – Data matching at $t = 30000\Delta t$

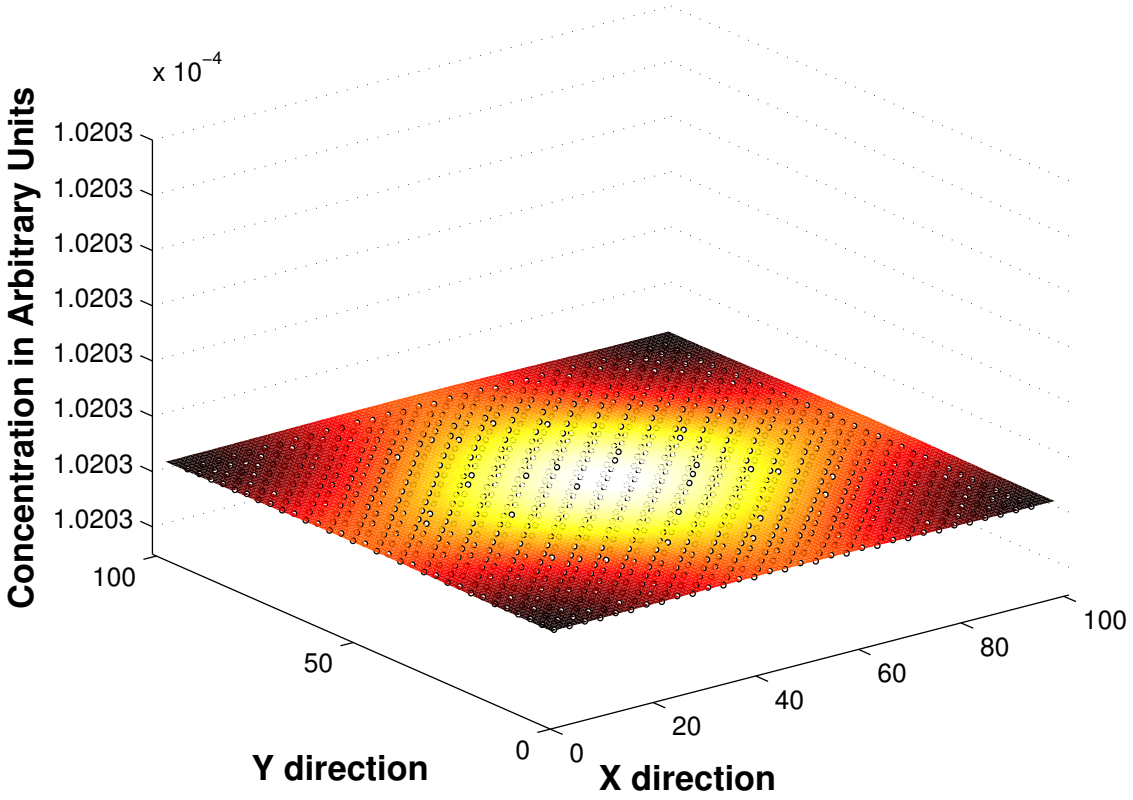


Figure A.7 – Data matching at $t = 60000\Delta t$

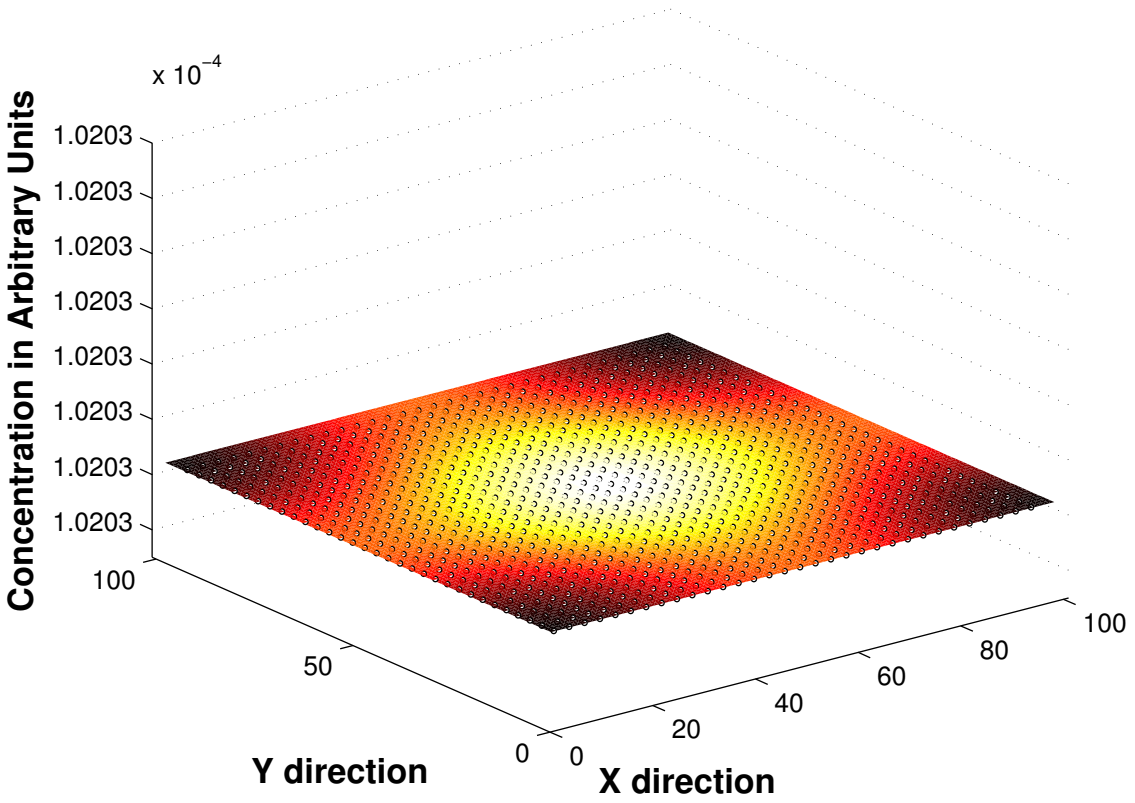


Figure A.8 – Data matching at $t = 100000\Delta t$

Appendix B

Simulation Source Code

This appendix includes the raw source code for the simulation. Everyone is welcome to use and modify the code, provided that references are made to the author. The code consists of several modules. The modules are generally broken down into subroutines. The code is compatible with any modern Fortran compilers including gFortran and G95. The direct, commandline (or terminal) compilation with gFortran, for example, usually reads "gFortran LBM.f90 Flagging.f90 Collision.f90 Collision_Regular.f90 Phase.f90 Streaming.f90 INOUT.f90 Setup.f90 -O3". For architectural tunings on, for example, modern Intel Core i-series CPUs, one can add in addition, "-mtune=corei7-avx" for increased execution speed. Tunings for other architectures are also possible, consult the documentation of the compiler of choice for specific commands. An excellent Integrated Development Environment (IDE) in Windows and Linux is Code::Blocks for Fortran (`http://darmar.vgtu.lt/`), it is compatible with a large variety of languages (specifically Fortran-inclusive for the Fortran version) and compilers (including the best free compilers, gFortran and G95).

The main program body is LBM.f90, which uses and calls all other subroutines in corresponding modules. The main variables and parameters are initialized in LBM.f90.

Streaming.f90 is the streaming portion of the simulation. Its function is to transfer at every time step, the distributions to neighbouring cells. This includes the handling of boundaries (simulation grid, obstacles, objects and crystals).

Collision.f90 consists of the multi-phase implementation of the collision operator, and is responsible for phase separation. It also includes the routines for the perturbation process.

Collision_Regular.f90 is the standard (one of the standard) LBM collision operators. It

relaxes both phases in the mixture, and is not cognizant of phase interactions.

Phase.f90 contains the phase transition portion of the routine. All the stochastics of phase transition and epitaxy are in this module.

Setup.f90 manages all the liquid and solid initializations. There are several instances of liquid and solid initializations, for different purposes such as testing. There is also a random number generator seeding routine based on system time.

Flagging.f90 performs all the cell type book-keepings by marking flags when the grid content has undergone changes due to phase transitions, for example.

INOUT.f90 contains all the routines that reads in and writes out in the simulation. Outputs to the terminal/commandline such as the current average standard deviation in the liquid state (state of separation), the total mass of liquid remaining, etc are also in this module. The images that are written out are in ".pnm" format, and can be directly opened with Gimp (`http://www.gimp.org/`), ImageMagick (`http://www.imagemagick.org/script/index.php`) and ImageJ (`http://rsb.info.nih.gov/ij/`).

B.1 LBM.f90

```
1  PROGRAM LBM
2  USE SETUP
3  USE STREAMING
4  USE COLLISION
5  USE COLLISION2
6  USE INOUT
7  USE FLAGGING
8  USE PHASE
9
10 IMPLICIT NONE
11 !Set Size of the Simulated Grid via S1 and S2
12 !Effective Size of the grid is (S1-2)x(S2-2) because 1 extra row and column at
       each side is used to create boundaries
13 !S1 corresponds to index "j" or vertical direction
14 Integer, Parameter:: S1=1003,S2=1003
15 !Neighbourhood radius (same as "n" in thesis) can be calculated or specified,
       rounded down. Careful with sizing, sizes larger than physically possible
       leads to instabilities.
16 Integer, Parameter:: S_neighbourhood=2
17 !Integer, Parameter:: S_neighbourhood=NINT(REAL(S1+S2)/800.)
18 !Aspect ratio (Asr) and resulting p2 table are below. User defines an Asr value,
        and the algorithm calculates all necessary coefficients in tables.
19 Integer, Parameter:: Asr=32
20 Real, Parameter:: nd=0.5
21 Real, Parameter:: p2min=1./(2.*REAL(Asr)+4.),p2max=Asr*p2min,p2diag=nd*p2min
22 Real, Parameter:: p2_1(9)=[0.,p2min,p2diag,p2max,p2diag,p2min,p2diag,p2max,p2diag
       ]
23 Real, Parameter:: p2_2(9)=[0.,p2max,p2diag,p2min,p2diag,p2max,p2diag,p2min,p2diag
       ]
24 Real, Parameter:: r_tolerance=0.99*min(minval(p2_1(2:9)),minval(p2_2(2:9)))
25 !Omega close to 2 => poor diffusion, Omega close to 0.5 => fast diffusion, stay
       awary from extremes, especially if the perturbation parameters are used to
       keep max in bounds.
26 !Define Pi as a frequently used parameter.
27 Double Precision, Parameter:: Pi=2*acos(0.D0)
28 !Double Precision, Parameter:: Root2=sqrt(2.D0)
```

```
29 !Set Omega(1) and Omega(2). There are 2 sets, for quick switching if needed for
       some scenarios, or for viscosity coupled to other parameters like temperature
       .
30 Double Precision, Parameter:: Omega(2)=[1.76D0,1.76D0], Omega2(2)=[1.D0,1.D0]
31 !Set the Gaussian parameters (mu,sigma). Mu is the same as "b" in thesis, but
       here changed in name to allow perturbation parameter to be "B" instead.
32 Double Precision, Parameter:: mu=1.0, sigma=0.15, Af=1/(1/sqrt(2*Pi*(sigma)**2))
33 !Set surface tension parameters, 1 for each phase, and mixing length Beta0, as
       well as perturbation parameters (A,B,C).
34 Double Precision, Parameter:: A0(2)=[0.000000016D0,0.000000016D0], Beta0=0.15D0,
       A=0.0, B=0.0, C=1.0D0
35 !Set R, the ratio of Rho_1 to (Rho1+Rho2)
36 Double Precision, Parameter:: R=0.35D0
37 !Set a_2, the epitaxy threshold.
38 Real, Parameter:: r_epi_nucl=0.01
39 !Set LSTimeStepRatio (coupling ratio "q" in thesis) for Scaling between Fluid
       Simulation and Solid Simulation
40 !Set LiquidOnlyTime ("Sigma Delta t_liquid" in thesis) and image output interval
       (10k good for actual runs, much less for diagnosis and tests).
41 !Set MaxSolidTime (to limit how long the simulation runs, or to ensure consistent
        # of timesteps).
42 Integer, Parameter:: LSTimeStepRatio=12, LiquidOnlyTime=2000, OutputInterval
       =10000, MaxSolidTime=200000
43 !D2Q9 specific weights, inverse directions, and their vector directions
       normalized are given below, NOT FOR CHANGING.
44 Integer, Parameter:: f(9)=[1,6,7,8,9,2,3,4,5],Ex(9)=[0,1,1,0,-1,-1,-1,0,1],Ey(9)
       =[0,0,1,1,1,0,-1,-1,-1]
45 Double Precision, Parameter:: w(9)=[4.D0/9.D0,1.D0/9.D0,1.D0/36.D0,1.D0/9.D0,1.D0
       /36.D0,1.D0/9.D0,1.D0/36.D0,1.D0/9.D0,1.D0/36.D0]
46 !Initialize a few more variables.
47 Integer TotalTimeSteps, DONE,NOW,PAST,t
48 Double Precision, allocatable::Lattice(:,:,:,:)
49 Integer, Allocatable::Flags(:,:),Q(:,:)
50 Double Precision:: std,std_threshold,totalmass(2)
51 Real clocktime(2)
52 Integer hours,minutes,seconds
53 Integer loadprevious
54 Logical reload_initials
```

```
55
56 !Reload initial conditions if liquid state conditions are found ?
57 reload_initials=.FALSE.
58 !reload_initials=.TRUE.
59
60 !write(*,*) S_neighbourhood
61
62 Call SaveCHECK(S1,S2,Omega,R,LiquidOnlyTime,loadprevious)
63
64
65 ! Initialize Some Variables
66 Allocate(Lattice(S1,S2,9,2,2))
67 Allocate(Flags(S1,S2))
68 Allocate(Q(S1,S2))
69 DONE=0
70 NOW=1
71 PAST=2
72 t=0
73
74
75 ! Initialize Random Number Generator and Create Directories for Storage of
       Results
76 Call init_random_seed()
77 Call system('mkdir IMAGES')
78 Call system('mkdir INIT_CONS')
79 Call system('mkdir DATA')
80
81
82 !Write(*,*) 'Enter Total Time Steps'
83 !Read(*,*) TotalTimeSteps
84
85
86
87 if ((loadprevious.eq.1).and.(reload_initials)) then
88   Call CPU_TIME(clocktime(1))
89   Write(*,*) 'Loading Initial Condition File'
90   Call FluidLOAD(Lattice,Flags,S1,S2,w,Omega,R,std,LiquidOnlyTime,t)
91   Call OutputPPM(Lattice,Flags,S1,S2,LiquidOnlyTime)
```

```
92
93 else
94    Call CPU_TIME(clocktime(1))
95    Write(*,*) 'Setting Up Liquid State From Scratch'
96
97    !Call StripTest(Lattice,Flags,S1,S2,w,TotalTimeSteps,Omega,R)
98    !Call ChannelTest(Lattice,Flags,S1,S2,w,TotalTimeSteps,Omega)
99    !Call FluidTest(Lattice,Flags,S1,S2,w,TotalTimeSteps,Omega)
100   Call FluidSETUP(Lattice,Flags,S1,S2,w,TotalTimeSteps,Omega,R)
101   !Call GradientSETUP(Lattice,Flags,S1,S2,w,TotalTimeSteps,Omega,R)
102
103   Call OutputPPM(Lattice,Flags,S1,S2,t)
104   Call ComputeSTD(Lattice,Flags,S1,S2,NOW,PAST,std,totalmass)
105   write(*,'(a,i7.7,a,f20.17,a,f28.17)') 't=',t,' |STD=',std,' |Total Mass=',
          totalmass(1)+totalmass(2)
106
107   !std_threshold=0.265
108   !t=1
109   !do while (std<std_Threshold)
110
111
112   do t=1,LiquidOnlyTime,1
113
114      !Call Collide2(Lattice,Flags,S1,S2,Ex,Ey,w,f,NOW,PAST,Omega2)
115      Call Collide(Lattice,Flags,S1,S2,Ex,Ey,w,f,NOW,PAST,Omega,Beta0,A0,A,B,C,
             S_neighbourhood)
116      Call Stream(Lattice,Flags,S1,S2,Ex,Ey,w,f,NOW,PAST)
117
118
119      if (mod(t,OutputInterval).eq.0) then
120
121         Call OutputPPM(Lattice,Flags,S1,S2,t)
122
123         Call ComputeSTD(Lattice,Flags,S1,S2,NOW,PAST,std,totalmass)
124         write(*,'(a,i7.7,a,f20.17,a,f28.17)') 't=',t,' |STD=',std,' |Total Mass=',
               totalmass(1)+totalmass(2)
125
126      end if
```

```
127       !t=t+1
128
129     end do
130   end if
131
132
133   !Compute and display total CPU time so far.
134   Call CPU_TIME(clocktime(2))
135   hours=int((clocktime(2)-clocktime(1))/3600.)
136   minutes=int(((clocktime(2)-clocktime(1))-hours*3600)/60.)
137   seconds=int(((clocktime(2)-clocktime(1))-hours*3600)-minutes*60)
138   write(*,*) '
        ------------------------------------------------------------------------',
139   write(*,*) 'TOTAL TIME FOR LIQUID STATE OF',LiquidOnlyTime, ' TIME STEPS WAS:'
140   write(*,*) hours,'hour(s)',minutes,'minute(s)',seconds,'second(s)'
141   write(*,*) '
        ------------------------------------------------------------------------',
142
143
144   !Save the fluid state if it wasn't a reloaded one for future use.
145   if (loadprevious.ne.1) Call FluidSAVE(Lattice,Flags,S1,S2,w,Omega,R,std,
        LiquidOnlyTime)
146   Call FluidSAVE(Lattice,Flags,S1,S2,w,Omega,R,std,LiquidOnlyTime)
147
148   !Setup the solid state.
149   Call SolidSETUP(Q,Flags,S1,S2)
150   std_threshold=std
151
152   !Pausing the simulation here allows only liquid state outputs without
        modifications elsewhere.
153   !pause
154
155   write(*,*) 'Phase Transition Will Run For At Least ',MaxSolidTime,' Time Steps!'
156
157   t=1
158   do while ((Done.eq.0).and.(t<MaxSolidTime))
159
160
```

```
161    Call CHANGE(Lattice,Q,Flags,S1,S2,Ex,Ey,w,f,NOW,PAST,mu,sigma,Af,Pi,r_epi_nucl,
           p2_1,p2_2,r_tolerance)
162    Call ModifySolidFlags(Flags,Q,S1,S2,DONE)
163
164    if (mod(t,OutputInterval).eq.0) then
165
166      write(*,*) 'Now Solid State At t=',t
167      Call OutputPPM(Lattice,Flags,S1,S2,t+LiquidOnlyTime)
168      !Call CompleteSAVE2(Lattice,Flags,S1,S2,w,Omega,R,std,t+LiquidOnlyTime)
169
170    end if
171
172    if (mod(t,LSTimeStepRatio).eq.0) then
173
174      Call ComputeSTD(Lattice,Flags,S1,S2,NOW,PAST,std,totalmass)
175      write(*,'(a,i7.7,a,f20.17,a,f28.17)') 't=',t,' |STD=',std,' |Total Mass=',
             totalmass(1)+totalmass(2)
176
177      !if(std<std_Threshold) then
178        Call Collide(Lattice,Flags,S1,S2,Ex,Ey,w,f,NOW,PAST,Omega,Beta0,A0,A,B,C,
               S_neighbourhood)
179      !else
180      ! Call Collide2(Lattice,Flags,S1,S2,Ex,Ey,w,f,NOW,PAST,Omega,S_neighbourhood)
181      !end if
182
183
184      Call Stream(Lattice,Flags,S1,S2,Ex,Ey,w,f,NOW,PAST)
185
186
187      !Call QFlagsPPM(Flags,S1,s2,t)
188
189      !if(mod((t/LSTimeStepRatio),OutputInterval).eq.0) then
190        !Call QFlagsPPM(Flags,S1,S2,t/LSTimeStepRatio)
191        !Call OutputPPM(Lattice,Flags,S1,S2,t+LiquidOnlyTime)
192
193      !end if
194
195
```

```
196
197     end if
198
199     t=t+1
200 End Do
201
202
203 !write(*,*) 'Performing Post-Simulation AnalysisÉ.'
204
205 Call OutputPPM(Lattice,Flags,S1,S2,t+LiquidOnlyTime)
206 Call CompleteSAVE_ALL(Lattice,Flags,S1,S2,w,Omega,R,std,t+LiquidOnlyTime)
207 !call ComputeCorrelations(Lattice,Flags,S1,S2,w,Omega,R,std,t+LiquidOnlyTime,Pi)
208
209
210 write(*,*) '!!!!!_____.D.O.N.E._____!!!!!'
211
212 END PROGRAM LBM
```

B.2 Streaming.f90

```
1  MODULE STREAMING
2  IMPLICIT NONE
3  Integer i,j,l,inv,inew,jnew
4
5  CONTAINS
6  Subroutine Stream(Lattice,Flags,S1,S2,Ex,Ey,w,f,NOW,PAST)
7  Integer, Intent(IN):: S1,S2,Ex(9),Ey(9),f(9)
8  Integer, Intent(IN):: Flags(S1,S2)
9  Integer, Intent(IN):: NOW,PAST
10 Double Precision, Intent(IN):: w(9)
11 Double Precision, Intent(INOUT):: Lattice(S1,S2,9,2,2)
12
13
14 do i=2,S2-1,1
15   do j=2,S1-1,1
16     do l=1,9,1
17       inv=f(l)
18       if(Flags(j+Ey(inv),i+Ex(inv)).eq.2) then
19         Lattice(j,i,l,1:2,NOW)=Lattice(j+Ey(inv),i+Ex(inv),l,1:2,PAST)
20       end if
21
22       if(Flags(j+Ey(inv),i+Ex(inv)).eq.3) then
23         Lattice(j,i,l,1:2,NOW)=Lattice(j,i,inv,1:2,PAST)
24       end if
25
26       if((Flags(j+Ey(inv),i+Ex(inv)).eq.4)) then
27         Lattice(j,i,l,1:2,NOW)=Lattice(j+Ey(inv),i+Ex(inv),l,1:2,PAST)
28       end if
29
30       if(Flags(j+Ey(inv),i+Ex(inv)).eq.1) then
31         Lattice(j,i,l,1:2,NOW)=Lattice(j,i,inv,1:2,PAST)
32       end if
33
34       if(Flags(j+Ey(inv),i+Ex(inv))>4) then
35         Lattice(j,i,l,1:2,NOW)=Lattice(j,i,inv,1:2,PAST)
36       end if
```

```
37
38       end do
39     end do
40 end do
41
42
43 End Subroutine Stream
44
45 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
46
47 END MODULE
```

B.3 Collision.f90

```
1  MODULE COLLISION
2  IMPLICIT NONE
3  INTEGER i,j,l,CURRENT,k,counter,ll,jj,ii
4  Double Precision Ux,Uy
5
6  CONTAINS
7  Subroutine Collide(Lattice,Flags,S1,S2,Ex,Ey,w,f,NOW,PAST,Omega,Beta0,A0,A,B,C,
       S_neighbourhood)
8  Integer, Intent(IN):: S1,S2,Ex(9),Ey(9),f(9)
9  Integer, Intent(IN):: Flags(S1,S2),S_neighbourhood
10 Integer, Intent(INOUT):: NOW,PAST
11 Double Precision, Intent(IN):: w(9),Omega(2),Beta0,A0(2),A,B,C
12 Double Precision, Intent(INOUT):: Lattice(S1,S2,9,2,2)
13 Double Precision fr(2), fg(2), N0(9,2), N(9,2)
14 Double Precision PHI,OmegaAVG,Ak(2),Beta,Ns,Nl
15
16
17
18 do i=2,S2-1,1
19   do j=2,S1-1,1
20     if((Flags(j,i).eq.2).or.(Flags(j,i).eq.4)) then
21       !can set a scenario for moving lids for testing
22       if((i.eq.2).or.(i.eq.S1-1)) then
23         Ux=0.D0
24         Uy=0.D0
25         fr=0.D0
26       else
27         Ux=0.D0
28         Uy=0.D0
29         fr=0.D0
30       end if
31
32
33       do l=1,9,1
34         fr(1)=fr(1)+Lattice(j,i,l,1,NOW)
35         fr(2)=fr(2)+Lattice(j,i,l,2,NOW)
```

```
36            Ux=Ux+Lattice(j,i,l,1,NOW)*Ex(l)+Lattice(j,i,l,2,NOW)*Ex(l)
37            Uy=Uy+Lattice(j,i,l,1,NOW)*Ey(l)+Lattice(j,i,l,2,NOW)*Ey(l)
38          end do
39
40          Ux=Ux/(fr(1)+fr(2))
41          Uy=Uy/(fr(1)+fr(2))
42          PHI=(fr(1)-fr(2))/(fr(1)+fr(2))
43          OmegaAVG=0.5*(1.+PHI)*Omega(1)+0.5*(1.-PHI)*Omega(2)
44
45          Nl=0
46          Ns=0
47
48
49          do ii=i-S_neighbourhood,i+S_neighbourhood,1
50            do jj=j-S_neighbourhood,j+S_neighbourhood,1
51                        if(((jj>1).and.(jj<S1).and.(ii>1).and.(ii<S2)).and.(Flags(jj,
                              ii).ne.1)) then
52                          if(Flags(jj,ii)>=5) then
53                              Ns=Ns+1
54                          end if
55                          if((Flags(jj,ii)<=4).and.(Flags(jj,ii).ne.1)) then
56                              Nl=Nl+1
57                          end if
58                        end if
59                  end do
60              end do
61
62
63
64          OmegaAVG=OmegaAVG+(OmegaAVG*A)*(Ns/(Ns+Nl))
65          Ak=A0+(A0*B)*(Ns/(Ns+Nl))
66          Beta=Beta0+C*(1.D0-Beta0)/((Ns+Nl-1.D0)/(Ns+Nl))*(Ns/(Ns+Nl))
67
68
69          do l=1,9,1
70            N0(l,1)=(1.D0-OmegaAVG)*(Lattice(j,i,l,1,NOW))+OmegaAVG*f0(0.D0,0.D0,Ex(l
                  ),Ey(l),w(l),fr(1))
```

```
71          N0(l,2)=(1.D0-OmegaAVG)*(Lattice(j,i,l,2,NOW))+OmegaAVG*f0(0.D0,0.D0,Ex(l
                ),Ey(l),w(l),fr(2))
72
73          N(l,1)=(1.D0-OmegaAVG)*(Lattice(j,i,l,1,NOW))+OmegaAVG*f0(Ux,Uy,Ex(l),Ey(
                l),w(l),fr(1))
74          N(l,2)=(1.D0-OmegaAVG)*(Lattice(j,i,l,2,NOW))+OmegaAVG*f0(Ux,Uy,Ex(l),Ey(
                l),w(l),fr(2))
75        end do
76
77
78        fg=0.D0
79        do k=2,9,1
80          if(((j+Ey(k))>1).and.((j+Ey(k))<S1).and.((i+Ex(k))>1).and.((i+Ex(k))<S2))
                then
81          !if(Flags(j+Ey(k),i+Ex(k))<5) then
82            fr=0.D0
83            do l=1,9,1
84              fr(1)=fr(1)+Lattice(j+Ey(k),i+Ex(k),l,1,NOW)
85              fr(2)=fr(2)+Lattice(j+Ey(k),i+Ex(k),l,2,NOW)
86            end do
87
88            fg(1)=fg(1)+Ex(k)*(fr(1)-fr(2))
89            fg(2)=fg(2)+Ey(k)*(fr(1)-fr(2))
90          !end if
91          end if
92        end do
93
94
95
96
97        fr=0.D0
98        do l=1,9,1
99          N(l,1)=N(l,1)+Nk2(Ex(l),Ey(l),Ak(1),fg,l,w(l))
100         N(l,2)=N(l,2)+Nk2(Ex(l),Ey(l),Ak(2),fg,l,w(l))
101         fr(1)=fr(1)+N(l,1)
102         fr(2)=fr(2)+N(l,2)
103       end do
104
```

```
105         do l=1,9,1
106           Lattice(j,i,l,1,PAST)=f000R(fr(1),fr(2),N(l,1)+N(l,2),Beta,N0(l,1)+N0(l
                ,2),fg,Ex(l),Ey(l))
107           Lattice(j,i,l,2,PAST)=f000B(fr(1),fr(2),N(l,1)+N(l,2),Beta,N0(l,1)+N0(l
                ,2),fg,Ex(l),Ey(l))
108         end do
109
110       end if
111     end do
112   end do
113
114
115
116   End Subroutine Collide
117
118
119   !Original LBM F0 operator
120   Double Precision Function f0(Ux,Uy,Ex,Ey,w,fr)
121   Double Precision, Intent(IN):: Ux,Uy,w,fr
122   Integer, Intent(IN):: Ex,Ey
123   f0=w*fr*(1.D0-1.5D0*(Ux*Ux+Uy*Uy)+3.D0*(Ex*Ux+Ey*Uy)+4.5D0*(Ex*Ux+Ey*Uy)*(Ex*Ux+
        Ey*Uy))
124   End Function f0
125
126
127
128   Double Precision Function Nk1(Ux,Uy,Ex,Ey,w,l,fr,ak)
129   Double Precision, Intent(IN):: Ux,Uy,w(l),fr,ak
130   Integer, Intent(IN):: Ex,Ey,l
131   if(l.eq.1) then
132     Nk1=fr*(ak-2./3.*(Ux*Ux+Uy*Uy))
133   end if
134   if((l.eq.2).or.(l.eq.4).or.(l.eq.6).or.(l.eq.8)) then
135     Nk1=fr*((1.-ak)/5.+w(l)*(3.*(Ex*Ux+Ey*Uy)+4.5*(Ex*Ux+Ey*Uy)*(Ex*Ux+Ey*Uy)-1.5*(
          Ux*Ux+Uy*Uy)))
136   end if
137   if((l.eq.3).or.(l.eq.5).or.(l.eq.7).or.(l.eq.9)) then
```

```
138     Nk1=fr*((1.-ak)/20.+w(l)*(3.*(Ex*Ux+Ey*Uy)+4.5*(Ex*Ux+Ey*Uy)*(Ex*Ux+Ey*Uy)
            -1.5*(Ux*Ux+Uy*Uy)))
139   end if
140
141   End function Nk1
142
143   Double Precision Function Nk2(Ex,Ey,A,fg,l,w)
144   Double Precision, Intent(IN):: A,fg(2),w
145   Integer, Intent(IN):: Ex,Ey,l
146   Double Precision:: ff, cf
147
148   ff=fg(1)*fg(1)+fg(2)*fg(2)
149   cf=Ex*fg(1)+Ey*fg(2)
150
151   if(ff>0.D0) then
152     if(mod(l,2).eq.0) then
153       Nk2=0.5D0*A*sqrt(ff)*(w*cf*cf/(ff)-(2.D0/27.D0))
154     end if
155     if((mod(l,2).eq.1).and.(l.ne.1)) then
156       Nk2=0.5D0*A*sqrt(ff)*(w*cf*cf/(ff)-(5.D0/108.D0))
157     end if
158     if(l.eq.1) then
159       Nk2=0.5D0*A*sqrt(ff)*(w*cf*cf/(ff)+(4.D0/27.D0))
160     end if
161
162   else
163     Nk2=0.D0
164   end if
165
166   End Function Nk2
167
168
169   Double Precision Function f000R(R,B,N,Beta,N0,fg,Ex,Ey)
170   Double Precision, Intent(IN):: R,B,N,Beta,N0,fg(2)
171   Integer, Intent(IN):: Ex,Ey
172   Double Precision:: uu,cu,ff,cf,cc
173
174   uu=(Ux*Ux+Uy*Uy)
```

```
175 cu=(Ex*Ux+Ey*Uy)
176 cc=(Ex*Ex+Ey*Ey)
177 ff=fg(1)*fg(1)+fg(2)*fg(2)
178 cf=Ex*fg(1)+Ey*fg(2)
179
180 if((cc.ne.0).and.(ff.ne.0)) then
181    f000R=R/(R+B)*N+Beta*N0*R*B/((R+B)**2)*((cf)/(sqrt(dble(cc))*sqrt(ff)))
182 end if
183 if((cc.eq.0).and.(ff.ne.0)) then
184    f000R=R/(R+B)*N
185 end if
186 if((cc.ne.0).and.(ff.eq.0)) then
187    f000R=R/(R+B)*N
188 end if
189 if((ff.eq.0).and.(ff.eq.0)) then
190    f000R=R/(R+B)*N
191 end if
192
193 End Function f000R
194
195 Double Precision Function f000B(R,B,N,Beta,N0,fg,Ex,Ey)
196 Double Precision, Intent(IN):: R,B,N,Beta,N0,fg(2)
197 Integer, Intent(IN):: Ex,Ey
198 Double Precision:: uu,cu,ff,cf,cc
199
200 uu=(Ux*Ux+Uy*Uy)
201 cu=(Ex*Ux+Ey*Uy)
202 cc=(Ex*Ex+Ey*Ey)
203 ff=fg(1)*fg(1)+fg(2)*fg(2)
204 cf=Ex*fg(1)+Ey*fg(2)
205
206 if((cc.ne.0).and.(ff.ne.0)) then
207    f000B=B/(R+B)*N-Beta*N0*R*B/((R+B)**2)*((cf)/(sqrt(dble(cc))*sqrt(ff)))
208 end if
209 if((cc.eq.0).and.(ff.ne.0)) then
210    f000B=B/(R+B)*N
211 end if
212 if((cc.ne.0).and.(ff.eq.0)) then
```

```
213     f000B=B/(R+B)*N
214  end if
215  if((ff.eq.0).and.(ff.eq.0)) then
216     f000B=B/(R+B)*N
217  end if
218
219  End Function f000B
220
221  END MODULE
```

B.4 Collision_Regular.f90

```
1  MODULE COLLISION2
2  IMPLICIT NONE
3  INTEGER i,j,l,CURRENT,k,counter,ll,jj,ii
4  Double Precision Ux,Uy
5
6  CONTAINS
7  Subroutine Collide2(Lattice,Flags,S1,S2,Ex,Ey,w,f,NOW,PAST,Omega,S_neighbourhood)
8  Integer, Intent(IN):: S1,S2,Ex(9),Ey(9),f(9)
9  Integer, Intent(IN):: Flags(S1,S2),S_neighbourhood
10 Integer, Intent(INOUT):: NOW,PAST
11 Double Precision, Intent(IN):: w(9),Omega(2)
12 Double Precision, Intent(INOUT):: Lattice(2,S1,S2,9,2)
13 Double Precision PHI,OmegaAVG,Ns,Nl,fr(2),Ux,Uy
14
15
16
17 do i=2,S2-1,1
18   do j=2,S1-1,1
19     if((Flags(j,i).eq.2).or.(Flags(j,i).eq.4)) then
20       Ux=0.
21       Uy=0.
22       fr=0.
23       do l=1,9,1
24         fr(1)=fr(1)+Lattice(j,i,l,1,NOW)
25         fr(2)=fr(2)+Lattice(j,i,l,2,NOW)
26         Ux=Ux+Lattice(j,i,l,1,NOW)*Ex(l)+Lattice(j,i,l,2,NOW)*Ex(l)
27         Uy=Uy+Lattice(j,i,l,1,NOW)*Ey(l)+Lattice(j,i,l,2,NOW)*Ey(l)
28       end do
29
30       Ux=Ux/(fr(1)+fr(2))
31       Uy=Uy/(fr(1)+fr(2))
32       PHI=(fr(1)-fr(2))/(fr(1)+fr(2))
33       OmegaAVG=0.5*(1.+PHI)*Omega(1)+0.5*(1.-PHI)*Omega(2)
34
35
36
```

```
37
38
39
40          do l=1,9,1
41             Lattice(j,i,l,1,PAST)=(1.D0-OmegaAVG)*(Lattice(j,i,l,1,NOW))+OmegaAVG*f0
                   (0.D0,0.D0,Ex(l),Ey(l),w(l),fr(1))
42             Lattice(j,i,l,2,PAST)=(1.D0-OmegaAVG)*(Lattice(j,i,l,2,NOW))+OmegaAVG*f0
                   (0.D0,0.D0,Ex(l),Ey(l),w(l),fr(2))
43
44
45          end do
46
47
48       end if
49     end do
50  end do
51
52
53  End Subroutine Collide2
54
55  Double Precision Function f0(Ux,Uy,Ex,Ey,w,fr)
56  Double Precision, Intent(IN) :: Ux,Uy,w,fr
57  Integer, Intent(IN) :: Ex,Ey
58  f0=w*fr*(1.-1.5*(Ux*Ux+Uy*Uy)+3.*(Ex*Ux+Ey*Uy)+4.5*(Ex*Ux+Ey*Uy)*(Ex*Ux+Ey*Uy))
59  End Function f0
60
61  END MODULE
```

B.5 Phase.f90

```
1  MODULE PHASE
2  IMPLICIT NONE
3  Integer i,j,k,l,inv,jj,ii
4
5  CONTAINS
6
7
8  Subroutine CHANGE(Lattice,Q,Flags,S1,S2,Ex,Ey,w,f,NOW,PAST,mu,sigma,Af,Pi,
       r_epi_nucl,p2_1,p2_2,r_tolerance)
9  Integer, Intent(IN):: S1,S2,Ex(9),Ey(9),f(9),NOW,PAST
10 Integer, Intent(INOUT):: Flags(S1,S2),Q(S1,S2)
11 Double Precision, Intent(IN)::Lattice(S1,S2,9,2,2),w(9),mu,sigma,Af,Pi
12 Real, Intent(IN):: r_epi_nucl,p2_1(9),p2_2(9),r_tolerance
13 Real::Rate(2),Kine(2),rho(2)
14 Integer::TempQ(S1,S2),Ns
15 Real:: AnisoRate, TwinProb(2), GrowthRate
16
17 TempQ=0
18
19 !p2(1:9,1) - vertical elongation, p2(1:9,2) - horizontal elongation
20
21
22 !Begin phase transition
23
24
25 do i=2,S2-1,1
26   do j=2,S1-1,1
27         if(Flags(j,i).eq.4) then
28
29       rho=0.
30       do l=1,9,1
31         rho(1)=rho(1)+Lattice(j,i,l,1,NOW)
32         rho(2)=rho(2)+Lattice(j,i,l,2,NOW)
33       end do
34
35       AnisoRate=((Af/sqrt(2*Pi*(sigma)**2)*exp(-((rho(1)-mu)**2/2/(sigma)**2))))
```

```
36
37
38          TwinProb=0
39          Ns=0
40          do l=1,9,1
41             if(Flags(j+Ey(l),i+Ex(l)).eq.5) then
42                TwinProb(1)=TwinProb(1)+p2_1(l)
43                Ns=Ns+1
44             end if
45             if(Flags(j+Ey(l),i+Ex(l)).eq.6) then
46                TwinProb(2)=TwinProb(2)+p2_2(l)
47                Ns=Ns+1
48             end if
49          end do
50
51
52          if(Ns.eq.0)TwinProb=0.5
53
54
55          GrowthRate=AnisoRate*(TwinProb(1)+TwinProb(2))
56
57             Call Random_Number(Rate)
58
59
60             if(Rate(1)<GrowthRate) then
61
62                      Call Random_Number(Kine)
63
64                      if((AnisoRate-TwinProb(1)-TwinProb(2))<r_epi_nucl) then
65
66                   if(abs(TwinProb(1)-TwinProb(2))>=r_tolerance) then
67                      if(TwinProb(1)>TwinProb(2)) then
68                         TempQ(j,i)=1
69                      else
70                         TempQ(j,i)=-1
71                      end if
72                   end if
73
```

```
74                if((abs(TwinProb(1)-TwinProb(2))<r_tolerance)) then
75                  if(kine(1)<0.5) then
76                    TempQ(j,i)=1
77                  else
78                    TempQ(j,i)=-1
79                  end if
80                end if
81
82              else
83
84                if(abs(TwinProb(1)-TwinProb(2))>=r_tolerance) then
85                  if(TwinProb(1)>TwinProb(2)) then
86                    TempQ(j,i)=-1
87                  else
88                    TempQ(j,i)=1
89                  end if
90                end if
91
92                if((abs(TwinProb(1)-TwinProb(2))<r_tolerance)) then
93                  if(kine(2)<0.5) then
94                    TempQ(j,i)=-1
95                  else
96                    TempQ(j,i)=1
97                  end if
98                end if
99
100             end if
101           end if
102
103
104
105         end if
106   end do
107 end do
108
109 Q=Q+TempQ
110
111
```

```
112
113 End Subroutine CHANGE
114
115
116
117 END MODULE
```

B.6 Setup.f90

```
1  MODULE SETUP
2  IMPLICIT NONE
3
4  CONTAINS
5
6  SUBROUTINE init_random_seed()
7  INTEGER :: i, n, clock
8  INTEGER, DIMENSION(:), ALLOCATABLE :: seed
9  CALL RANDOM_SEED(size = n)
10 ALLOCATE(seed(n))
11 CALL SYSTEM_CLOCK(COUNT=clock)
12 seed = clock+37*(/(i - 1,i = 1,n)/)
13 CALL RANDOM_SEED(PUT = seed)
14 DEALLOCATE(seed)
15 END SUBROUTINE
16
17 !Flag types => 1=Bounce back walls, 2=Fluid State, 3=Stream Through Borders, 4=
       Solid Liquid Interface, 5=Solid Type 1, 6=Solid Type 2, 9=initial nuclei
18
19
20 Subroutine FluidTest(Lattice,Flags,S1,S2,w,TotalTimeSteps,Omega)
21 Integer, Intent(IN):: S1,S2,TotalTimeSteps
22 Double Precision, Intent(IN):: w(9),Omega(2)
23 Integer, Intent(INOUT):: Flags(S1,S2)
24 Double Precision, Intent(INOUT):: Lattice(S1,S2,9,2,2)
25 INTEGER:: i, j, l
26 Double Precision:: r
27
28
29 Flags(1:S1,1)=1
30 Flags(1:S1,S2)=1
31 Flags(1,1:S2)=1
32 Flags(S1,1:S2)=1
33
34 do i=1,S2,1
35   do j=1,S1,1
```

```
36
37      if(Flags(j,i).ne.1) then
38        !Call Random_Number(r)
39          Flags(j,i)=2
40          do l=1,9,1
41            !if(r>0.5) then
42            if(mod(j,2).eq.0) then
43              Lattice(j,i,l,1,1:2)=w(l)
44              Lattice(j,i,l,2,1:2)=0.
45            else
46              Lattice(j,i,l,1,1:2)=0.
47              Lattice(j,i,l,2,1:2)=w(l)
48            end if
49
50
51
52          end do
53      else
54        Lattice(j,i,1:9,1:2,1:2)=0.
55      end if
56
57
58    end do
59  end do
60
61
62  End Subroutine FluidTest
63
64  Subroutine FluidSETUP(Lattice,Flags,S1,S2,w,TotalTimeSteps,Omega,R)
65  Integer, Intent(IN):: S1,S2,TotalTimeSteps
66  Double Precision, Intent(IN):: w(9),Omega(2),R
67  Integer, Intent(INOUT):: Flags(S1,S2)
68  Double Precision, Intent(INOUT):: Lattice(S1,S2,9,2,2)
69  INTEGER:: i, j, l
70  Real:: x,grad
71
72
73  Flags(1:S1,1)=1
```

```
74 Flags(1:S1,S2)=1
75 Flags(1,1:S2)=1
76 Flags(S1,1:S2)=1
77
78 do i=1,S2,1
79   do j=1,S1,1
80
81     if((Flags(j,i).ne.1)) then
82       Call Random_Number(x)
83         do l=1,9,1
84           grad=sqrt(real((j-ANINT(S1/2.))**2+(i-ANINT(S2/2.))**2))/((S1+S2)/4)
85           !grad=(real(abs(j-ANINT(S1/2.))+abs(i-ANINT(S2/2.))))/((S1+S2)/4)
86           if((x>0.).and.(grad<0.03)) then
87             Lattice(j,i,l,1,1:2)=w(l)*(1-grad)
88             Lattice(j,i,l,2,1:2)=w(l)*grad
89             Flags(j,i)=2
90           else
91             Lattice(j,i,l,1,1:2)=(R+(x-0.5)*0.01)*w(l)
92             Lattice(j,i,l,2,1:2)=((1-R)-(x-0.5)*0.01)*w(l)
93             Flags(j,i)=2
94           end if
95
96           !Lattice(j,i,l,1,1:2)=(Double Precision(j)/Double Precision(S1))*w(l)
97           !Lattice(j,i,l,2,1:2)=w(l)-Lattice(1:2,j,i,l,1)
98
99           !if ((Double Precision(abs(j-ceiling(S1/2.-1))**1+abs(i-ceiling(S2
                 /2.-1))**1))<((S1+S2)/8.)) then
100          ! Lattice(j,i,l,1,1:2)=w(l)
101          ! Lattice(j,i,l,2,1:2)=0.
102          ! Flags(j,i)=2
103          !else
104          ! Lattice(j,i,l,2,1:2)=w(l)
105          ! Lattice(j,i,l,1,1:2)=0.
106          ! Flags(j,i)=2
107          !end if
108
109
110          !Lattice(j,i,l,1,1:2)=0.5*w(l)
```

```
111             !Lattice(j,i,l,2,1:2)=0.5*w(l)
112
113
114           end do
115       else
116         Lattice(j,i,1:9,1:2,1:2)=0
117       end if
118
119
120     end do
121   end do
122
123   End Subroutine FluidSETUP
124
125   Subroutine GradientSETUP(Lattice,Flags,S1,S2,w,TotalTimeSteps,Omega,R)
126   Integer, Intent(IN):: S1,S2,TotalTimeSteps
127   Double Precision, Intent(IN):: w(9),Omega(2),R
128   Integer, Intent(INOUT):: Flags(S1,S2)
129   Double Precision, Intent(INOUT):: Lattice(S1,S2,9,2,2)
130   INTEGER:: i, j, l
131
132
133
134   Flags(1:S1,1)=1
135   Flags(1:S1,S2)=1
136   Flags(1,1:S2)=1
137   Flags(S1,1:S2)=1
138
139   do i=1,S2,1
140     do j=1,S1,1
141
142       if(Flags(j,i).ne.1) then
143         do l=1,9,1
144           Lattice(j,i,l,1,1:2)=w(l)*real(S2-i)/real(S2)
145           Lattice(j,i,l,2,1:2)=w(l)*(1.-real(S2-i)/real(S2))
146           Flags(j,i)=2
147         end do
148       else
```

```
149         Lattice(j,i,1:9,1:2,1:2)=0
150       end if
151
152
153     end do
154   end do
155
156   Flags(S1-1,2:S2-1)=5
157
158   End Subroutine GradientSETUP
159
160
161   Subroutine SolidSETUP(Q,Flags,S1,S2)
162   Integer, Intent(IN):: S1,S2
163   Integer, Intent(INOUT):: Flags(S1,S2)
164   Integer, Intent(INOUT):: Q(S1,S2)
165   INTEGER:: i, j
166
167   !Center Growth Below
168
169   write(*,*) 'Solid Initializing'
170
171   !do i=1,S2,1
172   ! do j=1,S1,1
173   !    if ((Double Precision((j-ceiling(S1/2.-1))**2+(i-ceiling(S2/2.-1))**2))<((S1+
          S2)/20.)**2) then
174         !if(Flags(j,i).eq.2) then
175           Flags(ceiling(S1/2.),ceiling(S2/2.))=9
176         !end if
177   !    end if
178   ! end do
179   !end do
180
181
182   !Side Growth Below
183   !do j=1,S1,1
184   ! Flags(j,2)=mod(j,2)+5
185   !end do
```

```
186
187
188 Q=0
189
190 do i=1,S2,1
191   do j=1,S1,1
192
193     if(Flags(j,i)>4) then
194       if(Flags(j+1,i).eq.2) Flags(j+1,i)=4
195       if(Flags(j-1,i).eq.2) Flags(j-1,i)=4
196       if(Flags(j,i+1).eq.2) Flags(j,i+1)=4
197       if(Flags(j,i-1).eq.2) Flags(j,i-1)=4
198
199       if(Flags(j-1,i-1).eq.2) Flags(j-1,i-1)=4
200       if(Flags(j-1,i+1).eq.2) Flags(j-1,i+1)=4
201       if(Flags(j+1,i-1).eq.2) Flags(j+1,i-1)=4
202       if(Flags(j+1,i+1).eq.2) Flags(j+1,i+1)=4
203     end if
204
205   end do
206 end do
207
208 do i=1,S2,1
209   do j=1,S1,1
210
211     if(Flags(j,i).eq.5) then
212       Q(j,i)=1
213     end if
214
215     if(Flags(j,i).eq.6) then
216       Q(j,i)=-1
217     end if
218
219   end do
220 end do
221
222 End Subroutine SolidSETUP
223
```

```
224 Subroutine ChannelTest(Lattice,Flags,S1,S2,w,TotalTimeSteps,Omega)
225 Integer, Intent(IN):: S1,S2,TotalTimeSteps
226 Double Precision, Intent(IN):: w(9),Omega(2)
227 Integer, Intent(INOUT):: Flags(S1,S2)
228 Double Precision, Intent(INOUT):: Lattice(S1,S2,9,2,2)
229 INTEGER:: i, j, l
230 Integer:: r
231
232 r=NINT(0.02*min(S1,S2))
233
234 write(*,*) 'width of channel is ',2*r
235
236 Flags(1:S1,1:S2)=2
237 Flags(1:S1,1)=1
238 Flags(1:S1,S2)=1
239 Flags(1,1:S2)=1
240 Flags(S1,1:S2)=1
241
242 do l=1,9,1
243    Lattice(2:S1-1,2:S2-1,l,1,1:2)=0.
244    Lattice(2:S1-1,2:S2-1,l,2,1:2)=w(l)
245    Lattice(NINT(S1/2.)-r:NINT(S1/2.)+r,2:S2-1,l,1,1:2)=1.*w(l)
246    Lattice(NINT(S1/2.)-r:NINT(S1/2.)+r,2:S2-1,l,2,1:2)=0.*w(l)
247    Lattice(2:S1-1,NINT(S2/2.)-r:NINT(S2/2.)+r,l,1,1:2)=1.*w(l)
248    Lattice(2:S1-1,NINT(S2/2.)-r:NINT(S2/2.)+r,l,2,1:2)=0.*w(l)
249 end do
250
251 do i=2,S2-1,1
252   do j=2,S1-1,1
253
254
255      if(((j-i)<2*r).and.((j-i)>-2*r).or.((j-(S1-i))<2*r).and.((j-(S1-i))>-2*r))
           then
256
257        do l=1,9,1
258          Lattice(j,i,l,1,1:2)=1.*w(l)
259          Lattice(j,i,l,2,1:2)=0.*w(l)
260        end do
```

```
261       end if
262
263     end do
264   end do
265
266   End Subroutine ChannelTest
267
268   Subroutine StripTest(Lattice,Flags,S1,S2,w,TotalTimeSteps,Omega,R)
269   Integer, Intent(IN):: S1,S2,TotalTimeSteps
270   Double Precision, Intent(IN):: w(9),Omega(2),R
271   Integer, Intent(INOUT):: Flags(S1,S2)
272   Double Precision, Intent(INOUT):: Lattice(2,S1,S2,9,2)
273   INTEGER:: i, j, l
274   Real:: x
275
276
277   Flags(1:S1,1:S2)=2
278   Flags(1:S1,1)=1
279   Flags(1:S1,S2)=1
280   Flags(1,1:S2)=1
281   Flags(S1,1:S2)=1
282
283   do l=1,9,1
284     Lattice(2:S1-1,2:S2-1,l,1,1:2)=0.5*w(l)
285     Lattice(2:S1-1,2:S2-1,l,2,1:2)=0.5*w(l)
286     !Lattice(1:2,NINT(S1/2.)-r:NINT(S1/2.)+r,2:S2-1,l,1)=1.*w(l)
287     !Lattice(1:2,NINT(S1/2.)-r:NINT(S1/2.)+r,2:S2-1,l,2)=0.*w(l)
288     !Lattice(1:2,2:S1-1,NINT(S2/2.)-r:NINT(S2/2.)+r,l,1)=1.*w(l)
289     !Lattice(1:2,2:S1-1,NINT(S2/2.)-r:NINT(S2/2.)+r,l,2)=0.*w(l)
290
291   end do
292
293   do i=2,S2-1,1
294     do j=2,S1-1,1
295
296       !if((MOD(NINT(real(j)/20.),2).eq.0).or.(MOD(NINT(real(i)/20.),2).eq.0)) then
297
298       if(j>3*NINT(real(S1)/4)) then
```

```
299         call Random_Number(x)
300
301         do l=1,9,1
302            Lattice(j,i,l,1,1:2)=(R+(x-0.5)*0.01)*w(l)
303            Lattice(j,i,l,2,1:2)=((1-R)-(x-0.5)*0.01)*w(l)
304         end do
305         Flags(j,i)=5
306       end if
307
308
309     end do
310  end do
311
312  !Flags(NINT(S1/2.)-r:NINT(S1/2.)+r,2:S2-1)=5
313  !Flags(NINT(S1/2.)-r:NINT(S1/2.)+r,2:S2-1)=5
314  !Flags(2:S1-1,NINT(S2/2.)-r:NINT(S2/2.)+r)=5
315  !Flags(2:S1-1,NINT(S2/2.)-r:NINT(S2/2.)+r)=5
316
317
318  End Subroutine StripTest
319
320
321  END MODULE
```

B.7 Flagging.f90

```
1  MODULE FLAGGING
2  IMPLICIT NONE
3  Integer i,j,l,k
4
5  CONTAINS
6  Subroutine ModifySolidFlags(Flags,Q,S1,S2,DONE)
7  Integer, Intent(IN):: S1,S2
8  Integer, Intent(INOUT):: Flags(S1,S2),DONE
9  Integer FlagTemp(S1,S2), FlagTemp2(S1,S2)
10 Integer, Intent(INOUT):: Q(S1,S2)
11
12   DONE=1
13   FlagTemp=Flags
14
15   do i=2,S2-1,1
16     do j=2,S1-1,1
17       if(Flags(j,i).eq.4) then
18
19         if(Q(j,i).eq.1) then
20           FlagTemp(j,i)=5
21         end if
22         if(Q(j,i).eq.-1) then
23           FlagTemp(j,i)=6
24         end if
25
26       end if
27     end do
28   end do
29
30   FlagTemp2=FlagTemp
31
32   do i=2,S2-1,1
33     do j=2,S1-1,1
34       if(FlagTemp(j,i)>4) then
35
36
```

```
37          if((FlagTemp(j+1,i).eq.2).or.(FlagTemp(j+1,i).eq.4)) then
38            FlagTemp2(j+1,i)=4
39            DONE=0
40          end if
41          if((FlagTemp(j-1,i).eq.2).or.(FlagTemp(j-1,i).eq.4)) then
42            FlagTemp2(j-1,i)=4
43            DONE=0
44          end if
45          if((FlagTemp(j,i+1).eq.2).or.(FlagTemp(j,i+1).eq.4)) then
46            FlagTemp2(j,i+1)=4
47            DONE=0
48          end if
49          if((FlagTemp(j,i-1).eq.2).or.(FlagTemp(j,i-1).eq.4)) then
50            FlagTemp2(j,i-1)=4
51            DONE=0
52          end if
53          if((FlagTemp(j+1,i+1).eq.2).or.(FlagTemp(j+1,i+1).eq.4)) then
54            FlagTemp2(j+1,i+1)=4
55            DONE=0
56          end if
57          if((FlagTemp(j+1,i-1).eq.2).or.(FlagTemp(j+1,i-1).eq.4)) then
58            FlagTemp2(j+1,i-1)=4
59            DONE=0
60          end if
61          if((FlagTemp(j-1,i+1).eq.2).or.(FlagTemp(j-1,i+1).eq.4)) then
62            FlagTemp2(j-1,i+1)=4
63            DONE=0
64          end if
65          if((FlagTemp(j-1,i-1).eq.2).or.(FlagTemp(j-1,i-1).eq.4)) then
66            FlagTemp2(j-1,i-1)=4
67            DONE=0
68          end if
69
70        end if
71
72      end do
73    end do
74
```

```
75     Flags=FlagTemp2
76
77 End Subroutine ModifySolidFlags
78
79
80
81 END MODULE
```

B.8 INOUT.f90

```
1  MODULE INOUT
2  IMPLICIT NONE
3  Integer j,i,l, k
4
5
6  CONTAINS
7
8  Subroutine ComputeTotalMass(Lattice,Flags,NOW,S1,S2)
9  Integer, Intent(IN):: S1,S2,Flags(S1,S2),NOW
10 Double Precision, Intent(IN):: Lattice(S1,S2,9,2,2)
11 Double Precision rho1, rho2
12
13 rho2=0
14 do i=2,S2-1,1
15 do j=2,S1-1,1
16    if((Flags(j,i).eq.2).or.(Flags(j,i).eq.4).or.(Flags(j,i).eq.3)) then
17       rho1=0
18       do l=1,9,1
19          rho1=rho1+Lattice(j,i,l,1,NOW)+Lattice(j,i,l,2,NOW)
20       end do
21       rho2=rho2+rho1
22    end if
23 end do
24 end do
25
26 write(*,*) 'Total Grid Mass:  ', rho2
27
28 End Subroutine ComputeTotalMass
29
30 Subroutine ComputeSTD(Lattice,Flags,S1,S2,NOW,PAST,std,totalmass)
31 Integer, Intent(IN):: S1,S2,Flags(S1,S2),NOW,PAST
32 Double Precision, Intent(INOUT):: std,totalmass(2)
33 Double Precision, Intent(IN):: Lattice(S1,S2,9,2,2)
34 Double Precision:: rho(2),mean,sumofsquares
35 Integer:: ctr
36
```

```
37 sumofsquares=0.D0
38 ctr=0
39 mean=0.D0
40 totalmass=0.D0
41
42 do i=2,S2-1,1
43   do j=2,S1-1,1
44
45     if ((Flags(j,i)>=2).and.(Flags(j,i)<=4)) then
46       rho=0.D0
47       ctr=ctr+1
48       do l=1,9,1
49         rho(1)=rho(1)+Lattice(j,i,l,1,NOW)
50         rho(2)=rho(2)+Lattice(j,i,l,2,NOW)
51       end do
52       mean=mean+(rho(1)-rho(2))
53       sumofsquares=sumofsquares+(rho(1)-rho(2))**2
54       totalmass=totalmass+rho
55     end if
56
57
58
59   end do
60 end do
61
62
63
64 if(ctr.ne.0) then
65   sumofsquares=sumofsquares/dble(ctr)
66   mean=mean/dble(ctr)
67 end if
68
69 if(sumofsquares>(mean**2)) then
70   std=sqrt(sumofsquares-mean**2)
71 else
72   std=0.D0
73 end if
74
```

```
75 End Subroutine ComputeSTD
76
77
78 Subroutine QFlagsPNM(Flags,S1,S2,t)
79 Integer, Intent(IN):: S1,S2,t
80 Integer, Intent(IN):: Flags(S1,S2)
81 Character (LEN=4) Out(S1,S2)
82 Character (80) Oname
83
84 write(Oname,'(a,i4.4,a,i4.4,a,i6.6,a)') 'Solid_GrayScale',S1,'x',S2,'_at_',t,'.
       pnm'
85
86
87 OPEN(140,FILE=Oname,STATUS='unkNOWn')
88
89 write(140,*) 'P2'
90 write(140,*) S2,S1
91 write(140,*) '255'
92
93 do i=1,S2,1
94   do j=1,S1,1
95       !if (Flags(j,i).eq.0) Out(j,i)=' 000'
96       if (Flags(j,i).eq.1) Out(j,i)=' 000'
97     if (Flags(j,i).eq.2) Out(j,i)=' 100'
98     if (Flags(j,i).eq.3) Out(j,i)=' 200'
99     if (Flags(j,i).eq.4) Out(j,i)=' 150'
100    if (Flags(j,i).eq.5) Out(j,i)=' 250'
101    if (Flags(j,i).eq.6) Out(j,i)=' 000'
102    if (Flags(j,i).eq.7) Out(j,i)=' 200'
103  end do
104 end do
105
106
107 do j=1,S1,1
108   write(140,*) (Out(j,i), i=1,S2)
109 end do
110
111 Flush(140)
```

```
112 Close(140)
113
114
115 End Subroutine QFlagsPNM
116
117 Subroutine QFlagsPPM(Flags,S1,S2,t)
118 Integer, Intent(IN):: S1,S2,t
119 Integer, Intent(IN):: Flags(S1,S2)
120 Character (LEN=12):: Out(S1,S2)
121 Character(255)::Oname
122 Character(12)::FlagString
123 Integer::r,g,b
124
125 write(Oname,'(a,i4.4,a,i4.4,a,i6.6,a)') 'Solid_',S1,'x',S2,'_at_',t,'.pnm'
126
127 OPEN(10,FILE=Oname,STATUS='unkNOWn')
128
129 write(10,*) 'P3'
130 write(10,*) S2,S1
131 write(10,*) '255'
132
133
134 do i=1,S2,1
135   do j=1,S1,1
136
137         if (Flags(j,i).eq.1) Out(j,i)=' 000 000 000'
138     if (Flags(j,i).eq.2) Out(j,i)=' 100 100 100'
139     if (Flags(j,i).eq.3) Out(j,i)=' 000 000 000'
140     if (Flags(j,i).eq.4) Out(j,i)=' 255 000 255'
141     if (Flags(j,i).eq.5) Out(j,i)=' 255 255 255'
142     if (Flags(j,i).eq.6) Out(j,i)=' 192 192 192'
143     if (Flags(j,i).eq.7) Out(j,i)=' 128 128 128'
144     if (Flags(j,i).eq.8) Out(j,i)=' 000 000 000'
145     if (Flags(j,i).eq.9) Out(j,i)=' 000 000 255'
146     if (Flags(j,i).eq.0) Out(j,i)=' 000 255 000'
147
148   end do
149 end do
```

```
150
151
152 do j=1,S1,1
153    write(10,*) (Out(j,i), i=1,S2)
154 end do
155
156
157 Flush(10)
158 Close(10)
159
160 End Subroutine QFlagsPPM
161
162 Subroutine OutputPPM(Lattice,Flags,S1,S2,t)
163 Integer, Intent(IN):: S1,S2,t
164 Integer, Intent(IN):: Flags(S1,S2)
165 Double Precision, Intent(IN):: Lattice(S1,S2,9,2,2)
166 Character(LEN=12)::Out(S1,S2)
167 Character(255):: Oname
168 Character(12):: DensityString
169 Character(500):: path
170 Double Precision::rho(2)
171 Integer:: r,g,b,maxval,display_oversaturations,stat
172 maxval=255
173 !set below variable to 1 to display over saturations, 0 to ignore.
174 display_oversaturations=0
175 !change below location and string to match desired location to store images and
       operating system syntax.
176 CALL GETCWD(path,stat)
177 !write(*,*) TRIM(path),stat
178
179
180 if(stat.eq.0) then
181    write(Oname,'(a,i4.4,a,i4.4,a,i7.7,a)') TRIM(path)//'/IMAGES/Combined_',S1,'x',
          S2,'_at_',t,'.pnm'
182 else
183    write(Oname,'(a,i4.4,a,i4.4,a,i7.7,a)') 'Combined_',S1,'x',S2,'_at_',t,'.pnm'
184 end if
185
```

```
186
187
188 OPEN(140,FILE=Oname,STATUS='unkNOWn')
189
190 write(140,'(a2)') 'P3'
191 write(140,'(i4.4,a,i4.4)') S2,' ',S1
192 write(140,'(i3.3)') maxval
193
194
195 do i=1,S2,1
196   do j=1,S1,1
197     rho=0
198     do l=1,9,1
199       rho(1)=rho(1)+Lattice(j,i,l,1,1)
200       rho(2)=rho(2)+Lattice(j,i,l,2,1)
201     end do
202
203
204     if((Flags(j,i).eq.2).or.(Flags(j,i).eq.4)) then
205       r=NINT(rho(1)*255.)
206       g=0
207       b=NINT(rho(2)*255.)
208       if (display_oversaturations.eq.1) then
209         if(r>255) then
210           r=255
211           b=0
212           g=255
213         end if
214         if(b>255) then
215           r=0
216           b=255
217           g=255
218         end if
219       else
220         if(r>255) then
221           r=255
222           b=0
223           g=0
```

```
224           end if
225           if(b>255) then
226             r=0
227             b=255
228             g=0
229           end if
230         end if
231
232
233         write(DensityString,'(a,i3.3,a,i3.3,a,i3.3)') ' ',r,' ',g,' ',b
234         Out(j,i)=DensityString
235       end if
236
237       if (Flags(j,i).eq.1) Out(j,i)=' 000 000 000'
238
239       if (Flags(j,i).eq.3) Out(j,i)=' 000 000 000'
240
241       !if (Flags(j,i).eq.4) Out(j,i)=' 255 000 255'
242
243       if (Flags(j,i).eq.5) then
244           r=128-NINT(127.*rho(1))
245           g=128-NINT(127.*rho(1))
246           b=128-NINT(127.*rho(1))
247           write(DensityString,'(a,i3.3,a,i3.3,a,i3.3)') ' ',r,' ',g,' ',b
248           Out(j,i)=DensityString
249       end if
250
251       if (Flags(j,i).eq.6) then
252           r=128+NINT(127.*rho(1))
253           g=128+NINT(127.*rho(1))
254           b=128+NINT(127.*rho(1))
255           write(DensityString,'(a,i3.3,a,i3.3,a,i3.3)') ' ',r,' ',g,' ',b
256           Out(j,i)=DensityString
257       end if
258
259       if (Flags(j,i).eq.7) then
260           r=128+NINT(127.*rho(1))
261           g=128+NINT(127.*rho(1))
```

```
262           b=0
263           write(DensityString,'(a,i3.3,a,i3.3,a,i3.3)') ' ',r,' ',g,' ',b
264           Out(j,i)=DensityString
265       end if
266
267       if (Flags(j,i).eq.8) then
268           r=0
269           g=128+NINT(127.*rho(1))
270           b=128+NINT(127.*rho(1))
271           write(DensityString,'(a,i3.3,a,i3.3,a,i3.3)') ' ',r,' ',g,' ',b
272           Out(j,i)=DensityString
273       end if
274
275       if (Flags(j,i).eq.9) then
276         Out(j,i)=' 000 000 255'
277       end if
278
279     end do
280   end do
281
282
283
284   do j=1,S1,1
285     write(140,*) (Out(j,i), i=1,S2)
286   end do
287
288
289   Flush(140)
290   Close(140)
291
292
293   End Subroutine OutputPPM
294
295
296
297   Subroutine FluidSAVE(Lattice,Flags,S1,S2,w,Omega,R,std,LiquidOnlyTime)
298   Integer, Intent(IN):: S1,S2,LiquidOnlyTime
299   Double Precision, Intent(IN):: w(9),Omega(2),R,std
```

```
300 Integer, Intent(IN):: Flags(S1,S2)
301 Double Precision, Intent(IN):: Lattice(S1,S2,9,2,2)
302 INTEGER:: i, j, l, stat
303 Character(255):: Oname
304 Character(500):: path
305 Double Precision:: rho(2)
306
307 CALL GETCWD(path,stat)
308 !write(*,*) TRIM(path),stat
309
310
311 if(stat.eq.0) then
312    write(Oname,'(a,i4.4,a,i4.4,a,i7.7,a)') TRIM(path)//'/INIT_CONS/Liquid_',S1,'x'
         ,S2,'_at_',LiquidOnlyTime,'.txt'
313 else
314    write(Oname,'(a,i4.4,a,i4.4,a,i7.7,a)') 'Liquid_',S1,'x',S2,'_at_',
         LiquidOnlyTime,'.txt'
315 end if
316
317 OPEN(10,FILE=Oname,STATUS='unkNOWn')
318
319 write(10,'(i4,1x,i4,1x,i5,1x,f4.2,1x,f4.2,1x,f4.2,1x,d16.8)') S1,S2,
      LiquidOnlyTime,Omega(1),Omega(2),R,std
320
321 do i=1,S2,1
322    do j=1,S1,1
323       rho=0.
324       do l=1,9,1
325          rho(1)=rho(1)+Lattice(j,i,l,1,1)
326          rho(2)=rho(2)+Lattice(j,i,l,2,1)
327       end do
328       write(10,'(i4,1x,i4,1x,d16.8,1x,d16.8,1x,i1)') j,i,rho(1),rho(2),Flags(j,i)
329    end do
330 end do
331
332 Flush(10)
333 Close(10)
334
```

```
335 End Subroutine FluidSAVE
336
337 Subroutine FluidLOAD(Lattice,Flags,S1,S2,w,Omega,R,std,LiquidOnlyTime,t)
338 Integer, Intent(IN):: S1,S2,LiquidOnlyTime
339 Double Precision, Intent(IN):: w(9),Omega(2),R
340 Integer, Intent(INOUT):: Flags(S1,S2),t
341 Double Precision, Intent(INOUT):: Lattice(S1,S2,9,2,2),std
342 INTEGER:: i, j, l, stat,f,ss1,ss2,elapsedtime
343 Character(255):: Oname
344 Character(500):: path
345 Double Precision:: rho(2),O(2),ratio
346
347 CALL GETCWD(path,stat)
348 !write(*,*) TRIM(path),stat
349
350
351 if(stat.eq.0) then
352   write(Oname,'(a,i4.4,a,i4.4,a,i7.7,a)') TRIM(path)//'/INIT_CONS/Liquid_',S1,'x'
        ,S2,'_at_',LiquidOnlyTime,'.txt'
353 else
354   write(Oname,'(a,i4.4,a,i4.4,a,i7.7,a)') 'Liquid_',S1,'x',S2,'_at_',
        LiquidOnlyTime,'.txt'
355 end if
356
357 OPEN(10,FILE=Oname,STATUS='unknown')
358
359 REWIND(10)
360
361 read(10,'(i4,1x,i4,1x,i5,1x,f4.2,1x,f4.2,1x,f4.2,1x,d16.8)') ss1,ss2,elapsedtime,
      O(1),O(2),ratio,std
362 !write(*,*) ss1,ss2,elapsedtime,O(1),O(2),ratio
363
364 stat=0
365 do while(stat.eq.0)
366   READ(10,'(i4,1x,i4,1x,d16.8,1x,d16.8,1x,i1)',iostat=stat) j,i,rho(1),rho(2),f
367   !write(*,*) j,i,rho(1),rho(2),f
368   do l=1,9,1
369     Lattice(j,i,l,1,1:2)=rho(1)*w(l)
```

```
370       Lattice(j,i,l,2,1:2)=rho(2)*w(l)
371     end do
372     Flags(j,i)=f
373   end do
374
375   Close(10)
376
377   End Subroutine FluidLOAD
378
379   Subroutine SaveCHECK(S1,S2,Omega,R,LiquidOnlyTime,loadprevious)
380   Integer, Intent(IN):: S1,S2,LiquidOnlyTime
381   Double Precision, Intent(IN):: Omega(2),R
382   Integer, Intent(INOUT)::loadprevious
383  INTEGER:: stat
384   Logical:: ex
385   Character(255):: Oname
386   Character(500):: path
387
388  CALL GETCWD(path,stat)
389   !write(*,*) TRIM(path),stat
390
391
392   if(stat.eq.0) then
393     write(Oname,'(a,i4.4,a,i4.4,a,i7.7,a)') TRIM(path)//'/INIT_CONS/Liquid_',S1,'x'
          ,S2,'_at_',LiquidOnlyTime,'.txt'
394   else
395     write(Oname,'(a,i4.4,a,i4.4,a,i7.7,a)') 'Liquid_',S1,'x',S2,'_at_',
          LiquidOnlyTime,'.txt'
396   end if
397
398  INQUIRE(FILE=Oname,EXIST=ex)
399
400   if(ex) then
401     loadprevious=1
402     write(*,*) 'Initial Condition Found In Saved Files'
403   else
404     loadprevious=0
405     write(*,*) 'No Initial Condition Files Found for Specified Parameters'
```

```
406 end if
407
408 End Subroutine SaveCHECK
409
410 Subroutine CompleteSAVE_L_S(Lattice,Flags,S1,S2,w,Omega,R,std,t)
411 Integer, Intent(IN):: S1,S2,t
412 Double Precision, Intent(IN):: w(9),Omega(2),R,std
413 Integer, Intent(IN):: Flags(S1,S2)
414 Double Precision, Intent(IN):: Lattice(S1,S2,9,2,2)
415 INTEGER:: i, j, l, stat
416 Character(255):: Oname
417 Character(500):: path
418 Double Precision:: rho(2)
419
420 CALL GETCWD(path,stat)
421 !write(*,*) TRIM(path),stat
422
423
424 if(stat.eq.0) then
425   write(Oname,'(a,i4.4,a,i4.4,a,i7.7,a)') TRIM(path)//'/RESULTS/Solid_',S1,'x',S2
        ,'_at_',t,'.txt'
426 else
427   write(Oname,'(a,i4.4,a,i4.4,a,i7.7,a)') 'Solid_',S1,'x',S2,'_at_',t,'.txt'
428 end if
429
430 OPEN(10,FILE=Oname,STATUS='unkNOWn')
431
432 if(stat.eq.0) then
433   write(Oname,'(a,i4.4,a,i4.4,a,i7.7,a)') TRIM(path)//'/RESULTS/Liquid_',S1,'x',
        S2,'_at_',t,'.txt'
434 else
435   write(Oname,'(a,i4.4,a,i4.4,a,i7.7,a)') 'Liquid_',S1,'x',S2,'_at_',t,'.txt'
436 end if
437
438 OPEN(20,FILE=Oname,STATUS='unkNOWn')
439
440 write(10,'(i4,1x,i4,1x,i7,1x,f4.2,1x,f4.2,1x,f4.2,1x,d16.8)') S1,S2,t,Omega(1),
      Omega(2),R,std
```

```
441 write(20,'(i4,1x,i4,1x,i7,1x,f4.2,1x,f4.2,1x,f4.2,1x,d16.8)') S1,S2,t,Omega(1),
        Omega(2),R,std
442
443 do i=1,S2,1
444   do j=1,S1,1
445     if(Flags(j,i)>4) then
446       rho=0.
447       do l=1,9,1
448         rho(1)=rho(1)+Lattice(j,i,l,1,1)
449         rho(2)=rho(2)+Lattice(j,i,l,2,1)
450       end do
451       write(10,'(i4.4,1x,i4.4,1x,d16.8,1x,d16.8,1x,i1)') j,i,rho(1),rho(2),Flags(
            j,i)
452     end if
453     if(Flags(j,i)<=4) then
454       rho=0.
455       do l=1,9,1
456         rho(1)=rho(1)+Lattice(j,i,l,1,1)
457         rho(2)=rho(2)+Lattice(j,i,l,2,1)
458       end do
459       write(20,'(i4.4,1x,i4.4,1x,d16.8,1x,d16.8,1x,i1)') j,i,rho(1),rho(2),Flags(
            j,i)
460     end if
461   end do
462 end do
463
464 Flush(10)
465 Close(10)
466 Flush(20)
467 Close(20)
468
469 End Subroutine CompleteSAVE_L_S
470
471
472 Subroutine CompleteSAVE_ALL(Lattice,Flags,S1,S2,w,Omega,R,std,t)
473 Integer, Intent(IN):: S1,S2,t
474 Double Precision, Intent(IN):: w(9),Omega(2),R,std
475 Integer, Intent(IN):: Flags(S1,S2)
```

```
476 Double Precision, Intent(IN):: Lattice(S1,S2,9,2,2)
477 INTEGER:: i, j, l, stat
478 Character(255):: Oname
479 Character(500):: path
480 Double Precision:: rho(2)
481
482 CALL GETCWD(path,stat)
483 !write(*,*) TRIM(path),stat
484
485
486 if(stat.eq.0) then
487    write(Oname,'(a,i4.4,a,i4.4,a,i7.7,a)') TRIM(path)//'/DATA/Combined_',S1,'x',S2
         ,'_at_',t,'.txt'
488 else
489    write(Oname,'(a,i4.4,a,i4.4,a,i7.7,a)') 'Combined_',S1,'x',S2,'_at_',t,'.txt'
490 end if
491
492 OPEN(10,FILE=Oname,STATUS='unkNOWn')
493 write(10,'(i4,1x,i4,1x,i7,1x,f4.2,1x,f4.2,1x,f4.2,1x,d16.8)') S1,S2,t,Omega(1),
      Omega(2),R,std
494
495 if(stat.eq.0) then
496    write(Oname,'(a,i4.4,a,i4.4,a,i7.7,a)') TRIM(path)//'/DATA/MATRIX',S1,'x',S2,'
         _at_',t,'.txt'
497 else
498    write(Oname,'(a,i4.4,a,i4.4,a,i7.7,a)') 'MATRIX',S1,'x',S2,'_at_',t,'.txt'
499 end if
500
501 OPEN(15,FILE=Oname,STATUS='unknown')
502
503
504
505 do i=1,S2,1
506    do j=1,S1,1
507       if(Flags(j,i).ne.1) then !exclude borders (flags=1)
508          rho=0.
509          do l=1,9,1
510             rho(1)=rho(1)+Lattice(j,i,l,1,1)
```

```
511           rho(2)=rho(2)+Lattice(j,i,l,2,1)
512         end do
513         write(10,'(i4.4,1x,i4.4,1x,d16.8,1x,d16.8,1x,i1)') j,i,rho(1),rho(2),Flags(
                j,i)
514       end if
515     end do
516 end do
517
518
519 Flush(10)
520 Close(10)
521
522 do j=1,S1,1
523    write(15,*) (Flags(j,i),i=1,S2)
524 end do
525
526
527 Flush(15)
528 Close(15)
529
530 End Subroutine CompleteSAVE_ALL
531
532
533
534 Subroutine ComputeCorrelations(Lattice,Flags,S1,S2,w,Omega,R,std,t,Pi)
535 Integer, Intent(IN):: S1,S2,t
536 Double Precision, Intent(IN):: w(9),Omega(2),R,std,Pi
537 Integer, Intent(IN):: Flags(S1,S2)
538 Double Precision, Intent(IN):: Lattice(S1,S2,9,2,2)
539 INTEGER:: i, j, l, stat,k,center(2),Euclid_Distance, Angle,jj,ii,rr,rr_max,rr_min
540 Character(255):: Oname
541 Character(500):: path
542 Double Precision, Allocatable:: radius(:,:), theta(:,:), theta2(:,:),theta3
        (:,:,:)
543
544
545 Allocate(radius(INT(REAL(MIN(S1,S2)/2.))-1,3))
546 Allocate(theta(360,3))
```

```
547 Allocate(theta2(360,3))
548 Allocate(theta3(360,NINT(REAL(min(S1,S2))/2.)-1,5))
549
550 center(1)=AINT(REAL(S1)/2.)
551 center(2)=AINT(REAL(S2)/2.)
552 radius=0.
553 theta=0.
554 theta2=0.
555 theta3=0.
556
557 !Set max radius for computations, correlations are computed for each rr value
        such that rr<rr_max.  Larger values can take a long time.
558 rr_max=25
559
560 CALL GETCWD(path,stat)
561
562
563 if(stat.eq.0) then
564   write(Oname,'(a,i4.4,a,i4.4,a,i7.7,a)') TRIM(path)//'/DATA/RCorrelations_',S1,'
          x',S2,'_at_',t,'.txt'
565 else
566   write(Oname,'(a,i4.4,a,i4.4,a,i7.7,a)') 'RCorrelations_',S1,'x',S2,'_at_',t,'.
          txt'
567 end if
568
569 OPEN(20,FILE=Oname,STATUS='unkNOWn')
570
571 if(stat.eq.0) then
572   write(Oname,'(a,i4.4,a,i4.4,a,i7.7,a)') TRIM(path)//'/DATA/TCorrelations_',S1,'
          x',S2,'_at_',t,'.txt'
573 else
574   write(Oname,'(a,i4.4,a,i4.4,a,i7.7,a)') 'TCorrelations_',S1,'x',S2,'_at_',t,'.
          txt'
575 end if
576
577 OPEN(30,FILE=Oname,STATUS='unkNOWn')
578
579 if(stat.eq.0) then
```

```
580    write(Oname,'(a,i4.4,a,i4.4,a,i7.7,a)') TRIM(path)//'/DATA/TCor_Struct_Type_',
           S1,'x',S2,'_at_',t,'.txt'
581  else
582    write(Oname,'(a,i4.4,a,i4.4,a,i7.7,a)') 'TCor_Struct_Type_',S1,'x',S2,'_at_',t,
           '.txt'
583  end if
584
585  OPEN(40,FILE=Oname,STATUS='unkNOWn')
586
587  if(stat.eq.0) then
588    write(Oname,'(a,i4.4,a,i4.4,a,i7.7,a)') TRIM(path)//'/DATA/TCor_R_Type_',S1,'x'
           ,S2,'_at_',t,'.txt'
589  else
590    write(Oname,'(a,i4.4,a,i4.4,a,i7.7,a)') 'TCor_R_Type_',S1,'x',S2,'_at_',t,'.txt
           '
591  end if
592
593  OPEN(50,FILE=Oname,STATUS='unkNOWn')
594
595  if(stat.eq.0) then
596    write(Oname,'(a,i4.4,a,i4.4,a,i7.7,a)') TRIM(path)//'/DATA/TR_SS_Type_',S1,'x',
           S2,'_at_',t,'.txt'
597  else
598    write(Oname,'(a,i4.4,a,i4.4,a,i7.7,a)') 'TR_SS_Type_',S1,'x',S2,'_at_',t,'.txt'
599  end if
600
601  OPEN(60,FILE=Oname,STATUS='unkNOWn')
602
603
604  do i=1,S2,1
605    do j=1,S1,1
606
607      Euclid_Distance=NINT(sqrt(REAL((j-center(1))**2+(i-center(2))**2)))
608
609      if(Euclid_Distance<INT(REAL(MIN(S1,S2)/2.))) then
610      if(Euclid_Distance>0) then
611
612        radius(Euclid_Distance,1)=Euclid_Distance
```

```
613         radius(Euclid_Distance,2)=radius(Euclid_Distance,2)+ANINT(REAL(Flags(j,i)
                /10.))
614         radius(Euclid_Distance,3)=radius(Euclid_Distance,3)+1.
615
616       end if
617
618
619
620         Angle=ANINT(atan2(REAL(j-center(1)),REAL(i-center(2)))*180./Pi)+180.
621         if(Angle.eq.0) Angle=360
622
623           theta(Angle,1)=Angle
624           theta(Angle,2)=theta(Angle,2)+ANINT(REAL(Flags(j,i)/10.))
625           theta(Angle,3)=theta(Angle,3)+1.
626           theta2(Angle,1)=Angle
627                     if(Flags(j,i).eq.5) theta2(Angle,2)=theta2(Angle,2)+1.
628                     if(Flags(j,i).eq.6) theta2(Angle,2)=theta2(Angle,2)-1.
629                     theta2(Angle,3)=theta2(Angle,3)+1.
630
631         !write(*,*) j,i,Euclid_Distance, Angle
632       end if
633
634     end do
635   end do
636
637
638   do k=1,INT(REAL(MIN(S1,S2)/2.))-1,1
639     if(radius(k,3).ne.0) radius(k,2)=radius(k,2)/radius(k,3)
640     write(20,*) INT(radius(k,1)),'  ',radius(k,2)
641   end do
642
643   do k=1,360,1
644     if(theta(k,3).ne.0) theta(k,2)=theta(k,2)/theta(k,3)
645     write(30,*) INT(theta(k,1)),' ',theta(k,2)
646   end do
647
648   do k=1,360,1
649     if(theta2(k,3).ne.0) theta2(k,2)=theta2(k,2)/theta2(k,3)
```

```
650     write(40,*) INT(theta2(k,1)),'    ',theta2(k,2)
651 end do
652
653
654
655 do rr=1,rr_max,1
656
657 do i=rr_max,S2-rr_max,1
658    do j=rr_max,S1-rr_max,1
659
660    if(Flags(j,i)>=5) then
661
662
663    do ii=i-rr,i+rr,1
664       do jj=j-rr,j+rr,1
665
666            if((jj>1).and.(jj<S1).and.(ii>1).and.(ii<S2)) then
667               if(NINT(sqrt(REAL((jj-j)**2+(ii-i)**2))).le.rr) then
668
669               !if(Flags(jj,ii)>=5) then
670                        Angle=ANINT(atan2(REAL(jj-j),REAL(ii-i))*180./Pi)+180.
671               if(Angle.eq.0) Angle=360
672
673               theta3(Angle,rr,1)=Angle
674               theta3(Angle,rr,2)=theta3(Angle,rr,2)+ANINT(REAL(Flags(jj,ii)/10.))
675               theta3(Angle,rr,3)=theta3(Angle,rr,3)+1.
676
677               if(Flags(jj,ii)>=5) then
678                 if(Flags(j,i).eq.Flags(jj,ii)) theta3(Angle,rr,4)=theta3(Angle,rr
                      ,4)+1.
679                 if(Flags(j,i).ne.Flags(jj,ii)) theta3(Angle,rr,4)=theta3(Angle,rr
                      ,4)-1.
680                 theta3(Angle,rr,5)=theta3(Angle,rr,5)+1.
681               end if
682
683             !end if
684          end if
685        end if
```

```
686
687       end do
688     end do
689
690     end if
691
692     end do
693   end do
694
695   write(*,*) 'Computing Radius ',rr
696
697   end do
698
699
700   do rr=1,rr_max,1
701     do k=1,360,1
702       if(theta3(k,rr,3).ne.0) theta3(k,rr,2)=theta3(k,rr,2)/theta3(k,rr,3)
703       if(theta3(k,rr,5).ne.0) theta3(k,rr,4)=theta3(k,rr,4)/theta3(k,rr,5)
704     end do
705   end do
706
707   write(50,*) (k,k=0,rr_max)
708
709   do k=1,360,1
710     write(50,*) k,(theta3(k,rr,2), rr=1,rr_max)
711   end do
712
713   write(60,*) (k,k=0,rr_max)
714
715   do k=1,360,1
716     write(60,*) k,(theta3(k,rr,4), rr=1,rr_max)
717   end do
718
719
720   Flush(20)
721   Close(20)
722   Flush(30)
723   Close(30)
```

```
724 Flush(40)
725 Close(40)
726 Flush(50)
727 Close(50)
728 Flush(60)
729 Close(60)
730
731 End Subroutine ComputeCorrelations
732
733
734 END MODULE
```