The Evolution of Cellular Structures via Curvature Flow

Emanuel A. Lazar

A Dissertation Presented to the Faculty of Princeton University in Candidacy for the Degree of Doctor of Philosophy

Recommended for Acceptance by the Program in Applied and Computational Mathematics Advisers: Robert D. MacPherson and David J. Srolovitz

November 2011

© Copyright by Emanuel A. Lazar, 2011. All Rights Reserved

Abstract

This dissertation explores cellular structures that evolve over time, primarily through curvature flow. This models *coarsening* in isotropic polycrystalline materials, an energy-minimizing process in which small cells gradually disappear and the average cell size increases.

Chapter 1 is an informal introduction and is accessible to general readers without a particular mathematical or scientific background. In it we introduce cell structures, curvature flow, steady states, and universal steady states. We use many examples to motivate questions we consider in later chapters.

Chapter 2 investigates one-dimensional cell structures that evolve through a variety of evolution equations. We show that many cell structures evolve towards universal steady states that depend only on the evolution equations and that are generally independent of initial conditions. Chapter 3 considers two-dimensional systems that evolve via curvature flow. We describe a simulation method, present analysis of its numerical accuracy, and provide a large set of results from simulations. Chapter 4 investigates three-dimensional systems that evolve via mean curvature flow. We describe a simulation method, present analysis of its numerical accuracy, and provide large amounts of data from simulations. We also introduce methods of characterizing the combinatorial structure of individual cells.

Chapter 5 compares and contrasts the curvature-flow evolution of two- and three-dimensional cell structures; we also consider properties of two-dimensional cross-sections of three-dimensional structures which have evolved through curvature flow. We conclude in Chapter 6 with a list of directions for further exploration. Appendix A describes a linear measure called the mean width, which plays a crucial role in describing systems which evolve through mean curvature flow. Appendices B and C describe technical details of the two- and three-dimensional simulation methods.

Acknowledgements

First thanks are owed to my advisers, Robert D. MacPherson and David J. Srolovitz. The opportunity to work with both of them has been the most positive and formative experience of my time in Princeton. Their talents are surpassed only by their modesty.

I owe a tremendous debt of gratitude to Jeremy Mason. Our many long discussions over the last two years have added to the work presented hear immeasurably. I would also like to thank Richard Ehrenborg, Frank Lutz, and Aaron Koolyk for many fruitful discussions and Chanoch Goldfeder for crucial assistance in developing some of the two-dimensional code.

Princeton University and its Program in Applied and Computational Mathematics have been wonderfully hospitable over the last five years. I would especially like to thank Ingrid Daubechies, who was exceptionally supportive during my first years in Princeton. I am grateful to the U.S. Department of Defense for their generous support through their NDSEG Fellowship. I would also like to thank the Institute for Advanced Study in Princeton, NJ for their hospitality during the 2010-2011 academic year and A*STAR (Singapore) for their hospitality for two weeks during the summer of 2010.

Last, none of this could have been possible without the constant love, support, and encouragement of my parents, my siblings, my friends, my teachers, and, in the last two years, Abby.

To my family, my friends, my teachers, and to Abby.

Contents

	Abs	tract	iii
	Ack	nowledgements	iv
1	Info	ormal Introduction	1
	1.1	Cellular structures	2
	1.2	Dynamical cell structures	5
	1.3	Curvature and curvature flow	6
	1.4	Curvature flow on cell structures	14
	1.5	The "shape" of a cell structure	16
	1.6	Steady states and universal steady states	20
2	1D	Systems	21
	2.1	Characterization	22
	2.2	Initial conditions	23
	2.3	Dynamics	28
	2.4	Simulation method	30
	2.5	Error analysis	31
	2.6	Results	33
	2.7	Conclusions	49
3	2D	Systems	52
	3.1	Motivation	52
	3.2	Characterization	54
	3.3	Curvature flow on 2D cell structures	57
	3.4	Simulation method	63
	3.5	Error analysis	71

	3.6	Results	74
	3.7	Conclusions	80
4	3D	Systems	82
	4.1	Motivation	82
	4.2	Mean curvature flow on 3D cell structures	84
	4.3	Simulation method	88
		4.3.1 Previous models	88
		4.3.2 Initial conditions	90
		4.3.3 Data Structure and Algorithm	91
		4.3.4 Error analysis	95
		4.3.5 Topological changes	101
		4.3.6 Technical details of the simulation	103
	4.4	Results	108
	4.5	Conclusions	130
5	Cor	nparing 2D and 3D Systems	133
	5.1	Point quantities	136
	5.2	Quantities correlated over metric distance	146
	5.3	Quantities correlated over topological distance	148
	5.4	Conclusions	151
6	Сот	nclusions and Future Work	152
A	Ske	tch of Mean Width	155
	A.1	Sizes, Dimensions, and Measures	155
	A.2	Steiner's Formula	156
	A.3	Measures and their Properties	158
	A.4	Integrating Mean Curvature	159
	A.5	Integrating Euler Widths	161
	A.6	Numerical Evaluation of the Mean Width	164
в	Sin	nulation Details for 2D	166
С	Sim	nulation Details for 3D	171

Chapter 1

Informal Introduction

This thesis focuses on some problems that arise while studying the mathematics and physics of many common materials. If you look at a piece of metal under a microscope, you might be surprised to find that its structure looks remarkably similar to that of a foam that might fill your sink after washing dishes. In most common metals, atoms arrange themselves into very orderly groups called



Figure 1.1: (a) A piece of stainless steel under 200X magnification; black lines separate distinct crystals [1]. (b) A simple two-dimensional foam made using water, soap, and red food coloring [2].

crystals, many of which pack together to form a larger piece of metal. A sample of stainless steel and its constituent crystals can be seen in Figure 1.1(a). A typical foam is comprised of individual bubbles, each of which is a small pocket of air, surrounded by a thin film of soap. Figure 1.1(b) shows a small foam made from water, soap, and some food coloring.

Although the two examples differ in many important respects, they share two important features. First, both exhibit a cellular structure, in which a larger sample is divided into many smaller pieces, or cells. Cells in each of these systems can be seen in Figure 1.1. A second common feature of metals and foams is the way they evolve over time. Both systems change in a process called coarsening, though at very different rates. In **coarsening**, some cells grow while others shrink and disappear. As more cells disappear, the average size of the remaining cells steadily increases.

Coarsening can be observed in a very simple, table-top experiment. Take a clear, empty bottle and pour in a little water and soap. Shake the bottle for a few seconds until the bottle is filled with a white soapy foam. Now let the bottle sit. After a few hours, the foam changes dramatically. Differences in pressure between neighboring bubbles leads to diffusion of air through the soap films that separate the bubbles from each other. Large bubbles grow and smaller ones disappear, and the average size of a bubble increases. Eventually, only one bubble will remain, though this might take a *very* long time.

Metals evolve via a very similar process. At room temperatures, this process generally occurs at rates much too slow to observe over a short period of hours or days, or even months or years. With additional heat, this process can be significantly sped up, though samples are generally too hot to directly observe at these very high temperatures. In coarsening, thermodynamics causes atoms to bounce across a plane that separates neighboring crystals, and eventually causes larger crystals to grow and smaller ones to shrink and disappear. Like the bubbles, the average size of a crystal steadily increases with time. Aside from affecting the average crystal size, coarsening also changes the microstructure of a metal in other important ways. Because the microstructure of a metal impacts many of its physical properties, improving our understanding of how coarsening affects changes the microstructure of a metal is an important goal in studying the science of materials. This chapter provides a general picture of cell structures, curvature, and the role that curvature plays in coarsening metals and foams. The end of this chapter also describes what occurs to cell structures that evolve for a long period of time, developing the idea of a steady state and a universal steady state. It is intended to be almost entirely accessible to educated readers without a particular mathematical or scientific background.

1.1 Cellular structures

Roughly speaking, a **cellular structure** is a large space divided into many smaller pieces that we call **cells**. In a cellular structure, or cell structure for short, no two cells overlap, and together the cells fill up the entire space. As a simple example, consider the geographical landmass that constitutes continental Europe, Asia, and Africa. Figure 1.2 shows a crude early sixteenth century map that



illustrates a possible breaking up of this space into smaller pieces [3]. Ignoring the accuracy of the

Figure 1.2: A map [3].

map, we can ask many simple questions about this particular cell structure: On average, how many neighbors does each country have? If a country has many neighbors, will its neighboring countries also have many neighbors? The answer to the first question is surprisingly simple: on average, countries have roughly six neighbors. This beautiful result stems from a theorem in topology and a few facts about the energy associated with borders between neighboring countries. The answer to the second question is much more complicated and has evaded many attempts to completely solve it (for more about both questions, see Section 3.2).

We can also ask about geometric aspects of the map's cell structure: Do countries tend be similarly sized, or do we have a few very large countries and many very small ones? What is the relationship between a country's land area and the total length of its borders? Is there a correlation between a country's land area and its number of neighbors? Do large countries with many neighbors tend to border other large countries?

These questions are certainly not limited to studying large geographical regions, but can certainly be asked about all sorts of cellular structures. Indeed, it is these questions, and many more like them, motivate much of our discussion in this thesis.

To get a better sense of the ubiquity of cell structures in nature, we consider four very different examples found in nature. Figure 1.3(a) shows a tissue sample from the *Plagiomnium affine* plant, a moss native to humid climates. If we ignore the green circles inside the cells (the chloroplasts), we can plainly see the hexagonal cell structure of the sample. Figure 1.3(b) shows a closeup of broken piece of glazed ceramic, seen on a door of an old building in Safed, Israel. Figures 1.3(c) shows a sample of a zinc oxide ceramic magnified roughly 200 times. Figure 1.3(d) shows a foam sample composed of bubbles made from soap and water. These four examples help illustrate some



Figure 1.3: Four examples of cellular structures that are roughly two-dimensional.

of the features that cell structures share in common, as well as much of the diversity that can exist. Cell structures can appear relatively ordered or disordered, their boundaries can appear straight or curved, smooth or crooked. Cell structures exist on scales we can observe with our eyes, and they can exist on scales only visible through a microscope. And although each of these cell structures have very different properties, all of the questions we asked before about the map we can ask about these as well.

Until now, all examples we have seen were two-dimensional in nature. However, we can also consider three-dimensional cell structures. Figure 1.4(a) shows a standard three-dimensional soap-



Figure 1.4: A simple three-dimensional foam made using water and soap [8].

and-water foam, whose structure is decidedly three-dimensional. Although the basic definition of the three-dimensional structure is very similar to the two-dimensional one, we will see in later chapters that the move from two to three dimensions significantly complicates and enriches this subject.

1.2 Dynamical cell structures

Section 1.1 dealt mostly with static cell structures. However, we can also consider features of cell structures that change over time. The frames in Figure 1.5 show results of a computer model of a simple cell structure that evolves over time. This particular system, about which we will say more later, evolves in a way that slowly decreases the total length of the cell boundaries. While doing this, the computer slowly shrinks small cells until they disappear, usually as slightly rounded triangles. As more of the cells shrink, the average size of a cell grows, as can be readily observed.

The bubbles in many foams behave like this: small cells shrink, large cells grow, and over time the average size of a bubble increases. Another important and similar dynamical system is found in common metals and in many ceramics. Figure 1.3(c) shows a two-dimensional cross section of a piece of a zinc oxide polycrystal, a ceramic, at room temperature. When left alone, polycrystalline structures evolve in a way that decreases the energy associated with their cell boundaries. Smaller cells slowly disappear and over time the average cell volume steadily increases. Because many of a material's properties depend on the average size of its cells, their orientations, and other microstructural properties, understanding how these systems evolve, and understanding what these cell structures look like, is crucial to a complete understanding of these materials.



Figure 1.5: A cell structure evolves through a coarsening process, in which small cells disappear and the average cell size increases. By the final frame, almost two thirds of the cells have disappeared; the average cell area has thus almost tripled.

1.3 Curvature and curvature flow

Aside from cell structures, an important part of our work concerns curvature and curvature flow. The general term curvature describes the way in which something deviates from being flat. We might say that a straight line has zero curvature, or that the surface of a chalkboard has zero curvature. Both of these objects are flat in a very intuitive sense. On the other hand, a circle or a sphere are not straight or flat. In mathematics we have a very good way of quantifying this which lets us formulate precise statements about how objects are curved. Understanding how an object is curved can tell us much about its other properties. We will first explain how we quantify curvature in two and three dimensions, and then explain what it means for curvature to "flow".

Two dimensions

Consider a curve drawn in the plane such as that illustrated in Figure 1.6. Looking at the shape briefly, we can agree that certain parts seem more "curved" than others. If we were driving on a



Figure 1.6: A simple, closed curve, adapted from a drawing in [9].

road shaped like that curve, then at certain points in our drive we would need to turn the steering wheel a little bit to the left, sometimes we might need to turn it very sharply to the right, and sometimes we would be able to coast straight ahead without turning the wheel at all. Intuitively, the ways in which we need to turn the steering wheel tells us something about the way in which the road itself is curved. How can we quantify that?

The following is one way that is often used in mathematics to quantify curvature. Let us consider a particular point along the curve. Figure 1.7 shows the same curve with a particular point chosen.



Figure 1.7: A simple, closed curve, with a point highlighted.

We draw a red dot to make that point stand out. We can draw a circle that runs through the point and which is tangent to the curve near that point. This circle best approximates the curve near that point. Figure 1.8 shows the curve, the highlighted point, and a circle that runs through the point and which is tangent to the curve near that point. In a similar manner, we can repeat the same exercise for *any* point along the curve. Figure 1.9 shows a number of points highlighted with tangent circles drawn. Each of these tangent circles says something about the way in which the curve bends at those points—the smaller the circle, the sharper the turn at that point and the more curved the road is at that point. Very large circles indicate points at which the road is not sharply curved. All



Figure 1.8: A simple, closed curve, with a point highlighted and a tangent circle drawn.



Figure 1.9: A simple, closed curve, with points highlighted and tangent circles drawn.

of this leads us to define curvature in terms of these circles as follows. At each point along the curve, we measure the radius of the associated tangent circle and call that **the radius of curvature** r at that point. Figure 1.10 shows the curve, along with tangent circles and their respective radii labeled.



Figure 1.10: A simple, closed curve, with points highlighted, and tangent circles and radii drawn.

We then define the **curvature** at a point to be the reciprocal of the radius of curvature. We use the Greek letter κ to refer to the curvature, and so we have $\kappa = \frac{1}{r}$. If the curve is completely flat near a given point, then in theory we would need a circle infinitely large $(r = \infty)$ to approximate the curve at the point. Because the reciprocal of ∞ is not defined, we define the curvature to be 0 at those points. This matches our intuition in which flat pieces of space have no curvature. A straight line would have zero curvature at all points. A circle with radius r would have the same curvature, $\kappa = \frac{1}{r}$, at all points. Oftentimes we will want to distinguish between points whose tangent circles lie "inside" the curve and those that lie "outside". One way of "orienting" the curvatures is by saying that points whose tangent circles lie inside the curve have positive curvature, and points whose tangent curves lie outside the shape have negative curvature. These might correspond to turning the steering wheel left or right while driving down a road. When considering curvature in this oriented manner, we use the letter k to refer to the signed curvature. Therefore we have $k = \kappa$ or $k = -\kappa$ depending on which way the shape is curved. We continue to use κ when referring to the unsigned curvature.

Three dimensions

This is a rough picture of curvature in two dimensions. What happens when we consider objects in three dimensions. For example, how do we talk about curvature when we consider something like a chalkboard or a sphere? Fortunately, we can still use techniques that are similar to those we have used until now. Figure 1.11 shows a saddle-shaped surface. Here we choose a specific point



Figure 1.11: A saddle shaped surface, adapted from [10].

in the middle of the surface, where all three dotted lines appear to intersect. We now draw some planes that intersect the surface at the given point and that are perpendicular to the surface at that point. Figure 1.12 shows two such perpendicular planes. If we consider the planes and the intersection of the surface with those planes, then we can recast our three-dimensional problem as a family of two dimensional ones. Figure 1.13 shows the two planes drawn flat, and the curves where the planes intersect with the saddle-surface drawn in yellow. In these cases we can again draw



Figure 1.12: A saddle shaped surface with some perpendicular planes drawn in, adapted from [10].

tangent circles and calculate the curvature for those planes. Every intersecting plane will have an associated curvature. In the two planes drawn, the two yellow intersection curves have the same curvature but with opposite sign. We use k_1 to refer to the minimum of all signed curvatures of these intersecting curves, and k_2 to refer the maximum. These are known as the **principal radii** of curvature. There are different ways of using the two principal radii of curvature to give us a



Figure 1.13: Two planes perpendicular to the saddle surface, with intersection curves drawn.

single number to describe how a surface is curved at a point. The product of the principal radii of curvature, $k_1 \times k_2$, is known as the Gaussian curvature and it has important meaning in many areas of mathematics. For example, the famous Gauss-Bonet theorem tells us that by adding up all of

this curvature over an entire surface, we can figure out how many holes that surface has! For many applications, this Gaussian curvature, an *intrinsic* property of the surface, is what matters most.

For some problems, though, the more relevant quantity is the sum of the two principal curvatures. We call this quantity the **mean curvature** and use the letter H to refer to it, so $H = k_1 + k_2$. This quantity tells us something about how the volume enclosed by a surface will change if we deform its surface a little bit. This will eventually be very important in understanding how the volume of a cell changes when its boundaries move through curvature flow.

Although quite rough, this description of curvature is enough to allow us to understand what it means for curvature to "flow". Understanding how curvature can flow can tell us much about the nature of the surface itself.

Curvature flow

Consider the simple, closed, smooth curve shown before in Figure 1.6 and shown again here in Figure 1.14. In theory, we can calculate the curvature at every point along this curve in the manner



Figure 1.14: A simple, closed curve, adapted from a drawing in [9].

described above. Using that information, we can also calculate a curvature vector at every point along the curve. A **curvature vector** is an arrow that points towards the center of the associated tangent circle and has a length proportional to the curvature at that point. Figure 1.15 shows our curve with a few curvature vectors drawn in. The idea now is to change the curve a little bit by moving every point on the curve in the direction of its curvature vector and by an amount proportional to its length. As time goes by, the trail continuously changes shape. Figure 1.16 shows what the curve would look like as time passes; the curve becomes rounder and rounder, until it turns into a circle and disappears. In fact, a beautiful result reported in [11, 9] says that *any* simple, smooth curve that evolves by curvature flow will always stay smooth, becoming rounder and rounder



Figure 1.15: A curve with some curvature arrows drawn.

before it disappears as a perfectly round circle! No two parts of the curve will ever cross, the curve

Figure 1.16: A smooth curve embedded in the plane evolves via curvature flow. The area bounded by the curve decreases at a constant rate, and the curve becomes progressively more circular; the curve eventually disappears in finite time.

will *never* develop any sort of pinch, and the curve will *never* break apart into two separate curves. This is one beautiful result that tells us something about the *global* evolution of the curve based on limited knowledge of the initial path and of a simple *local* rule of motion.

The story is more complicated in three dimensions and the results are not nearly as nice. In three dimensions, we define a *mean curvature vector* at every point along a surface, which is an arrow that points in a direction perpendicular to the surface, and whose signed magnitude is the mean curvature H, or the sum of the two principal curvatures, as explained above. At every point in time, each point on the surface moves with the velocity given by the mean curvature vector. This is called mean curvature flow.

In mean curvature flow, convex objects eventually become more and more round and disappear as spheres [12]. However, objects that are not as round can develop serious problems. A simple example of a nice surface that does *not* disappear into a single sphere is provided in [13] and shown in a series of frames in Figure 1.17. The object starts as a relatively smooth dumbbell-shaped surface,



Figure 1.17: A dumbbell-shaped surface splits into two before disappearing. Images are taken from the website of Uwe F. Mayer [14].

and evolves through a curvature flow similar to the one explained above. Before it disappears, the object rips apart in the middle. After this point, the remaining pieces seem to disappear as nice,

round spheres.

1.4 Curvature flow on cell structures

The primary motivation of our work is understanding the evolution of polycrystalline metals, which involves a combination of cell structures and curvature flow. Most ordinary metals are polycrystalline in nature, which means that they are composed of many individual crystals packed together, as seen before and shown again in Figure 1.18. This is an ordinary piece of stainless steel, though an



Figure 1.18: A piece of polycrystalline stainless steel under 200X magnification. Black lines show boundaries between neighboring crystals. This sample has been prepared by polishing the surface of an ordinary piece of stainless steel [1].

ordinary piece of aluminum, bronze, or silver will exhibit a very similar pattern when observed under similar magnification. If we look at the sample from even closer range, we will see that each of the crystals is a large collection of atoms arranged in very orderly patterns. Figure 1.19 shows a simulated sample of a two-dimensional polycrystalline material in which both the individual atoms, as well as the larger crystals, can be made out. Looking carefully, one can make out individual crystals, each of which contain many atoms arranged in a very orderly fashion. While all crystals in a particular sample will have the same patterned structure, a triangular lattice in this case, each of the crystals will be oriented slightly differently; this might be slightly harder to see, but can be done with some practice. Figure 1.20 shows the same simulated sample as before, though this time the atoms are colored by the orientation of the crystal. Deciding which atom is part of which crystal is not always easy, nor is there always a "right" answer. Sometimes there is no fact about whether a particular atom should be considered part of one crystal or another, or even whether two groups of



Figure 1.19: A simulated piece of a two-dimensional polycrystalline material. Atoms are drawn as black dots. Notice the triangular patterns in which most of the "atoms" are arranged [15].



Figure 1.20: A simulated piece of a two-dimensional polycrystalline material, with atoms colored by the orientation of the crystal [15].

atoms should be considered one crystal or two. For example, we might consider joining the green and purple crystals into one. Although the exact patterns that form in two- and three-dimensional crystals will vary from material to material, all polycrystalline materials will show some sort of orderly patterns at the atomic scale.

At this point we can better understand the patterns we see in Figure 1.18. The individual crystals we see there are large collections of atoms, all of which are arranged in very orderly patterns. The black lines in between them are crystal boundaries. They correspond to the atoms in Figure 1.20 that either lie on a crystal's boundary, or else belong to no crystal at all and are thus colored gray.

This description of polycrystalline materials allows us to understand the role that curvature

plays in the evolution of cell structures like metals. We must begin by pointing out that even at very low temperatures, atoms are always in motion and never sit perfectly still. Most of the time, atoms vibrate around a central location. Figure 1.20 thus shows the "average location" of an atom near some point in time. However, every so often an atom has enough energy to move to a new location. Atoms along crystal boundaries will periodically collect enough energy to allow them to "jump" from one crystal to another. For example, look at some of the uncolored atoms that lie between the blue and brown crystals. Notice how a small push will allow these atoms to easily become part of either the brown or blue crystal. Notice also that these atoms tend to have more brown neighbors than blue neighbors, resulting from the way in which the boundary is curved. This imbalance will cause the boundary atoms to preferentially jump towards the brown crystal, thus causing the boundary between the blue and brown crystals to gradually move. Left alone, the blue crystal here will eventually dissolve into the brown crystal. This is one example of what is known as **curvature driven motion**. This process will eventually stop when either there remains only one crystal left in the system, and hence no boundaries, or else all boundaries are flat, and there is no imbalance pushing the boundary in one direction or the other.

1.5 The "shape" of a cell structure

One of the most exciting phenomena we explore in this thesis arises when cell structures evolve under curvature flow for a long period of time. Aside from coarsening, cell structures gradually relax and take on a fixed "shape" that stays constant throughout the remainder of the evolution. What is particularly interesting is that *almost all* cell structures, if allowed to evolve for a long enough period of time, will reach the *same* long-time shape, regardless of what the structure looked like in the beginning. Before explaining this phenomenon in more detail, we need to explain what we mean by the "shape" of a cell structure.

The top row of Figure 1.21 shows three cell structures on the unit square; the bottom row shows three "relaxed" cell structures. For now we ignore details of how these particular structures are generated, but point out that each structure in the bottom row is obtained from the structure directly above it by deforming its vertices and edges in a very controlled way. For example, we are careful to never break apart edges from vertices or join new ones together. Therefore, if a cell in the top structure has n sides, then it will still have n sides after we deform the vertices and edges, and if two cells are touching then they will continue to touch. Moreover, deforming the edges is done in such a way that preserves the area of each cell. In some ways, then, each cell structure in the top



Figure 1.21: The first row shows three cell structures in the unit square. The second row shows three "relaxed" cell structures in the unit square. Structures in the first column contain 100 cells, those in the second column contain 200 cells, those in the third column contain 300 cells.

row is very similar to the structure directly below it.

However, in some intuitive sense the cell structures in the top row share more in common with each other than with the structures directly below them. In some intuitive sense, the structures in the top row have one "shape" and those in the bottom row have a different one. We would like to make this more quantitative.

Historically, highly ordered cell structures have received the lion's share of attention from the mathematical and scientific communities. Theoretical tools such as group theory and algebra, and experimental ones such as X-ray diffraction, allow us to completely describe all sorts of very ordered cell structures. Disordered cell structures, on the other hand, such as those considered here, have received considerably less attention and fewer tools, both theoretical and experimental, are available to help us understand them.

Figure 1.22 shows four different cell structures which we have considered previously. We might ask, how can we describe the "shape" of each cell structure? What does it mean to say that the "shape" of one cell structure is different from that of another?

A first attempt at describing the shape of a cell structure involves recording a number of its basic quantitative properties. We first consider how many sides each of its cells has. We notice that all cells in Figure 1.22(a), for example, have exactly six sides. Figures 1.22(b), 1.22(c), and 1.22(d),



Figure 1.22: Four examples of cellular structures that are roughly two-dimensional.

however, contain many cells with fewer or more than six sides. One way, then, in which we can quantitatively describe a cell structure is by recording what fraction of its cells have three sides, four sides, and so forth.

While this quantitative description of cell structures is certainly not limited to two-dimensional ones, it is much more complicated in three dimensions. Consider for example the two cells illustrated in Figure 1.23. Although both cells have exactly five faces, the two cells are quite different. The one on the left has one rectangular face and four triangular faces; the cell on the right has three rectangular faces and two triangular ones. As the number of faces increases, the number of possible cells with that number of faces increases very quickly. As a simple example of how quickly this number grows, we note that there are over a thousand different cell with 11 faces, and over two million with 15 faces. A complete description of a cell in three-dimensional structures therefore requires substantially more information than just its number of faces, which *is* sufficient in describing



Figure 1.23: Two cells with 5 faces, a square pyramid and a prism over a triangle.

a two-dimensional cell.

A second way in which we can describe a cell structure involves recording the distribution of cell sizes. For example, all cells in Figure 1.22(a) have roughly the same area. Figure 1.22(d), on the other hand, has a number of very small cells. The cells in Figures 1.22(b) and 1.22(c) are not all the same size, but we also do not find many cells that are much smaller than the average. A second way, then, in which we can quantitatively describe a cell structure is by recording what fraction of its cells are very small or very large when compared to the rest of the cells in the structure.

However, even both of these descriptions fail to distinguish between some cell structures whose "shapes" appear quite different. For example, we have already noted that the distribution of sides per cell and distribution of cell areas of structures in the top row of Figure 1.21 are identical to those of structures in the bottom row. However, the structures in the top row look markedly different from those in the bottom row. Another property of a structure we therefore consider is how "round" are its cells. One quick look back at Figure 1.21 shows that the cell structures in the top row contain many very long and narrow cells, whereas almost all cells in the structures in the bottom row are quite round.

In the chapters that follow, we will use a number of statistics including the distributions of cell sides, sizes, and roundnesses to help characterize the *shape* of a cell structure. This rough definition of shape will allow us to compare and contrast cell structures with one another. Two cell structures have the same (or similar) shape if their distribution of cell sides, sizes, roundnesses and other properties are the same (or similar), and have different shapes if these distributions are different. We can now use this very rough notion of shape and similarity to better understand steady states and universal steady states of cell structures that evolve under curvature flow.

1.6 Steady states and universal steady states

What is particularly amazing about curvature flow on cell structures is the way in which it changes the shape of a cell structure. When a cell structure is allowed to evolve, its shape, as explained in the previous section, often changes. The period of time during which the shape of an evolving structure changes is called the *transient period*. What is very surprising is that in almost all cases this period is limited, and the shape of a cell structure eventually *stops* changing, even as the structure continues to evolve and coarsen. That is, after a certain point in time, although cells continue to disappear and the structure itself continues to change, the *shape* of the cell structure remains the same. When the shape of a cell structure eventually stops changing and remains constant with time, we call that shape a *steady state*. The term steady state refers primarily to a particular shape, though we occasionally use it to refer to the period of time in which the cell structure has this particular shape. It is certainly not obvious that all evolving cell structures will reach a steady state, but all those we consider do. This is the first extremely fascinating feature we observe in evolving cell structures.

A second incredibly fascinating feature we observe involves a type of universality principle. We find that not only do evolving cell structures always reach steady states, but almost all of them always reach the *same* steady state! This means that we can take any two initial cell structures, whose shapes are as different as possible, and if we allow both of them to evolve under curvature flow for enough time, then eventually the two structures will take on the same shape! We call this shape, which is a particular example of a steady state, the *universal steady state* of curvature flow.

In the chapter that immediately follows we consider one-dimensional dynamical cell structures. In one dimension, there is no good notion of curvature and instead we consider other ways in which one-dimensional cell structures can evolve. We find that each method of evolving a system has its own universal steady state. To illustrate the universality of these steady states, we use different initial shapes and watch as each of them gradually evolves to the same, universal steady state.

In Chapters 3 and 4 we consider two- and three-dimensional cell structures that evolve via curvature and mean curvature flow, respectively. In each dimension, we consider only one particular initial shape. Although this does not allow us to observe the universality of the steady state, we do observe the gradual evolution of systems towards steady states. We spend parts of those chapters and the majority of Chapter 5 describing the shape of the universal steady state of two-dimensional and three-dimensional cell structures that evolve via curvature flow.

Chapter 2

1D Systems

The simplest cell structures we consider are those embedded in one-dimensional spaces such as the real line or the unit circle. The limitation of such low-dimensional spaces is that they do not allow us to study curvature-driven motion, since there is no obvious way to define curvature in this dimension. However, the upside is that in this dimension, the geometry and topology of cell structures are considerably simpler than in higher dimensions, which allows us to study a broad range of initial conditions and coarsening dynamics. Studying dynamical cell structures in one dimension provides us with insight into dynamical cell structures in general, which will help us in our quest to understand the evolution of cell structures through curvature flow.

This chapter illustrates the steady state and the universal steady state phenomena described towards the end of the last chapter. There we stated that evolving cell structures tend to pass from a transient state, in which many of its statistical properties are changing, to a *steady state*, in which these statistical characterizations of the cell structure remain constant with time. We also noted a universality feature of these steady states. It is almost always the case that if two cell structures evolve under the same rules of evolution, such as curvature or mean curvature flow, then the two cell structures will both reach the *same* steady state. We call this steady state the *universal stead state*.

In this chapter we consider four different initial conditions and four different dynamics that act on these systems. We find that each dynamic, or method of evolving a system, has its own universal steady state to which it eventually brings all initial cell structures.

2.1 Characterization

We consider one-dimensional dynamical cell systems on spaces such as the real line \mathbb{R} or the circle \mathbb{S}^1 . Cells are intervals on these spaces and cell boundaries are simply the 0-dimensional boundary points where two adjacent cells meet.

A simple example of a 1-dimensional cell structure is provided in Figure 2.1. In order to help visualize the system, we have drawn small bars to indicate points at which two cells meet. Both

·····

Figure 2.1: A one-dimensional cell structure.

the geometry and topology of these systems are fully described by these points, and in this sense their topology and geometry are trivial. And while initially these systems may not seem particularly interesting, we will see that dynamical systems built on them can display amazingly complex and beautiful trajectories and properties. Studying 1-dimensional systems can also shed light on some general properties of dynamical cell systems.

Before considering their evolution, we must address the question of characterizing static onedimensional cell structures. Figure 2.2 illustrates four different cell structures, each with 100 cells. Although none of these four examples here are identical to the one in Figure 2.1, some of them look



Figure 2.2: Four different cell structures.

more similar to it than others. Is there a good way to make this more precise? How can we quantify similarity between two different one-dimensional cell structures?

In one dimension, the topology of cells is trivial, and there is nothing interesting to say about the cell boundaries. We therefore consider the distribution of cell sizes. Upon inspecting the different examples, you will notice that some systems have more very small cells than the other examples, or more very large cells. In the third example, all cells appear to be similar in size.

A second property of a cell system we might consider is a correlation function that describes a relationship between neighboring cells: given a cell of a certain size, what is the expected size of its neighbors? It is perhaps difficult to see this from the figure, but in the first example, there is no correlation between the size of a cell and the expected size of its neighboring cells. In the second example there is a positive correlation, and in the fourth example there is a negative correlation.

Although the cell size distribution and correlation function are insufficient by themselves to completely characterize the structure of one-dimensional systems, we leave for another time the problem of determining sufficient criteria for doing so. In what follows, we focus on the cell-size distributions of cell structures, though we will also take time to consider the correlation function, analogues of which will certainly play an important role in describing order in higher dimensional cell structures.

2.2 Initial conditions

In our simulations, we consider four different initial conditions, illustrations of which were provided above. Each initial condition will have a particular distribution of cell sizes and a particular "correlation" function between the sizes of neighboring cells. In the sections following, we will consider how these systems evolve under various dynamics imposed on them.

First a word about notation. We use letters (A, B, C, D) to denote the four initial condition. We use numbers (1, 2, 3, 4) to denote the four dynamics which we impose on the various cell structures. We use pairs of these to refer to specific dynamical systems. For example, C2 refers to a dynamical cell system that has begun in initial state C and that evolves under dynamic 2.

Although in theory we can consider many more initial conditions and many more dynamics, we limit ourselves to four each for reasons of time and computational resources. The four initial conditions are each quite different, and it is surprising that after a long time, each of them will always converge to a steady state that depends only on the dynamic that acts on it. One property which these four initial conditions share is scale- and translation-invariance. That is, neither scaling the entire system by a constant factor nor translating it by any Euclidian motion will change any of the system's properties which we described earlier. We leave for another time the study of dynamical systems that begin in states that do not satisfy these conditions.

Initial State A: Random Distribution with Uniform Density

The first initial condition we consider is obtained by randomly distributing N points on the unit interval with a uniform probability density. The cells are then the intervals between two adjacent points. An example of such a system is drawn at the top of Figure 2.3. One fruitful way of viewing this initial condition is as the result of a homogenous Poisson process with intensity $\lambda = N$. Cells correspond to *waiting times* between two consecutive events. One elementary byproduct of this



Figure 2.3: Initial State A: Random distribution with uniform density. At the top is an example of a structure in Initial State A. Below shows the distribution of cell sizes and the "correlation" function between neighboring cells.

analogy is that the cell sizes, which correspond to waiting times between two consecutive events, are exponentially distributed. This can be seen in a graph of the probability distribution of cells sizes, Figure 2.3. We always normalize the data so that the mean of the distribution is 1 and the probability density function is exactly $P(x) = e^{-x}$.

A second elementary result of the comparison with Poisson processes is the correlation between sizes of adjacent cells. Because waiting times between consecutive events are always independent in a Poisson process, the size of a cell is completely independent of the sizes of its neighbors. This property can be seen by plotting the expected size of neighboring cells as a function of the size of a fixed cell, as shown in Figure 2.3. Notice that the expected value of a neighboring cell is always 1, independent of the size of the known cell.

Initial State B: Voronoi Tessellation

The second initial condition we consider also originates from a random distribution of points on the unit interval. However, instead of using the points as the cell boundaries, we instead use them as seeds for a Voronoi tessellation of the interval. In a *Voronoi tessellation* of a space, a *seed* is just a point in that space. With each seed we associate all points in the space which are closer to that seed than to any other seed. All points in the space associated with a particular seed constitute a single cell. This construction can be done in any dimension, and indeed it is the primary initial

condition we use in two- and three-dimensional systems. An example of such a system is provided in Figure 2.4. This system can be viewed as a smoothing of Initial State A, since each cell in the new structure corresponds to the average of two neighboring cells in the previous structure.



Figure 2.4: Initial State B: Voronoi. At the top is an example of a structure in Initial State B. Below shows the distribution of cell sizes and the "correlation" function between neighboring cells.

In this system, the sizes of cells are distributed differently from the previous example. Because we are "smoothing out" the cells, we no longer have as many very small or very large cells. These difference can be seen already in the systems illustrated in Figure 2.2 above. Although the mean cell size remains unchanged, the distribution is much narrower and the standard deviation is smaller. The exact equation for the probability density function of this distribution is $P(x) = 4xe^{-2x}$. The mode of the distribution is 0.5, where the probability density function peaks at approximately 0.736. The correlation function between adjacent cells is also considerably different. Here there is a positive correlation between the size of a cell and the expected size of its neighboring cells. Both characterizations can be seen in Figure 2.4.

Initial State C: Lightly Perturbed Lattice

A third initial condition we consider is one in which all neighboring points are similarly, but not identically, spaced. If all neighboring points were identically spaced, then all cells would be identical in size, and under all dynamics we consider below, these systems would remain stationary in an unstable equilibrium. To avoid this uninteresting state, we perturb the system by displacing every point by a random value in $(-\epsilon, \epsilon)$. In Initial State C, we set ϵ to one hundredth of the average cell size, or $\frac{1}{100N}$. Before perturbing the system, each cell is identical in size. Since each boundary can move anywhere between $-\epsilon$ and ϵ , each cell (which has two boundary points) can change anywhere between -2ϵ and 2ϵ . If the average cell size is normalized to 1, then the size of a cell must then lie in the interval $(1 - 2\epsilon, 1 + 2\epsilon)$. The cell size distribution function is then:

$$P(x) = \begin{cases} 0 & : x < 1 - 2\epsilon \\ (x - 1 + 2\epsilon)/(4\epsilon^2) & : x \in [1 - 2\epsilon, 1) \\ (1 - x + 2\epsilon)/(4\epsilon^2) & : x \in [1, 1 + 2\epsilon) \\ 0 & : x \ge 1 + 2\epsilon \end{cases}$$
(2.1)

The correlation function is only defined on the interval $(1 - 2\epsilon, 1 + 2\epsilon)$; on that interval, if the size of a cell is x, then the expected size of its neighbor is $\frac{3-x}{2}$. This system turns out to be particularly interesting, exhibiting beautiful behavior in the initial stages of evolution under many dynamics. A graph of the distribution density function can be seen in Figure 2.5.



Figure 2.5: Initial State C: Lightly Perturbed. At the top is an example of a structure in Initial State C. Below shows the distribution of cell sizes and the "correlation" function between neighboring cells.

Initial State D: Heavily Perturbed Lattice

The final initial condition we consider is identical in construction to Initial State C, except that we use a substantially larger perturbation, setting $\epsilon = \frac{1}{2N}$, half the average cell size. The cell-size distribution function is again given by Equation 2.1. This leaves us with a much wider range of possible cell sizes. As was the case for Initial State C, if the size of a cell is x, then the expected size of its neighbors is $\frac{3-x}{2}$. This function is now defined on the much larger domain (0, 2).



Figure 2.6: Initial State D: Heavily Perturbed. At the top is an example of a structure in Initial State D. Below shows the distribution of cell sizes and the "correlation" function between neighboring cells.

Figure 2.6 shows an example of such a system as well as the two characterizations of this system. Although in some sense this system is very similar to Initial State C, we will see later that its evolution is remarkably different.

Other Initial States

As noted before, the four initial conditions described here are not comprehensive and we could in theory consider a much larger set of initial conditions. However, we limit ourselves to these four, which allow us to demonstrate some of the more interesting properties of dynamical cell systems without requiring excessive resources and without requiring copious amounts of information. One set of initial states in particular that might be worth considering and which we leave for another time, are systems which are not spatially homogeneous. By spatially homogeneous, we mean the property that if we look at any contiguous subset of the larger space, then that the properties of that subset will be similar to that of the larger space. Figure 2.7 shows an example of a space that is *not* spatially homogeneous. We might then ask about how quickly particular dynamics bring certain

Figure 2.7: A cell structure that is not spatially homogeneous.

initial conditions into spatially homogenous ones. All cell structures considered in this thesis are spatially homogeneous.

2.3 Dynamics

After choosing an initial condition, we choose a dynamic to control the evolution of the structure. Throughout our work, all dynamics focus on controlling the motion of the cell boundaries; cell interiors evolve as implicit consequences of the evolution of cell boundaries. This is true in all dimensions we consider.

For one-dimensional systems, we choose a set of equations that governs the motion of every boundary point for all times. Unlike in higher dimensional systems, here we do not have a good way to define curvature, and so we do not consider curvature-driven motion. However, we can still associate an energy with a system and can design dynamics under which a system evolves to reduce its total energy. The simplest way to define the energy of a system is by associating a constant energy with every boundary point. The net energy of a system is then proportional to its number of boundary points or, equivalently, its cells. We then allow a system to evolve in various ways that reduce this total energy. When only one cell remains in the system, there are no cell boundaries (recall we are working on S or \mathbb{R}), and the energy of the system vanishes.

As with initial conditions, here too we can choose from many possibilities, though we restrict ourselves to studying four *coarsening* dynamics. In a coarsening dynamic, smaller cells become smaller with time, larger ones grow, and the average cell size monotonically increases with time.

It is not clear whether the four dynamics we consider have any physical meaning, but they certainly tell us much about dynamical cell systems. Each dynamic we consider is defined by an equation of motion that control each boundary point x at each point in time. We use L and R to refer to the lengths of the cells to the left and right, respectively, of the point x.

1. Dynamic 1: $\frac{dx}{dt} = L - R$

This dynamic is relatively simple to compute. The velocity of each boundary point is the difference between the size of the cell on its left and the size of the cell on its right.

2. Dynamic 2: $\frac{dx}{dt} = \ln L - \ln R$

This dynamic is more computation-intensive though interesting. Because $\ln cL - \ln cR = \ln L - \ln R$ for all L, R, c > 0, this equation provides an interesting example of a scale-independent equation of motion.

3. Dynamic 3: $\frac{dx}{dt} = \frac{1}{R} - \frac{1}{L}$

A third dynamic we consider uses the reciprocal of the cell sizes to determine the velocity of

boundary points. In some sense, this is analogous to the dynamics used for curvature-driven growth in two and three dimensions, as described in subsequent chapters.

4. Dynamic 4: $\frac{dx}{dt} = 1$ if L < R and -1 if L > R.

In some sense the last dynamic may initially appear the least interesting. At any point in the time, the velocity of any cell boundary is either -1, 0, or 1. And yet, this dynamic leads to incredibly interesting behavior.

It is easy to see that dx/dt is positive when L > R, negative when L < R, and zero when L = Rfor all of the dynamics considered. In a random finite structure, at almost all times there will be a unique smallest cell. If there is at least one other cell in the system, this cell will necessarily shrink and disappear. Since this is true for all times, the system will evolve until there remains only one cell left in the system. Thus each provides a *coarsening* dynamic for our cell structures.

Both the initial condition and the dynamic effect how long a system will take to evolve to a steady state. This in turn affects how large the initial system must be to obtain accurate statistics from the long-time steady state. We often found that if we began a system with too few cells in an initial structure, the system had completely coarsened before it had reached a steady state. We therefore begin each simulation with between 1 and 250 million cells, depending on these factors. Moreover, some dynamics take considerably longer to compute than others, which further limits the size of the simulations.

As with the initial states, we could have considered many other dynamics to place on the structures. There are certainly others that lead to interesting phenomena. Time and resources limited us to study a few examples. In future work, we might consider systems in which a stochastic variable plays a role in the evolution of a system; in this work, we only considered deterministic evolutions.

A few dynamics that we considered that were not particularly interesting were $dx/dt = e^L - e^R$ and $dx/dt = \sin L - \sin R$. These particular examples did not seem terribly interesting given that for small $x, e^x \approx x$ and $\sin x \approx x$, but we did not study these cases more fully. We also considered the dynamic $dx/dt = L^2 - R^2$, which produces interesting dynamics but which we could not study completely because it would always finish coarsening before reaching a steady state. More details about this dynamic are presented at the end of the chapter.
2.4 Simulation method

We use the unit interval [0,1] in which 0 and 1 are identified as the underlying space of the cell structure. The periodic boundary condition helps us avoid some phenomena that might occur in a non-compact space. Cells are intervals $(a,b) \in [0,1)$ or are the union of two intervals $[0,b) \cup (a,1)$. We choose N distinct boundary points $x_1, x_2, ...x_N \in [0,1)$ using a routine built on the standard random number generator rand(). The natural distance on [0,1] induces a distance here, namely:

$$d(x, y) = \min\{|x - y|, 1 - |x - y|\}$$

The size of a cell with boundary points a, b is b - a, except where the cell contains the point 0, in which case it is b + 1 - a.

To determine the motion of each boundary point x in the system, we consider the two cells that meet at that point. We use L = L(x) to denote the size of the cell to the left of the boundary point and R = R(x) to denote the size of the cell to the right. The velocity dx/dt of a boundary point xat any time is a function of only L and R, i.e. dx/dt = f(L, R). Each dynamical system we consider corresponds to using a different choice of f to describe the equations of motion of the points. To ensure that larger cells grow and smaller ones shrink — and so that we indeed have a coarsening dynamic — we choose functions f that are positive when L > R, negative when L < R, and zero when L = R. This ensures that a cell that is smaller than both of its neighbors will shrink, and that a cell that is larger than both of its neighbors will grow. In three of our four dynamics, we choose a function g of one variable such that f(L, R) = g(L) - g(R). If g is an increasing function on the interval $(0, \infty)$, then f > 0 when L > R, f < 0 when L < R, and f = 0 when L = R. In this manner, creating a coarsening dynamic reduces to choosing any increasing function on [0, 1].

At each step, we use a forward Euler method to determine the position of each point at the next time step. We calculate this quantity for all points and then move all points to their determined positions. We choose a variable step size that is small enough to ensure that no two boundary points will cross. This ensures that no cell ever has a negative size. When a cell becomes very small compared to either of its neighbors, one ten-thousandth its size, we collapse that cell, and split its length between its two neighboring cells according to their relative sizes. Every 100 steps we record the distribution of cell sizes and calculate a cell-size correlation function for the system. The cell sizes are normalized in a way so that the average cell size is always 1. As the systems coarsen under their provided dynamics, we watch how these cell structures evolve.

2.5 Error analysis

We use a linear approximation of the velocities over each time step. The time step was chosen to be half the maximum step size allowed over all cell boundary points. The step size needs to be small enough to guarantee that at no point in time will two boundary points cross each other. For each cell, we calculate the relative velocities of the two boundary points. If two adjacent points are moving closer to each another, then the cell between them is "shrinking" and we need to guarantee that in one time step they will not cross. We set the global time step to be half this maximum velocity over *all* cells.

The accuracy of this approximation depends on the dynamic. One way to examine the accuracy of a numerical approximation using a fixed time step involves considering a simplified cell structure consisting of exactly two cells and one moving boundary point. Figure 2.8 illustrates such a system. For each dynamic we can determine the exact equation of motion that governs the point located at x.



Figure 2.8: System with only two cells. Only x can move; its velocity depends on L and R.

If x lies at the midpoint of l and r, then both cells will be equal in size, and dx/dt = 0 independent of the dynamic chosen; the system will not evolve. For others value of $x \in (0, 1)$, the system will evolve until the moving point reaches 0 or 1, at which point L or R will disappear and the system will stop changing. We can calculate how long it will take to reach this point, analytically and when using finite step sizes. This allows us to study how the size of the time step affects the accuracy of the simulation for the various dynamics.

We define the error after one step to be the difference between the result provided by the method, and the exact solution. For two of the dynamics, we can calculate the exact solution, and thus we can calculate the error; for the other two dynamics we cannot calculate the exact solution, and thus provide only rough sketches of what the error might be.

The numerical analysis for Dynamic 1 and Dynamic 4 are the most straightforward. In Dynamic 1, the velocity of a point x is:

$$\frac{dx}{dt} = L - R = 2x - l - r \tag{2.2}$$

This dynamic lends itself to an exact solution. Because each point has a velocity that is linear in the positions of its neighbors, this problem reduces to solving a system of N linear ordinary differential

equations. We can solve this system exactly for all points, though the computation involved would be relatively difficult. For this reason we use this approximation at every time step.

We keep l and r fixed in our analysis. Solving Equation 2.2 for an initial condition $x(t = 0) = x_0$, we find that the location of the point x at any time t is:

$$x(t) = \frac{l+r}{2} + \left(x_0 - \frac{l+r}{2}\right)e^{2t}$$
(2.3)

Therefore, the exact position of the point at time Δt should be:

$$x(\Delta t) = \frac{l+r}{2} + \left(x_0 - \frac{l+r}{2}\right)e^{2\Delta t}$$

$$(2.4)$$

However, using our linear approximation, after one time step of size Δt our point is located at:

$$x_1 = x_0 + (2x_0 - l - r)\Delta t.$$
(2.5)

We define the error ϵ to be the difference between the exact and numerical solutions:

$$\begin{aligned} \epsilon &= x(\Delta t) - x_1 \\ &= \frac{l+r}{2} + \left(x_0 - \frac{l+r}{2}\right) e^{2\Delta t} - (x_0 + (2x_0 - l - r)\Delta t) \\ &= \left(x_0 - \frac{l+r}{2}\right) \left(e^{2\Delta t} - 1 - 2\Delta t\right) \\ &= \left(x_0 - \frac{l+r}{2}\right) \left(\frac{(2\Delta t)^2}{2!} + \frac{(2\Delta t)^3}{3!} + \dots\right), \end{aligned}$$
(2.6)

obtaining the last line by taking the power series expansion of the $e^{2\Delta t}$ term. Our error over a single time step is then linear in the displacement of x from the midpoint of r and l, and of order $(\Delta t)^2$ in the time-step. The error accumulated over all time steps turns out to be linear in the time step itself.

Dynamic 4 has the simplest error analysis. Over short times, the velocity of every point in the system is constant. If a cell is smaller than both of its neighbors it will shrink at a constant rate of -2. If it is larger than both neighbors it will grow at a rate of +2. If it is larger than one neighbor and smaller than its other neighbor its size will stay constant. If a cell is the same size as one neighbor and larger or smaller than its other neighbor, then it will grow or shrink at a rate of +1 or -1, respectively. The velocity of a boundary point can only change when a cell in the system dies. This change can affect other cells nearby and can affect the velocities of other points. However,

between those points in time, no changes can occur. Thus, we are free to use a step size that is up to half of the size of the smallest cell in the structure with no error. After one time step of that size, the smallest cell in the structure will be at least 0 in size and will be removed.

The error involved in Dynamic 2 and Dynamic 3 are difficult to analyze. It seems that the error in both is similar to the other cases for which we can provide more rigorous results.

2.6 Results

Dynamic 1: L - R

In this section we look at dynamical cell systems constructed using one of the four initial states and one of the four dynamics described above. We begin by considering a system that begins in Initial State A with 1 million cells and that evolves through Dynamic 1.

Figure 2.9 shows the cell-size distribution and correlation function after the system has evolved until only 500,000 cells remain. Figures 2.10 and 2.11 show similar data when 200,000 and 100,000 cells remain, respectively. The distribution of cell sizes remains roughly fixed over time, as does the correlation function between the size of a cell and the expected size of its neighbors. From these figures it appears that under this dynamic, the cell structure maintains a self-similar state as it evolves.



Figure 2.9: The cell-size distribution function, and the cell-size correlation function, after the system has evolved. At this point in time, t = 0.346 and 500,000 cells remain.

We introduce here an alternate way of presenting data, one that allows us to present significantly more data in the same amount of space. Instead of showing the cell-size distribution at one point in time, we attempt to present the cell-size distribution for all time. To do this, we plot our data as follows. The x-axis now corresponds to the time variable. The y-axis continues to correspond to



Figure 2.10: The cell-size distribution function, and the cell-size correlation function after the system has evolved. At this point in time t = 0.804 and 200,000 cells remain.



Figure 2.11: The cell-size distribution function, and the cell-size correlation function after the system has evolved. At this point in time t = 1.151 and 100,000 cells remain.

the y values of a probability distribution. However, we now use curve to represent the y values for a bin with a single x-value as the y values changes over time; we call these curve probability density contours. We use x-values in $\{0.05, 0.15, 0.25...\}$, corresponding to x-values of bars in the histograms used until now.

This graphical representation allows us to see how the histogram of the cell size distribution changes over time. The drawback of this method is that identifying particular curves with particular x-values is not possible. For example, there is no way of knowing which curve corresponds to cells that are smallest in size, or to those that are twice the average cell size. The upside is that this plot allows us to see when the distribution of cell sizes is changing with time and when it has settled. Eventually, these plots will allow us to "watch" as cell structures evolve from initial states to long-time steady states.

Figure 2.12 shows the time-evolution of a system beginning in Initial State A with 1 million cells and evolving through Dynamic 1. Although the curves fluctuate slightly over time, they stay



Figure 2.12: The evolution of the cell-size distribution function with time for system A1. This system begins with 1,000,000 cells and at t = 1 there remain about 136,000 cells. Each curve corresponds to the x-value of a single bar in a histogram with an x-value in $\{0.05, 0.15, 0.15, ...\}$.

roughly constant. The small black circles indicate the exact values of the function $y = e^{-x}$ for values of $x = \{0.05, 0.15, 0.25...\}$; these correspond to we place them at certain points to make comparison easy. From this figure it appears that the distribution of cell sizes remains constant even while the system coarsens. Indeed, it appears that the system is in a steady state from the very beginning. Before attempting to explain why this is, it is worthwhile to consider what happens to other initial conditions that evolve through this dynamic.

We now consider the evolution of three other initial states using the same dynamic. Figure 2.13 shows the evolution of a system which begins with 250,000,000 cells in a Voronoi state, described and illustrated in Section 2.2. The distribution of cell sizes in the initial Voronoi state can be seen in Figure 2.4. This system evolves using Dynamic 1, in which the velocity of each boundary point is dx/dt = L - R, where L and R are the sizes of the cells to the left and the right of a boundary point x. Unlike in the previous example, here we find a system whose probability distribution changes under this dynamic, and which reaches a steady state different from its initial one. By the time that only 30,500 cells remain, it seems that the system has reached a steady state identical to Initial



Figure 2.13: System B1 begins with 250,000,000 cells. At t = 5, there remain about 30,500 cells; when the graph ends at t = 6, there remain 4000 cells. Black horizontal bars are drawn at corresponding values of e^{-x} to help the reader observe the system reaching steady state.

State A. We draw line segments on the right-hand side of the graph of values of e^{-x} , to help the reader observe the system reaching steady state. Here, even when 99% of the cells have already disappeared, at $t \approx 2.77$, the system has still not reached steady state; this forced us to use a very large initial systems in simulations. We will see later that other dynamics reach steady states many times faster and require significantly smaller simulations to observe.

Initial State C, the lightly perturbed lattice, provides a fascinating case in unstable equilibrium. Before perturbing the system, all cells are identical in size and none of the boundary points move. As soon as that symmetry is broken, by perturbing all of nodes very lightly, the entire system quickly evolves away from the initial state. The distribution of cell sizes in the initial Slightly Perturbed State can be seen in Figure 2.5. Figure 2.14 shows the evolution of a system which begins with 250,000,000 cells in the Slightly Perturbed State, which is described and illustrated in Section 2.2. The system evolves under Dynamic 1, in which the velocity of each boundary point is dx/dt = L - R, where L and R the sizes of the cells to the left and the right of a boundary point. At t = 0, half the cells are slightly larger than 1 and half the cells are slightly smaller than 1. Thus the cells are evenly split between the two bins which border the cell size of 1. As soon as the system begins evolving, the cell-size distribution changes rapidly — the distribution changes tremendously even before many



Figure 2.14: Above shows the entire evolution of a C1 system beginning with 250,000,000 cells. Below shows small sections of C1 evolution. The system has 41,000,000 cells at t = 1.5, 9,000,000 when t = 2.5, 350,000 cells at t = 6, and 10,000 cells at t = 8.

of the cells have disappeared. The bottom part of Figure 2.14 shows small sections of the same evolution. This allows us to see some of the semi-chaotic behavior of the system immediately after it begins evolving and before it enters a more relaxed period of slow convergence to the steady state. It is also clear that the system has not reached a steady state even when 99.999% of initial cells have disappeared, by $t \approx 7.75$. This very slow convergence prevents us from observing a C1 system reach steady state, though we expect that large simulations will verify that this system indeed reaches a steady state similar to that of the A1 and B1 systems.

Our final example begins with 250,000,000 cells in the Heavily Perturbed State. Before perturbing the structure, all cells are identical in size. After the perturbation, which is much stronger than in the last case, the cells range in size from 0 to just under twice the average cell size. The distribution of cell sizes in this state can be seen in Figure 2.6. As with the three prior examples, here too the system relaxes to the same steady state. Figure 2.15 shows a system that begins with 250,000,000



cells and has evolved until only 10,000 cells remain. Because the initial condition is different from

Figure 2.15: System D1 begins with 250,000,000 cells. At t = 6, there remain roughly 24,000 cells.

any of the three previous ones, the relaxation is necessarily different as well. If we needed to describe the relaxation, we might say that it is considerably smoother than that of system C1. However, this system also relaxes very slowly towards the steady state. By the time that 99.999% of the initial cells have disappeared (at $t \approx 7.25$), the system has still not reached a steady state. Much larger simulations will be necessary to verify that this system indeed reaches a steady state similar to that of the A1 and B1 systems.

Although we do not observe all four example systems reach steady-states, we expect that large enough simulations will verify that they all do. Moreover, we expect that all four will reach a steady state which is independent of initial condition. It seems that the universal steady state which they reach is identical to Initial Condition A, the state constructed by randomly placing boundary points on the unit interval with a uniform density distribution and which is exponential in shape, $P(x) = e^{-x}$. We do not have a good understanding of why this is, though we might consider that this distribution maximizes the entropy of the cell-size distribution. It is not clear though why this should matter. In what follows, we look at how the same four initial states evolve under three other dynamics. Each dynamic brings all four initial conditions to the same steady state. Each dynamic leads to a unique universal steady state.

Dynamic 2: $\ln L - \ln R$

The second set of dynamical systems we study uses the natural logarithm of the cell sizes to determine the motions of the boundary points. We considered this dynamic because scaling the system by a constant factor leaves the motions unchanged. We thought that this dynamic might provide interesting results.

Figure 2.16 shows the evolution of an A2 system beginning with 50,000,000 cells and evolving until roughly 738,000 cells remain. Figure 2.17 shows the evolution of a B2 system that begins with



Figure 2.16: System A2 begins with 50,000,000 cells; when the graph ends at $t = 1 \times 10^{-6}$, there remain 738,000 cells.

50,000,000 cells and evolving until roughly 3,700,000 cells remain. Figure 2.18 shows the evolution of a C2 system that beings with 50,000,000 cells and evolving until roughly 1,700,000 cells remain. Figure 2.19 shows the evolution of a D2 system that begins with 50,000,000 and evolving until roughly 1,600,000 cells remain. The graphs in Figure 2.20 show the two characteristic functions of the universal steady state of this dynamic. The *y*-values show the average taken over the four systems A2, B2, C2, and D2; error bars indicate the standard deviation among the four samples.



Figure 2.17: System B2 begins with 50,000,000 cells; when the graph ends at $t = 2 \times 10^{-7}$, there remain 3,700,000 cells.



Figure 2.18: System C2 begins with 50,000,000 cells; when the graph ends at $t = 5 \times 10^{-7}$, there remain 1,700,000 cells.



Figure 2.19: System D2 begins with 50,000,000 cells; when the graph ends at $t = 5 \times 10^{-7}$, there remain 1,600,000 cells.



Figure 2.20: At the top is an example of a universal steady state structure of dynamic 2. Below shows the distribution of cell sizes, and the correlation function between neighboring cells, averaged over the steady state of the A2, B2, C2, and D2 systems; error bars indicate the standard deviation among the four samples.

Dynamic 3: $\frac{1}{R} - \frac{1}{L}$

The third dynamic we consider uses the inverse of the cell sizes to determine the velocity of boundary points. Again it seems that all initial conditions lead to the same steady state.

Figure 2.21 shows the evolution of an A3 system beginning with 5,000,000 cells and evolving until roughly 135,000 cells remain. Figure 2.22 shows the evolution of a B3 system beginning with



Figure 2.21: System A3 begins with 5,000,000 cells; when the graph ends at $t = 2 \times 10^{-11}$, there remain 135,000 cells.

5,000,000 cells and evolving until roughly 360,000 cells remain. Figure 2.23 shows the evolution of a C3 system beginning with 5,000,000 cells and evolving until roughly 140,000 cells remain. Figure 2.24 shows the evolution of system D3 beginning with 5,000,000 cells and evolving until roughly 615,000 cells remain. The graphs in Figure 2.25 show the two characteristic functions of the universal steady state of this dynamic.



Figure 2.22: System B3 begins with 5,000,000 cells; when the graph ends at $t = 3 \times 10^{-12}$, there remain about 360,000 cells.



Figure 2.23: System C3 begins with 5,000,000 cells; when the graph ends at $t = 2 \times 10^{-11}$, there remain 140,000 cells.



Figure 2.24: System D3 begins with 5,000,000 cells; when the graph ends at $t = 1 \times 10^{-12}$, there remain 614,000 cells.



Figure 2.25: At the top is an example of a universal steady state structure of dynamic 3. Below shows the distribution of cell sizes, and the correlation function between neighboring cells, averaged over the steady state of the A3, B3, C3, and D3 systems; error bars indicate the standard deviation among the four samples.

Dynamic 4: 1 if L < R and -1 if L > R

Although the last dynamic might appear the least interesting, it in fact leads to some of the most interest behavior observed this far. Under this dynamic, the velocity of a cell boundary is always -1, 0, or 1. We ran a number of large simulations that began with 10,000,000 cells, one in each of the four initial states. Figure 2.26 shows the results from a system that begins in Initial State A, the system with uniformly, independently distributed cell boundaries. At the end of the simulation, $t = 1 \times 10^{-5}$, there remain roughly 75,000 cells, and it seems that the system has reach a steady state.



Figure 2.26: System A4 begins with 10,000,000 cells; when the graph ends at $t = 1 \times 10^{-5}$, there remain 75,000 cells.

Figure 2.27 shows a system evolving from Initial State B under the same dynamic. The simulation begins with 10,000,000 cells; the right end of the graph, at $t = 1 \times 10^{-5}$, there remain 100,000 cells. The system seems to settle at around $t = 5 \times 10^{-6}$, at which point 200,000 cells remain, or 2% of the original cells. Despite some similarity between this figure and the last one, looking closely at the two will show different trajectories at very early times in the evolution.

Figure 2.28 shows a system evolving from the Initial State C under the same dynamic. The simulation begins with 10,000,000 cells; at the right end of the graph, at $t = 1 \times 10^{-5}$, there remain roughly 75,000 cells. The system seems to settle at around $t = 5 \times 10^{-6}$, at which point 150,000 cells remain. Here the evolution of the system looks considerably more "choppy", though it seems



Figure 2.27: System B4 begins with 10,000,000 cells; when the graph ends at $t = 1 \times 10^{-5}$, there remain 100,000 cells.

to settle to the same long-time steady-state.

Figure 2.29 shows a system evolving from Initial State D under the same dynamic. The simulation begins with 10,000,000 cells and evolves until roughly 75,000 cells remain at time $t = 1 \times 10^{-5}$. The system seems to settle at roughly $t = 2 \times 10^{-6}$, or when there are about 200,000 cells remaining. Figure 2.30 we show a small segment of the evolution of the D4 system. The immediate evolution from the initial state seems chaotic, though as can be seen from 2.29, this system too eventually relaxes towards a steady state. The graphs in Figure 2.31 show the two characteristic functions of the universal steady state of this dynamic.



Figure 2.28: System C4 begins with 10,000,000 cells; when the graph ends at $t = 1 \times 10^{-5}$, there remain 75,000 cells.



Figure 2.29: System D4 begins with 10,000,000 cells; when the graph ends at $t = 1 \times 10^{-5}$, there remain roughly 75,000 cells.



Figure 2.30: System D4 begins with 10,000,000 cells, using Initial State D.



Figure 2.31: At the top is an example of a universal steady state structure of dynamic 4. Below shows the distribution of cell sizes, and the correlation function between neighboring cells, averaged over the steady state of the A4, B4, C4, and D4 systems; error bars indicate the standard deviation among the four samples.

Coarsening Rates

After each of the systems has reached a steady state, we can measure the rate at which it coarsens. This rate should be independent of the initial condition and depend only on the dynamic. For each dynamic considered, we report the estimated coarsening rate. In most cases, data were averaged over the four samples generated by the four initial conditions. In the table that follows $\langle l \rangle$ denotes the average size of a cell; we use the $X \pm x$ notation to indicate that amongst the four samples we considered, none had a value higher than X + x or lower than X - x.

Dynamic	Estimated coarsening rate in steady state
1. $L - R$	$\langle l \rangle = l_0 e^{2t}$
2. $\ln L - \ln R$	$\langle l \rangle = (1.235 \pm 0.003)t$
3. $1/R - 1/L$	$\langle l \rangle = (1.61 \pm 0.02)\sqrt{t} + l_0$
4. ±1	$\langle l \rangle = (1.309 \pm 0.003)t$

We note that for Dynamic 1, only one system (Initial State A) relaxed to the steady state, and so data were taken from only one sample.

2.7 Conclusions

We study dynamical cell structures that begin with one of four initial conditions and evolve through one of four dynamics. We consider the rates at which these system coarsen and the long-time steady states which they eventually reach.

The important observation we make is that each dynamic evolves the cell structures to a particular steady state independent of the initial condition, and which depends only on the particular dynamic imposed. This provides "experimental" support for a conjecture that a large set of coarsening processes lead to steady state cell structures that are generally independent of initial conditions.

Much work is left to be done. It is hoped that more analytical results can be obtained to help understand why the studied dynamics lead to particular steady states. Can we find a mapping between the set of dynamics, or a particular subset of them, and the set of universal steady state distributions of cell sizes, for example?

All of the dynamics considered here are defined very locally, depending only on the two adjacent cells. In future work we might investigate dynamics which consider longer-range interactions as well. For example, we can consider each cell boundary point as a point-mass and consider the dynamical system in which each point attracts other points with a force that decays as $1/d^2$, with d being

the distance between the pair of points. Simulating these systems will require considerably more computational investment. We might also consider dynamics with a stochastic component.

In the following chapters we consider cell structures in two and three dimensions. In each dimension, we look at only one particular dynamic, namely curvature flow in two dimensions and mean curvature flow in three dimensions. The reason for this is that in two and three dimensions, these dynamics have a very natural, physical meaning. Also, the simulations are more complicated, requiring more time to set up and considerably more processing power and memory to execute.

Supplementary notes

We mentioned earlier that we have done some work for a fifth dynamic, where the velocity of a boundary point is $dx/dt = L^2 - R^2$. Preliminary work indicated that we would need simulations larger than we were able to conduct at the time. We leave for future work determining the universal steady state of this and other dynamics.

The error analysis for this dynamic is quite similar to that of Dynamic 1. The velocity of a boundary point is:

$$\frac{dx}{dt} = L - R = (x - l)^2 - (r - x)^2$$
(2.7)

If $x(0) = x_0$, then the equation that describes the location x(t) at any point in time is:

$$x(t) = \frac{l+r}{2} + \left(x_0 - \frac{l-r}{2}\right)e^{2(r-l)t}$$
(2.8)

Therefore, the exact position of the point at time Δt should be:

$$x(\Delta t) = \frac{l+r}{2} + \left(x_0 - \frac{l-r}{2}\right)e^{2(r-l)\Delta t}$$
(2.9)

However, using our linear approximation, after one time step of size Δt our point is located at:

$$x_1 = x_0 + ((x-l)^2 - (r-x)^2)\Delta t.$$
(2.10)

We define the error ϵ to be the difference between the exact and numerical solutions:

$$\begin{aligned} \epsilon &= x(\Delta t) - x_1 \\ &= \frac{l+r}{2} + \left(x_0 - \frac{l-r}{2}\right) e^{2(r-l)\Delta t} - (x_0 + ((x-l)^2 - (r-x)^2)\Delta t) \\ &= \left(x_0 - \frac{l+r}{2}\right) \left(e^{2(r-l)\Delta t} - 1 - 2(r-l)\Delta t\right) \\ &= \left(x_0 - \frac{l+r}{2}\right) \left(\frac{(2(r-l)\Delta t)^2}{2!} + \frac{(2(r-l)\Delta t)^3}{3!} + \dots\right), \\ &= \left(r-l\right) \left(x_0 - \frac{l+r}{2}\right) \left(\frac{(2\Delta t)^2}{2!} + \frac{(2\Delta t)^3}{3!} + \dots\right), \end{aligned}$$
(2.11)

obtaining the penultimate line by taking the power series expansion of the $e^{2(r-l)\Delta t}$ term. Our error after a single time step is then linear in the sizes of the two adjacent cells (r-l), linear in the displacement of x from the midpoint of r and l, and of order $(\Delta t)^2$ in the time-step. The error accumulated over all time steps turns out to be linear in the time step itself.

Chapter 3

2D Systems

Two-dimensional cell structures are the simplest cell structures whose topology is non-trivial. They are also the lowest-dimensional cell structures for which a natural notion of curvature can be defined. This chapter focuses on two-dimensional cell structures that evolve via curvature flow. We begin by explaining the materials science background that motivates work on these problems. We then explore some problems and questions that arise in characterizing two-dimensional cell structures and suggest a number of ways of approaching them. Next we describe in detail what it means for a cell structure to evolve through curvature flow, and consider some consequences of this. Last, we describe a method for simulating these systems, analyze the error involved, and report results of our simulations.

3.1 Motivation

The microstructure of most common metals and many ceramics is cellular in nature, as described briefly and illustrated in Chapter 1. It has long been recognized [17, 18] that curvature plays a crucial role in the evolution of these structures, because grain¹ boundaries tend to migrate toward their centers of curvature to reduce their interfacial energy [19]. It was shown that this boundary migration occurs only when grain faces are curved or when grains meet at non-equilibrium angles [20]. When a boundary between two grains is curved, thermal motion causes individual atoms to preferentially migrate away from the convexly shaped grain towards its neighbor, which in effect shifts the position

 $^{^{1}}$ In the materials science literature, individual cells are often called *grains*, their boundaries *grain boundaries* and so forth. In this and the following chapters, we use the terms cell and grain interchangeably.

^{*}The content of this chapter has been adapted from [16].

of the boundary itself away from the convexly-shaped grain. When grain boundaries are flat, thermal motion continues but there is no net migration over time, except than that expected from a random walk. Likewise, when grain boundaries meet at angles other than 120° in the isotropic case, grains with acute angles quickly lose atoms at their sharp tips, which causes angles to change so that they approach an equilibrium in which these angles are 120° . Smith later showed [21] that not only are flat grain boundaries stable, but so are any surfaces whose *mean curvature* is zero. Smith also showed that in the more general case, the particular orientations of the grains will impact the surface tension of the grain boundaries, and consequently the equilibrium angles diverge from 120° in this anisotropic case [22]. Figure 3.1 shows a cell structure in which each grain has an identical pattern, though each grain is oriented differently. Circles represent individual atoms sitting on the triangular lattice. This models real polycrystalline materials, in which all grains have identical crystalline structure, though each grain is oriented differently.



Figure 3.1: Many grains with identical triangular patterns, each with a different orientation. Circles here represent atoms in the individual crystals.

It has also been observed for a long time that polycrystalline materials which evolve through curvature flow exhibit statistical self-similarity. For example, polycrystalline materials have a definite distribution of grain shapes and, after normalization, grain sizes. It is understood that curvature flow shapes cell structures in a way that provides order amongst the otherwise disordered arrangement of grains in the material. Because many of a material's properties depend strongly on the material's microstructure — for example its strength and electrical conductivity — a complete understanding of this statistically "universal" structure is very much desired. This requires understanding how cell structures evolve under curvature flow.

The bulk of our work focuses on the problem of *isotropic* grain growth, the simplified case



Figure 3.2: A Voronoi tessellation of the unit square torus and its dual Delaunay triangulation.

where the grain boundary energy γ and mobility M are uniform throughout the system. Although almost all materials exhibit some *anisotropy* associated with the oriented nature of individual grains and the relative misorientation between adjacent ones, the isotropic case is still worth considering. From a purely mathematical perspective, this problem raises many interesting questions. Moreover, understanding the isotropic case might provide insight into the more general anisotropic case. Last, the anisotropy of many systems is relatively mild, and so solutions to the isotropic case are reasonable approximations of the real solution.

3.2 Characterization

Here we describe some problems involved in characterizing two-dimensional cell structures. In most physical systems, the number of cells at the system's boundary are far outnumbered by those in its interior. We therefore would like consider a space without boundary. For computational purposes we do not use \mathbb{R}^2 , and because we would like to use an intrinsically flat space we do not consider the standard two-sphere \mathbb{S}^2 . The flat two-torus \mathbb{T}^2 turns out to be the best space to use and in practice we model that using $[0, 1]^2$ with periodic boundary conditions.

Completely characterizing two-dimensional cell structures involves characterizing both their topological² and geometrical features. Much work in describing two-dimensional cell structures has traditionally focused on triangulations, which are roughly dual to the simple cell structures that we consider. Figure 3.2 shows a Voronoi cell decomposition of a flat torus and its dual Delaunay triangulation. Although the number of triangulations of \mathbb{T}^2 are finite for a fixed number of vertices, this number grows exponentially with the number of triangles or cells in the structure. If we only count

²By topological features, we mean those of the cell structure, and not of the underlying space, \mathbb{T}^2 in our case. That is, how the cells are connected and so forth. We sometimes refer to this as the topological characterization, sometimes as the combinatorial characterization. In both cases we mean the same.

cell decompositions with 15 or fewer cells, we already find 1,618,768,888 combinatorially distinct structures! We would like a way of comparing cell structures and describing which ones are more similar or less similar, and not only saying whether or not two are isomorphic.

Before continuing, we point out that the cell structures we consider in this chapter are slightly more limited and slightly more general than those considered elsewhere. First, we only deal with *simple* cell structures. This condition limits the way in which edges can meet: at most three cells can meet at any one point. This would exclude, for example, a map of the United States as divided into states, which includes the four-way meeting between Utah, Arizona, Colorado, and New Mexico. We will explain the motivation for this restriction in Section 3.3. In this sense, the cell structures we



Figure 3.3: Map of the United States with a disallowed, non-simple vertex highlighted.

consider are more narrowly defined than those considered elsewhere. We note that triangulations *do* impose this restriction.

On the other hand, the cell structures we consider here are slightly more general than those considered elsewhere. We explicitly allow cells with only two sides, also known as *digons*. Figure 3.4 shows a simple cell structure with a digon in the middle. Although digons do not violate the *simple* condition, they are often avoided for a number of other reasons. A graph containing a digon (or the tessellation of a digon) cannot be 3-connected and therefore, via Steinitz's theorem [23], cannot be the graph of a convex polyhedron. Moreover, the presence of digons introduces irregularities in boundaries of adjacent cells. Without digons, two distinct cells either intersect along an edge or not at all. After digons are introduced, we must generalize this statement: two distinct cells either intersect along a series of edges or not at all. In the figure above, the two cells that are adjacent to the digon intersect one another along more than one edge. These are some of the reasons that digons are often disallowed.



Figure 3.4: A cell structure containing a digon.

With all of these "problems", why do we allow cell structures containing digons? The reason is because digons actually appear in many physical systems including those we study. Although they complicate our analysis, we must allow them if our systems are to model real physical systems.

One condition we do not impose explicitly yet seems to always be satisfied is this: Every edge is bounded by two vertices. This condition excludes "circular island" cells. We have not found these in any system and we believe that they cannot arise in the physical systems we study.

A few remarks can be made about these systems. According to Euler's theorem, the number of vertices, edges, and cells are related as follows:

$$\chi = V - E + F,\tag{3.1}$$

where V is the number of vertices, E the number of edges, and F the number of cells; χ is known as the Euler characteristic of the underlying space, and in our case $\chi(\mathbb{T}^2) = 0$. Because every vertex is incident with three edges and every edge is incident with two vertices, we have 2E = 3V. We can then conclude that E/F = 3. Because every edge is adjacent to two cells, the average number of sides per cell is twice that value, i.e. the average number of sides per cell is 6. This places a significant restriction on our cell structures, though still allows for much variety in the distribution of sides per cell.

One way to characterize cell structures is by considering P(n), the probability distribution of cells with $n \ge 2$ number of sides; we can consider this distribution for both finite and infinite systems. We say that a particular probability distribution P is *realizable* if there exists some cell structure with that probability distribution. A natural question to ask is what probability distributions are realizable? Is any discrete probability distribution whose mean is 6 realizable? We leave for another place answering this question, though point out that a promising lead to answering this question in the affirmative might be found in the work of [24].

Aside from characterizing the combinatorics of a particular cell structure, we also characterize its geometry. That is, even if two cell structures have the same distributions of cell shapes, their geometries can be quite different. In this chapter and later chapters we consider a number of geometric descriptions of cell structures, including the distribution of cell areas and roundness measures. We have noted before that formulating criteria by which we could say that two cell structures are similar or different would be an important accomplishment.

3.3 Curvature flow on 2D cell structures

Curvature flow on curves

In contrast to the last chapter, now we consider only one dynamic that acts on cell structures curvature flow. Before explaining how curvature flow affects cell structures, we begin by introducing curvature flow on manifolds, the more traditional setting for geometrical flows.

Much interest has arisen over the last thirty years in various types of geometrical flows. The general idea is to consider a differentiable manifold whose metric structure evolves over time via a set of partial differential equations. Inverse mean curvature flow is one particular example that has proven successful in shedding light on general relativity [25] and black holes [26]. A possibly more prominent example is the Ricci flow, which "smooths out" the metric of a Riemannian manifold in a very particular and controlled manner. When considered properly, this study of geometry can help us learn much about the topology of an object. Grigori Perelman recently used this tool to help solve the longstanding Poincaré conjecture regarding the classification of 3-manifolds [27, 28, 29].

In this thesis, we focus on *curvature flow* of planar curves and two-dimensional cell structures, and *mean curvature flow* of embedded surfaces and three-dimensional cell structures. Both of these areas have proven fruitful areas of research in the last thirty years, and we provide a rough sketch of the general ideas and of a few basic, but very beautiful, results in the field. We focus now on curvature flow of planar curves and two-dimensional cell structures, and leave mean curvature flow of surfaces and three-dimensional cell structures for further discussion in the next chapter.

A curve in the plane can be defined as a continuous mapping: $\alpha : I = [a, b] \to \mathbb{R}^2$. We say that the curve is *closed* if $\alpha(a) = \alpha(b)$. We say that a curve is *simple* if for all $t, u \in (a, b), \alpha(t) \neq \alpha(u)$. This condition prohibits a curve from crossing itself. Last, a curve is *regular* if $\alpha'(t) \neq 0$ for all $t \in I$.

The arc-length s of a curve α mapped from the interval [a, b] is defined: $s(\alpha) = \int_a^b |\alpha'(t)| dt$. Since the arc-length of a curve does not depend on its parameterization, we choose a parameterization in a way so that the arc-length s of a curve α mapped from the interval [a, b] is always exactly b - a. This is called the *arc-length parameterization* of α .

If an arc-length parameterized curve α is simple, closed, and at least twice-differentiable, then $|\alpha'| = 1$ and we can define a notion of curvature as follows. We use $\mathbf{T}(s) = \alpha'(s)$ to denote the unit vector tangent to α at a point $\alpha(s)$ and pointing in the direction in which we traverse the curve. We use $\mathbf{N}(s)$ to refer to the unit vector normal pointing outward from α . We define the unsigned curvature $\kappa(s) = ||\mathbf{T}'(s)|| = ||\alpha''(s)||$. In the plane we can also give the curvature a sign, depending on which direction the tangent direction is turning. If we are traversing the curve in a counterclockwise fashion, then if the unit tangent vector is turning clockwise then the curvature is negative; if it is turning counter clockwise then the curvature is positive. We use k(s) to denote this signed curvature at a point $\alpha(s)$.

In Figure 3.5 we draw a picture of a curve with a number of $k\mathbf{N}$ vectors drawn at various points along the curve. It can be seen that when the unit tangent changes more sharply, the vectors are



Figure 3.5: A curve with some curvature arrows drawn.

longer (owing to a large κ), and when the unit tangent changes less sharply, the vectors are shorter.

Curvature flow uses this curvature vector $k\mathbf{N}$ to define an equation governing the evolution of the curve. We let $\alpha(\cdot, 0) : S^1 \to \mathbb{R}^2$ be an embedded, closed planar curve that is at least twice differentiable. We define $\alpha : S^1 \times [0, T) \to \mathbb{R}^2$ and require that it satisfy the differential equation:

$$\frac{\partial \alpha}{\partial t} = Ck\mathbf{N} \tag{3.2}$$

As the parameter t moves through [0,T), the curve "evolves through time". The variable C allows

the introduction of a constant that might depend on physical properties of the system. Generally speaking we consider isotropic systems in which this variable is uniform throughout a system and over time. Figure 3.6 shows an embedded curve at different points in its evolution. We note two features



Figure 3.6: A smooth curve embedded in the plane evolves via curvature flow. The area bounded by the curve decreases at a constant rate, and the curve becomes progressively more circular; the curve eventually disappears in finite time.

of this evolution, the first is observable from the illustrations, the second not. One curious feature about this curve is that it seems to evolve toward a circle as it shrinks. A beautiful result of Hamilton, Gage, and Grayson [11, 9] states that *every* embedded curve that evolves through curvature flow becomes convex and asymptotically closer to a circle as it disappears, without developing singularities before disappearing.

A second key feature of the evolution is that way in which its area changes — the area decreases with time at a constant rate independent of the shape of the curve or the area of the bounded region. We adapt a proof of this theorem from the paper of Mullins [30].

Consider a simple smooth curve evolving via curvature flow as described above. We can consider the curve given by polar coordinates, $r(\theta, t)$, where r and θ are the polar coordinates and t is time. To define the signed curvature we choose to traverse the curve counterclockwise and regard the tangent vector as directed in this sense. We use s to refer to the arc length along the curve and β the angle measured in a counterclockwise manner between the positive x-axis and the directed tangent; we use ψ to measure the angle measured in a counterclockwise manner between the polar radius vector and the directed tangent. Any point on the curve moves toward its center of curvature in a manner described by curvature flow, Equation 3.2. We set C in that equation to $M\gamma k$, where M is the mobility of the grain boundary, γ is its surface tension, and k is the signed curvature at that point. Figure 3.7 shows a solid curve at time t and another dashed curve at a time $t + \Delta t$. It



Figure 3.7: A simple closed curve adapted from [30]. The solid closed curve shows $r(\cdot, t)$; the shorter, dashed curve shows $r(\cdot, t + \Delta t)$.

can be seen from the top left part of the illustration that $\Delta r \sin \psi = -M\gamma k\Delta t$. From differential geometry we know that $k = \partial \beta / \partial s$ and that $\sin \psi = r(\partial \theta / \partial s)$. We can then calculate:

$$\frac{\partial r}{\partial t} = -M\gamma \frac{k}{\sin\psi} = -M\gamma \frac{1}{r} \frac{\partial\beta}{\partial\theta}$$
(3.3)

The area enclosed by a simple closed curve is given by $A = \frac{1}{2} \oint r^2 d\theta$ where the integral is taken in a counterclockwise sense around the curve. Together with the previous equation, we can calculate the rate of change of the enclosed area:

$$\frac{dA}{dt} = \oint \frac{\partial r}{\partial t} r d\theta = -M\gamma \oint \frac{\partial \beta}{\partial \theta} d\theta = -M\gamma \oint d\beta = -2\pi M\gamma$$
(3.4)

The area inside a closed curve thus changes by a constant rate determined only by M and γ . This is the theory of curvature flow on simple closed planar curves. This result will be vital when studying the evolution of cell structures that evolve via curvature flow.

In the next section we describe how curvature flow affects cell structures and derive the von Neumann-Mullins relation which states that the rate of change of cell areas is proportional to six less than its number of sides.

Curvature flow on cell structures

Before deriving the general von Neumann-Mullins relation for cell structures evolving via curvature, we point out and explain two particular features of curvature flow cell structures. Figure 3.8 shows a typical cell structure which has evolved through curvature flow. The first feature worth noting is the



Figure 3.8: A small cell structure.

way in which edges meet: there is no point at which more than three edges meet because such points are "unstable" under curvature flow. Curvature flow is the gradient flow of the length functional and thus minimizes the total length of a curve, which can be seen from the frames in Figure 3.6. In cell systems, curvature flow minimizes the sum of all edge lengths in the structure. For this reason, we almost never find four or more edges meeting at a single vertex, since the total length of this configuration can almost always be reduced by splitting the vertex into two and introducing a new edge between them. Figure 3.9 shows a vertex where four edges meet, and a picture after the vertex



Figure 3.9: Splitting a vertex where four or more edges meet can almost always decrease the total length of the edges.

has been "split" into two. The total length of the edges is smaller after the vertex has been split

and a new edge created. Vertices are thus unstable under this edge length-minimizing flow.

Another feature of interest is the way in which three edges meet at a vertex — the angle between any two edges is always 120°! The explanation for this phenomenon goes back to an old problem from the 17th century. In a letter to his student, Fermat asked the following question. Consider three points in the plane such that the edges between them do not form any angle larger than 120°. Figure 3.10 shows such a picture. Now consider adding a fourth point and measure the sum of the



Figure 3.10: A set of three points in the plane such that the edges between them do not form any angle α_i greater than 120°. The sum of edge lengths of all segments from the three points to a fourth point inside the triangle is minimized when the angles between the three segments are all 120°.

distances between the new point and the three initial points. Fermat asked his student Torricelli to find the point which minimizes this sum. It turns out that there is one point that minimizes this sum, and at this point all angles between the edge segments will be 120° ! Since curvature flow minimizes the sum of the edge lengths, vertices will always move in a way that brings them closer to equilibria positions, where the angles are 120° . We note that although grain boundaries move at a finite rate, in an infinitesimal neighborhood of a point, the edges can move infinitely fast, and so the edges can always meet at 120° , even while the structure is evolving at a finite rate.

We can now generalize Equation 3.4 to cases of cells in cell structures, giving us the von Neumann-Mullins relation [31, 30]. Equation 3.4 states that the rate of change of the area enclosed by a simple smooth curve is proportional to the integral of the signed curvature around the curve. Positively signed curvature contributes to the cell's shrinking while negatively signed curvature contributes to the cell's shrinking while negatively signed curvature contributes to the cell's growth. When integrated around the entire curve, the integral of the signed curvature is always 2π . In cells that are part of cell structures and that include non-differentiable "corners", this integral must be reconsidered. Figure 3.11 show a typical example of a cell in a cell structure that evolves via curvature flow. Although the shape is somewhat irregular, internal angles between adjacent edges are always 120° . Therefore, the discrete angle at which the tangent changes at this point is $\pi/3$, as illustrated. Since the points at which three edges meet are in local equilibrium, they do not move and therefore do not contribute to the change in area of the cell. We must then



Figure 3.11: A cell with five sides; each pair of cell edges meets at a vertex with an internal angle of 120° .

calculate how much curvature is left around the edges. The signed curvature along the edges and the discrete turning angles must sum to 2π . If each of the *n* discrete turns is $\pi/3$, then the curvature along the edges themselves is $2\pi - n\frac{\pi}{3}$. Our new equation of motion is then $dA/dt = -M\gamma(2\pi - n\frac{\pi}{3})$, or rewritten in terms of n - 6.

$$\frac{dA}{dt} = \frac{\pi}{3}M\gamma(6-n). \tag{3.5}$$

This form allows us to readily see that grains with more than six sides will grow, those with fewer than six sides will shrink, and those with exactly six sides will neither grow nor shrink (although their shapes may change). This equation is known as the von Neumann-Mullins relation [31, 30] and has deeply influenced understanding of grain growth.

In the next section, we describe a front-tracking method for modeling cell structures that evolve via curvature flow and analyze the error associated with this method. We show that this error is remarkably small, even for a rather coarse discretization of the structure mesh.

3.4 Simulation method

Previous models

Over the last quarter century, numerous methods have been developed to study grain growth in two and three dimensions, including Monte Carlo Potts models [32, 33], cellular automata [34, 35], phase field models [36, 37], vertex models [38, 39], front tracking models [40], finite element models [41], molecular dynamics [42, 43, 15] and level set methods [44, 45, 46, 47]. Each approach has advantages and disadvantages. The Monte Carlo Potts model is both simple, easily implementable and extendable to a wide range of grain growth phenomena. Phase field models, like the Potts model, are based upon well-founded microscopic physics but have the advantage of being formulated in terms of continuum descriptions. Vertex models have the most compact data sets and, arguably, are based on the most fundamental objects in the microstructure — triple-junctions. Front tracking models have the advantage of well-defined equations of motion for boundary elements. Finite element methods naturally carry all material point information. Molecular dynamic simulations require an accurate understanding of the forces between atoms in the material and require enormous resources of memory and time to simulate macro-scale phenomena. In all cases, a discretization of the microstructure is involved, which necessarily compromises our ability to model the requisite grain growth physics in full fidelity.

A widely used numerical scheme for studying the evolution of surfaces, including those of cell structures, is the front tracking method as realized in the robust and versatile program Surface Evolver, developed by Brakke [48]. This program can track the evolution of grain boundaries moving via curvature flow in any number of dimensions. Several papers report grain growth simulation results based upon this method [49, 50, 51]. In this section, we develop a new approach for simulating grain growth that is based on front tracking ideas and that satisfies the von Neumann-Mullins relation at all times with very small error, regardless of discretization. In this chapter, we implement this new grain growth method in two dimensions and compare our results with those obtained using the Surface Evolver program. In Chapter 4 we will extend this method to three dimensions.

Proposed model

In Chapter 2 we considered four different initial conditions from which we began our simulations. This allowed us to demonstrate the existence not only of steady states but also of universal steady states, i.e. states to which almost all initial systems evolved after they had evolved for sufficient time. The increase in dimension significantly limits our simulations and in this chapter and the next we focus on data collected from simulations that begin with only one particular initial condition.

In two dimensions, we begin all simulations from a Voronoi tessellation. This construction involves placing "seeds" in the unit square and associating with each seed a particular region of the space. In practice, we choose N pairs of random variables from a uniform probability distribution on the interval [0, 1). These two random numbers become the x and y coordinates for a seed. With each seed we associate all points in the unit square that are closer to that seed than to any other seed. All points associated with one seed constitute a single cell. The simplicity of this construction allows its use in arbitrary dimension, and indeed we use this construction in the next chapter, in studying three-dimensional systems, as well. Figure 3.12 illustrates the construction of such a system with 1000 cells.



Figure 3.12: Constructing a Voronoi tessellation of $[0, 1]^2$ with periodic boundary conditions. Here we choose 100 points independently and randomly distributed, from a uniform probability density distribution on $[0, 1]^2$; these points are *seeds*. To each seed we associate all points in the space that are closer to that seed than to any other seed. All points associated with one seed constitute a single cell. We draw lines to show the boundaries between cells. We then discard the original seeds.

Data representation

Every grain is represented as an ordered list of points which lie along the boundary of the grain. Points located where three grains meet are called *triple-nodes*; points located along grain boundaries but that are adjacent to only two grains are called *boundary nodes*. Straight line segments connect adjacent boundary nodes and are called *edge segments*. Figure 3.13 shows two 5-sided grains surrounded by a few other grains. Larger dots and smaller dots are used to indicate triple-nodes and boundary nodes, respectively. Boundary nodes are occasionally added to make the discretization



Figure 3.13: Two 5-sided grains with a few neighboring cells. Each edge is broken into a few edge segments. Larger dots indicate the triple-nodes; smaller dots indicate the boundary nodes.
smoother, or removed when neighboring ones become too close. Each node in the system is represented by a data object which stores the node's x and y coordinates, its velocity, and some data about its neighboring grains and nodes. Each grain is represented by a data object which stores an ordered list of the nodes which make up the grain's boundary, as well as data about the grain's area, perimeter, and information about its neighboring grains.

Node motions

We develop a method for moving nodes in a manner that ensures that the von Neumann-Mullins relation is satisfied for all grains at all times with minimal error. To do this, we discretize the time variable t from Equation 3.5 into time steps Δt and solve for the change in grain area ΔA :

$$\Delta A = \frac{\pi}{3} M \gamma \left(6 - n\right) \Delta t. \tag{3.6}$$

For simplicity, we consider a discretization of a single isolated grain, illustrated in Figure 3.14. All nodes here should be considered boundary nodes. Moving the node σ changes the area of the shaded



Figure 3.14: An isolated grain with no triple-nodes.

triangle by the same amount as it changes the area of the entire grain. This ability to localize changes of area enables us to express each side of Equation 3.6 as a sum over all nodes surrounding a grain. The sum of the exterior angles around a discretized grain (a polygon) is exactly 2π . If we ensure that the area of each triangle changes by an amount $-\alpha M\gamma \Delta t$, where α is the exterior angle at its apex, then after each time step the area of the entire polygon will change by $-2\pi M\gamma \Delta t$, with an error of order $(\Delta t)^2$, owing to overlap between triangles associated with adjacent nodes; this error will be elaborated in detail in the next section. We point out that if moving a boundary node changes the area of one adjacent grain by ΔA , then it also changes the area of the other adjacent grain by $-\Delta A$.

We now shift attention to the case of triple-nodes, which we treat as boundary nodes with one correction. That is, we move triple-nodes in a way that changes the area of each neighboring grain by an amount $-(\alpha_i - \frac{\pi}{3})M\gamma\Delta t$. Since a grain with *n* neighbors has *n* triple-nodes, the area of a grain with *n* neighbors will change by an amount $-(2\pi - n\frac{\pi}{3})M\gamma\Delta t$, or $\frac{\pi}{3}M\gamma(n-6)\Delta t$, which is exactly the discretized von Neumann-Mullins relation for *n*-sided bodies, Equation 3.6.

We now provide a more precise description of the motion of each node, beginning with boundary nodes. Consider a boundary node σ with edges $\mathbf{e_1}$ and $\mathbf{e_2}$, as shown in Figure 3.15. The exterior angle between the two edges is $\alpha = \cos^{-1}(-\frac{\mathbf{e_1} \cdot \mathbf{e_2}}{|\mathbf{e_1}||\mathbf{e_2}|})$. Our goal is to move node σ by a displacement



Figure 3.15: A schematic of an isolated grain, emphasizing a boundary node σ and directed edges $\mathbf{e_1}$ and $\mathbf{e_2}$.

vector \mathbf{dv} that will change the area of the shaded triangle by $\Delta A = -\alpha M \gamma \Delta t$. Although we could move the node in many different directions (with an appropriate magnitude) to achieve the desired area change, for reasons of numerical stability we move the node in the direction of $\mathbf{e_1} + \mathbf{e_2}$. If α is the turning angle, or exterior angle, between the two edges, then to change the area of the triangle by exactly $-\alpha M \gamma \Delta t$, we let:

$$\mathbf{dv} = \alpha M \gamma \Delta t \frac{\mathbf{e_1} + \mathbf{e_2}}{|\mathbf{e_1} \times \mathbf{e_2}|}.$$
(3.7)

We next consider a triple-node τ where three grains meet, as shown in Figure 3.16. To satisfy the von Neumann-Mullins relation, displacement of the triple-node must change the area of each neighboring grain by $\Delta A_i = -(\alpha_i - \frac{\pi}{3})M\gamma\Delta t$, where α_i is the exterior angle at the triple-node with respect to grain *i*. Unlike boundary nodes, which have a degree of freedom in their solution, triple-nodes have exactly one solution. If $\alpha_1 = \cos^{-1}(-\frac{\mathbf{e}_1 \cdot \mathbf{e}_2}{|\mathbf{e}_1||\mathbf{e}_2|})$, $\alpha_2 = \cos^{-1}(-\frac{\mathbf{e}_2 \cdot \mathbf{e}_3}{|\mathbf{e}_2||\mathbf{e}_3|})$, and $\alpha_3 =$ $\cos^{-1}\bigl(-\frac{\mathbf{e_3}\cdot\mathbf{e_1}}{|\mathbf{e_3}||\mathbf{e_1}|}\bigr)$ then solving:

$$\begin{bmatrix} \mathbf{e_1} - \mathbf{e_2} \\ \mathbf{e_2} - \mathbf{e_3} \\ \mathbf{e_3} - \mathbf{e_1} \end{bmatrix} \mathbf{dv} = 2M\gamma\Delta t \begin{bmatrix} \alpha_1 - \frac{\pi}{3} \\ \alpha_2 - \frac{\pi}{3} \\ \alpha_3 - \frac{\pi}{3} \end{bmatrix}$$
(3.8)

for \mathbf{dv} will tell us exactly how to move the triple-node τ . Although this system may initially appear overdetermined, inspection will show that exactly two equations are independent, and so there will always be exactly one solution for \mathbf{dv} .

Equations 3.7 and 3.8 then describe the motions of all boundary nodes and all triple-nodes, respectively. These equations represent the displacement vectors for each node at every time step. More details regarding the two-dimensional method are provided in Appendix B.



Figure 3.16: A triple-node τ where Grains 1, 2, and 3 meet, where $\mathbf{e_i}$ are the vectors from the triple-node to the nearest nodes on each of the three boundaries, and α_i are the "turning angles", or exterior angles with respect to each of the three grains.

It is worth explaining here an important difference between our method and other front-tracking methods. At first sight, the motions described above may not resemble motion by curvature flow: we do not attempt to move nodes in their normal directions and with a magnitude proportional to their local curvature. Indeed, a more naive approach may attempt to define a normal direction at every node, and move each node in that direction by an amount proportional to the mean curvature at that point. The problem is that although this method might accurately describe the displacements of a finite set of points in the system, where the nodes are placed, it does not adequately describe the motion of any other point in the system. In the current method, rather than considering the velocities of a finite set of points and moving the surface as those points require, we consider the way in which the area of a grain changes with the movement of entire sections of the boundary. In this way, our method allows us to satisfy the exact von Neumann-Mullins relation without resorting to an arbitrarily refined surface mesh.

Topological changes

Aside from calculating the velocities of all nodes in the system, we also must change the local "topology" from time to time. Edges and entire grains shrink and disappear, and the network must be adjusted accordingly. In two dimensions, the number of topological changes is fairly limited. The most frequent topological change is the switching of an edge lying between two grains and adjacent to two other grains. Figure 3.17(a) shows the "neighbor-switching", or T1, process [52]. The second



Figure 3.17: Illustrations of T1 and T2 processes; figures taken from [52].

most frequent topological change is the removal of a three-sided grain, illustrated in Figure 3.17(b). In this process, three edges and triple-nodes shrink down to one triple-node.

Weaire and Kermode, the authors who first named the T1 and T2 processes [53, 52] noted that C.S. Smith [54] had asserted many years earlier that all shrinking cells eventually become three-sided and then disappear as such. In other words, Smith believed that a two-sided grains could not arise in a cell-structure evolving through curvature flow if it did not exist in the initial structure. In an appendix to that same paper [53], the authors point out that experimental results of two-dimensional soap foams reported in [55] contain no two-sided grains and they argue based on theoretical grounds why this should generally be the case. Although they report two-sided grains in their own simulations, they attribute this to numerical error.

However, other early papers [56, 57, 58] have noted the presence of two-sided grains and their disappearance in computer simulations, and believe that their occurrence is not due merely to numerical error. We also find such grains in our simulations and believe that such grains can arise

in real systems. The disappearance of a two-sided grain is often called a T3 process [58, 59, 60, 61] and is illustrated in Figure 3.18.



Figure 3.18: T3 process.

Implementing each of these changes in our code is relatively simple. If we observe that a certain edge is shrinking too fast, we decide to implement a T1 change and begin by removing all boundary nodes along that edge. Next, we remove the two triple-nodes associate with the edge that we are deleting, and create two new ones. We attach edges appropriately, as illustrated in Figure 3.17(a). The locations of the two new nodes are such that the new edge is perpendicular to the old edge and its length is roughly the same. Occasionally we run into trouble with the local geometry and adjust the length of the new edge in a way to make it "fit nicely" into the local structure.

Implementing a T2 change is also straightforward. We first remove all boundary nodes along the grain's boundary. Next we create a new triple-node at the geometric center of the grain's three corner nodes. We then attach the three nodes which had been the corners of this grain to the new triple-node, and then adjust the data structures of the three adjoining grains appropriately. Last, we remove the data object associated with the grain itself.

Implementing the T3 change is also very straightforward. We begin by removing all boundary nodes along the grain's boundary. Next we create a new edge between the two corner nodes which had been the corners of the this grain and adjust accordingly the neighboring grains. Last, we remove the data object associated with the two-sided grain.

More complicated topological changes can be decomposed into series of such changes as already noted in [52]. For example, a four-sided grain that is shrinking can be removed by a T1 process, followed by a T2 process. In a similar way it seems that all topological changes to the system can be decomposed into a series of these three changes. In theory, we could decompose a T2 change into a T1 and T3 change. That is, a triangle can be first changed into a two-sided grain using a T1 change and then removed as a two-sided grain with a T3 change. However, it seems that T2 changes indeed occur, and their decomposition seems unnatural. It is not clear whether or not a 4- or 5-sided grain can really disappear as such, or whether the occurrences of these in simulations is only a result of numerical error. It would be nice to prove what topological events can occur during curvature flow on cell structures in a continuous setting.

3.5 Error analysis

One strength of our method is the low error inherent in the refinement of the time step and the surface mesh. We first deal with the error associated with the time step, which is of order $O(\Delta t^2)$. We also derive the error associated with Brakke's method [48], and find that error to be of order $O(\Delta t)$.

We define the error as follows. After one time-step, the area of an individual grain *should* change exactly:

$$\Delta A = -2\pi M \gamma \left(1 - \frac{n}{6}\right) \Delta t. \tag{3.9}$$

as explained earlier. We define the error to be the difference between this exact, theoretical result and what actually happens to a grain after one time-step. If ΔA_B is the change in area of a grain after one time-step of size Δt using the Brakke method, then the absolute error is $E_B = |\Delta A - \Delta A_B|$; the relative error is $\epsilon_B = |E_B/\Delta A|$. Likewise, if ΔA_P is the change in area of a grain after one time-step using the proposed method, then the absolute and relative errors are $E_P = |\Delta A - \Delta A_P|$ and $\epsilon_P = |E_P/\Delta A|$, respectively.

We first show that the absolute numerical error involved in our method is of order $O(\Delta t^2)$. To do so, we consider moving an individual node to change the area of a grain by $-\alpha M\gamma \Delta t$ in the case of a boundary node or $-(\alpha_i - \frac{\pi}{3})M\gamma\Delta t$ in the case of a triple-node. If we keep all other nodes fixed, there is no error (up to machine precision). This is because the proposed method moves the nodes in a manner that is consistent with the exact von Neumann-Mullins relation, as explained above.

If we move all nodes simultaneously, errors result from "interference" between motions of neighboring nodes. Consider for example the edge segment shown in Figure 3.19. If we move only node n_1 by $\mathbf{dn_1}$ and leave all other nodes fixed, the area of the grain will change by exactly -(a+b+c). Similarly, if we move only n_2 by a motion $\mathbf{dn_2}$, leaving all other nodes fixed, the area of the grain changes by -(c+e+f). When we move both nodes simultaneously, we want to change the area of the grain by a+b+2c+e+f. However, the simultaneous motions of nodes n_1 and n_2 instead change the area of the grain by a+b+2c+e+f. This produces an absolute error $|d-c| = \frac{1}{2} |\mathbf{dn_1} \times \mathbf{dn_2}|$. Since each $\mathbf{dn_i}$ is linear in Δt , the cross product $\mathbf{dn_1} \times \mathbf{dn_2}$, and hence the the error resulting from the "interference" of the two motions, is linear in $(\Delta t)^2$.



Figure 3.19: Representation of part of an edge between two bodies.

Similar errors occur for all pairs of adjacent nodes moving simultaneously. Because all grain have a finite number of these "interference" errors, the total error involved in the motion of all nodes of one grain will be $O(\Delta t)^2$. More precisely, the error is given by:

$$E_P = \frac{1}{2} \Big| \sum_{i=1}^{N} \mathbf{dn_i} \times \mathbf{dn_{i+1}} \Big|, \qquad (3.10)$$

where the summation is over all node motions dn_i of a particular grain.

In the Brakke method, the displacements of the nodes are also linear in Δt , and so it too produces "interference" errors of order $(\Delta t)^2$. However, in the Brakke case, the displacement of each node results in errors that are $O(\Delta t)$ even when all other nodes are fixed. When summed over all nodes in a grain, we have a total error that is also of order $O(\Delta t)$. We demonstrate this for the case of a grain shaped like a regular polygon.

Consider an isolated grain represented by a regular polygon of m sides and radius r, as shown in Figure 3.20. The area of this grain is $\frac{1}{2}mr^2 \sin\left(\frac{2\pi}{m}\right)$. After one time-step, the area of this shape should change by $\Delta A = -2\pi M \gamma \Delta t$, as explained above. In the Brakke scheme³, evolution by one time-step changes the radius r of the grain by $-\frac{M\gamma\Delta t}{r\cos(\frac{\pi}{m})}$, while in the proposed method, evolution by one time-step changes the radius by $-\frac{2\pi M\gamma\Delta t}{mr\cos(\frac{2\pi}{m})}$. The corresponding changes in the grain area

³We used the **area normalization** and **effective area** options in Surface Evolver. These options are meant to approximate motion by mean curvature. With these options, resistance to motion of a node is proportional to the component of the area associated with that node which is also perpendicular to the force on the node. This method is explained in greater detail in Appendix B and in [62].



Figure 3.20: Regular polygonal grain with m sides and of radius r.

are

$$\Delta A_B = -M\gamma \left[2m\sin(\frac{\pi}{m})\Delta t - \frac{m}{r^2}\tan(\frac{\pi}{m})(\Delta t)^2 \right]$$
(3.11)

$$\Delta A_P = -M\gamma \left[2\pi\Delta t - \frac{2\pi^2 (\Delta t)^2}{mr^2 sin(\frac{2\pi}{m})} \right].$$
(3.12)

The absolute and relative errors are therefore

$$E_B = M\gamma \left[\left(2\pi - 2m\sin(\frac{\pi}{m}) \right) \Delta t + \frac{m}{r^2} \tan(\frac{\pi}{m}) (\Delta t)^2 \right]$$
(3.13)

$$E_P = M\gamma \left[\frac{2\pi^2 (\Delta t)^2}{mr^2 sin(\frac{2\pi}{m})} \right]$$
(3.14)

$$\epsilon_B = \left(1 - \frac{m}{\pi}\sin(\frac{\pi}{m})\right) + \frac{m}{2\pi r^2}\tan(\frac{\pi}{m})(\Delta t)$$
(3.15)

$$\epsilon_P = \frac{\pi \Delta t}{m r^2 \sin(\frac{2\pi}{m})}.$$
(3.16)

The absolute error in the Brakke method has a leading order term proportional to Δt for all m; the relative error has a leading order term entirely independent of the time-step Δt ! On the other hand, in the proposed method, the leading order term in the absolute and relative errors are $(\Delta t)^2$ and Δt , respectively.

To show the difference in scale between ϵ_B and ϵ_P , we evaluate Equations 3.15 and 3.16 for a fixed radius r and time step Δt taken from a typical simulation when 20,000 grains remain. With 20,000 grains, the approximate average radius of a grain is r = 0.003989L and the step size is $\Delta t = 1.21 \times 10^{-9} M \gamma L^2$. Table 3.1 shows relative errors calculated for the two methods with different values of m. Clearly, the relative error in the proposed method is several orders of magnitude smaller than in the Brakke method.

m	ϵ_B	ϵ_P
3	17.31%	0.00919%
4	9.97%	0.00597%
5	6.46%	0.00502%
6	4.51%	0.00459%
7	3.33%	0.00436%
8	2.55%	0.00422%
9	2.02%	0.00413%
10	1.64%	0.00406%
11	1.36%	0.00401%
12	1.14%	0.00398%

Table 3.1: Relative errors ϵ calculated for the change in area of a grain discretized into *m* segments using the Brakke method and that proposed here, using Equations 3.15 and 3.16. The errors are calculated for a regular polygon grain with radius r = 0.003989L and a step size $\Delta t = 1.21 \times 10^{-9} M \gamma L^2$.

Similar results can be obtained for grains with different numbers of neighbors. These calculations show that for both vertex and non-vertex nodes, the Brakke scheme produces an absolute error that is, to leading order, proportional to Δt , while in the proposed method the absolute error is proportional to $(\Delta t)^2$. Examination of Equations 3.13 through 3.16 shows that both methods lead to identical errors in the limit where the number of sides m tends to infinity.

3.6 Results

Microstructure Evolution

The evolution of a typical microstructure using the proposed method is shown in Figure 3.21; the microstructure began as a Voronoi tessellation based on 1000 randomly distributed seed points. The initial microstructure has grains with straight edges and triple-nodes where three edges do not generally meet at $2\pi/3$. However, after a short time, the triple-node angles become very close to $2\pi/3$ and many of the boundaries are curved. The structure coarsens over time resulting in fewer grains and a larger average grain size. Figure 3.22 shows a comparison of two microstructures, starting from the same Voronoi initial condition state, where one has been evolved using the method developed by Brakke and the other has been evolved using the method proposed here. While the microstructures appear similar, a grain-by-grain comparison shows that they are microscopically quite different. In the following sections, we offer more quantitative descriptions of these systems that highlight the differences between the two evolution methods.



Figure 3.21: Temporal evolution of a microstructure based upon the proposed method for $M\gamma L^2 = 1$. This microstructure was initialized as a Voronoi tessellation of the unit square into 1000 grains.



Figure 3.22: Microstructures evolved from a single Voronoi tessellation of 1000 grains after half of the grains have been consumed, using (a) the Brakke method and (b) the proposed method.

Grain Size Evolution

While the microstructure generated using the Brakke method and that generated using the method proposed here appear similar, the comparison presented above is not quantitative. In this section we look at how the average grain size changes over time using these two approaches. To this end, we simulate the evolution of four different microstructures using the two approaches, each initialized by Voronoi tessellations based on random distributions of 25,000 points. For each method we considered two cases: a refined system, where each grain boundary is represented by approximately 5 line segments (i.e., placing 4 boundary nodes between each pair of triple-nodes) and an unrefined system, where nodes are placed only at points where three grains meet. We evolved these system until 1000 grains remained.

Figure 3.23 shows the change of the average grain area over time, averaged over four runs, for each of the four cases described above: Brakke method-refined, Brakke method-unrefined, proposed method-refined, and proposed method-unrefined. In all four cases, the average grain size appears to



Figure 3.23: The temporal evolution of the average grain area $\langle A \rangle$ for the four cases described above; four samples were run for each case and the averages are plotted. Each system began with 25,000 grains; when t = 0.0008, there remain slightly more than 1000 grains. The typical error is about two or three times the size of the dots; the errors are not shown for the sake of clarity.

grow linearly with time, albeit at slightly different rates. For the refined cases shown in Figure 3.23, the slopes of the curves are 1.067 and 1.092 for the Brakke and proposed simulations, respectively. While these slopes are close to 1, we know of no rigorous analytical results that predict them; both are close to the 1.12 ± 0.04 reported in [58]. See also [63] which records slopes ranging from 0.5 to 20 obtained by various other methods and reported elsewhere.

While these results show the evolution behavior of entire systems, the exact von Neumann-Mullins relation (Equation 3.5) describes how each individual grain evolves; i.e., at a constant rate that depends only on its number of sides. Figure 3.24 shows the area growth rates at a single time step for each of the 20,000 grains in a system that was evolved from a 25,000 grain Voronoi microstructure using the Brakke and proposed methods together with a refined discretization. When



Figure 3.24: Area growth rates $\Delta A_i/\Delta t$ for each grain in a 20,000 grain system for one time step using a refined discretization using (a) the Brakke method and (b) the proposed method. We assign a random number to each grain. In this simulation $M\gamma = 1$.



Figure 3.25: Area growth rates $\Delta A_i/\Delta t$ for each grain in a 20,000 grain system for one time step using an unrefined discretization using (a) the Brakke method and (b) the proposed method. We assign a random number to each grain. In this simulation $M\gamma = 1$.

 $M\gamma = 1$, these figures should show sharp, horizontal lines at integer values of $\frac{3\Delta A}{\pi\Delta t}$, where each line corresponds to a different number of grain neighbors n. Figure 3.24(b) is an excellent description of the results for the proposed method. However, there is discernible scatter in the data for the Brakke method results, Figure 3.24(a). Furthermore, the average values of $\frac{3\Delta A}{\pi\Delta t}$ differ slightly from the von Neumann-Mullins prediction that these should all be integers.

Figure 3.25 shows results similar to those shown in Figure 3.24, but for the unrefined discretizations which contain boundary nodes. The proposed method shows results that accurately match the von Neumann-Mullins relation even in this unrefined discretization. However, the scatter in the results from the Brakke calculations is very much increased compared with that for the refined discretization. Even more problematic is that the mean position of each set of horizontal lines in Figure 3.25(a) is in strong disagreement with the prediction of the von Neuman-Mullins exact relation. This again emphasizes the necessity for maintaining a sufficiently refined discretization in the Brakke calculations. The robustness of the proposed method for *any* discretization is one of its main advantages.

The data in Figures 3.25(a) and 3.25(b) can be summarized in a plot of $\Delta A/\Delta t$ versus n. Figure 3.26 shows data collected from a microstructure beginning with 10,000 grains and evolving until 5000 grains remain, using both refined and unrefined discretizations. The best fit line through each set of data is $\frac{3\Delta A}{\pi t} = 0.99997n - 5.9998$. The errors here, provided by comparing this equation with Equation 3.6, are several orders of magnitude smaller than the 1 - 3% errors reported in [58, 64]. That the results are so accurate even for the unrefined microstructure demonstrates another strength of the proposed method.



Figure 3.26: Validation of the von Neumann-Mullins relation from data obtained using the proposed simulation method with both a refined and an unrefined discretization; $M\gamma$ is set to 1. The von Neumann-Mullins relation predicts that the points will fall on the line shown. The points here represent data averaged over all grains and time steps in simulations that began with 10,000 grains and ended with 5000. The error bars here show the standard deviations magnified by a factor of 100.

Distributions

We also examine the distribution of grain topologies (number of sides) and areas for the different systems. Figures 3.27 and 3.28 show these distributions after the systems have evolved from 25,000 grains until only 5000 grains remain. Examination of the distribution of grain topologies in Figure 3.27 and of grain areas in Figure 3.28 shows that the Brakke and proposed methods yield nearly

identical results when refined. However, when unrefined, both methods yield results that diverge from their refined versions. In particular, the unrefined Brakke method produces more grains with small numbers of faces than both refined versions, whereas the unrefined proposed method produces fewer such grains. Likewise, in considering grain areas, the unrefined Brakke method produces too many very small grains. whereas the unrefined version of the proposed method produces too few, though the discrepancy here is much smaller.



Figure 3.27: Topological (number of sides, n) distributions for microstructures evolved from four Voronoi tessellations of 25,000 grains until only a fifth of the grains remain.



Figure 3.28: Normalized grain area distributions $(A/\langle A \rangle)$ for microstructures evolved from four Voronoi tessellations of 25,000 grains until only a fifth of the grains remain.

The observation that simulations that produce too many grains of small areas also produce too

many grains of few sides and visa versa is not surprising in light of Lewis's Law [65, 66], which states that the area of a grain is linearly related to its number of sides. Moreover, the excess of small grains in the Brakke method on an unrefined mesh can be understood by reference to Figure 3.25. This figure shows that grains with few sides shrink more slowly than they should according to the von Neumann-Mullins exact result. We also note that this figure shows that large grains grow too slowly. This is also consistent with the distributions, although this effect is weaker.

3.7 Conclusions

The primary purpose of this chapter was to introduce curvature flow on two-dimensional cell structures and a method we use to simulate such systems. Although we report some data in this chapter, the bulk of our results are reported in Chapter 5, where we compare results from two- and threedimensional systems.

In future work we might consider in greater detail simulations that begin from various initial conditions. J.K. Mason has done some work [67] in simulating systems that begin from a variety of initial conditions, including hexagonal cell structures into which some topological defects have been introduced (a system which exhibits very beautiful evolution). Considering various initial conditions will help us understand how curvature flow evolves cell structures towards a steady state that is independent of initial conditions.

We might also consider other dynamics on two-dimensional cell structures. One particular dynamic worth considering is the coarsening of foams. Although curvature flow also plays a role here, the evolution of these systems is qualitatively different from that considered in this chapter. This is due to the ability of gas to diffuse within a bubble rapidly and to allow the boundary to assume a minimal surface area with a fixed volume at a time-scale much faster than the diffusion of air through boundaries. J.K. Mason has performed some work on these systems, though the intrinsically non-local nature of this dynamic allows for the simulation of only relatively small systems [68].

One area that is also worthwhile pursing is the case of anisotropic boundary energy and boundary mobility. That is, what happens when we consider systems in which the speed of a boundary's motion depends on the misorientation between two neighboring grains? How will this effect the steady state of our dynamical cell systems? This area has been studied extensively [69, 70, 71], though not with a precise front-tracking model like the one described here.

In a more theoretical vein, we would like to understand what types of events can occur in these

systems. We have pointed out that for curvature flow on closed, smooth curves, no singularities arise before the curve disappears as a circle. We can ask similar questions regarding continuous cell structures that evolve via curvature flow. For example, we have noticed in our simulations that edges never intersect except at triple points. In theory, can edges intersect at other places? In a similar vein we notice in our simulations that grains disappear with two or three sides. In theory, can grains with four or five sides disappear before becoming triangular? Once a system has reached steady state, at what frequencies do T1, T2, and T3 changes occur?

Chapter 4

3D Systems

In this chapter, we describe mean curvature flow of surfaces and three dimensional cell structures. We extend the simulation method introduced in Chapter 3 to three-dimensional cell structures that evolve through mean curvature flow, and show that this method satisfies the MacPherson–Srolovitz relation with high accuracy. We demonstrate this method can simulate large systems with as many as 100,000 grains, large enough to collect significant statistics well after the systems have reached the steady-state. We report statistics from a number of simulations beginning with a total of either 800,000 or 2.5 million grains and reaching steady states with more than 80,000 or 250,000 grains.

4.1 Motivation

One of the persistent objectives in materials science is the characterization of three-dimensional microstructures. Current knowledge of grain growth in polycrystalline materials relies on a combination of experimental observations of grain growth microstructures, developments in the theory of thermodynamics and kinetics of materials, and more recently, simulations of microstructural evolution. This understanding is limited by the difficulty of experimentally accessing the details of a three-dimensional microstructure, by the anisotropy of properties that govern grain boundary motion, and by impurities and inhomogeneities in real materials. Hence, despite recent advances in three-dimensional microscopy [73, 74], in characterizing the anisotropy of grain boundary energies and mobilities [75, 76], and in analyzing the effects of impurities [77, 78], the importance of theory and simulations developed for simplified and idealized microstructures cannot be overstated. Indeed, a model that assumes material isotropy, a grain boundary motion governed only by interfacial ther-

^{*}The content of this chapter has been adapted from [72].

modynamics, and the absence of impurities provides fertile ground for investigating basic features of the complicated grain growth process. As an example, these idealizations have permitted an exact solution for the rate of growth of any given grain within a microstructure. This was done first in two dimensions by von Neumann [31] and Mullins [30], and more recently in three and higher dimensions by MacPherson and Srolovitz [79].



Figure 4.1: An example of a simulated three-dimensional microstructure with 250 grains.

Nonetheless, even with this idealized model and the accompanying exact solutions, our understanding of the properties of the evolving microstructure remains limited. Difficulties in developing a theory of the broader microstructure arise principally in understanding the interactions between grains and from the many degrees of freedom of a grain boundary network in three dimensions. Although simulations have provided important insight into the mechanisms and energetics of grain growth [80, 70], contemporary methods do not rigorously adhere to the same assumptions of isotropy and curvature-driven grain boundary motion that have proven instrumental in developing the theory for individual grains. In this chapter, we extend the method introduced in Chapter 3 to threedimensional systems. This method enables us to simulate grain growth in an idealized isotropic microstructure that evolves via curvature-driven grain boundary motion. This new method for three-dimensional grain growth obeys the exact relation for the rate of growth of every grain, is more accurate and more efficient than prior methods, and is capable of investigating microstructural properties previously overlooked.

4.2 Mean curvature flow on 3D cell structures

Mean curvature flow on surfaces

We mentioned in the last chapter the recent interest in geometrical flows and some results of this work. In this section we first describe mean curvature flow on differentiable manifolds, the more traditional setting for this flow. Afterwards we explain how mean curvature flow can be applied to cell structures, which are clearly not manifolds.

We let $\{M_t\}$ be a family of surfaces embedded in \mathbb{R}^3 . At each point $x \in M_t$ we define a mean curvature vector $\mathbf{H}(x) = H\nu$, where the mean curvature $H = (k_1 + k_2)$ is the sum of the two principal curvatures k_1 and k_2 and ν is the outer unit normal at x. We say that M_t evolves by mean curvature if it satisfies the differential equation:

$$\frac{\partial}{\partial t}x = -c\mathbf{H}(x),\tag{MCF}$$

where $c \in \mathbb{R}^+$ is some constant factor that might depend on physical properties of a particular system.

A very simple example is given by a one-parameter family of shrinking spheres $M_t \subset \mathbb{R}^3$ centered at the origin. If the radius of the sphere is given by r(t), then every point $x \in M_t$ moves towards the center of the sphere with a velocity $-c_{\overline{r^2}}^2 \mathbf{x}$ at all times. The radius r thus changes:

$$\frac{dr}{dt} = -\frac{2c}{r}.\tag{4.1}$$

If $r(0) = r_0$, then the radius at time t is

$$r(t) = \sqrt{r_0^2 - 4ct}.$$
(4.2)

The sphere then disappears as a point at time $\frac{r_0^2}{4c}$; the flow is not defined after this time.

A number of beautiful results about mean curvature flow on smoothly embedded closed surfaces have been obtained. It is known that any bounded convex surface will stay convex and become asymptotically spherical before disappearing, without developing other singularities [12]. It is also known that some non-convex surfaces will develop singularities before disappearing. For example, some dumbbell-shaped surfaces will pinch off at the neck well before the volume or surface area disappears [13]. An illustration of this type of singularity can be found in Section 1.3.

Unlike the rate of change of area bounded by a simple closed curve that evolves via curvature

flow, the rate of change of volume bounded by a smoothly embedded closed surface that evolves via mean curvature flow is *not* constant. The reason for this difference is that in two dimensions, the integral of the signed curvature around a closed curve is always 2π . The change in area of the bounded region, which depends on this integral, is therefore also constant. In three dimensions, the mean curvature of a surface (unlike the Gaussian curvature) is an extrinsic property and depends on the particular embedding of the surface. The rate of change of volume of the bounded region is therefore not constant. However, we can still derive a similar formula for the change of volume in terms of the integral of the mean curvature. If V is the volume bounded by a smoothly embedded closed surface S, c is a constant controlling the speed of the surface, H is the mean curvature, and dA is an element of the boundary, then:

$$\frac{dV}{dt} = -c \int_{S} H dA. \tag{4.3}$$

This equation describes the way in which the volume bounded by a smoothly embedded closed surface changes as the surface evolves via mean curvature flow.

We return to the example of an embedded 2-sphere S^2 of radius r(t) evolving via mean curvature flow. At what rate does the volume bounded by the sphere change? The principal curvatures of a sphere are uniformly 1/r and 1/r, and so the mean curvature is $\frac{1}{r} + \frac{1}{r} = \frac{2}{r}$. We therefore have:

$$\frac{dV}{dt} = -c \int_{S^2} \frac{2}{r} dA$$
$$= -4\pi cr^2 \frac{2}{r}$$
$$= -8\pi cr$$

Substituting $\sqrt[3]{\frac{3V}{t\pi}}$ for r and solving for V(t) we have:

$$V(t) = \frac{\sqrt{3}}{9} \left(3V_0^{2/3} - 16\pi c \sqrt[3]{3/4\pi t} \right)^{3/2}.$$
(4.4)

Solving for V(t) = 0, we see that the sphere will disappear at time $t = \frac{r_0^2}{4c}$, as derived earlier.

We now turn to three-dimensional cell structures. Two important features of cell structures which evolve via mean curvature flow are the way in which faces meet at edges, and the way that edges meet at vertices. At most three faces meet along an edge, and they always meet locally at angles of 120°. Figure 4.2 shows a typical edge in a cell structure that is evolving via mean curvature flow. The cross-section shows the way in which three faces meet at equal angles at any point along an



Figure 4.2: On the left is a typical edge at which three faces meet. A cross-section of the faces in a direction perpendicular to the edge shows that three faces always meet at angles of 120° . Four edges meet at a quadruple point, shown on the bottom left and bottom right, and the angles between any two edges meeting there is $\cos^{-1}(-1/3) \approx 109.5^{\circ}$.

edge. The reason for this is that mean curvature flow is the gradient flow of the area functional and thus minimizes the total surface area of the structure. This forces faces to meet at equal angles much in the same way that curvature flow in the plane forces edges to meet at equal angles, as explained in Section 3.3. This area-minimizing geometry can be seen as a higher-dimensional analogue of the Fermat problem explained there. Moreover, the total surface area of a structure with four faces meeting along a single edge can almost always be reduced by "splitting" the edge into two, much in the same way that "splitting" a vertex almost always reduces the sum of edge lengths of four edges meeting at a point.

Likewise, at most four edges meet at a point, and they almost always meet locally at angles of $\cos^{-1}(-1/3) \approx 109.5^{\circ}$. This also results from the area-minimizing effect of mean curvature flow. These two results will be important in understanding how the volume of a grain changes with time. We point out that although finite surfaces can only move at a finite rate, an infinitesimal neighborhood around an edge or a point can move infinitely fast, ensuring that these conditions are satisfied at all times to a first order approximation.

Mean curvature flow on cell structures

We now generalize Equation 4.3 to the case of an individual cell in a cell structure where surfaces evolve via mean curvature flow. Equation 4.3 states that the volume of a region bounded by a smoothly embedded closed surface which is evolving via mean curvature flow changes at a rate proportional to the integral of the mean curvature over that surface. The next step is to understand how this can be generalized to cells which have "edges" and "vertices", which are not differentiable. The important point to bear in mind is that to a first order approximation, edges and vertices are energetically stable, as explained above. Therefore, in a first order approximation they do not move and hence do not contribute to the change in volume of any adjacent grains. Measuring the way in which the volume of a grain changes over time involves integrating the mean curvature over all faces of the cell and ignoring the edges and vertices, which to a first approximation do not contribute to the change in volume. We then have:

$$\frac{dV}{dt} = -c \sum_{f \in F} \int H dA, \tag{4.5}$$

where F is the set of faces, H is the mean curvature, and dA is an element of a face. In grain growth, the constant c is generally set to $M\gamma$, where M and γ describe the grain boundary mobility and grain boundary energy, respectively.

A possibly more elegant, and perhaps more enlightening, way of writing this equation uses a linear measure called the mean width. The *mean width* is a natural additive measure that describes the linear size of an object. Defined most generally, the mean width is a complicated integral based on the Euler characteristic of various cross-sections of an object. For our purposes, it suffices to state its relationship with the integral of the curvature over the boundary of a grain. If $e_i(\mathbf{D})$ are the lengths of all edges of a grain \mathbf{D} and all turning angles along the edges are $\pi/3$, then the integral of mean curvature over the grain boundary $\partial \mathbf{D}$ is given by:

$$\int_{\partial \mathbf{D}} H dA = 2\pi \left(\mathcal{L}(\mathbf{D}) - \frac{1}{6} \sum_{i} e_i(\mathbf{D}) \right)$$
(4.6)

This identity is proven in the Supplementary Information section of [79]. We combine Equations 4.3 and 4.6, setting $c = M\gamma$, to give us the rate of change of volume of a grain **D**:

$$\frac{dV(\mathbf{D})}{dt} = -2\pi M\gamma \left(\mathcal{L}(\mathbf{D}) - \frac{1}{6} \sum_{i} e_i(\mathbf{D}) \right)$$
(4.7)

The last term on the RHS is a direct result of the equilibrium angles of the edges, $\pi/3$. Because they are stable to a first approximation, they do not contribute to volume changes, and thus need to be subtracted from the mean width, which includes the curvature concentrated there.

4.3 Simulation method

4.3.1 Previous models

Prior simulations of three-dimensional grain growth can be roughly divided into three categories. One of the earliest and most frequently implemented is the Monte Carlo Potts model [33, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91], where a three-dimensional volume is broken up into voxels and every voxel is assigned a label. Contiguous voxels with the same label constitute a grain, and the energy of the system is proportional to the number of interactions between voxels with dissimilar labels. The microstructure is evolved by randomly reassigning the label of a randomly chosen voxel, and the relabeling is accepted or rejected based on the change in the energy of the system. While implementing this method is relatively straightforward, various complications quickly arise. The anisotropy of the underlying lattice structure effectively induces an anisotropic boundary energy [33, 92, 93], though this may be mitigated by increasing the interaction length of the voxels. Meanwhile, the discretization of space weakens the vertex angle boundary conditions [33, 92], causes boundary pinning and alters the grain growth kinetics [33, 92, 93], and significantly complicates the measurement of topological and geometric features [91]. Nevertheless, several of the most extensive three-dimensional grain growth simulations appearing in the literature use the Monte Carlo Potts model.

Closely related to the Monte Carlo Potts model is the cellular automaton model [34, 94, 95, 96]. This uses the same discretization of space and therefore inherits many of the same limitations as the Potts model, including the possibility of faceting and lattice pinning due to the lattice-based discretization of the microstructure. However, this model allows for more flexibility in defining the energetics and transition probabilities. Recent work has been able to simulate systems with roughly 30,000 grains [96].

The second broad category of simulations may be described as phase-field models [37, 97, 98, 99]. The fundamental idea of these models is to represent the volume occupied by a grain as the region where the value of a nominally continuous function goes to one. Every one of these functions is regarded as a distinct order parameter, and the evolution of the system is governed by the time-dependent Ginzburg-Landau equations [100]. Modifications to the energy functional allow for the evolution to depend on a variety of conditions including gradients in the stress and temperature [101]. Practically speaking, this is a coupled system of nonlinear partial differential equations that is not necessarily stable [102, 99] and must be solved on every time step. This solution generally involves the use of finite difference approximations and the discretization of space on a scale significantly finer

than the Monte Carlo Potts model, since the kinetics depend sensitively on the profile across the boundary [103, 104]. Furthermore, the computational resources required for a naive implementation of this model with an appreciable number of grains quickly become prohibitive, limiting the extent of the microstructure [37, 97, 98]. Certain variations on these models exist, known as the continuum-field [37, 98, 99] or multi-phase-field [97] models, though the difference in the results is slight [105].

The simulation of three-dimensional grain growth by diffusion-generated motion [106, 107] shares certain characteristics with the phase-field models, particularly the representation of a grain by a function defined over a lattice in the ambient space. This approach appears to be unconditionally stable, and its relative simplicity allows for the simulation of a significant number of grains. However, this model was developed only recently and has not been widely adopted.

The final category of simulations are front-tracking methods, including vertex models [108, 39, 58, 109], finite element models [110, 111, 112, 113, 114], and the algorithm introduced in [16, 72] and employed in this thesis. These models rely on the observation that the evolution of the microstructure is defined principally by the motion of the grain boundaries. The grain boundary network may therefore be explicitly triangulated and evolved in continuous space without devoting any memory or processing power to the grain interiors. While computationally efficient and convenient for the measurement of geometric and topological features, this approach requires that all topological events that naturally occur while the system evolves must be predicted and handled explicitly in the implementation. This complexity, with the added complication of mesh management in three dimensions, limits the widespread implementation of these models.

Numerous variations on this basic principle exist, depending on the triangulation of the interfaces and the equations of motion for the boundaries. Some of the earliest are referred to as vertex models [108], since they avoid triangulating the grain boundaries themselves and only follow the motion of the triple points in two dimensions and the quadruple points in three dimensions. However, since the quadruple points around a given face do not necessarily sit in a plane, simulations of the three-dimensional case quickly led to more sophisticated triangulations [39, 109] and have continued to become more flexible [111, 113, 114]. Some of the more recent ones triangulate the entire microstructure including the grain interiors [110, 112]. However, the computational expense associated with preserving the mesh quality has limited these simulations to fewer than 5000 grains.

The majority of front-tracking algorithms derive equations of motion for the grain boundaries by observing that the energy of the system is proportional to the total surface area and attempt to evolve systems in the direction of steepest descent in the energy [108, 39, 58, 109, 110, 111, 112, 114]. This would be equivalent to mean curvature flow in a continuous setting, though defining mean curvature flow for a triangulated surface is a delicate matter. One of the notable features of our simulations [16, 72] is that our equations of motion instead derive from the von Neumann–Mullins [31, 30] relation in two dimensions and the MacPherson–Srolovitz relation [79] in three dimensions, meaning that our simulations satisfy the constraints imposed by these relations to high accuracy. Furthermore, the fact that the edges of a grain in two dimensions and the faces of a grain in three dimensions are generally not highly curved [108] justifies our use of a restricted triangulation to reduce the computational requirements. This enables us to perform simulations beginning from a total of 10,000,000 grains in two dimensions and 2,500,000 grains in three dimensions, well beyond the range of contemporary front-tracking models [109, 111] and rivaling the most extensive of grain growth simulations by any method [97, 89, 107].

In Chapter 3, we introduced a front-tracking method for two-dimensional grain growth that accurately satisfies the von Neumann-Mullins relation [31, 30]. In this chapter, we extend this method to three dimensions, maintaining the same high level of accuracy in satisfying the exact MacPherson–Srolovitz relation [79]. Specifically, the error involved in displacing nodes during a time step Δt is of order $O(\Delta t^2)$, and the relative error in the rate of change of the volume of a grain from the discretization of the surface mesh is found to be small. Furthermore, our implementation enables us to model large systems, observe them as they evolve into steady-state microstructures, and examine many of their topological and geometrical properties in a very straightforward manner. In this chapter and the next, we provide compiled statistics from systems that are larger than those currently appearing in the literature.

4.3.2 Initial conditions

We begin all simulations from Voronoi tessellations of the unit cube with periodic boundaries. This construction involves placing "seeds" in the unit cube and associating with each seed a particular region of the space. In practice, to construct an initial structure with N cells, we choose N triplets of random variables from a uniform probability distribution on the interval [0, 1). These three random numbers become the x-, y-, and z-coordinates for a seed. To each seed we associate all points in the unit cube that are closer to that seed than to any other seed. All points associated with one seed constitute a single cell. This method of generating initial cell structures guarantees that more than three grains will almost never meet along a single edge and that more than four edges will almost never meet at a single point.

Different parts of this chapter and the next were written at different times, and so different data

sets were used when compiling different sets of statistics. Earlier in our research we had analyzed a smaller set of 8 systems. Each system had begun with 100,000 Voronoi cells and was allowed to evolve until it had reached a steady-state; generally, this occurred when roughly 10,700 grains remained in each structure. Later in our research we analyzed a set of 25 systems, each of which had begun as a Voronoi system with 100,000 cells and which had reached a steady state when roughly 10,700 grains remained.

4.3.3 Data Structure and Algorithm

The algorithm described below accurately evolves a triangulation of a grain boundary network. In a real grain growth microstructure, adjacent grains meet along two-dimensional surfaces called faces. Three adjacent faces meet along one-dimensional curves called edges or triple edges. Four adjacent edges meet at points called vertices or quadruple points. Many of these features can be seen in Figure 4.1.

This algorithm requires that each grain be represented as a closed collection of faces, though allows considerable variation in the triangulation used. Each face is represented by a collection of triangular facets with *face nodes* on the interior of the face and *boundary nodes* around its perimeter. Boundary nodes located at vertices are called *vertex nodes*; boundary nodes located along edges are called *edge nodes*. Line segments that connect adjacent boundary nodes are *triple edge segments*, while line segments connected to at least one face node are *regular edge segments*. An example of a typical discretized grain can be seen in Figure 4.3. Although our simulations use a single face node



Figure 4.3: A single discretized grain. Dark edges are triple edge segments and are lines along which three grains meet. One face node is located at the center of each face. Regular edge segments connect the face node with boundary nodes located along the grain's edges.

located in the center of a given face to simplify the implementation and to reduce the computational

requirements, we note that the algorithm described below can handle arbitrary triangulations of the surface as well. Edge nodes along the triple edges are added to make the discretization smoother, and occasionally removed when neighboring ones become too close. Two adjacent boundary nodes and a neighboring face node then determine a small, flat triangular facet of a face, as can be seen in the figure.

The fundamental equation that governs normal grain growth in three-dimensional isotropic polycrystalline materials is the MacPherson–Srolovitz relation [79] which describes the volume evolution of individual grains:

$$\frac{dV(\mathbf{D})}{dt} = -2\pi M\gamma \left(\mathcal{L}(\mathbf{D}) - \frac{1}{6}\mathcal{M}(\mathbf{D}) \right), \qquad (4.7)$$

where $V(\mathbf{D})$ is the volume of grain \mathbf{D} , $\mathcal{L}(\mathbf{D})$ is a one-dimensional measure of the grain called the *mean width*, and $\mathcal{M}(\mathbf{D})$ is the sum of the lengths of all edges of the grain; M and γ are constants describing the grain boundary mobility and grain boundary energy, respectively.

The general algorithm for evolving the microstructure involves two parts. First, we calculate the appropriate displacements for all nodes at every time step and move the nodes of the triangulation accordingly. Second, we implement topological changes that occur while the system evolves, such as the disappearance of edges and faces. We first describe the node displacements. The analytic solution for the volume evolution of each grain is given by Equation 4.7. We discretize the time and volume variables in this equation and solve for the change in grain volume $\Delta V(\mathbf{D})$:

$$\Delta V(\mathbf{D}) = -2\pi M \gamma \left(\mathcal{L}(\mathbf{D}) - \frac{1}{6} \mathcal{M}(\mathbf{D}) \right) \Delta t.$$
(4.8)

Next, we localize this equation for each node of the grain. That is, for each grain \mathbf{D} we calculate "local" versions of $\mathcal{L}(\mathbf{D})$ and $\mathcal{M}(\mathbf{D})$ for every node, and then change the volume of grain \mathbf{D} near each node by an appropriate amount. Done carefully, this will ensure that Equation 4.8 is satisfied for every grain at every time step with an error proportional to the square of the time step Δt .

We briefly explain the terms $\mathcal{L}(\mathbf{D})$ and $\mathcal{M}(\mathbf{D})$. The first is a linear measure known as the *mean* width and which can be seen as measuring the total mean curvature on the surface of a grain; this is a key quantity in describing systems that evolve via mean curvature flow.¹ Because our discretized grains are piecewise linear, the mean width \mathcal{L} reduces to a sum over all edge segments of a grain. If e_i is the length of the *i*th edge segment, and α_i is the exterior angle at that edge segment with

¹More information about the mean width can be found in Appendix A, as well as in [79], and especially in its Supplementary Information. To date, there is no standard reference on this subject.

respect to the same grain, then:

$$\mathcal{L}(\mathbf{D}) = \frac{1}{2\pi} \sum_{i} e_i \alpha_i, \tag{4.9}$$

where *i* is indexed over *all* edge segments (both regular and triple) of grain **D**. Figure 4.4 shows the exterior angle α of a typical edge.



Figure 4.4: Illustration of an exterior angle α with respect to the grain sitting below the two faces for a typical edge.

The second term $\mathcal{M}(\mathbf{D})$ measures the length of all triple edges around grain \mathbf{D} , i.e. all edges at which three different grains meet. If e_j is the length of a triple edge segment j of grain \mathbf{D} then:

$$\mathcal{M}(\mathbf{D}) = \sum_{j} e_j,\tag{4.10}$$

where j is indexed only over triple edge segments of the grain. In this description, $\mathcal{L}(\mathbf{D})$ and $\mathcal{M}(\mathbf{D})$ are then defined as sums over the edge segments of a grain. However, we can also define these quantities as sums over the nodes of a grain. We define "local" versions of $\mathcal{L}(\mathbf{D})$ and $\mathcal{M}(\mathbf{D})$ at every node n of a grain \mathbf{D} :

$$\mathcal{L}_n(\mathbf{D}) = \frac{1}{4\pi} \sum_i e_i \alpha_i, \qquad (4.11)$$

$$\mathcal{M}_n(\mathbf{D}) = \frac{1}{2} \sum_j e_j,\tag{4.12}$$

where *i* is indexed over *all* edge segments (both regular and triple) of grain **D** that are incident with a node *n*, and *j* is indexed only over the respective triple edge segments. We then rewrite Equations 4.9 and 4.10 as sums over nodes: $\mathcal{L}(\mathbf{D}) = \sum_k \mathcal{L}_k(\mathbf{D})$ and $\mathcal{M}(\mathbf{D}) = \sum_k \mathcal{M}_k(\mathbf{D})$ where *k* is indexed over all nodes of a grain. Because the faces of our grains are piecewise flat, the amount by which the volume of a grain will change by moving node *n* by **dn** is

$$\Delta V_n(\mathbf{D}) = \frac{1}{6} \sum_{i,j} (\mathbf{e}_i \times \mathbf{e}_j) \cdot \mathbf{dn}, \qquad (4.13)$$

where \mathbf{e}_i and \mathbf{e}_j are all ordered pairs of consecutive edge vectors pointing from a node n to its

neighboring nodes, ordered clockwise with respect to an outward-pointing normal; here no distinction is made between regular and triple edge segments.

We use these "local" versions of $\mathcal{L}(\mathbf{D})$, $\mathcal{M}(\mathbf{D})$, and $\Delta V(\mathbf{D})$ to define a local version of Equation 4.8:

$$\Delta V_n(\mathbf{D}) = -2\pi M \gamma \left(\mathcal{L}_n(\mathbf{D}) - \frac{1}{6} \mathcal{M}_n(\mathbf{D}) \right) \Delta t.$$
(4.14)

Since all terms on the RHS are determined, we have only to substitute for $\Delta V_n(\mathbf{D})$ using Equation 4.13 and solve for **dn** to find a displacement that will satisfy the MacPherson–Srolovitz relation locally at every node of the grain **D**.

The actual motion of a node is given by three unknown spatial variables. The grains adjacent to face nodes, edge nodes, and vertex nodes then provide two, three and four constraints of the type described above, respectively. Not all of these constraints are independent though since the total volume of all grains incident with a node is conserved with that node's displacement. This leaves three unknowns and one, two and three constraints for each face node, edge node, and vertex node, respectively. Consequently, there remains one degree of freedom when moving edge nodes and two degrees of freedom when moving face nodes.

Node displacements are decomposed into two parts. First, we move nodes by distances that appropriately change the volumes of the adjacent grains. Specifically, we move each face node in the unique direction that minimizes the magnitude of the node displacement; this direction is roughly perpendicular to a planar approximation the face. Likewise, we move each edge node in the unique direction that minimizes the magnitude of that node displacement; this direction is orthogonal to the line segment connecting the edge node with the two neighboring boundary nodes. The direction of the vertex node displacement is completely determined. Because each individual displacement satisfies the local, discretized version of the MacPherson–Srolovitz relation (Equation 4.14), the only error involved in these displacements are "interference" errors that result from the simultaneous movement of adjacent nodes. In the next section we show that the this error is small.

In order to maintain numerical stability, we then attempt to move each face node as close as possible to the center of mass of the face's boundary without changing the volumes of the two adjacent grains. We also attempt to move each edge node as close as possible to the center of its two adjacent boundary nodes without changing the volumes of its three adjacent grains. These displacements prevent face nodes from wandering towards face boundaries and edge nodes from becoming too close to one another, both of which can cause numerical instability. Although individual face nodes and edge nodes are moved in a way that preserves the volumes of the neighboring grains, the simultaneous motion of all of the face nodes and edge nodes in this manner creates a second set of "interference" errors similar to the ones mentioned above. Since this error is small, we believe that it is outweighed by the substantial improvement in numerical stability.

At first sight, the displacements described in this section may not resemble motion by mean curvature, since we do not attempt to move nodes in their "normal directions" and with magnitudes proportional to their local curvature. Indeed, some approaches attempt to define a normal direction an mean curvature at every node, and move each node accordingly point. The difficulty with this approach is that while that method might adequately describe the displacements of a finite set of points commensurate with the nodes, it does not adequately describe the motion of other points of the surface. Consequently, the volume does not change in accordance with Equations 4.3 and 4.7. In the current method, rather than considering the velocities of a finite number of points and moving the surface as those points require, we consider the way in which the volume of a grain changes with the movement of entire elements of the surface. This allows us to accurately satisfy the exact MacPherson–Srolovitz relation without resorting to arbitrarily refined surface meshes.

4.3.4 Error analysis

We now consider the error associated with evolving a given triangulation one time step by the above equations. This error is measured by comparing the volume change of a grain in one time step of the simulation method with the volume change predicted by the the discretized MacPherson–Srolovitz relation (Equation 4.8). We demonstrate that our method produces errors from the discretization of time that are proportional to $(\Delta t)^2$ to leading order, independent of the triangulation used.

Consider a simple tetrahedron like that shown in Figure 4.5. Imagine one of its corners, \mathbf{c} , placed in the center of a discretized grain, and the other three corners, \mathbf{v}_1 , \mathbf{v}_2 , and \mathbf{v}_3 , placed at the three corners of a triangular facet of that grain. Every grain can be decomposed into a finite number of these tetrahedra, with one tetrahedron associated with each triangular facet of the grain.

Before moving any of the vertices, the volume of the tetrahedron can be determined from the edge vectors \mathbf{e}_i : $V = \frac{1}{6}\mathbf{e}_1 \cdot (\mathbf{e}_2 \times \mathbf{e}_3)$. If we only move \mathbf{v}_1 by $\mathbf{d}\mathbf{v}_1\Delta t$, then the volume would change by $\frac{1}{6}(\mathbf{e}_2 \times \mathbf{e}_3) \cdot \mathbf{d}\mathbf{v}_1\Delta t$. Likewise, if we only move \mathbf{v}_2 or \mathbf{v}_3 , then the volume would change by $\frac{1}{6}(\mathbf{e}_2 \times \mathbf{e}_3) \cdot \mathbf{d}\mathbf{v}_1\Delta t$. Likewise, if we only move \mathbf{v}_2 or \mathbf{v}_3 , then the volume would change by $\frac{1}{6}(\mathbf{e}_3 \times \mathbf{e}_1) \cdot \mathbf{d}\mathbf{v}_2\Delta t$ or $\frac{1}{6}(\mathbf{e}_1 \times \mathbf{e}_2) \cdot \mathbf{d}\mathbf{v}_3\Delta t$, respectively. An "exact" solution involved in moving all three vertices should then change the volume by $\Delta V = \frac{1}{6}[(\mathbf{e}_2 \times \mathbf{e}_3) \cdot \mathbf{d}\mathbf{v}_1 + (\mathbf{e}_3 \times \mathbf{e}_1) \cdot \mathbf{d}\mathbf{v}_2 + (\mathbf{e}_1 \times \mathbf{e}_2) \cdot \mathbf{d}\mathbf{v}_3]\Delta t$. However, if we move all three vertices at the same time, we find that the volume *actually* changes by $\Delta V_a = \frac{1}{6}(\mathbf{e}_1 + \mathbf{d}\mathbf{v}_1\Delta t) \cdot ((\mathbf{e}_2 + \mathbf{d}\mathbf{v}_2\Delta t) \times (\mathbf{e}_3 + \mathbf{d}\mathbf{v}_3\Delta t)) - \frac{1}{6}\mathbf{e}_1 \cdot (\mathbf{e}_2 \times \mathbf{e}_3)$. The difference between



Figure 4.5: A volume element associated with a triangular facet on the grain boundary.

these two numbers is the "interference" error involved in these node displacements:

$$\Delta V_a - \Delta V = \frac{1}{6} \mathbf{e}_1 \cdot (\mathbf{d} \mathbf{v}_2 \times \mathbf{d} \mathbf{v}_3) \Delta t^2 + \frac{1}{6} \mathbf{e}_2 \cdot (\mathbf{d} \mathbf{v}_3 \times \mathbf{d} \mathbf{v}_1) \Delta t^2 + \frac{1}{6} \mathbf{e}_3 \cdot (\mathbf{d} \mathbf{v}_1 \times \mathbf{d} \mathbf{v}_2) \Delta t^2 + \frac{1}{6} \mathbf{d} \mathbf{v}_1 \cdot (\mathbf{d} \mathbf{v}_2 \times \mathbf{d} \mathbf{v}_3) \Delta t^3.$$
(4.15)

Since the \mathbf{e}_i and \mathbf{dv}_i are independent of Δt , the error for these overlapping displacements is, to leading order, proportional to Δt^2 . Since each grain can be decomposed into a finite number of these tetrahedra, the total error for each grain is a sum of these interference errors and is thus also of order $O(\Delta t)^2$.

One way to visualize the magnitude of the error involved in this method is by plotting $\Delta V/\Delta t$ for each grain against the rate of change of the volume predicted by the MacPherson–Srolovitz relation. Figure 4.6 shows these values for the current method and for Surface Evolver [48], a prominent front-tracking program that is used for grain-growth simulations [115, 111]. It is clear from the data that Surface Evolver produces much larger errors from the discretization of time than does our method.

Aside from the error associated with the discretization of time, we also consider the error associated with the discretization of space. For a triangulation where the linear size s of the individual triangles is small and the triangles are equiaxed, the error in the mean width introduced by discretizing a continuous grain **D** scales as $(s/\mathcal{L}(\mathbf{D}))^2$ [79]. In theory then, the algorithm presented above can also achieve an arbitrarily small space-discretization error by refining the surface mesh into small equiaxed triangles.



Figure 4.6: We graph the rate of change of volume against the rate of change of the volume predicted by the MacPherson–Srolovitz relation for (a) Surface Evolver and (b) the current method. Data are collected from one time-step in a system with 1000 grains.

The triangulation used in our simulations, however, do not satisfy the equiaxed condition and calculating the space-discretization error is therefore more complicated. One reason for the difficulty in calculating this error is the difficulty in deriving an analytic description of even relatively simple grains, such as a maximally symmetric shrinking tetrahedral grain. Without a complete understanding of these simple cases, it is difficult to compare them to our numerical models.

Nevertheless, we can analyze the error involved in our method discretization by considering a spherical cap, which can reasonably approximate a continuous representative face in a real system [116]. In our method, such a face is represented using exactly one node in the middle of a face and other nodes along its boundary, as described above. Figure 4.7 shows an approximation of a representative continuous face F_c by a spherical cap and by a discretized face as it might be realized in our simulation method. The spherical cap surface is completely determined by the radius r of its circular base and the angle β of its intersection with the base. Following our discretization, the analogue of this face appears on the right of Figure 4.7. The base of the discrete face F_d is a regular polygon that is defined by the number of edges n and the inradius r, while the height of the single face node is adjusted to make the triangular facets intersect the base at the angle β . As n increases, the polygonal base will asymptotically approach the circular base of the continuous face.

Since the rate of change of volume of a grain is directly proportional to the integral of the mean curvature over the faces, we compare the integrals of the mean curvature \mathcal{M} over the continuous and discrete faces to estimate the relative error in the rate of change of volume of the corresponding



Figure 4.7: A single face of a grain, and the analogue corresponding to the discretization used by the current method. The quantity r is the radius of the base of the continuous grain and the inradius of the n-sided polygonal base of the discrete grain, as shown.

grains. That is, we estimate the relative error η as

$$\eta = \frac{\left|\int_{F_d} \mathcal{M} dA - \int_{F_c} \mathcal{M} dA\right|}{\int_{F_c} \mathcal{M} dA}.$$
(4.16)

Consider the integral of the mean curvature over the continuous face. We construct a spherical coordinate system with the origin on the center of the sphere coincident with the surface, and with the polar axis extending through the center of the face. The constraint on the angle of intersection with the base requires that the face be described by the polar coordinate $0 \le \theta \le \beta$ and radial coordinate $r \csc(\beta)$. Integrating gives $\pi r^2 \sec^2(\beta/2)$ as the area of the surface, and since the mean curvature is the sum of the principal curvatures, the mean curvature is uniformly $2\sin(\beta)/r$. This gives

$$\int_{F_c} \mathcal{M} \mathrm{d}A = 4\pi r \tan(\beta/2) \tag{4.17}$$

as the integral of the mean curvature over the continuous face.

For the calculation of the integral of the mean curvature over the discrete face, we use the relation

$$\int_{\partial D} \mathcal{M} dA = 2\pi \left(\mathcal{L}(D) - \frac{1}{6} \mathcal{M}(D) \right), \qquad (4.18)$$

with the simplification that the contributions to \mathcal{L} and \mathcal{M} from the edges around the polyhedral base cancel. As evidence of this, consider that the height of the discrete face is adjusted to make the angle of intersection of the triangular facets with the polygonal base equal the corresponding angle of the continuous face. Since the faces of a continuous grain meet along a triple line with an exterior angle of $\pi/3$, triangular facets of the corresponding discretized grain will meet along triple edge segments with the same exterior angle. Equations 3 and 4 of the article reveal that the contributions to \mathcal{L} and \mathcal{M} of the triple edge segments around the polygonal base of the discrete grain cancel in the above relation. This leaves only the contribution of the regular edge segments to the integrated mean curvature. Since the length e_0 and exterior angle α_0 of all n of these edges are identical by construction, Eq. (4.18) reduces to

$$\int_{\partial F_d} \mathcal{M} \mathrm{d}A = n \cdot e_0 \cdot \alpha_0 \tag{4.19}$$

for the single discrete face described above. This leaves the calculation of the quantities e_0 and α_0 as functions of β and n.

Geometric considerations give the circumradius of the polygonal base as $r \sec(\pi/n)$ and the height of the single face node as $r \tan(\beta)$. The length of the regular edge segments is therefore

$$e_0 = r\sqrt{\tan^2(\beta) + \sec^2(\pi/n)}.$$
 (4.20)

Meanwhile, we construct a Cartesian coordinate system with the origin at the center of the discrete grain and write down a set of explicit coordinates for the edge nodes and face nodes of the discrete grain, allowing the calculation of α_0 . Since the calculation is lengthy and not particularly revealing, we provide only the result that

$$\alpha_0 = \arccos\left[\sin^2(\beta)\cos(2\pi/n) + \cos^2(\beta)\right]. \tag{4.21}$$

Substituting these into Eq. (4.18) gives the expression for the integral of the mean curvature over the discrete face as

$$\int_{\partial F_d} \mathcal{M} dA = nr \sqrt{\tan^2(\beta) + \sec^2(\pi/n)} \arccos\left[\sin^2(\beta)\cos(2\pi/n) + \cos^2(\beta)\right].$$
(4.22)

Since n is restricted to a subset of the positive integers, the expression for the integral of the mean curvature is not amenable to asymptotic analysis for small values of n. The limiting function as nincreases may be found though, and is

$$\lim_{n \to \infty} \int_{\partial F_d} \mathcal{M} dA = 2\pi r \tan(\beta).$$
(4.23)



Figure 4.8: The mean curvature, integrated over a representative face. Solutions for the faces on the left and right of Figure 4.7 are given as the thick straight line and points, respectively. The dashed line is the limiting integrated mean curvature of the discretized face for large n.

Hence, by combining Eqs. (4.16), (4.17) (4.23), we see that the relative error η in the mean curvature of a face that is due purely to the use of a single face node is roughly

$$\lim_{n \to \infty} \eta(\beta, n) = \frac{|2\pi r \tan(\beta) - 4\pi r \tan(\beta/2)|}{4\pi r \tan(\beta/2)}$$
(4.24)

$$=\frac{\beta^2}{4} + O(\beta^4).$$
 (4.25)

That is, the relative error scales as the square of the maximum angular separation of the normals at two distinct points on the face.

Of course, this gives neither the behavior of the error as a function of n, nor the magnitude of the error for an average face in our simulation. From our simulations, the effective value of β , averaged over all the faces in a steady state structure, is roughly 0.123 radians. For this particular value of β , Figure 4.8 shows the value of the integrated mean curvature for a continuous face as the solid blue line, the limiting value of the integrated mean curvature for a discrete face as the dashed black line, and the behavior of the integrated mean curvature of a discrete face as a function of n as the red points. The most striking feature of this figure is that the error due to the discrete polygonal base falls quickly with increasing n. Finally, since the average number of edges per face is about five, and the average number of segments per edge in our simulations is about five, the average number of segments around the perimeter of a face is roughly 25. Explicitly evaluating the relative error in the integral of the mean curvature given by Eq. (4.16) for this value of n and the above value of β gives 0.009 for the average face, or slightly less than one percent. The relative error in the rate of

change of the volume of a grain is expected to be of roughly the same magnitude.

The question of whether this is a significant error is another matter entirely. For instance, there does not appear to be any straightforward means to predict the effect of this error on the evolution of features of the microstructure. Furthermore, we cannot compare the error introduced by our triangulation to that of other simulations, since the corresponding error estimates do not seem to be reported elsewhere in the literature.

We stress again that this error is a consequence of the particular triangulation, and not of the algorithm, which is entirely amenable to the use of a finer surface mesh with equiaxed triangles. Furthermore, this error in our implementation is offset by the fact that our choice has allowed us to perform a front-tracking simulation containing more than a factor of ten more grains than other comparable simulations [111, 113, 114]. Finally, we point out that the time- and space-discretization errors are not generally reported for other simulation methods, and therefore cannot be compared with ours.

4.3.5 Topological changes

Aside from calculating displacements for all nodes at every step we must also describe appropriate topological changes that occur during coarsening, namely, the disappearance of grains and the disappearance and creation of edges and faces. In three-dimensional systems, we have found that in our simulations all topological events can be decomposed into combinations of five basic topological changes: the disappearances of (a) edges, (b) triangular faces, (c) two-sided faces, (d) tetrahedral grains and (e) three-faced grains. These events occur when the length of an edge, the area of a face, or the volume of a grain approaches zero. Since none of these quantities can be negative, we are forced to make appropriate changes in the system by removing these entities and creating new ones in their place. Figure 4.9 illustrates these five topological changes.

Figure 4.10 illustrates how a more complicated topological event can be decomposed into a series of events from this group. We have the disappearance of a grain that has two faces with two sides and two faces with four sides. Such a grain is collapsed by first removing a two-sided face. The remaining grain, which has three two-sided faces, is then removed. All topological events that we have observed in the evolution of our systems may be decomposed in this manner, meaning that the coding is simplified by requiring only a small number of topological events to be implemented.

The two most frequent changes that occur in our simulations of evolving systems are the disappearance and creation of edges and triangular faces. When an edge disappears a new triangular face


Figure 4.9: Five basic topological changes: (a) An edge disappears and becomes a triangular face. (b) A triangular face disappears and becomes an edge. (c) A two-sided face disappears; two neighboring faces merge. (d) A grain with four triangular faces disappears into a vertex. (e) A grain with three two-sided faces disappears into a triple edge.



Figure 4.10: This quickly shrinking grain is removed in two steps: first we remove a two-sided face, then we remove the remaining three-faced grain.

is created, and when a triangular face disappears a new edge is created. In this sense these changes can be considered duals and are "reversible"; the other three topological changes are not reversible.

Although we did not initially expect them, the occurrence of two-sided faces and three-faced grains during normal grain growth is widely reported in the literature, including in numerous experimental studies [117, 54, 118, 119] and several recent simulations [111, 113, 114]. Curiously, these topological events do not seem to appear in some otherwise detailed discussions of three-dimensional grain growth, including [120].

It is important to note that we have no a priori way of knowing that these five topological changes are the only ones that can occur throughout the coarsening of a three-dimensional cell structure. For example, it is possible that the mean-curvature evolution of a cell structure will witness a cube shrinking until its volume vanishes, all while remaining a cube. However, our success in implementing a code that appears stable while implementing only these five lead us to believe that no more than these are necessary. In a similar vein, these five topological changes can be further decomposed into combinations of only three fundamental events—the removal of an edge, a two-sided face, and a grain with three faces. For example, a tetrahedron could be removed by first removing one edge, which would leave it as a three-sided grain. However, it seems that each of these five topological changes occurs during the evolution, and decomposing them into only three would seem unnatural. It would be nice to prove what topological events can occur during mean curvature flow on cell structures.

4.3.6 Technical details of the simulation

All simulations used the $[0,1]^3$ unit cube with periodic boundaries.

Step size and maximum triple edge segment length

If N is the number of cells in the system, the size of one time step is:

$$\Delta t = \frac{1}{2500N} \tag{4.26}$$

This choice of step-size seemed to provide the most stable evolution. The maximum triple edge segment length ℓ_{max} scales with the linear dimension of the system:

$$\ell_{max} = 0.15\sqrt[3]{1/N} \tag{4.27}$$

Every five steps we check all triple edge segments in the system. If any are longer than ℓ_{max} we add a boundary node at the midpoint of that triple edge segment, thus creating two edge segments each of which is half the length of the original one. This allows for a smoother discretization of the cells.

Removing a boundary node

Edge nodes along the triple edges are occasionally removed when neighboring ones become too close, as numerical instabilities arise if the mesh is overly refined for a given time step size. Every five steps we check all boundary nodes in the system to see whether they are moving "too fast". In one time step, a vertex x with a velocity dx/dt will move a total distance of $||dx/dt||\Delta t$. Each boundary node has exactly two neighbors n_1 and n_2 that lie along the same triple edge; these nodes might be other boundary nodes or else quadruple points. We say that a boundary node is moving "too fast" with respect to its neighboring nodes if:

$$\frac{||dx/dt||\Delta t}{||n_1 - n_2||} \ge 0.02. \tag{4.28}$$

If a boundary node x is moving too fast, then we remove the boundary node x and connect n_1 and n_2 to each other. Figure 4.11 illustrates the local structure near a node before and after it is removed. Removing the node requires making adjustments in the data structures associated with



Figure 4.11: A node x which is moving "too fast" is removed; neighboring nodes n_1 and n_2 are now connected.

 n_1 and n_2 , other neighboring nodes, and three neighboring grains. The volumes of all neighboring grains are also recalculated, as are the velocities of all nodes which were adjacent to the removed node; we do *not* check again whether or not these nodes need be removed.

Removing a triple edge

On every step we check all triple edges in the system to make sure they are not shrinking "too fast". We only consider edges which are represented by exactly one edge segment and have no intermediate boundary nodes between their two bounding quadruple points. We use the velocities of the two quadruple points that lay on either end of the edge to determine dl/dt, the rate at which an edge is growing or shrinking. We say that a triple edge with length l is shrinking too fast if:

$$\frac{dl}{dt}\frac{\Delta t}{l} \le -0.2. \tag{4.29}$$

If an edge is shrinking too quickly, then we replace it with a triangular face. This topological change was illustrated in Figure 4.9(a) and is redrawn here. The two grains which sat at opposite ends of this triple edge now meet at the new triangular face. The two quadruple points at each end of the edge are removed, and three new quadruple points, corners of the new triangular face, are introduced. Each of the new quadruple points is placed at a location determined by the geometric center of the previous edge, as well as by the locations of the two edges which are adjacent to it; further details can be found in Appendix C. Data structures associated with all neighboring nodes



Figure 4.12: A shrinking triple edge is removed and replaced by a triangular face.

and grains are adjusted appropriately. Volumes of all adjacent grains are recalculated, as are the velocities of all neighboring nodes whose velocities will be affected.

Although in most situations, removing a triple edge is a "safe" procedure, if the edge is part of a face which has exactly two sides, removing the edge will cause serious problems. In this situation we do not remove the edge here and instead leave it to be removed by other topological procedures, including ones that will remove two-sided faces or grains with three faces, in the near future.

Removing a small face

On every step we check all faces in the system to make sure they are not shrinking "too fast" or else are "very small". Using the velocities of all nodes that make up a face, we determine the amount that the area of a face will change in one time step. A face with area A is shrinking too fast if its change in area ΔA in one time step is such that:

$$\frac{\Delta A}{A} \le -0.2. \tag{4.30}$$

A face is also considered to be too small if its area is less than 1×10^{-6} times the average face area. In a situation where a face is part of a grain that has only three faces then we do not remove this face directly. This would cause the creation of a grain with two faces, something our simulation program cannot handle and should not occur. We therefore leave these faces to other other topological procedures including the removal of grains with three faces.

Once we have decided to erase a particular face, we begin by removing all boundary nodes along its perimeter. We then count how many sides the face has. If it has more than three sides, we remove one or more of its sides until the face has only three. To do this, we remove the shortest edge of the face that can be safely removed using the procedure outlined above. We repeat this procedure until there remain only three edges in this face.

We have separate procedures for removing two-sided and three-sided faces. The simpler procedure

to execute is the removal of a face with three sides. This topological change was illustrated in Figure 5.7(b) and is redrawn here. Two new quadruple points and an edge joining them are created. This



Figure 4.13: A shrinking triangular face is removed and replaced by a triple edge.

edge is drawn perpendicular to the face being erased; the two grains that had shared the triangular face are now connected only by this edge. The three others grains which shared a triple edge with this triangular face now meet along the new triple edge. Data structures associated with neighboring nodes and with the five neighboring grains must be adjusted appropriately.

The removal of a face with two sides is more complicated both to visualize and to code. This procedure was illustrated in Figure 4.9(c) and is redrawn here. A face with two sides (shaded dark



Figure 4.14: The disappearance of a face with two sides, shaded dark gray.

gray) is adjacent to four grains: two lay on opposite sides of the face, and two more share a triple edge with this face. Referring to Figure 4.14 we will call these the top and bottom grains, and the front and back grains, respectively. After removing this face, the top and bottom grains no longer meet. Also, two faces that had initially divided the front and back grains (shaded light gray) merge into one face as pictured. Data structures associated with neighboring nodes and with the four neighboring grains must be adjusted appropriately.

Removing a small grain

The last set of topological changes are those involving the removal of entire grains. We again use the velocities of the nodes to determine the amount that the volume of a grain will change in one time step. A grain with volume V is shrinking too fast if its change in volume ΔV in one time step is such that:

$$\frac{\Delta V}{V} \le -0.4. \tag{4.31}$$

We also remove a grain if its volume is less than 1×10^{-6} times the average grain volume. At this point we remove all boundary nodes from the grain in question. Next we count how many faces the grain has. If it has three or four faces, then we remove the grain directly. If the grain has more than four faces, then we remove faces until there are only four remaining. We begin by removing the face with the fewest number of sides and then amongst those, the one with the smallest area. We repeat this procedure until the grain has only three or four faces remaining.

We have separate procedures for removing grains with three faces and those with four triangular faces. The simpler procedure to execute is the removal of grains with four triangular faces. This procedure was illustrated in Figure 4.9(d) and is redrawn here. In this case, four neighboring grains



Figure 4.15: The disappearance of a tetrahedrally-shaped grain.

each lose one triangular face. We place a new quadruple point in the geometric center of the four quadruple points that were the "corners" of the tetrahedrally-shaped grain. We then connect the incoming edges to the new quadruple point as necessary. Data structures associated with neighboring nodes and with the four neighboring grains must be adjusted appropriately. Also, the data structure associated with the tetrahedral grain is removed from memory.

The last topological change that we implement when evolving a system is the removal of a grain with three faces. This procedure was illustrated in Figure 4.9(e) and is redrawn here. In this case, three neighboring grains each lose one two-sided face. We draw a new edge between the two quadruple points associated with this grain. Data structures associated with neighboring nodes and with the three neighboring grains must be adjusted appropriately. Also, the data structure associated with the three-faced grain is removed from memory.

Although it is not obvious that these five topological processes are enough to describe the evolution of all three-dimensional cell structures, we have found them sufficient in all of our simulations.



Figure 4.16: The disappearance of a tetrahedrally-shaped grain.

We believe that all topological changes in the system can be implemented using these five basic topological changes.

4.4 Results

Microstructure Evolution

We report here results from our implementation of the algorithm described above. The initial condition for all simulations is a Voronoi tessellation of 100,000 points distributed with a uniform probability distribution in a unit cube with periodic boundary conditions. Setting the length of the unit cell to L, we measure time in dimensionless units of $L^3/M\gamma$, where M and γ are the grain boundary mobility and energy, respectively.

In the initial Voronoi tessellation, all grains have flat faces and straight edges. The evolution of a typical microstructure is illustrated by a series of still images in Figure 4.17. After a short time, the faces and edges begin to curve: angles between adjacent faces at triple edges quickly approach $2\pi/3$ and the angles between triple edges at vertex nodes approach the tetrahedral angle $\cos^{-1}(-1/3) \approx 109.5^{\circ}$. The microstructure gradually coarsens, resulting in fewer grains and a larger average grain size.

Once a structure has reached a steady state, the average linear dimension of a grain should increase linearly with $t^{1/2}$ [22]; the volume should thus increase linearly with $t^{3/2}$. This behavior can be derived analytically using dimensional analysis [121]. Figure 4.18 shows that the average grain volume $\langle V \rangle$ increases linearly with $t^{3/2}$ after the initial transient period.

Steady State

During grain growth the microstructure evolves towards a statistically self-similar state, i.e. a state in which properties are statistically time-invariant up to a change in scale. One way of observing this



Figure 4.17: Temporal evolution of a microstructure using the current method. This microstructure was initialized as a Voronoi tessellation of 10,000 points randomly distributed in the unit cube with periodic boundary conditions.



Figure 4.18: The average grain volume $\langle V \rangle$ as a function of $t^{3/2}$. Each system began with 100,000 grains; when $t^{3/2} = 0.001$, there remain slightly fewer than 500 grains. Data points are averaged from eight simulations; error bars indicate standard deviation from the mean.

process is by observing scale-independent properties as they relax towards asymptotic, steady-state values. Figures 4.19 and 4.20 show how the average number of faces per grain and the average number of edges per face change as the system evolves. It is easy to see the initial values, a transient period in which both of these numbers are decreasing, and a point after which these numbers remain fixed.



Figure 4.19: The average number of faces per grain decreases as the system evolves away from the initial Voronoi structure. As the system approaches the steady state, this number approaches a fixed value close to 13.769 ± 0.016 . Data are averaged from eight simulations; error bars indicate standard deviation from the mean.

A frequent prediction in the literature [122, 123, 116] is that the average number of faces per grain should be about 13.397 in the steady-state, while Coxeter [124] provided some justification that this quantity should be around 13.564. We observe that the average number of faces per grain and edges per face remain fixed in the steady-state at 13.769 ± 0.016 and 5.128 ± 0.001 respectively, close to but measurably different from both of the above predictions. These values were obtained from the data at time t = 0.00125, when approximately 10,700 grains remained in each simulation.

Figures 4.21 and 4.23 show the distribution of faces per grain and the distribution of edges per face in the steady state structure. To highlight the smoothness of our data plots, it is worth comparing them to results collected from other methods. In Figure 4.22 we reproduce a graph from a 2002 paper [37] which collects statistics from five different methods, including Potts models [33, 82], front tracking methods [109, 111], and its own phase field method. The distribution of faces and edges highlight some differences between the steady-state and Voronoi microstructures. This difference is best highlighted by contrasting the distribution of grain volumes in the two structures. Figure



Figure 4.20: The average number of edges per face decreases as the system evolves away from the initial Voronoi structure. As the system approaches the steady state, this number approaches a fixed value close to 5.128 ± 0.001 . Data are averaged from eight simulations; error bars indicate standard deviation from the mean.



Figure 4.21: Distribution of grains for initial and steady-state microstructures averaged from eight simulations; error bars indicate standard deviation from the mean.

4.24 shows the distribution of grain volumes in each system. In the Voronoi case, there are very few extremely large and extremely small grains. In the evolved microstructure, on the other hand, there are a very large number of extremely small grains, and there are a consider number of very large grains too. The distribution appears to be either exponential or very close to exponential. Figure 4.25 plots the same data as Figure 4.24 and highlights the exponential nature of the distribution.



Figure 4.22: A graph from a 2002 paper [37] showing the distribution of grains with different numbers of faces from structures evolved using two Monte Carlo Potts models [33, 82], two front tracking methods [111, 109], and its own phase field method.



Figure 4.23: Distribution of faces with different numbers of edges from Voronoi configurations and from steady-state microstructures averaged from eight simulations; error bars indicate standard deviation from the mean.

Indeed, the density distribution of grains in the evolved state appear to be very well described by e^{-x} , where is x is the volume of a grain. These data are consistent with earlier simulations of smaller systems using other three-dimensional grain growth simulation methods [109, 111].

We do not currently understand why this appears to be the limiting distribution of cell structures

that evolve through mean curvature flow. We only point out that the exponential distribution $\exp(-x)$ has the largest entropy of all distributions with support $[0, \infty)$ and mean 1. In future work we might consider how curvature flow impacts the entropy of a system, which might help explain this interesting result.

These results represent the largest three-dimensional grain growth simulations performed to date using a front-tracking method. Large data sets are important for several reasons, namely: (1) they ensure that simulations evolve long enough to reach the steady-state microstructure, (2) they ensure that sufficient grains remain in the steady-state to capture reasonable statistics of the microstructure, and (3) they ensure that the features of the statistical distributions are sufficiently accurate for comparison with experiments, other simulations and theory.



Figure 4.24: Distribution of normalized grain volumes $V/\langle V \rangle$ for the initial, Voronoi, and steadystate microstructures averaged over eight samples; error bars indicate standard deviation from the mean.



Figure 4.25: Distribution of normalized grain volumes $V/\langle V \rangle$ for the steady-state microstructure averaged over eight samples; error bars indicate standard deviation from the mean. The function $\exp(-x)$ is drawn behind the data for comparison.

Further Characterization

It is clear that the distributions presented so far are very coarse descriptions of a microstructure. One strength of our method is the way in which it allows us to easily collect data about various geometrical and combinatorial features of grain growth microstructures. For example, one way in which we might further analyze our microstructures is by looking at the distribution of vertex nodes, as characterized by the sum of their four adjacent grains' numbers of faces, as plotted in Figure 4.26. A fascinating difference we immediately see between the Voronoi and the steady-state systems is



Figure 4.26: Distribution of vertices, as characterized by the sum of their four adjacent grains' numbers of faces. Data are averaged from eight simulations; error bars indicate standard deviation from the mean.

that while the Voronoi systems have an apparent preference for vertices with even sums for numbers in the middle of the distribution, the steady-state microstructure shows no such bias.

Another example of a scale-invariant property we consider is the distribution of topologically distinct polyhedra. For example, instead of grouping together all grains with a given number of faces (as was done for Figures 4.21 and 4.22), we might consider specifying more exactly the combinatorial structure of the grains. In two-dimensional systems, the combinatorial structure of an individual cell can be completely described by its number of sides. That is, any two cells with the same number of sides can be continuously deformed into one another without a need to change its topology. Three-dimensional grains, whose boundaries are homeomorphic to the 2-sphere, are significantly more complicated, and reporting only their number of faces fails to completely characterize their entire combinatorial structure. To illustrate this point, consider the two distinct polyhedra with

six faces shown in Figure 4.27. Although both polyhedra have six faces, the cube on the right is



Figure 4.27: Two distinct polyhedra with six faces. These happen to be the *only* possible polyhedra with six faces, each of which has three or more sides.

combinatorially distinct from the shape on the left: the shape on the left has 2 three-sided faces, 2 four-sided faces, and 2 five-sided faces, while the cube on the right has 6 four-sided faces. Because the distribution of these shapes in a steady-state microstructure is a scale-invariant property of the microstructure, a complete understanding of grain growth structures must include an understanding of how these shapes are distributed. We would therefore like to devise an efficient way of describing the combinatorial structure of a grain. In the subsections that follow, we describe two ways, one partial and one complete, to characterize the combinatorial structure of grains.

Describing Polyhedra

A first attempt to characterize grains more completely might use what are sometimes called p-vectors, a term taken from [125]. To each polyhedral cell C with n faces we associate an $n \times 1$ vector $p(C) = (p_1, p_2...p_n)$, such that p_i is the number of faces of the cell that have i sides. We point out that if f(C) is the number of faces of a cell C, then $f(C) = \sum_i p_i$. Although the p-vector is a more refined measurement of the combinatorial structure of a grain than a count of sides, we will see later that it too is incomplete.

Table 4.1 shows a list of the 25 most frequently occurring *p*-vectors in both the Voronoi structures as well as in the evolved steady state structures. Two discrepancies between the Voronoi and steady state microstructures immediately stand out. First, the Voronoi structure has a significantly larger number of grains with 3- and 7-sided faces than does the steady state. All except for two of the 25 most common *p*-vectors have a positive value for p_3 and p_7 . In contrast, in the steady state, very few of the 25 most common *p*-vectors have 3-sided or 7-sided faces.

A second, perhaps more striking difference between the structures is the way in which shapes are

Vo	ronoi		Stead	ly state	
<i>p</i> -vector	frequency	σ	p-vector	frequency	σ
(0, 1, 3, 4, 3, 1, 0)	0.00385	0.00019	(0, 0, 4, 4, 0, 0, 0)	0.0283	0.0021
(0, 1, 3, 4, 2, 1, 0)	0.00337	0.00018	(0, 0, 4, 4, 2, 0, 0)	0.0237	0.0012
(0, 1, 4, 3, 3, 2, 0)	0.00301	0.00017	(0, 0, 4, 4, 3, 0, 0)	0.0186	0.0011
(0, 1, 4, 2, 3, 1, 0)	0.00290	0.00013	(0, 0, 3, 6, 0, 0, 0)	0.0186	0.0010
(0, 1, 3, 4, 4, 1, 0)	0.00289	0.00015	(0, 0, 4, 4, 1, 0, 0)	0.0161	0.0013
(0, 2, 3, 3, 3, 1, 1)	0.00282	0.00012	(0, 0, 5, 2, 0, 0, 0)	0.0149	0.0014
(0, 1, 3, 3, 2, 0, 0)	0.00273	0.00012	(0, 0, 3, 6, 3, 0, 0)	0.0146	0.0014
(0, 1, 3, 5, 2, 2, 0)	0.00270	0.00017	(0, 0, 3, 6, 1, 0, 0)	0.0141	0.0010
(0, 2, 2, 3, 3, 1, 0)	0.00264	0.00021	(0, 1, 3, 3, 0, 0, 0)	0.0138	0.0011
(0, 0, 4, 4, 2, 0, 0)	0.00262	0.00020	(0, 0, 3, 6, 2, 0, 0)	0.0138	0.0011
(0, 1, 3, 5, 3, 2, 0)	0.00258	0.00013	(0, 1, 3, 3, 2, 0, 0)	0.0138	0.0011
(0, 1, 4, 2, 2, 1, 0)	0.00254	0.00013	(0, 1, 3, 3, 1, 0, 0)	0.0136	0.0011
(0, 2, 3, 4, 2, 2, 1)	0.00252	0.00014	(0, 0, 3, 6, 4, 0, 0)	0.0116	0.0010
(0, 2, 3, 3, 2, 1, 1)	0.00252	0.00014	(0, 0, 2, 8, 1, 0, 0)	0.0108	0.0010
(0, 1, 4, 3, 2, 2, 0)	0.00252	0.00019	(0, 0, 4, 4, 4, 0, 0)	0.0097	0.0009
(0, 2, 2, 4, 2, 2, 0)	0.00248	0.00015	(0, 2, 2, 2, 0, 0, 0)	0.0093	0.0010
(0, 1, 4, 4, 3, 1, 1)	0.00242	0.00016	(0, 0, 2, 8, 2, 0, 0)	0.0092	0.0006
(0, 2, 3, 2, 2, 2, 0)	0.00241	0.00014	(0, 0, 6, 0, 0, 0, 0)	0.0086	0.0011
(0, 0, 4, 4, 3, 0, 0)	0.00238	0.00014	(0, 0, 2, 8, 0, 0, 0)	0.0079	0.0007
(0, 1, 4, 4, 2, 1, 1)	0.00235	0.00015	(0, 0, 4, 5, 4, 1, 0)	0.0078	0.0007
(0, 2, 3, 4, 3, 2, 1)	0.00233	0.00020	(0, 0, 5, 2, 2, 0, 0)	0.0076	0.0007
(0, 1, 4, 2, 4, 1, 0)	0.00226	0.00015	(0, 0, 4, 5, 3, 1, 0)	0.0076	0.0010
(0, 1, 4, 3, 4, 2, 0)	0.00220	0.00015	(0, 0, 4, 5, 2, 1, 0)	0.0069	0.0007
(0, 2, 2, 4, 3, 2, 0)	0.00215	0.00015	(0, 2, 3, 0, 0, 0, 0)	0.0068	0.0008
(0, 2, 3, 2, 3, 2, 0)	0.00210	0.00013	(0, 1, 2, 5, 2, 0, 0)	0.0063	0.0005

Table 4.1: A list of the 25 most frequently occurring *p*-vectors in the Voronoi and steady state systems. The first column provides an abbreviated *p*-vector: $(p_2, p_3, p_4, p_5, p_6, p_7, p_8)$; none of the most frequently occurring *p*-vectors had faces with fewer than 2 sides or more than 8 sides. Data are averaged from 25 systems each beginning with 100,000 grains and each of which had relaxed when roughly 10,700 grains had remained. The frequency has been averaged over the 25 runs; σ is the standard deviation from the mean of the frequency amongst the runs.

distributed. In the steady state, the 25 most popular shapes constitute 30.4% of all grains in the system. In the Voronoi system, on the other hand, this number is only 6.5%. In other words, whereas both structures seem to prefer some shapes over others, the "preference" is much stronger in the steady state microstructure than in the Voronoi one. Although we may have expected grain growth, which can access a greater range of shapes (such as those with two-sided faces), to distribute the shapes more widely than the Voronoi case, it appears that mean curvature flow imposes additional order on the structure, thereby strongly influencing the distribution of grains.

Although the *p*-vector certainly provides us with a more detailed classification than that provided by a count of faces alone, it too is incomplete in describing the entire combinatorial structure of an individual grain. Polyhedra with identical *p*-vectors can have distinct combinatorial structure. The simplest example of a *p*-vector that can be realized in two distinct ways can be seen with two grains with 8 faces and a p-vector (0, 2, 2, 2, 2, 0, 0), where the first index corresponds to the number of two-sided faces, and so forth. Figure 4.28 shows plane graphs of two polyhedra with 8 faces. Both polyhedra have two triangular faces, two rectangular faces, two pentagonal faces, and



Figure 4.28: Two graphs of polyhedra that have two triangular faces, two rectangular faces, two pentagonal faces, and two hexagonal faces (including the "exterior" face). However, these graphs are combinatorially distinct. On the left, both triangles are adjacent to both hexagons; on the right, each triangle is adjacent to only one hexagon.

two hexagonal faces (one must also count the "exterior" face which bounds the remaining graph). However, the two graphs are combinatorially distinct as can be seen by looking at the triangles and hexagons. In the graph on the left, both triangles are adjacent to both hexagons; on the right, each triangle is adjacent to only one hexagon. Of the two types of grains, the one on the left appears significantly more often in a steady state microstructure than the one on the right. In the 25 steady state structures investigated, there were a total of 1917 grains (out of 269,555 total grains), which a *p*-vector (0, 2, 2, 2, 2, 0, 0). Only 16 of those, or less than one percent, were of the type on the right.

Tables 4.2, 4.3, 4.4, 4.5, and 4.6 show the most frequently occurring p-vectors as broken up into number of faces. Collecting accurate data for grains with more than 20 faces is difficult for two reasons. First, the number of grains with more than 20 faces is relatively small compared to the number of grains with fewer faces. Second, the total number of p-vectors with a given number of faces increases quickly with the number of faces, as described below. Thus, the distribution of p-vectors with a fixed number of faces becomes wider with increasing number of faces.

Vo	ronoi		Steady state	
p-vector	frequency	σ	<i>p</i> -vector frequent	$\sigma \sigma$
3 faces -	_	-	3 faces (3, 0, 0, 0, 0, 0, 0) 1.0000	1.0000
4 faces (0, 4, 0, 0, 0, 0, 0) -	1.0000	0.0000	4 faces (0, 4, 0, 0, 0, 0, 0) 0.9661 (2, 0, 2, 0, 0, 0, 0) 0.0339	0.0457 0.0457
5 faces (0, 2, 3, 0, 0, 0, 0) - -	1.0000 - -	0.0000		$\begin{array}{c} 0.0216 \\ 0.0174 \\ 0.0072 \end{array}$
6 faces (0, 2, 2, 2, 0, 0, 0) (0, 0, 6, 0, 0, 0, 0) - - -	0.7317 0.2683 - - -	0.0713 0.0713 - - -		$\begin{array}{c} 0.0386\\ 0.0397\\ 0.0085\\ 0.0043\\ 0.0015\end{array}$
$\begin{array}{c} \textbf{7 faces} \\ (0, 1, 3, 3, 0, 0, 0) \\ (0, 0, 5, 2, 0, 0, 0) \\ (0, 2, 2, 2, 1, 0, 0) \\ (0, 3, 0, 3, 1, 0, 0) \\ (0, 2, 3, 0, 2, 0, 0) \end{array}$	$\begin{array}{c} 0.3501 \\ 0.2105 \\ 0.1720 \\ 0.1538 \\ 0.1135 \end{array}$	0.0324 0.0285 0.0267 0.0222 0.0220	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	$\begin{array}{c} 0.0273\\ 0.0215\\ 0.0136\\ 0.0117\\ 0.0150\end{array}$
$\begin{array}{c} \textbf{8 faces} \\ (0, 1, 3, 3, 1, 0, 0) \\ (0, 0, 4, 4, 0, 0, 0) \\ (0, 2, 2, 2, 2, 2, 0, 0) \\ (0, 3, 1, 2, 1, 1, 0) \\ (0, 1, 4, 1, 2, 0, 0) \end{array}$	$\begin{array}{c} 0.2705 \\ 0.1719 \\ 0.1296 \\ 0.1063 \\ 0.0870 \end{array}$	$\begin{array}{c} 0.0166\\ 0.0140\\ 0.0115\\ 0.0124\\ 0.0114 \end{array}$	$\begin{array}{c c} \textbf{8 faces} \\ (0, 0, 4, 4, 0, 0, 0) \\ (0, 1, 3, 3, 1, 0, 0) \\ (0, 2, 2, 2, 2, 2, 0, 0) \\ (0, 1, 4, 1, 2, 0, 0) \\ (0, 0, 6, 0, 2, 0, 0) \end{array} \begin{array}{c} 0.5177 \\ 0.2481 \\ 0.0722 \\ 0.0722 \\ 0.0551 \\ 0.0336 \end{array}$	$\begin{array}{c} 0.0278\\ 0.0159\\ 0.0125\\ 0.0095\\ 0.0093\end{array}$

Table 4.2: Five most frequent *p*-vectors for grains with 3, 4, 5, 6, 7, and 8 faces. The frequency here is with respect to all grains with N faces. Data are averaged from 25 runs, each beginning with 100,000 grains and ending with roughly 11,700. The standard deviation σ is measured from the mean of the 25 samples.

Vo	ronoi		Steady state	
<i>p</i> -vector	frequency	σ	<i>p</i> -vector frequency	σ
$\begin{array}{c} \textbf{9 faces} \\ (0, 1, 3, 3, 2, 0, 0) \\ (0, 1, 4, 2, 1, 1, 0) \\ (0, 0, 4, 4, 1, 0, 0) \\ (0, 2, 2, 3, 1, 1, 0) \\ (0, 1, 2, 5, 1, 0, 0) \end{array}$	0.1817 0.0851 0.0775 0.0708 0.0627	0.0076 0.0062 0.0069 0.0078 0.0079	$ \begin{array}{ c c c c c c c c c c c c c c c c c c c$	$\begin{array}{c} 0.0134\\ 0.0156\\ 0.0148\\ 0.0099\\ 0.0092\\ \end{array}$
$\begin{array}{c} \textbf{10 faces} \\ (0, 0, 4, 4, 2, 0, 0) \\ (0, 1, 4, 2, 2, 1, 0) \\ (0, 2, 2, 3, 2, 1, 0) \\ (0, 1, 2, 5, 2, 0, 0) \\ (0, 1, 3, 4, 1, 1, 0) \end{array}$	$\begin{array}{c} 0.0849 \\ 0.0828 \\ 0.0682 \\ 0.0660 \\ 0.0660 \end{array}$	$\begin{array}{c} 0.0061 \\ 0.0042 \\ 0.0048 \\ 0.0043 \\ 0.0041 \end{array}$	$ \begin{array}{ c c c c c c c c c c c c c c c c c c c$	0.0126 0.0102 0.0087 0.0067 0.0077
$\begin{array}{c} \textbf{11 faces} \\ (0, 1, 3, 4, 2, 1, 0) \\ (0, 1, 4, 2, 3, 1, 0) \\ (0, 2, 2, 3, 3, 1, 0) \\ (0, 2, 3, 2, 2, 2, 0) \\ (0, 0, 4, 4, 3, 0, 0) \end{array}$	$\begin{array}{c} 0.0647\\ 0.0556\\ 0.0504\\ 0.0466\\ 0.0455\end{array}$	0.0032 0.0026 0.0034 0.0031 0.0026	$ \begin{array}{ c c c c c c c c c c c c c c c c c c c$	0.0094 0.0093 0.0117 0.0080 0.0077
$\begin{array}{c} \textbf{12 faces} \\ (0,1,3,4,3,1,0) \\ (0,1,4,3,2,2,0) \\ (0,2,3,3,2,1,1) \\ (0,2,2,4,2,2,0) \\ (0,1,4,2,4,1,0) \end{array}$	$\begin{array}{c} 0.0500 \\ 0.0330 \\ 0.0329 \\ 0.0323 \\ 0.0295 \end{array}$	$\begin{array}{c} 0.0025\\ 0.0025\\ 0.0016\\ 0.0018\\ 0.0018\end{array}$	$\begin{array}{c c} \textbf{12 faces} \\ (0, 0, 3, 6, 3, 0, 0) \\ (0, 0, 4, 4, 4, 0, 0) \\ (0, 0, 2, 8, 2, 0, 0) \\ (0, 0, 4, 5, 2, 1, 0) \\ (0, 0, 5, 3, 3, 1, 0) \end{array} \begin{array}{c} 0.1622 \\ 0.1092 \\ 0.1036 \\ 0.0780 \\ 0.0689 \end{array}$	$\begin{array}{c} 0.0139 \\ 0.0104 \\ 0.0075 \\ 0.0086 \\ 0.0064 \end{array}$

Table 4.3: Five most frequent *p*-vectors for grains with 9, 10, 11, and 12 faces. The frequency here is with respect to all grains with N faces. Data are averaged from 25 runs, each beginning with 100,000 grains and ending with roughly 11,700. The standard deviation σ is measured from the mean of the 25 samples.

Vo	ronoi		Steady state	
<i>p</i> -vector	frequency	σ	<i>p</i> -vector frequency	σ
$\begin{array}{c} \textbf{13 faces} \\ (0, 1, 4, 3, 3, 2, 0) \\ (0, 1, 3, 4, 4, 1, 0) \\ (0, 2, 3, 3, 3, 1, 1) \\ (0, 1, 3, 5, 2, 2, 0) \\ (0, 1, 4, 4, 2, 1, 1) \end{array}$	$\begin{array}{c} 0.0302 \\ 0.0290 \\ 0.0283 \\ 0.0270 \\ 0.0234 \end{array}$	0.0017 0.0013 0.0012 0.0016 0.0015	$\begin{array}{c c} \textbf{13 faces} \\ (0, 0, 3, 6, 4, 0, 0) \\ (0, 0, 4, 5, 3, 1, 0) \\ (0, 0, 3, 7, 2, 1, 0) \\ (0, 0, 4, 4, 5, 0, 0) \\ (0, 0, 5, 3, 4, 1, 0) \end{array} \begin{array}{c} 0.1386 \\ 0.0907 \\ 0.0907 \\ 0.0753 \\ 0.0578 \end{array}$	$\begin{array}{c} 0.0126\\ 0.0101\\ 0.0093\\ 0.0094\\ 0.0070\end{array}$
$\begin{array}{c} \textbf{14 faces} \\ (0, 1, 3, 5, 3, 2, 0) \\ (0, 2, 3, 4, 2, 2, 1) \\ (0, 1, 4, 4, 3, 1, 1) \\ (0, 1, 4, 3, 4, 2, 0) \\ (0, 2, 4, 2, 3, 2, 1) \end{array}$	0.0224 0.0220 0.0210 0.0191 0.0182	0.0011 0.0012 0.0013 0.0012 0.0015	$\begin{array}{c c c c c c c c c c c c c c c c c c c $	0.0085 0.0117 0.0110 0.0081 0.0083
$ \begin{array}{ c c c c c c c c c c c c c c c c c c c$	$\begin{array}{c} 0.0194\\ 0.0158\\ 0.0154\\ 0.0153\\ 0.0147\end{array}$	0.0016 0.0013 0.0012 0.0010 0.0011	$ \begin{array}{ c c c c c c c c c c c c c c c c c c c$	$\begin{array}{c} 0.0143\\ 0.0085\\ 0.0062\\ 0.0090\\ 0.0076\end{array}$
$\begin{array}{c} \textbf{16 faces} \\ (0,2,3,4,4,2,1) \\ (0,1,4,5,3,2,1) \\ (0,2,4,3,3,3,1) \\ (0,1,4,4,4,3,0) \\ (0,1,5,3,4,2,1) \end{array}$	$\begin{array}{c} 0.0149 \\ 0.0148 \\ 0.0125 \\ 0.0111 \\ 0.0108 \end{array}$	$\begin{array}{c} 0.0009\\ 0.0010\\ 0.0009\\ 0.0010\\ 0.0010\end{array}$	$\begin{array}{c c} \textbf{16 faces} \\ (0,0,3,7,5,1,0) \\ (0,0,4,6,4,2,0) \\ (0,0,4,5,6,1,0) \\ (0,0,2,9,4,1,0) \\ (0,0,3,8,3,2,0) \end{array} \begin{array}{c} 0.0659 \\ 0.0636 \\ 0.0451 \\ 0.0372 \\ 0.0363 \end{array}$	$\begin{array}{c} 0.0090\\ 0.0096\\ 0.0066\\ 0.0070\\ 0.0078\end{array}$

Table 4.4: Five most frequent *p*-vectors for grains with 13, 14, 15, and 16 faces. The frequency here is with respect to all grains with N faces. Data are averaged from 25 runs, each beginning with 100,000 grains and ending with roughly 11,700. The standard deviation σ is measured from the mean of the 25 samples.

Vo	ronoi		Steady stat	e
<i>p</i> -vector	frequency	σ	<i>p</i> -vector frequ	ency σ
$ \hline \hline \begin{array}{ c c c c c c c c c c c c c c c c c c c$	$\begin{array}{c} 0.0120\\ 0.0106\\ 0.0101\\ 0.0100\\ 0.0096 \end{array}$	0.0011 0.0012 0.0009 0.0010 0.0009	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	515 0.0091 474 0.0072 430 0.0084 309 0.0051 307 0.0071
$\begin{tabular}{ c c c c c }\hline \hline 18 faces \\ (0, 2, 3, 5, 4, 3, 1) \\ (0, 1, 4, 5, 5, 2, 1) \\ (0, 1, 5, 4, 4, 3, 1) \\ (0, 2, 4, 4, 4, 2, 2) \\ (0, 1, 4, 6, 3, 3, 1) \\ \hline \end{tabular}$	0.0085 0.0085 0.0083 0.0083 0.0077	0.0010 0.0009 0.0010 0.0009 0.0009	$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	423 0.0082 374 0.0098 345 0.0066 306 0.0086 278 0.0079
$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	$\begin{array}{c} 0.0067\\ 0.0067\\ 0.0062\\ 0.0060\\ 0.0059\end{array}$	0.0012 0.0009 0.0009 0.0009 0.0009	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	350 0.0103 314 0.0081 297 0.0062 235 0.0070 221 0.0067
$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	$\begin{array}{c} 0.0061 \\ 0.0053 \\ 0.0050 \\ 0.0047 \\ 0.0046 \end{array}$	$\begin{array}{c} 0.0011 \\ 0.0009 \\ 0.0011 \\ 0.0010 \\ 0.0010 \end{array}$	$\begin{array}{c c} \textbf{20 faces} \\ (0, 0, 4, 7, 6, 3, 0) \\ (0, 0, 3, 8, 7, 2, 0) \\ (0, 0, 3, 9, 5, 3, 0) \\ (0, 0, 5, 6, 6, 2, 1) \\ (0, 0, 4, 8, 5, 2, 1) \end{array} \begin{array}{c} 0.02 \\ 0.02 \\ 0.03 \\ 0$	303 0.0096 251 0.0087 229 0.0076 227 0.0079 216 0.0068

Table 4.5: Five most frequent *p*-vectors for grains with 17, 18, 19, and 20 faces. The frequency here is with respect to all grains with N faces. Data are averaged from 25 runs, each beginning with 100,000 grains and ending with roughly 11,700. The standard deviation σ is measured from the mean of the 25 samples.

Vo	ronoi			Stead	ly state	
<i>p</i> -vector	frequency	σ	<i>p</i> -v	rector	frequency	σ
	$\begin{array}{c} 0.0046 \\ 0.0042 \\ 0.0041 \\ 0.0041 \\ 0.0041 \end{array}$	0.0011 0.0012 0.0011 0.0009 0.0010	$\begin{array}{c} \textbf{21 face} \\ (0, 0, 4, \\ (0, 0, 4, \\ (0, 0, 5, \\ (0, 0, 3, \\ (0, 0, 5, $	es (7, 7, 3, 0) (8, 6, 2, 1) (7, 5, 3, 1) (9, 6, 3, 0) (6, 7, 2, 1)	$\begin{array}{c} 0.0238 \\ 0.0205 \\ 0.0180 \\ 0.0171 \\ 0.0168 \end{array}$	0.0069 0.0086 0.0070 0.0078 0.0081
	$\begin{array}{c} 0.0041 \\ 0.0037 \\ 0.0035 \\ 0.0033 \\ 0.0033 \end{array}$	0.0016 0.0010 0.0014 0.0018 0.0016	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	25 7, 6, 3, 1) 7, 8, 3, 0) 8, 7, 2, 1) 8, 6, 4, 0) 9, 5, 3, 1)	$\begin{array}{c} 0.0211 \\ 0.0184 \\ 0.0174 \\ 0.0157 \\ 0.0156 \end{array}$	0.0094 0.0093 0.0079 0.0078 0.0064
	$\begin{array}{c} 0.0030\\ 0.0030\\ 0.0029\\ 0.0029\\ 0.0028\end{array}$	0.0022 0.0023 0.0016 0.0015 0.0018	23 face (0, 0, 4, (0, 0, 4, (0, 0, 5, (0, 0, 4, (0, 0, 3,	es 9, 6, 3, 1) 7, 9, 3, 0) , 7, 7, 3, 1) , 8, 8, 2, 1) , 9, 8, 3, 0)	$\begin{array}{c} 0.0191 \\ 0.0145 \\ 0.0133 \\ 0.0127 \\ 0.0126 \end{array}$	0.0102 0.0120 0.0115 0.0092 0.0110
$\begin{array}{c} \textbf{24 faces} \\ (0, 3, 4, 5, 6, 3, 2) \\ (0, 2, 6, 5, 4, 4, 2) \\ (0, 2, 5, 7, 3, 4, 2) \\ (0, 3, 5, 5, 4, 3, 3) \\ (0, 3, 5, 4, 5, 4, 2) \end{array}$	$\begin{array}{c} 0.0027\\ 0.0023\\ 0.0023\\ 0.0021\\ 0.0021 \end{array}$	0.0023 0.0020 0.0023 0.0023 0.0019	$\begin{array}{c} \textbf{24 face} \\ (0, 0, 4, 0, 0, 5, 0, 0, 0, 4, 0, 0, 5, 0, 0, 0, 3, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,$	es 8, 8, 4, 0) 8, 6, 4, 1) 9, 7, 3, 1) 7, 8, 3, 1) , 1, 0, 7, 4)	$\begin{array}{c} 0.0151 \\ 0.0145 \\ 0.0138 \\ 0.0111 \\ 0.0103 \end{array}$	$\begin{array}{c} 0.0093\\ 0.0098\\ 0.0083\\ 0.0119\\ 0.0092 \end{array}$

Table 4.6: Five most frequent *p*-vectors for grains with 21, 22, 23, and 24 faces. The frequency here is with respect to all grains with N faces. Data are averaged from 25 runs, each beginning with 100,000 grains and ending with roughly 11,700. The standard deviation σ is measured from the mean of the 25 samples.

Faces	Distinct shapes	Distinct p -vectors	Ratio
4	1	1	1
5	1	1	1
6	2	2	1
7	5	5	1
8	14	13	1.08
9	50	33	1.52
10	233	82	2.84
11	1249	372	3.36
12	7595	1088	6.98
13	49566	2500	19.83
14	339722	3911	86.86

Table 4.7: For a given number of faces $4 \le N \le 14$, we list the number of combinatorially distinct convex polyhedra and the number of realizable *p*-vectors. In the last column we provide the ratio between these two numbers; this number grows very quickly.

To provide a glimpse of how much information the *p*-vector provides, we consider the number of realizable *p*-vectors in comparison with the number of combinatorially distinct grains for a given number of faces. Table 4.7 lists the number of combinatorially distinct polyhedra with a given number of faces [126, 127, 128]. The table also lists the number of distinct *p*-vectors of these shapes; we have calculated these numbers from a list of dual triangulations of all of these polyhedra that can be in [129]. The ratio between these numbers, also listed in the table, appears to grows superexponentially. From this Table we can deduce that amongst all grains with 14 faces, there exist *p*-vectors with more than 86 combinatorially distinct realizations. It appears that as the number of faces increases, the number of combinatorially distinct polyhedral graphs grows significantly faster than the possible number of *p*-vectors.

The question then arises, how can we describe the combinatorial structure of a grain in a simple way that allows fast comparison with other grains? We have already seen that the number of faces of a grain is insufficient, even if we specify the particular number of each type of face. Can we find an invariant of a grain that uniquely determines its entire combinatorial structure? In the next subsection we introduce a grain invariant that uniquely describes its entire combinatorial structure. We use this data to help us investigate steady state cell structures.

Weinberg vectors

A complete method of characterizing three-dimensional cells can be built on the work of [130] and [131]. The motivation for those papers is the problem of comparing two graphs and determining whether or not they are isomorphic. Little is still known about the general case: although this is an NP problem, it is not known whether it belongs to P or NP-complete [132]; it is one of very few problems of this kind.

In [130], Louis Weinberg introduced a method of encoding the combinatorial structure of a 3connected² planar graph in a two-dimensional array of "codes". These codes, which can be generated relatively quickly, can then be used to determine whether two graphs are isomorphic. Two graphs G_1 and G_2 are said to be isomorphic if there is a bijection f between their vertex sets such that any two vertices u, v are adjacent in G_1 if and only if f(u), f(v) are adjacent in G_2 .

Although our graphs are always planar (they are embedded in a topological 2-sphere), they are not generally 3-connected. The reason for this is the presence of two-sided faces and tessellations thereof. Figure 4.29 shows a typical 2-sided face that can appear in a grain growth structure. It is



Figure 4.29: A typical two-sided face in a polyhedron.

clear that removing the two edges a and b will disconnect the two-sided face from the rest of the graph. Weinberg's algorithm cannot determine isomorphism of graphs that are not 3-connected. This, however, is actually preferred for the following reason.

Consider a piece of a graph shown in Figure 4.30. Two facts are clear. First, a graph containing



Figure 4.30: A tessellation of a two-sided face.

this as a subgraph is not 3-connected—removing edges a and b would leave the subgraph disconnected from the rest of the graph. It is also clear that rotating the subgraph about an axis that passes through edges a and b will leave the remaining graph isomorphic. However, it should also be clear that if this graph is associated with a grain, then the combinatorial structure of the grain changes with this rotation. For example, Face 1 will gain one side and Face 2 will lose one side. Weinberg's algorithm *does* distinguish between these two structures even though the graphs are isomorphic because it captures additional structure provided by the embedding of the graph on the sphere.

 $^{^{2}\}mathrm{A}$ graph is said to be 3-connected if it remains connected with the removal of fewer than three edges.



Figure 4.31: Rotating the tessellated two-sided face about an axis that passes through edges a and b leaves the graphs isomorphic but changes the combinatorial structure of the grain's surface.

Since we do want to distinguish between these two different structures, Weinberg's algorithm is actually ideally suited for describing the combinatorial structure of our grains even though they are not 3-connected.

Weinberg's algorithm works by constructing a set of Euler paths in a graph associated with each of our grains. An Euler path is a path in a graph that traverses every edge exactly once. Weinberg showed that two 3-connected, planar graphs have identical code arrays if and only if they are isomorphic. For our purposes, the combinatorial structure of two grains is identical if and only if their code arrays are identical. We construct these code arrays for each grains and use them to compare pairs of grains.

Euler long ago showed that a necessary condition for the existence of an Euler path in a graph is that every vertex must have an even number of edges adjacent to it. This is clearly necessary: a path segment passing through a vertex must "enter" and "exit" via exactly two edges. In 1873 Carl Hierholzer showed that this was also a sufficient condition. However, neither Euler nor Hierholzer explained how to construct such a path. Trmaux [133] introduced a method which involves building a path while traversing the graph, leaving marks at vertices after they have been "visited". One then uses these marks and follows a set of rules that determines how to continue along a path. Weinberg builds on this method, adding a numbering scheme to record the path. Instead of only marking a vertex as visited or not, vertices are also numbered. The numbers used in this Euler path then form a code which encodes all of combinatorial structure of the graph. This technique is repeated starting at different vertices in the graph and traveling along different initial edges. The codes are then ordered lexicographically. Two planar graphs with identical combinatoric structure will have identical sets of these codes.

We begin with an undirected graph composed of the vertices and edges of a grain in our system; the graph's embedding in the oriented grain surface allows us to speak of "turning right" or "turning left" when traveling along a path in this graph. To guarantee that each graph generates an Euler path, we travel along every edge twice, once in each direction. In effect we have a directed graph with a an equal number of inward edges at each vertex as outward edges; this guarantees the existence of an Euler path in this graph.

In our implementation of the algorithm, every vertex in the graph is labeled with a number in $\{0, 1, 2...N\}$, where N is the number of vertices in the graph; the algorithm begins with the label of each vertex initially set to 0. To construct each code, we begin by choosing one edge in the graph and a direction along which to travel. The direction allows us to identify the *initial* vertex of the edge, which we label 1 and append to the initially empty code. We then "travel" along the edge in the edge in the chosen direction doing the following:

- 1. When we reach a vertex whose label is 0, we relabel it with the next largest integer that has not been used; we append this label to our code. We then "turn right" and continue traveling in the graph.
- 2. When we reach a vertex whose label is not 0, we append its label to our code and then:
 - (a) If we reached this vertex by traveling along an edge that has not been traversed at all, then we return to the previous vertex along the same edge but in the opposite direction.
 - (b) If we reached this vertex by traveling along an edge that has been traversed (in the opposite direction), then we "turn right" and continue traveling; if that edge has previously been traversed in that direction, then we instead "turn left" and continue traveling; if that edge has also been previously traversed then we stop—we have traversed every edge in the graph twice, once in each direction.

While following rules (1), (2a), and (2b), we record a list of the vertices in the order that they have been visited. Each vertex will have been visited exactly three times except for the vertex from which we begin and at which we end, which is visited four times. We then repeat this procedure for each edge in the graph and in each direction. If there are E undirected edges in the graph, then we should then have 2E codes, one beginning at each directed edge. Because we would like the mirror-image of a grain to be considered identical to itself, we also repeat this procedure for all directed edges though changing the algorithm so that every right turn is replaced by a left turn, and visa versa. We then finish with 4E codes for each graph with E undirected edges.

After constructing this set of codes, we sort them lexicographically. Since the codes depend only on the combinatorial structure of the planar graph, and not on any prior labeling, two graphs with identical combinatorial structure will have identical sets of codes. Since any one of the codes is enough to reconstruct the graph (using a method derivable from the above algorithm), two plane graphs have identical combinatorial structure if and only if they have identical first lexicographical codes. After constructing these codes we then save only the first one.

In Figure 4.32 we provide an example of how we construct one code for an example graph. We



Figure 4.32: We construct a code associated with a graph of the edges and vertices of a grain. (a) In the very beginning, the label of every vertex is 0. (b) Here we choose to begin with the upper left vertex and begin constructing an Euler path by traversing the vertical edge on the left side; we label the first vertex 1 and the second one 2. (c-e) We continue traversing the edges, following rules (1) and (2), appending labels from the vertices as they are passed. (f) The code is completed when we return to our starting node and cannot travel along any other edge.

begin with all vertices labeled 0. For this particular code we begin from the upper left vertex and begin by traveling along the vertical edge on the left side of the figure. We travel along the edges according to rules (1) and (2), labeling the vertices as we reach them and recording the vertex labels as we cross vertices in our path. The final "code" for this particular path is 1231341424321. We then repeat this procedure for every directed edge in the graph, and then repeat this for the identical graph though modifying the algorithm by replacing every right turn with a left turn and visa versa.

One insightful outcome of this encoding method is what is says about the group's automorphism group. In this particular case, the code we constructed would have been the same regardless of which directed edge we used to start and regardless of whether we used the original algorithm or its mirror-image dual. Our set of codes would be 24 copies of the code 1231341424321. The graph in the example was that of the tetrahedron, and it is no coincident that the order of the tetrahedral group is 24. Each graph G with E edges has 4E codes. If $\Gamma(G)$ is the group of automorphisms associated with it and $ord(\Gamma)$ is the order of that group, then we will have exactly $4E/ord(\Gamma)$ distinct codes, each repeating $ord(\Gamma)$ times. Weinberg [131] uses this algorithm to prove that the maximum order of the group of a triply connected planar graph is 4E. He also shows that only the five Platonic graphs realize this upper bound.

In what follows, we report the most frequent combinatoric structures found in grains in both the initial Voronoi and the steady state structures. Tables 4.8 and 4.9 show a list of the 25 most frequently occurring combinatorial structures in the Voronoi and steady state structures. The tables list the number of faces of each combinatorial structure, an abbreviated p-vector, the first lexicographical code, which can be used to reconstruct the structure, the order of the associated automorphism group, and the frequency of this structure in the system. The frequency is averaged from data aggregated over 25 runs. We use letters instead of numbers to represent the codes because they use less space; of course, the particular symbols used is of no consequence.

The data in these tables allow us to make a number of observations about the Voronoi and steady state grain-growth structures:

- In the Voronoi structure, the 25 most frequent combinatorial types of grains contain many triangles and a considerable number of 7-sided faces; in the steady state, these grains have very few triangles and no 7-sided faces. Almost all faces of these very common grains have either 4, 5, or 6 sides. We made a similar observation when considering *p*-vector data.
- In the Voronoi structure, the 25 most frequent combinatorial types of grains have 8, 9, 10, or 11 faces each. In the steady state, the 25 most frequent combinatorial types of grains have 5, 6, 7, 8, 9, 10, 11, and 12 faces each.
- 3. The automorphism groups associated with grains of the Voronoi structure tend to be of relatively small order (≤ 8); in the steady state structure we have significantly more grains with larger associated symmetry groups. In both structures, there seems to be little correlation between the order of the associated automorphism group and the frequency of the particular combinatorial type.
- 4. In the Voronoi structure, the 25 most frequent combinatorial types of grains make up only 2.5% of the grains; in the steady state structure, this number is 26.1%. We made a similar observation when considering *p*-vector data.

These data are, of course, only a first step in attempting to understand how different shapes are "packed together" to form different three-dimensional cell structures. We don't know why steady state grains seem more symmetric than those in Voronoi tessellations. We don't understand why grain "shapes" are distributed so differently in the two structures. It is hoped that a more complete understanding of mean curvature flow on cell structures will help us understand the geometry and combinatorial structure of these incredibly simple yet complicated objects.

4.5 Conclusions

The MacPherson–Srolovitz relation provides an exact expression for the volume evolution of individual grains in isotropic polycrystalline microstructures. We have developed a simulation method for evolving these structures in a manner consistent with this exact relation. We demonstrate that the present method is simultaneously capable of high accuracy and computational efficiency. Furthermore, the present method accounts for all types of topological reactions that are observed to occur during normal grain growth.

We use this method to simulate large systems beginning with a total of 2.5 million grains. This provides us with ample data, larger systems than previously observed with any 3D simulation method. We report many statistics of grain growth structures including distributions of faces, edges, and volumes. We also introduce a method of completely describing the combinatorial structure of individual grains, allowing us to investigate the distribution of grain "shapes" in Voronoi and steady state structures.

In future work we might consider beginning with initial conditions and watching how they relax to identical steady state structures. This would help illustrate the existence of a universal steady state, as described in Chapter 1. We might also consider other dynamics on three-dimensional cell structures and the steady states which they produce, much like we did in Chapter 2 for onedimensional cell structures. One particularly dynamic of much interest is the anisotropic grain growth dynamic, where the grain boundary energies and mobilities are not uniform throughout a system. Since almost all physical systems have some anisotropy, understanding the impact of anisotropy on the long-time steady state microstructure is of great interest.

In Chapter 5, we compare and contrast grain growth in two and three dimensions, seeing how the dimensions affects various system properties. In that chapter we report much more data about three-dimensional cell structures.

freq.	0.00273	0.00159	0.00155	0.00127	0.00117	0.00101	0.00099	0.00098	0.00094	0.00093	0.00093	0.00092	0.00092	0.00091	0.00090	0.00089	0.00086	0.00081	0.00081	0.00079	0.00078	0.00077	0.00075	0.00071	0.00069
$ord(\Gamma)$		2	2	1	4	×		2	2		1	2		1	4	2			2	2	2	9	2	1	4
Weinberg code	ABCACDEFAFGHIBIJKDKLELMGMNHNJNMLKJIHGFEDCBA	ABCACDEFAFGHBHIJDJKEKLGLILKJIHGFEDCBA	ABCDADEFAFGHIBIJKCKLMEMNGNOHOPJPLPONMLKJIHGFEDCBA	ABCACDEFAFGHIBIJKLDLMEMNGNOPHPJPOKONMLKJIHGFEDCBA	ABCDADEFAFGHBHIJCJKLELMGMNINKNMLKJIHGFEDCBA	ABCDADEFAFGHBHIJCJKEKLGLILKJIHGFEDCBA	ABCACDEFAFGHIBIJKLDLMEMNOGOPHPJPONKNMLKJIHGFEDCBA	ABCDADEFGAGHIBIJKCKLMEMNFNOPHPQJQRLRORQPONMLKJIHGFEDCBA	ABCACDEFAFGHBHIJDJKLELMGMNINKNMLKJIHGFEDCBA	ABCDADEFAFGHBHIJKCKLMEMNOGOPIPQJQRLRNRQPONMLKJIHGFEDCBA	ABCACDEFGAGHIJBJKLMDMNENOPFPHPOQRIRKRQLQONMLKJIHGFEDCBA	ABCDADEFAFGHBHIJCJKLELMNGNOIOPKPMPONMLKJIHGFEDCBA	ABCACDEFAFGHIBIJKDKLELMNGNOHOPJPMPONMLKJIHGFEDCBA	ABCACDEFAFGHIBIJKLDLMNENGNMOPHPJPOKOMLKJIHGFEDCBA	ABCDADEFAFGHBHIJKCKLELMNGNINMJMLKJIHGFEDCBA	ABCACDEFAFGHIBIJKDKLMEMNGNOPHPJPOLONMLKJIHGFEDCBA	ABCACDEFAFGHIBIJKLDLMNENOGOPQHQJQPRKRMRPONMLKJIHGFEDCBA	ABCACDEFAFGHIBIJKLDLMNENOGOPKPMPONMLKJHJIHGFEDCBA	ABCACDEFAFGHIBIJKDKLMEMNGNOHOPJPLPONMLKJIHGFEDCBA	ABCACDEFAFGHIBIJKLDLMEMNGNKNMLKJHJIHGFEDCBA	ABCACDEFGAGHIJBJKLDLMEMNFNOHOPIPKPONMLKJIHGFEDCBA	ABCDADEFGAGHIBIJKCKLMEMNFNOHOPJPLPONMLKJIHGFEDCBA	ABCDADEFAFGHIBIJKCKLMEMNOGOPHPQJQRLRNRQPONMLKJIHGFEDCBA	ABCACDEFAFGHIJBJKLMDMNENOPGPHPOLONMLKIKJIHGFEDCBA	ABCACDEFAFGHIBIJKDKLELGLKJHJIHGFEDCBA
<i>p</i> -vector	(0,1,3,3,2,0,0)	(0,1,3,3,1,0,0)	(0,0,4,4,2,0,0)	(0,1,3,4,1,1,0)	(0,0,4,4,1,0,0)	(0, 0, 4, 4, 0, 0, 0)	(0,1,4,2,2,1,0)	(0,0,3,6,2,0,0)	(0,1,2,5,1,0,0)	(0,0,4,4,3,0,0)	(0,1,4,2,3,1,0)	(0,0,4,4,2,0,0)	(0,1,3,3,3,0,0)	(0,1,4,2,2,1,0)	(0,0,5,2,2,0,0)	(0,1,2,5,2,0,0)	(0,1,2,6,1,1,0)	(0,2,2,3,2,1,0)	(0,1,2,5,2,0,0)	(0,2,2,3,1,1,0)	(0,1,4,1,4,0,0)	(0,0,3,6,1,0,0)	(0,0,4,4,3,0,0)	(0, 3, 2, 1, 2, 2, 0)	(0,2,2,2,0,0)
Faces	6	×	10	10	6	×	10	11	6	11	11	10	10	10	6	10	11	10	10	6	10	10	11	10	×

Table 4.8: A list of the 25 most frequently occurring combinatorial structures in the Voronoi system. The first column F lists the number of faces; the second column lists an abbreviated *p*-vector: $(p_2, p_3, p_5, p_6, p_7, p_8)$; the third column lists the codes which can be used to reconstruct the combinatorial structure of the grain; the fourth column provides the order of the symmetry group associated with the particular combinatorial structure; the last column provides the frequency of this combinatorial structure in the Voronoi system. Data were take from 25 initial systems, each containing 100,000 grains.

freq.	0.0283	0.0186	0.0161	0.0152	0.0149	0.0141	0.0138	0.0138	0.0136	0.0115	0.0108	0.0093	0.0086	0.0079	0.0076	0.0076	0.0068	0.0065	0.0064	0.0063	0.0050	0.0048	0.0047	0.0045	0.0040
$ord(\Gamma)$	x	12	4	2	20	9	9		2	2	4	4	48	16	4	2	12	Η		2	4	2	2	4	4
Weinberg code	ABCDADEFAFGHBHIJCJKEKLGLILKJIHGFEDCBA	ABCDADEFGAGHIBIJKCKLELMFMNHNJNMLKJIHGFEDCBA	ABCDADEFAFGHBHIJCJKLELMGMNINKNMLKJIHGFEDCBA	ABCDADEFAFGHIBIJKCKLMEMNGNOHOPJPLPONMLKJIHGFEDCBA	ABCDADEFAFGHBHICIJEJGJIHGFEDCBA	ABCDADEFGAGHIBIJKCKLMEMNFNOHOPJPLPONMLKJIHGFEDCBA	ABCACDEFAFGHBHIDIJEJGJIHGFEDCBA	ABCACDEFAFGHIBIJKDKLELMGMNHNJNMLKJIHGFEDCBA	ABCACDEFAFGHBHIJDJKEKLGLILKJIHGFEDCBA	ABCDADEFGAGHIBIJKCKLMEMNFNOPHPQJQRLRORQPONMLKJIHGFEDCBA	ABCDADEFGAGHIBIJKCKLMEMNOFOPHPQJQRLRNRQPONMLKJIHGFEDCBA	ABCACDEAEFGBGHDHFHGFEDCBA	ABCDADEFAFGBGHCHEHGFEDCBA	ABCDADEFGAGHIBIJKCKLELMNFNOHOPJPMPONMLKJIHGFEDCBA	ABCDADEFAFGHBHIJKCKLELMNGNINMJMLKJIHGFEDCBA	ABCDADEFAFGHBHIJCJKLELMNGNOIOPKPMPONMLKJIHGFEDCBA	ABCACDEAEFBFDFEDCBA	ABCDADEFGAGHIBIJKCKLMEMNOFOPQHQRJRSTLTNTSPSRQPONMLKJIHGFEDCBA	ABCDADEFAFGHBHIJKCKLMEMNOGOPIPQJQRLRNRQPONMLKJIHGFEDCBA	ABCDADEFAFGHIBIJKCKLMEMNOGOPHPQJQRLRNRQPONMLKJIHGFEDCBA	ABCDADEFGAGHIBIJKLCLMNENOFOPQHQJQPRKRMRPONMLKJIHGFEDCBA	ABCDADEFGAGHIBIJKCKLMEMNOFOPHPQRJRSLSTNTQTSRQPONMLKJIHGFEDCBA	ABCDADEFGAGHIBIJKLCLMNENOPFPQHQRJRSKSTMTOTSRQPONMLKJIHGFEDCBA	ABCDADEFGAGHIBIJKLCLMNENOPFPQHQRJRSTKTMTSOSRQPONMLKJIHGFEDCBA	ABCDADEFAFGHBHIJKCKLMEMNOGOIONPJPLPNMLKJIHGFEDCBA
<i>p</i> -vector	(0, 0, 4, 4, 0, 0, 0)	(0, 0, 3, 6, 0, 0, 0)	(0, 0, 4, 4, 1, 0, 0)	(0,0,4,4,2,0,0)	(0,0,5,2,0,0,0)	(0,0,3,6,1,0,0)	(0,1,3,3,0,0,0)	(0,1,3,3,2,0,0)	(0,1,3,3,1,0,0)	(0,0,3,6,2,0,0)	(0,0,2,8,1,0,0)	(0,2,2,2,0,0,0)	(0, 0, 6, 0, 0, 0, 0)	(0, 0, 2, 8, 0, 0, 0)	(0,0,5,2,2,0,0)	(0,0,4,4,2,0,0)	(0, 2, 3, 0, 0, 0, 0, 0)	(0, 0, 3, 6, 3, 0, 0)	(0,0,4,4,3,0,0)	(0,0,4,4,3,0,0)	(0, 0, 4, 4, 3, 0, 0)	(0, 0, 2, 8, 2, 0, 0)	(0, 0, 3, 6, 3, 0, 0)	(0,0,2,8,2,0,0)	(0,0,5,2,3,0,0)
Faces	×	6	6	10	7	10	7	6	×	11	11	9	9	10	6	10	ъ	12	11	11	11	12	12	12	10

Table 4.9: A list of the 25 most frequently occurring combinatorial structures in the steady state system. The first column F lists the number of faces; the second column lists an abbreviated *p*-vector: $(p_2, p_3, p_4, p_5, p_6, p_7, p_8)$; the third column lists the codes which can be used to reconstruct structure; the last column provides the frequency of this combinatorial structure in the steady state system. Data were take from 25 systems each containing roughly 10,700 grains in a steady state; in total, data were take from almost 250,000 grains. the combinatorial structure of the grain; the fourth column provides the order of the symmetry group associated with the particular combinatorial

Chapter 5

Comparing 2D and 3D Systems

Having considered one-, two-, and three-dimensional grain growth systems, we are now in a position to make some remarks about dynamical cell structures in general and on the role which dimension plays. Because the primary motivation for the problems considered in this thesis involves curvature driven evolution, we focus primarily on the two- and three-dimensional systems, though make reference to one-dimensional systems where relevant.

We also use this opportunity to report data of two-dimensional cross-sections of three-dimensional structures. Experimentally, this type of data is the most accessible and is therefore frequently reported in papers of an experimental nature. Practically, it is substantially more difficult to directly observe the internal microstructure of a three-dimensional material than it is to observe a cross-sectional slice of one. Observations and analysis of two-dimensional systems and cross-sections of three-dimensional ones are often made with hope that they will shed light on the actual three-dimensional case [135, 58, 136].

Monte-Carlo simulations by Anderson [33], phase-field simulations by Kim [97] and vertex model simulations by Nagai [108] and Weygand [109] performed similar comparisons, though, the data presented here are taken from systems substantially larger than those considered in earlier works.

Figure 5.1 shows examples of a two-dimensional grain-growth structure, a three-dimensional grain-growth structure, and a cross-section of a three-dimensional grain-growth structure. Notice that the edges in the cross-section are much rougher than those in the genuine two-dimensional system. Also notice that the angles in the cross-section diverge from the 120° equal angles that appear in the two-dimensional system.

^{*}The content of this chapter has been adapted from [134].



Figure 5.1: Examples of a two-dimensional grain-growth structure, a three-dimensional grain-growth structure, and a cross-section of a three-dimensional grain-growth structure.

To study each of these steady-state systems, we begin with very large Voronoi tessellations of the unit square or unit cube with periodic boundaries, as described in previous chapters. The equations of motion, derived from the von Neumann–Mullins relation and the MacPherson–Srolovitz relation, induce curvature flow in two dimensions and mean curvature flow in three dimensions to simulate the process of normal grain growth. We allow the systems to evolve until all observed scale-invariant properties stabilize at asymptotic values. We associate these asymptotic values with the steady state which characterizes all, or almost all, isotropic grain-growth structures which evolve through mean curvature flow, and in this sense it is universal.

One measurable difference between one-, two-, and three-dimensional systems is the rate at which these systems relax towards a steady state from an initial Voronoi construction. To measure this, we consider what fraction of an initial cell structure's cells must disappear before its scale invariant properties reach asymptotic limits. Because this fraction depends only on the initial conditions and on the dynamic used (and not on the initial size of the system), we can use this measure to compare how long cell structures take to relax to asymptotic steady states in various dimensions. We consider the one-, two-, and three-dimension Voronoi structures as comparable initial conditions.

In one dimension, we found that by the distribution of cell sizes in a system that began from a Voronoi construction (Initial State B) did not reach asymptotic values until more than 99.98% of the initial cells had disappeared using Dynamic 1. In two dimensions, the distributions of edges per grain and of normalized areas reached asymptotic values by the time that only 96% of the initial grains had disappeared. In three dimensions, the distribution of faces per cell and of normalized volumes had all reached asymptotic values by the time that only 90% of the initial grains had disappeared. It appears that the increase in dimension decreases the time, as measured in fraction of grains that disappeared, necessary to reach steady states. This might be explained in one of two ways. It is

possible that the Voronoi construction is more "similar" to the steady state in higher dimensions. Alternatively, it is possible that someone

For two-dimensional simulation data, we begin with one system containing 10,000,000 grains, and allow it to evolve until it has reached a steady state when roughly 400,000 grains remain. Statistics from a total of 1,000,000 cells were collected from four different points in time after the system had reached that point.

For three-dimensional systems, we collected data from thirteen trials, each of which began with 100,000 grains and each of which had reached a steady state when roughly 10,700 grains remained. We also collected data from two larger systems, one which began with 250,000 grains and which reached a steady state when roughly 17,000 grains remained, and one which began with 500,000 grains and which reached a steady state when roughly 42,000 grains remained. We report statistics from roughly 200,000 grains in the steady state. These large sample sets are necessary for producing accurate statistics of the steady state structure.

For the cross-sectional data, we used cross-sections of all three-dimensional samples. In each system we took a series of cross-sections parallel to the xy-, yz-, and zx- planes. We spaced the cross-sections roughly five grain diameters apart, to decrease any correlation between nearby cross-sections. In total we sampled roughly 100,000 grains for this data. We use the term 3DX as shorthand to refer to data from these cross-sections.

We divide data for the three systems into three groups. First we report statistics that describe the distributions of individual grains in the system as classified by various single-grain quantities such as their their areas or volumes. We call these measurements *point quantities*. Next we report correlations between neighboring grains that are a certain metric distance apart. For example, we consider how the size of a grain is correlated with grains that are three grain diameters away. Last, we consider relationships between a grain and its topological neighbors. That is, we consider the correlation between nearest neighbors, second nearest neighbors, and so forth. We introduce a method of describing second and third nearest neighbors which is different from the standard method used in the literature, and show that this definition should be used in reporting data of this kind. These correlations between a grain and its neighbors can help us measure the local order inherent in grain-growth structures.

Where possible, we compare analogous results for two-dimensional simulations, three-dimensional simulations and cross-sections of three-dimensional simulations, referred to as 2D, 3D, and 3DX, respectively. Error bars in all plots indicate the standard error from the mean of these samples.

5.1 Point quantities

The statistics mostly frequently reported in the literature for steady state structures are distributions of grains characterized by their surface areas, volumes, number of faces, and other grain-specific quantities. This information can tell us whether a cell structure has a few very large grains and many very small grains, or whether most of the grains are roughly the same size. It can also tell us how circular or spherical grains are in a particular microstructure. This type of data is reported in the first section.

Edges and faces

Since the von Neumann–Mullins relation indicates that the rate of change of a grain's area in two dimensions depends only on its number of edges, a natural feature to consider is the distribution of the number of edges per grain for the 2D and 3DX structures. These appear in Figure 5.2(a). Analogous quantities in three dimensions include the number of edges per face and the number of faces per grain, both of which appear in Figures 5.2(a) and 5.2(b), respectively. Although considerations



Figure 5.2: (a) Probability distributions for the number of edges of a grain for 2D and 3DX and of a face for 3D. The means and standard deviations are 6.000 and 1.273 for 2D, 6.000 and 1.813 for 3DX, and 5.128 and 1.273 for 3D. Splines provide a guide for the eye, and error bars are smaller than the markers. (b) Probability distribution for the number of faces of a grain in 3D. The mean and standard deviation are 13.766 and 4.732, respectively. Error bars are smaller than the markers.

of physics and topology require that the average number of edges per grain for the 2D and 3DX structures be precisely six (c.f. Section 3.2), the constraint on the average number of edges per face and the average number of faces per grain in the 3D structure is weaker. According to Euler's

theorem, the number of grains, faces, edges, and vertices are related as follows:

$$\chi = G - F + E - V, \tag{5.1}$$

where G is the number of grains in a system, F the number of faces, E the number of edges, and V the number of vertices or quadruple points in the system; χ is known as the Euler characteristic of the underlying space, and in our case $\chi(\mathbb{T}^3) = 0$, where \mathbb{T}^3 is the cube with periodic boundary conditions. Since each vertex is incident with exactly four edges and each edge is incident with exactly two vertices, we have E = 2V. Because every edge is adjacent to three faces, the average number of edges per face is 3E/F; because every face is adjacent to two grains, the average number of faces per grain is 2F/G. If we use $\langle E \rangle = 3E/F$ to denote the average number of edges per face the average number of faces per grain is 2F/G. If we use $\langle E \rangle = 3E/F$ to denote the average number of edges per face the average number of faces per grain is 2F/G. If we use $\langle E \rangle = 3E/F$ to denote the average number of edges per face the average number of faces per grain is 2F/G. If we use $\langle E \rangle = 3E/F$ to denote the average number of edges per face the average number of faces per grain is 2F/G. If we use $\langle E \rangle = 3E/F$ to denote the average number of edges per face and $\langle F \rangle = 2F/G$ to denote the average number of faces per grain, then the relationship between these two averages is precisely:

$$\langle E \rangle = 6 - 12/\langle F \rangle. \tag{5.2}$$

Therefore, only one of these two quantities is independent. As a short aside, we point out that despite the simple nature of this relationship, a number of papers have reported data that are clearly inconsistent with it. For example, [96] reports that $\langle F \rangle \approx 13.6$ and $\langle E \rangle \approx 5.65$, two values that cannot be simultaneously correct. This might point to some the difficulty in consistently identifying edges and faces when using volume-based methods such as Potts and cellular automaton models. In a second example, a front-tracking model [109] reports that $\langle F \rangle \approx 13.8$ and $\langle E \rangle \approx 5.01$. There seems to be no easy to reconcile this data with itself. Both examples make use of periodic boundaries in their simulations.

In any case, it is not known what this number should be in the three-dimensional steady state structure, although a few conjectures have been made. Several publications predict that $\langle F \rangle \approx$ 13.397, for which $\langle E \rangle \approx 5.104$ [66, 122, 123, 137, 116] and one predicts $\langle F \rangle \approx 13.564$, for which $\langle E \rangle \approx 5.115$ [124]. Our simulations indicate that $\langle F \rangle \approx 13.766$ and $\langle E \rangle \approx 5.128$.

Likewise, there is no known analytic function that gives the probability distributions in Figure 5.2, despite some recent attempts [138] for the 3D case. Nevertheless, the similarity in the shapes of the distributions for edges per grain in 2D and for faces per grain in 3D is quite striking, as is the difference in the distributions of edges per grain in the 2D and 3DX systems. These data make clear the difficulty of directly comparing two-dimensional simulations with cross sections of three-dimensional experimental samples. The field of *stereology* studies the sort of data that can be
extracted from one of these systems to the other [139, 140].

The distribution of faces with different numbers of edges in two-dimensional structures has been reported elsewhere. Both of [141, 58] report very similar distributions to the ones we obtain; [33] is different from our results in that they show a high number of 5- and 4- sided faces, and not very many 6-sided shapes. Experimental work reported in [142] is fairly similar to our data, though simulation data by the same authors [56] is noticeably different in that it has significantly more 6-sided faces that does our. In fact, [56] reports that 40% of all grains have 6 sides.

The distribution of faces with different numbers of edges in cross-sections of three-dimensional structures is reported in [143, 144, 145] from experimental data and in [33, 108, 109] from simulation data. All experimental results show distributions where the number of 5-sided faces outnumber the 6-sided faces, the reverse of what we report here and what is reported in [108, 109]. It is not clear how to account for this discrepancy.¹ It is possible that the inherent anisotropy of the experimental systems might impact the distribution, though it is not clear why. The distribution of faces with different numbers of edges in three dimensional systems has been reported in [33, 111]; both results are very similar to ours.

The distribution of grains with different numbers of faces in three-dimensional systems has been reported in [33, 108, 39, 82, 109, 111, 37, 93, 97, 96], all from simulation data. The one with data most similar to ours is that in [39], though most of the results are somewhat similar to ours also. A notable exception is that of [33], which exhibits a very different shape. Beautiful experimental data reported in [119] and theoretical results predicted in [146] are fairly similar to the results presented here.

Despite the similarities between Figures 5.2(a) and 5.2(b), we should point out an important difference between what they measure. In two dimensions, the local combinatorics of a particular grain can be completely described by its number of faces, because a grain with n edges is combinatorially equivalent to any other grain with n edges. However, three-dimensional grains are substantially more complicated. Many combinatorially-distinct polyhedra can share the same number of faces, as explained at length in Section 4.4. The information, then, that is captured in Figure 5.2(b) is much less descriptive of a particular three-dimensional system than Figure 5.2(a) is of a two-dimensional

one.

¹We point out a puzzling result reported in Fig. 17 of [33]. The authors there present data from a two-dimensional simulation alongside data from cross-sections of a three-dimensional simulation. In light of the results reported here and in [109], it is difficult to believe that the two systems can have such similar distributions.

Areas and volumes

Another set of data that is easy to calculate and that is most often reported is the distribution of grain areas and volumes. This has often been reported using an "approximated radius" of a grain; i.e. the square root of the area in two dimensions and the cubed root of the volume in three dimensions. Figure 5.3 shows these distributions for two- and three-dimensional systems. The



Figure 5.3: (a) Distributions of approximate grain radii of face areas in 2D, 3D, and 3DX. (b) Distributions of approximate grain radii of grain volumes in 3D. Curves predicted by Hillert are shown for both the two- and three-dimensional data. Error bars for both plots are smaller than the markers.

reason for reporting sizes in this way is mostly historical. In the mid-1960's, Hillert [147] predicted the distribution of approximated radii in two- and three-dimensional grain-growth systems. Both of Hillert's predictions are shown on the graphs in Figure 5.3. Although neither of these predictions are supported by our simulations, it is worth noting that the mode of Hillert's predicted distribution in the three-dimensional system is identical to the mode of the distribution found in these simulations.

One curious feature about the 2D in Figure 5.3(a) is the large "dip" in the middle. Before explaining this, we attempt to present the same data in a more straightforward manner. In Figure 5.4, we plot the distribution of grain areas for both two-dimensional systems; for comparison, we also plot the distribution of face areas in the three-dimensional system. In each of the three data sets, the mean is set to 1. It is worth noting that the areas measured in the the 2D and 3DX systems are of entire two-dimensional grains. In 3D, the areas measured are those of individual faces, which are only parts of large, higher-dimensional, grains.

The data in Figure 5.4 show an interesting trend that clearly distinguishes the 2D system from the other two. Notice that data from the 2D system contain "bumps" towards the left side of the



Figure 5.4: (a) Distributions of grain and face areas in the three systems. (b) Distribution of areas of grains with a particular number of edges. Error bars are smaller than the points.

plot. The presence of a similar bump in this data is also reported in [141, 109]. These bumps appear to be real phenomena, and illustrate the impact of the combinatorial structure of the grains on their geometry. To help explain this phenomenon, in Figure 5.4 we plot the distribution of areas amongst faces with a fixed number of edges in the 2D system. What is particularly interesting about this plot is that when considered individually, each of the curves appears quite smooth. However, when the curves are added together, as they are on the left side, "bumps" appear! A similar decomposition method used to illustrate the origin of irregularities in the combined data can be found in [108, 97]. It is worth noting, however, that the area distribution for a two-dimensional experimental system reported in [142] do not indicate any sort of bump that we see in our simulations; it is possible that the anisotropic conditions of the aluminum samples is the source for this discrepency.

Cross-section radii and areas of simulated three-dimensional systems have been reported in [108, 39, 148, 109, 111, 89]. Data in those papers appear consistent with that reported here. Data for faces from three-dimensional structures have not been reported previously.

The most natural way to define the size of a three-dimensional grain is using its conventional volume. Figure 5.5 shows the distribution of volumes of three-dimensional grains in the steady state. The trend-line drawn is the function e^{-x} . Although data at the very beginning and very end of the plot diverges from the trend line, the trend line appears to generally fit the data well. Perhaps this exponential distribution is somehow "preferred" by this dynamical system because it maximizes the entropy with respect to all distributions of grain sizes. Yet, it is not clear how to explain this exponential distribution and to explain why to does not arise in the distribution of grain areas in two dimensions.

Other papers that report grain volume distributions as they appear in 5.3(b), include [111],



Figure 5.5: Distribution of grain volumes, with a regular and semi-log plots. The trend-line on the right shows the function e^{-x} . Error bars for both plots are smaller than the markers.

which reports a similar exponential distribution. Distributions of approximate grain radii for threedimensional simulated systems are reported in [148, 109, 37, 96, 97]. Distributions of approximate grain radii for three-dimensional experimental systems are reported in [119]. Data in all of these appear similar to the data we report here.

Lewis's Law and generalization

One of the earliest investigations into the relationship between the topology and geometry of grains in grain-growth structures was undertaken in [65]. In that paper, Lewis pointed out that in some two-dimensional cellular structures, the area of a cell appears to be proportional to its number of edges n, when n is large; this relationship is sometimes known as *Lewis's Law*. The left side of Figure



Figure 5.6: (a) Average area of a face with with a fixed number of edges. (b) Average area of faces with a fixed number of edges. Error bars show the standard error of the mean; error bars not shown are smaller than the data points drawn.

5.6 shows the way in which the area of a cell depends on its number of edges in each of the three systems. The right side of Figure 5.6 shows the way in which the perimeter of a cell depends on its number of edges in each of the three systems. Areas and perimeters are normalized in a way that their mean is 1. These data for areas in two-dimensional systems have been reported in [149, 58], where the data are similar to ours. Although the area seems to increase with the number of edges, it can be readily seen that Lewis's Law is not obeyed by grain-growth systems, as the curve is not linear.

We also consider generalizations of this relationship to three-dimensional systems by considering the dependance of a grain's volume on its number of faces. Figure 5.7 shows the dependance of a grain's volume and surface area on its number of faces. Although there is some noise in the data for



Figure 5.7: The average volume and surface area of grains with a fixed number of faces.

grains with large numbers of faces (of which there are very few), it is difficult to conclude that there is any linear correlation between a grain's number of faces and either its surface area or volume.

Data reported in [39, 150, 89] are quite similar to results of these simulations. Similar data can also be found in [82, 111, 93], where the approximated grain radius is used instead of volume.

Edge lengths, perimeters, and surface areas

Another way in which we can describe a cell structure is by measuring boundary elements of the grains. Figure 5.8 shows the distribution of edge lengths and perimeters in the three different systems. Data are normalized so that the mean of each distributions is 1. The 2D system shows the narrowest distribution of edge lengths and perimeters, the 3DX system shows the widest, and the 3D system has a distribution that is somewhere in between. This provides a second example



Figure 5.8: (a) Distribution of edge lengths of the three systems. (b) Distribution of face perimeters. Error bars are smaller than the data points and hence not shown.

of a substantial difference between a two-dimensional cross-section of a three-dimensional structure and a genuinely two-dimensional structure. These data have not been previously reported in the literature.

When considering boundary elements, we also look at boundary elements unique to threedimensional grains. Figure 5.9 shows the distribution of surface areas of grains. Data are normalized so that the mean is 1. The shape of this distribution is noticeably different from the distribution



Figure 5.9: Distribution of surface areas of grains. Error bars are smaller than the data points, and not shown.

of grain perimeters in two dimensional systems, shown in Figure 5.8. Both sets of data describe a measure of complete grain boundaries, and yet these numbers are significantly different in two and three dimensions. Similar distributions of grain surface areas appear in [111, 89].

Roundness measures

Another way of studying a system is by looking at how "round" are its grains. In two dimensions, we can measure how close grains are to circles; in three dimensions, we can measure how close grains are to spheres, or how close faces are to circles. We define the "roundness" of a two-dimensional grain or face as the ratio of its perimeter to the square root of its area. We normalize this quantity by dividing it by $2\sqrt{\pi}$, so that the roundness of a circle is 1. The isoparametric inequality states that this is the smallest possible "roundness" of any planar shape. Figure 5.10(a) shows the distribution of faces measured by this roundness measure. In three dimensions, we define the "roundness" of



Figure 5.10: (a) Distribution of a roundness measure of faces in two- and three-dimensional systems. P is the perimeter of a grain; A is its area. These values are normalized by dividing by $2\sqrt{\pi}$, which is the "roundness" of a circle. (b) Distribution of a roundness measure of grains in three-dimensional systems. A is the surface area of a grain; V is its volume. We normalized these values by dividing by the $2\sqrt{\pi}/\sqrt[3]{4\pi/3}$, which is the roundness of a sphere. Error bars are smaller than the data points and hence not drawn.

a grain as the ratio of the square root of its surface area to the cubed root of the volume. We normalize this quantity by dividing it by $2\sqrt{\pi}/\sqrt[3]{4\pi/3}$, so that the roundness of a sphere is 1. The isoparametric inequality states that this is the smallest possible "roundness" of any simple closed surface with a given surface area. Figure 5.10(b) shows the distribution of three-dimensional grains described by this roundness measure.

The average roundness of faces in the 2D, 3DX, and 3D systems are 1.081, 1.133, and 1.149, respectively. The average roundness of three-dimensional grains is 1.073. Three-dimensional grains are then slightly "rounder" than two-dimensional ones, though comparing data from different dimensions is difficult. The faces in the genuinely two-dimensional system are considerably rounder than those in both the 3D and 3DX systems. This shows another qualitative distinction between these systems. Similar results have not been previously reported.

Mean width and related quantities

One measure that is particular important in light of the MacPherson-Srolovitz relation is the mean width of individual grains. In Figure 5.11(a), we plot the distribution of grains measured by their mean width. As the importance of this measure has not been appreciated until relatively recently,



Figure 5.11: (a) The distribution of mean widths of grains, normalized so the average is 1. Error bars are smaller than the size of the points drawn. (b) The average mean width, the average sum of triple edges around a grain, and the average sum of those quantities over all grains with a fixed number of faces. From this graph it seems that on average, grains with 15 faces maintain a constant volume with time.

these data have not been previously reported in the literature. The first thing that jumps out from this plot is its similarity to Figure 5.3(b). In fact, the two plots look almost identical. It turns out that the mean width of a grain in the steady state can be approximated by the cubed root of its volume, when both quantities are normalized so that their means are 1. This may help us formulate an approximation for the MacPherson-Solovitz relation, Equation 4.7, using a grain's volume.

To help illustrate how the mean width of a grain and the sum of its triple edges interact to determine its growth rate, we consider averages of these quantities taken over all grains with a fixed number of faces. Figure 5.11(b) shows the average mean width and the average sum of triple edges taken over all grains with a fixed number of faces. By multiplying these by appropriate constants, we weight these values so that when added together, they will determine the average growth rate of grains with a fixed number of faces. The mean width of a grain D is given by $\mathcal{L}(\mathbf{D})$; the sum of a grain's triple edges is $\mathcal{M}(\mathbf{D})$. The rate of growth of an individual grain is provided by Equation 4.7, the MacPherson-Solovitz relation: $\frac{V(\mathbf{D})}{dt} = -2\pi M\gamma \left(\mathcal{L}(\mathbf{D}) - \frac{1}{6}\mathcal{M}(\mathbf{D})\right)$.

It is easy to see that both the expected mean width and the expected sum of the triple edges of a grain increase as its number of faces increases. In a grain with few faces, the mean width dominates this equation, and the grain shrinks. In a grain with many faces, the sum of the triple edges dominates this equation and the grain grows. The two terms seem to roughly cancel when a grain has 15 faces. The average rate of change of volume of a grain as a function of its number of faces, equivalent to the middle curve, is reported in [89].

5.2 Quantities correlated over metric distance

In addition to considering the distribution of various types of grains within a system, we might also consider relationships between neighboring grains. For example, if a particular grain is very large, should we expect its neighbors to also be large? Should we expect them to be significantly smaller? Unaffected? If a grain has many neighbors, should we expect grains in its neighborhood to also have many neighbors? These sorts of relationships can tell us something about the local order inherent in various grain-growth structures. In the following two sections, we consider the relationship between various properties of a grain and other grains in its neighborhood. One standard way to do this is to look at grains that are a given *topological distance* away from a central grain. For example, we consider all grains that are adjacent to a central grain, and consider the relationship between them. In the next section, we elaborate this method and consider relationships between a grain and its neighbors of various kind. We demonstrate why the standard why of defining second nearest neighbors fails to distinguish between two different types of such neighbors, and introduce an alternate definition.

In the present section, we consider grains and their neighborhoods as defined in a metric sense. Instead of considering first, second, third neighbors and so forth, we consider all grains that are a given *metric distance* away from a central grain. This will allow us to compare the correlation between grains and their neighbors, and see how that correlation changes from one system to another.

In each steady-state system, we begin by measuring the center of mass of each grain. We then look at all grains in the system which are a certain distance away from that center of mass. As long as any point inside a grain is a fixed distance away from another grain's center of mass we can say that that grain is that fixed distance away. For each fixed distance we create a vector of pairs: in the first entry we record some property of a grain, for example its number of sides, and in the second entry we record the same property of its neighboring grain. We have such a vector for any given distance. For each distance and for each property of interest, we use this set of pairs to calculate a Pearson correlation coefficient. A positive Pearson coefficient means the values are positively correlated; a negative value indicates that they are negatively correlated. Pearson coefficient correlation values are bounded between -1 and 1, with 0 indicated no correlation between the two data sets. The graphs below show how this correlation coefficient depends on the distance from the central grain. We calculate these values over all steady-state systems and report the averages.

In the following plots, we report data collected from all three systems. In each, the unit of measurement is one approximate radius: in 2D and 3DX we use the square root of the average grain area; in three dimensions, we use the cubed root of the average grain volume. Figure 5.12(a) shows the correlation between the number of edges of a grain and that of its neighbors. Figure 5.12(b) shows the correlation between the area or volume of a grain and that of its neighbors. Figure 5.12(c) shows the correlation between the size of a grain's boundary and that of its neighbors' boundaries. In all three systems, there is a positive correlation between a grain and its immediate neighbors. In



Figure 5.12: Graphs of the Pearson correlation between some measurement of a grain and that of its neighboring grains which are a certain distance (measured in average radii) away. Figure 5.12(a) shows the correlation between the topology of a grain, as measured in its number of edges (2D, 3DX) or faces (3D), with that of its neighbors. Figure 5.12(b) shows the correlation between the area (2D, 3DX) or volume (3D) of a grain and that of its neighbors. Figure 5.12(c) shows the correlation between the perimeter (2D, 3DX) or surface area (3D) of a grain and that of its neighbors.

part, this is due to the way we calculate these numbers: a grain is always its own neighbor for small radii, so the correlation at 0 distance is always 1. Further out, the correlations begin to drop. For all systems and properties, there is a negative correlation between a property of a central grain and that property of its neighboring grains. Properties for the 2D and the 3DX systems tend be similar. For all properties, the 3D system sees the most negative correlation at intermediate distances. By the time we reach a distance of 3 or 4 radii, there is no correlation between a grain and its neighbors.

There is a strange "bump" in all three curves of the 3D data near distance 0.5. It is unclear how to explain this phenomenon. No data comparable to that reported in this section have been previously reported.

5.3 Quantities correlated over topological distance

Another way of considering the local ordering of a system involves looking at correlations between grains that are a certain *topological* distance apart. That is, we can study the correlation between a particular grain and all neighbors that share a face with that grain, or with all neighbors that are connected by a single triple edge to that grain. Those who have made similar studies in the



Figure 5.13: A central grain (red) with different types of neighbors.

past have generally limited themselves to considering only nearest neighbor grains, or two adjacent grains. However, it is clear that we can also ask about second or third nearest neighbors. What is not clear, however, is how to define second and third nearest neighbors, and so forth. Those who considered second nearest neighbors generally defined them to be grains which are not adjacent, but which have a mutual first nearest neighbor. While this might seem like a reasonable definition, it fails to distinguish between two different types of neighbors. Figure 5.13 shows a central grain colored red, surrounded by neighboring grains of various types. Neighbors that share an edge with the central grain are colored blue. Both the green and purple grains share a mutual neighbor with the central grain, and therefore both might be considered "second neighbors". However, notice that while the green grains can be connected to the central one by an edge, the purple ones grain cannot. This illustrates an important difference between two different types of grains, both of which are sometimes called second nearest neighbors.

For this reason, we consider a slightly different definition of second and third nearest neighbors. In two dimensions, first nearest neighbors are still those that are adjacent. Second nearest neighbors now are those that do not touch, but can be connected by a single edge. Third nearest neighbors are those that require two edges to connect. The green grains will then be called second nearest neighbors, while those colored purple will be called third nearest neighbors. We will see a dramatic difference between the correlation with second nearest neighbors and third nearest ones. We hope that the data convince the reader of the need for this refined definition of nearest neighbors.

In three dimensions, we can likewise define nearest neighbors in this more detailed sense. Adjacent grains are first nearest neighbors. Second nearest neighbors are those that do not touch but can be connected by a single triple edge. Third nearest neighbors are those that require two triple edges to connect.

One relationship often considered is that between the number of neighbors of a grain, and the average number of neighbors of its neighbors. This relationship was first explored in [151] and [152]; it is oftentimes referred to as the Weaire-Aboav relationship, in honor of the two authors. Figure 5.14 shows this relationship for all three systems. In the 2D and 3DX systems, we look at the expected number of edges of a grain's neighbors as a function of the its number of edges. In the 3D system, we look at the expected number of neighbors of a grain's neighbors of a grain's neighbors as a function of its number of its number of its number of faces. Looking at the three plots, we notice similar trends in the three systems. In all



Figure 5.14: The average number of edges or faces of neighbors as a function of the number of edges or faces of a central grain.

three systems, grains with few edges of faces tend to have first nearest neighbors that have many edges or faces; grains with many edges or faces tend to have first nearest neighbors with fewer edges or faces. This correlation appears stronger in the 2D system than in the 3DX system, which in turn seems stronger than that in the 3D system. This correlation appears opposite in the 2D and 3D systems. In all three systems, there appears to be very little correlation between a grain and its third nearest neighbors.

Experimental data for first nearest neighbors in three-dimensional samples are reported in [119]. The contrast between the 2D and 3D systems might indicate that the increase in dimension has a dampening affect on local structure. We also consider the relationship between a grain's area or volume and that of its neighbors. Figure 5.15 shows the expected area or volume of a grain's neighbors as a function of its area or volume. In all three systems, small grains typically have large neighbors, and large grains have



Figure 5.15: Correlation of the area or volume of a central grain with grains related by a given topological feature. Areas and volumes are normalized so their means are 1.

average size neighbors. Although most pronounced for first nearest neighbors, this appears to be the case also for second and less so for third nearest neighbors.

Last we look at measurements of grain boundaries. In the 2D and 3DX system, we consider the expected perimeter of a grain's neighbors as a function of its perimeter. In the 3D system, we consider the expected surface area of a grain's neighbors as a function of its surface area. These data are presented in Figure 5.16. These data have not been previously reported.



Figure 5.16: Correlation of the perimeter or surface area of a central grain with grains related by a given topological feature. Perimeters and surface areas are normalized so their means are 1.

In each of these examples, data for the second nearest neighbors is noticeably different from that of third nearest neighbors. These examples illustrate the importance of carefully distinguishing between the two types of neighbors.

5.4 Conclusions

For many years, universal steady-state grain-growth structures have been conjectured to exist in two and three dimensions, and much work has been done to report statistics for each of them. An accurate method for simulating two- and three-dimensional systems that evolve via curvature flow was introduced and developed in Chapters 3 and 4. In this chapter, we reported data from large simulations, which provide us with more accurate statistics about grain growth structures than previously available.

We have shown a number of qualitative differences between two-dimensional grain growth structures and cross-sections of three-dimensional ones. We have also shown important differences between steady state structures in various dimensions.

Future work might attempt to derive analytic forms of some of the distributions and other relations reported above. One particular problem that shows some hope is explaining the exponential distribution of grain volumes in three dimensions. More data are needed to verify that that is indeed the correct form, and more work is necessary to explain why that is.

Another research direction might consider how mean curvature flow controls cell structures in higher dimensions. The algorithm presented in Chapters 3 and 4 is generalizable to these higher dimensions, though it is not clear if simulating these systems will be feasible. The amount of memory and processing power increases drastically with the increase in dimension. Moreover, the topological changes that occur during four- or five-dimensional grain growth might prove to complicated to implement.

One question which we have not addressed in this thesis is the impact of the initial condition on the evolution of two- and three-dimensional systems towards the steady state. In one dimension, we saw that various initial conditions greatly affected the way in which a system evolves. Interestingly, structures closest to perfect lattices exhibited some of the most interesting behavior at the initial stages of evolution. After long times, all systems look the same. More work is left to be done in studying two- and three-dimensional systems that begin in various initial states, including perturbed lattices. Such work might provide insight about the universal nature of grain growth steady states.

Chapter 6

Conclusions and Future Work

This thesis examines cell structures that change over time, primarily through curvature flow. Chapter 1 is an informal introduction to the topics of cell structures and dynamic cell structures. There we considered the concepts of an evolving cell structure, steady state, and universal steady state, which is associated to a particular dynamic imposed on evolving cell structures.

In Chapter 2 we simulated one-dimensional cell structures that evolve under a variety of dynamics. Although a relatively simple, this sort of work does not seem to have been previously. In this chapter we provide experimental evidence of the existence of steady states and universal steady states for one-dimensional cell structures that evolve under a variety of dynamics. We report coarsening rates and long-time behavior of these systems.

This area looks promising in terms of future work. The system itself is quite simple compared with systems in higher dimensions. There is hope that some problems in this area can be answered analytically, perhaps benefitting from insight gleaned from simulation results. In this chapter we considered only deterministic dynamics which only considered very local features in determining the motion of each boundary point. In future work we might consider both allowing non-local calculations as well as stochastic components of the motions.

In Chapter 3 we consider two-dimensional cell structures that evolve through curvature flow. We develop a method for simulating such systems that satisfies the von Neumann-Mullins relation with error of order $O(\Delta t)^2$, where Δt is the time-step. This surpasses in accuracy previous methods used for modeling two-dimensional isotropic grain growth. This method allows us to model very large systems and obtain accurate statistics about the steady state.

We have considered only isotropic systems, in which the boundary mobility and surface tension

are uniform throughout the system. Real systems, however, are often anisotropic and this must be considered in modeling any real problem. Future work might focus on generalizing the work here for these cases. Another direction of research might consider two-dimensional cell structures that evolve in various other manners. For example, the evolution of dry foams through gas diffusion through cell boundaries shares much in common with coarsening in polycrystalline materials, though it is different in substantial ways. The list of events that can occur throughout the evolution of such a structure appears to be different, as is the long-time steady state such a system reaches. Future work might consider a more general framework of two-dimensional dynamical cell structures.

In Chapter 4 we consider three-dimensional cell structures that evolve through mean curvature flow. We develop a method for simulating such systems that satisfies the MacPherson-Srolovitz relation with error of order $O(\Delta t)^2$, where Δt is the time-step; it is difficult to compare this with previous models as calculations of this error have not been reported. The method also allows us to compute important geometric and topological statistics of steady state structures much larger than have been previously reported. We find that the evolution of a three-dimensional cell structures can be captured by five basic topological changes; from our simulations it appears that more complex topological transformations can be composed of combinations of these five. We also develop in this chapter a method to characterize the combinatorial structure of individual grains. This allows us to investigate more detailed questions about what sort of grains appear in structures that have been formed through a mean curvature flow coarsening process. We observed that grain growth structures tend to have more symmetric grains (in both a geometric and combinatorial sense) than cells in a certain type of Voronoi construction.

The work in this chapter has been limited in a number of important ways. First, the discretization used here is not ideal. Faces in our simulations are triangulated by placing one node in the center and many nodes around the face boundary. This is not ideal because even in the most refined version of this method, triangles will remain large in at least one linear dimension. A more ideal discretization of the surfaces involves using equiaxed triangles to represent the surface. This would allow for arbitrarily small triangles, which would allow more accurate representation of the surfaces. In particular, it would allow us to measure important quantities of interest that are unavailable to us now, or else are available with only poor accuracy. One example is the total Gaussian curvature over the faces, a quantity that will suffer from poor accuracy using the current method. The reason that this has not been implemented here has mainly been the complexity of implementing such a representation. The triangulations we use here were considerably easier to implement. Although we think the impact of this choice of discretization is small, it is not clear what differences it makes to both the evolution of the system and to its long-time steady state. In future work we hope to rewrite the code in a way that will allow this superior representation to be utilized.

Another limitation of the three-dimensional code as developed here is its restriction to the isotropic case. This idealized model of grain growth fails to account for anisotropy in the grain boundary surface tension and mobility in real systems. In real polycrystalline materials, the orientation nature of individual grains leads to surface energies and mobilities that depend strongly on the way in which two neighboring grains are aligned and "misaligned" from one another. Although this anisotropy certainly effects the local laws of grain growth, it is not clear how this affects the long-time steady state of a typical material. Future work in modeling grain growth will need to eventually accurately account for these anisotropic conditions.

Another future direction worth pursuing is connected to complicated combinatorial nature of the individual cells. In the thesis itself we described a way to completely characterize the combinatorial structure of individual cells. We might consider using this to help us understand how individual grains evolve over time. For example, is there a "typical" trajectory that a certain kind of grain can be expected to travel? Do global properties such as the relative frequencies of basic topological transitions tell us something about the likelihood of finding grains with certain combinatorial structures? What can both of these tell us about the more global structure we might expect to see in typical grain growth structures?

In Chapter 5 we compare grain growth in different dimensions, focusing primarily on two and three dimensions. We use the opportunity to report results of large steady state structures. We also report statistics of two-dimensional cross-sections of three-dimensional structures. We observe that grain volumes in three-dimensional systems appear to be exponentially distributed, something we observe in some one-dimensional systems but which we do not observe in two-dimensional systems.

Appendix A

Sketch of Mean Width

A.1 Sizes, Dimensions, and Measures

The toolbox of every scientist and engineer is full of devices for measuring the size of objects of interest. Chemists measure the distance between atoms in Å, materials scientists often measure microstructure in μ m, and astronomers measure distance in parsecs. These are each linear measures of size that differ only by a constant. Areas and volumes are usually quoted in terms of the same quantities to the power of the dimension of the object being measured.

Clearly, the choice of unit depends on the magnitude of the distance being measured, and the choice of dimensionality of the measure—be it length, area, or volume—depends on the property of interest. For example, consider a fishtank. If we want to know how much water it can hold, we care about its volume. If we want to cover the glass with a reflective coating, we care about its surface area. If we want to know whether it will fit on the bookshelf between the encyclopedias, we care about its length.

We can easily talk about measures of an object in dimensions from one up to the intrinsic dimension of the object. For example, an egg is three-dimensional and DaVinci's Mona Lisa is, nearly, two-dimensional, yet we can still talk about the two-dimensional surface area of an egg or the one-dimensional distance between the tip of the Mona Lisa's nose and her right ear. The application determines the dimension of the measurement.

In many cases, the choice of measure is obvious. For example, the most natural three-dimensional measure of a three-dimensional object is probably volume. A natural one dimensional measure of a two dimensional object might be its perimeter. Sometimes, though, the choice of measure is not obvious. For example, what is a good one-dimensional measure of a three-dimensional object? Imagine measuring a pencil. If you are designing a high-end pocket protector, you care about the distance from eraser to tip. Yet, if you are designing a pencil sharpener, you probably care more about the pencil's diameter. Both are one-dimensional measures of a three-dimensional object, yet neither, generally speaking, seems more natural or intuitive.

In this chapter, we consider the question of whether some measures are intrinsically better than others. In particular, we show that like volume and surface area, there exists a natural choice for a one-dimensional measure of three-dimensional objects. Surprisingly, most scientists and engineers never use this measure, and as a result, one-dimensional measures of higher dimensional objects have proliferated. To see this, consider how many different ways biologists measure cell sizes or materials scientists measure grain sizes. Clearly, size does matter and how we measure it matters too.

A.2 Steiner's Formula

To see that not all measures of a given dimension are created equal, we consider an interesting mathematical relationship found by the Swiss geometer Jakob Steiner in the 1840's. Steiner considered convex objects growing at constant rates (more specifically, each point on the surface moves normal to the surface at a constant velocity, v). For example, picture an expanding chicken egg or a growing ice cube. How does the volume of the egg or the ice-cube evolve in time? Steiner proved that

$$V(t) = C_0 V(0) + C_1 S(0)(vt) + C_2 \mathcal{L}(0)(vt)^2 + C_3 (vt)^3,$$
(A.1)

where V(0) is the volume of an object at t = 0, S(0) is half of its surface area, and $\mathcal{L}(0)$ is a mysterious linear measure of the object called the *mean width*; the C_n are constants that can be easily determined as shown below [153]. Steiner's formula is exact, beautiful and remarkably simple. It tells us that at any time the volume depends only on vt and three measurements of the initial shape — the volume, surface area, and a linear measure \mathcal{L} we have not yet defined.

We now focus on two simple applications of Steiner's formula. Consider a sphere¹ of radius $r = r_0 + vt$ for all $t \ge 0$. The volume of a sphere is given by $\frac{4}{3}\pi r^3$, so we have:

$$V(t) = \frac{4}{3}\pi(r_0 + vt)^3 = \frac{4}{3}\pi r_0^3 + 4\pi r_0^2(vt) + 4\pi r_0(vt)^2 + \frac{4}{3}\pi(vt)^3$$
(A.2)

Knowing that $V(0) = \frac{4}{3}\pi r_0^3$ and $S(0) = \frac{1}{2}4\pi r_0^2 = 2\pi r_0^2$, we can compare Equation (A.1) with

 $^{^1\}mathrm{In}$ this chapter we use the term sphere to refer to the sphere and its interior.

Equation (A.2) to determine that $C_0 = 1$, $C_1 = 2$, and $C_3 = \frac{4}{3}\pi$. Because we have not yet defined \mathcal{L} , we cannot yet determine C_2 . We leave this until the next section.



Figure A.1: A growing cube for vt = 0, 0.25, 1, 2, and 5 in arbitrary units.

Now, consider a growing cube with initial edge length a, as shown in Figure (A.1) at various times. We can see in Figure (A.2) that the growing cube consists of an initial cube plus 6 parallelepipeds (yellow), 12 quarter cylinders (blue), and 8 eighth spheres (red). Therefore,

$$V(t) = a^{3} + (6)a^{2}(vt) + (12)a\pi(vt)^{2}/4 + (8)\frac{4}{3}\pi(vt)^{3}/8$$

$$= a^{3} + 6a^{2}(vt) + 3a\pi(vt)^{2} + \frac{4}{3}\pi(vt)^{3}.$$
 (A.3)

Because $V(0) = a^3$ and $S(0) = \frac{1}{2}6a^2 = 3a^2$, we can compare (A.3) with Steiner's formula (A.1) to



Figure A.2: Dissected growing cube.

see that $C_0 = 1$, $C_1 = 2$, and $C_3 = \frac{4}{3}\pi$. Remarkably, the C_n are the same for the growing cube as they were for the sphere! What is so beautiful about Steiner's formula is that the constants are just that — they are the same for all convex shapes. Note that we can still not determine C_2 because the mean width \mathcal{L} has yet to be defined. However, we can already begin to see that the mean width is as fundamental a measure of three-dimensional objects as volume and surface area. In the following section, we explore the mean width measure \mathcal{L} and then return to Steiner's formula.

A.3 Measures and their Properties

In the previous section, we saw the utility of a special one-dimensional measure of three-dimensional objects called the *mean width*. In this section we rigorously define this measure and discuss a few of its interesting properties. With this we will be able to calculate, among other things, the constant C_2 in Steiner's formula which we were unable to determine above.

By means of introduction, a brief discussion of measures is in order. A measure, roughly speaking, is a way of assigning a nonnegative number, or size, to an object. Among other things, defining a measure allows us to say that one object is bigger than another. Measures often have special rules that allow us to measure the combination of two objects in terms of the measures of the individual pieces. For example, we might have a piece of string that is 5 inches long and a second piece that is 8 inches long. If we splice them together, then their total length would be 5 + 8 = 13 inches minus any overlap of the two pieces. More generally, a measure *m* is considered to be additive if it satisfies $m(A \cup B) = m(A) + m(B) - m(A \cap B)$, where *A* and *B* are two objects being measured, $m(A \cup B)$ is their union, and $m(A \cap B)$ their overlap, or intersection. This is *additivity*.

Many measures that are familiar to us are additive. For example, consider the volumes V(A)and V(B) of two three-dimensional objects A and B. If volume is additive, then the volume of the union of A and B is $V(A \cup B) = V(A) + V(B) - V(A \cap B)$. Figure A.3 shows two spheres with radii 4 (red) and 6 (yellow) separated by a distance of 5; the volume of their intersection is $\frac{565}{12}\pi$. Because volume is additive, we can calculate the volume of the union of the two spheres: $\frac{4}{3}\pi 4^3 + \frac{4}{3}\pi 6^3 - \frac{565}{12}\pi = \frac{1305}{4}\pi$.



Figure A.3: Two intersecting spheres.

Another example of an additive measure is surface area, S. The surface area of the union of A and B is $S(A \cup B) = S(A) + S(B) - S(A \cap B)$. Returning to Figure A.3 and knowing that the surface area of the intersection is 46π , we can calculate the surface area of the union as $4\pi 4^2 + 4\pi 6^2 - 46\pi = 162\pi$. Although volume and surface area are additive, not all measures are so well behaved. For example, consider a two-dimensional measure $V^{2/3}$, where V is the conventional volume of the object. Using again the spheres as our objects (Figure A.3), we immediately see that $(\frac{1305}{4}\pi)^{2/3} \neq$ $(\frac{256}{3}\pi)^{2/3} + (\frac{864}{3}\pi)^{2/3} - (\frac{565}{12}\pi)^{2/3}$. Likewise, we might propose a one-dimensional 'length' measure of three-dimensional objects, $l = \sqrt[3]{V}$. Again, using our sphere example, it is easy to see that $l(A \cup B) \neq l(A) + l(B) - l(A \cap B)$. And so although we have additive 2- and three-dimensional measures of three-dimensional objects (surface area and volume), we have yet to identify an additive one-dimensional measure. We will later see that the mean width is such a measure.

Two more desirable properties of measures are rotational and translational invariance. Translating or rotating the union of spheres in Figure A.3 does not change its volume. While this may seem trivial, many measures do not possess this type of invariance. To illustrate this point, consider w(A)= maximal width of A along the x-axis. If the union of spheres in Figure A.3 was rotated about an axis coming out of the page by 90°, this measure $w(A \cup B)$ changes from 8 to 6. Clearly, $w(\cdot)$ is not a rotationally invariant measure.

One last useful property of a measure is *scalability*. If we enlarge an object by some factor c, we would like to see the measure of that object increase proportionally. For example, if we double the edge lengths of a cube, then its length increases by a factor of 2^1 , its surface area by a factor of 2^2 , and its volume by a factor of 2^3 . In the more general case, the standard measures (length, area, and volume) are each multiplied by a factor of c^d , where c is the uniform scaling factor and d is the dimensionality of the measure.

A.4 Integrating Mean Curvature

We suggested above that the measure \mathcal{L} , known as the *mean width*, is a natural one-dimensional measure of three-dimensional objects. We first consider a definition of \mathcal{L} as an integral of the mean curvature H over the surface of an object D:

$$\mathcal{L}(D) = \frac{1}{2\pi} \int_{\partial D} H dA.$$
(A.4)

where $H = k_1 + k_2$ is the mean curvature, k_1 and k_2 are the principal curvatures (the minimal and maximal curvatures of all curves on the surface passing through a point), ∂D is the surface of Dand dA is an element of area of the surface.² Since curvature measures how "curved" a surface is

²In some books the mean curvature is given by $\frac{1}{2}$ of the *H* we defined here.

at a point, this integral tells you about how "curved" a surface is overall. It can be shown that this measure is both additive and scalable (for the sake of simplicity and brevity we omit the proofs of these statements). The mean width is also rotationally and translationally invariant. Hence, the mean width is a *good* measure. In the 1950's, the Swiss mathematician Hugo Hadwiger proved that in fact, up to a constant multiple, the mean width is the *only* such measure [154].

Using this definition we can easily calculate the mean width of a sphere and a cube, and determine the constant C_2 . First we consider the mean width of a sphere S of radius r. The surface of a sphere of radius r has uniform mean curvature H = 1/r + 1/r = 2/r. Integrating over the surface we get,

$$\mathcal{L}(S) = \frac{1}{2\pi} \int_{\partial S} \frac{2}{r} dA = \frac{1}{2\pi} 4\pi r^2 \frac{2}{r} = 4r.$$
(A.5)

Since we now know that the mean width of the sphere of radius r is 4r, comparing Equations (A.1) and (A.2) shows us that $C_2 = \pi$.

We now return to the cube and calculate its mean width by integrating the mean curvature over its surface. For reasons which will become clear later, we first focus on the "growing cube" of Figure (A.2). The growing cube has 3 types of surfaces elements: 6 flat faces, each with surface area a^2 and constant mean curvature H = 0; 12 quarter-cylinder edges, each with surface area $\frac{\pi(vt)a}{2}$ and constant mean curvature H = 1/(vt); and 8 eighth-sphere corners, each with surface area $\frac{\pi(vt)^2}{2}$ and constant mean curvature H = 2/(vt). Hence,

$$\mathcal{L}(\text{growing cube}) = \frac{1}{2\pi} \left[\int_{faces} H dA + \int_{edges} H dA + \int_{corners} H dA \right]$$
(A.6)

$$= \frac{1}{2\pi} \left[6(a^2)0 + 12\left(\frac{\pi(vt)a}{2}\right) \frac{1}{(vt)} + 8\left(\frac{\pi(vt)^2}{2}\right) \frac{2}{(vt)} \right]$$
(A.7)

$$= 0 + 3a + 4(vt).$$
 (A.8)

At t = 0, the growing cube of Figure (A.2) reduces to a standard cube, and so:

$$\mathcal{L}(cube) = 3a. \tag{A.9}$$

Using this value of \mathcal{L} for the cube in Steiner's equation, Equation A.1, and comparison of the result with Equation (A.3) again shows that $C_2 = \pi$. Indeed, C_2 , like the rest of the C_n , is the same for every convex shape!

A.5 Integrating Euler Widths

While the above definition allows us to calculate \mathcal{L} for a number of simple shapes, it is not general enough to apply to one- and two- dimensional objects. A more general definition of the mean width can be obtained by integrating the Euler characteristic χ rather than the mean curvature. We present here a brief description of the Euler characteristic.

The Euler characteristic is a 0-dimensional measure which represents a particular type of counting. For polygons and for the surfaces of polyhedra, the Euler characteristic is defined as follows:

$$\chi = V - E + F,\tag{A.10}$$

where V, E, and F are the number of vertices, edges, and faces, respectively. As a simple example, a triangle in the plane has 3 vertices, 3 edges, and 1 face and an Euler characteristic $\chi = 3-3+1=1$; a rectangle has 4 vertices, 4 edges, and 1 face, and therefore and Euler characteristic $\chi = 4-4+1=1$. In fact, any solid polygon in the plane has $\chi = 1$. The surface of a cube has 8 vertices, 12 edges, and 6 faces and $\chi = 8 - 12 + 6 = 2$. Interestingly, any polyhedral surface homeomorphic³ to the sphere will have $\chi = 2$. In the plane, χ is equal to the number of connected objects minus the number of holes in those objects (note, χ need not be positive).



Figure A.4: Figures with different numbers of objects and holes.

A few illustrations (see Figure A.4) might help explain the Euler characteristic of objects em-

 $^{^{3}}$ Loosely speaking, two objects are homeomorphic if one can be deformed into the other without opening new holes or closing existing ones.

bedded in the plane. Figure (A.4a) shows only one object with no holes, so $\chi = 1$; Figure (A.4b) shows three objects, so $\chi = 3$; Figure (A.4c) shows a single object containing one hole, and so $\chi = 1 - 1 = 0$; and Figure (A.4d) has two objects and five holes, and so $\chi = 2 - 5 = -3$.

Using the Euler characteristic, we now define a one-dimensional measure of three-dimensional objects in the following manner. Take an object D and a line ℓ , and consider all cross-sections of D perpendicular to ℓ . Figure (A.5) shows an example with a few such cross-sections. Now, taking the Euler characteristic of each cross-section and integrating over *all* planes perpendicular to ℓ gives us the *Euler width* of D in the direction ℓ :

$$E_{\ell}(D) = \int_{\ell} \chi(D \cap P_{\ell}) d\ell, \qquad (A.11)$$

where a cross-section of D is the intersection of the three-dimensional object D with a plane P_{ℓ} perpendicular to ℓ , i.e. $D \cap P_{\ell}$.



Figure A.5: Torus with cross-section cuts.

As a simple example, consider a sphere of radius r. It is easy to see that the Euler width in *any* direction is just 2r. Slightly more complicated is the Euler width of the torus T of Figure (A.5) in the direction ℓ_1 . The radius of the tube is r and the inner radius of the torus is R. Every plane perpendicular to ℓ_1 and intersecting the torus has either $\chi = 1$ or $\chi = 2$; planes that do not intersect

the torus at all have $\chi = 0$. Figure (A.6) illustrates this for various regions of ℓ_1 . We then have:



$$E_{\ell}(T) = \int_{\ell} \chi(T \cap P_{\ell}) d\ell = (\chi = 1)2r + (\chi = 2)2R + (\chi = 1)2r = 4(R+r)$$

Figure A.6: Torus with values of χ .

Next, we consider the Euler width of the same torus but in the direction pointed directly through the hole of the ring. Every plane perpendicular to this line either fails to intersect the torus, in which case $\chi = 0$, or else intersects it to form an annulus, or a circle with a hole in its center, in which case again $\chi = 0$. Therefore, the Euler width in this direction is exactly 0. It should be clear that the value of the Euler width generally depends on the direction ℓ .

We now reintroduce the mean width in terms of the Euler width, without reference to the mean curvature. The mean width is twice the average Euler width over all lines ℓ passing through the origin; the particular choice of origin is immaterial as the mean width is invariant upon translation of the object or the origin. The mean width is then the mean *Euler* width.

In the case of a sphere of radius r, we saw above that the Euler width in every direction is 2r. Considering all lines through the origin (which is most conveniently chosen to be the center of the sphere), it is obvious that the average is 2r, and so $\mathcal{L} = 4r$, consistent with our earlier calculations.

One advantage of the definition based on the Euler characteristic over that based on the mean curvature is that it can be directly applied to objects that do not have smooth two-dimensional surfaces. As a simple example, consider a straight line segment of length L embedded in a threedimensional space and choose the origin to lie along the line segment. The Euler width in the direction of the line segment will be L, the Euler width in a direction perpendicular to the line segment will be 0, and the Euler width in all other directions ℓ will be $0 < E_{\ell} < L$. The average Euler width turns out to be L/2 and so the mean width \mathcal{L} , which is defined as twice the average Euler width, turns out to be exactly L. By this definition, it turns out that the mean width of any space curve with length L is always exactly L.

A.6 Numerical Evaluation of the Mean Width

The analysis above provides methods for the exact, analytic determination of the mean with. Unfortunately, the analytical determination of the mean width is rather difficult, except for a relatively small set of highly symmetric objects such as those considered here. Fortunately, there exists a relatively straightforward way to numerically approximate the mean width of an arbitrary shape provided that a piecewise linear discretization of the shape is available.

Consider the torus of Figure A.7. Calculating the Euler width in arbitrary directions is rather tricky even for such a relatively simple shape. However, numerically calculating the mean width of a discretized torus, as shown in Figure A.8, is relatively straightforward. Using analysis similar to what we used to calculate the mean width of the growing cube (Figure A.2), we can easily divide the integral into sums over the faces, edges, and vertices of this object. Because the triangles are flat, their mean curvature is 0 everywhere. All curvature of the object is concentrated in the edges. The integral of the mean curvature along each edges is $l_i\alpha_i$, where l_i is the length of edge *i* and α is the turning angle there between the two adjacent edges. This leaves us with:

$$\mathcal{L}(A) = \sum_{i}^{n} e_i \alpha_i \tag{A.12}$$

for a discretizated surface. As the discretization triangles become smaller, the approximating surface converges to the object, and the calculated mean width converges to exact mean width. In Chapter 4, we use Equation A.12 to calculate the mean width of grains with triangulated surfaces.



Figure A.7: Plain torus.



Figure A.8: Discretized torus.

Appendix B

Simulation Details for 2D

In this appendix we add some details two-dimensional simulation method. The details provided here describe the method used to obtain results reported in Chapters 3 and 5.

Brakke method

The Surface Evolver program [48] developed and maintained by Ken Brakke is a powerful collection of tools to study minimal surfaces in arbitrary dimension. As such, it is often used to model grain growth in two [155, 50, 49] and three dimensions [111, 156]. In Chapter 3 we compared the numerical error involved in using Brakke's method with that involved in using the method proposed in this thesis. We briefly describe the method used by Brakke in Surface Evolver.

The two options effective area and area normalization are used in order to most accurately simulate motion by curvature flow. We describe the motion of boundary nodes for this setting. This will allow us to calculate the way in which the radius of a regular polygon changes in one time step using this method, a result critical in Section 3.5 for calculating the relative and absolute errors involved in this method.

We consider the motion of boundary node σ in this setting. We begin by considering two unit vectors $\mathbf{e_1}$ and $\mathbf{e_2}$ pointing away from σ in the direction of the two adjacent edges. Without any normalization, node σ will move with a velocity $\mathbf{e_1} + \mathbf{e_2}$. However, in this setting there exists a resistance to this motion from the adjacent edges. We therefore scale this vector by dividing it by half of some measurement of the adjacent edges. Brakke points out that the resisting motion comes from the component of the edges that are normal to the direction of the motion. Therefore, if $p = ||\mathbf{p}_n(edges)||$ is the length of the projection of the adjacent edges onto the normal direction of $\mathbf{e_1} + \mathbf{e_2}$, then we scale the $\mathbf{e_1} + \mathbf{e_2}$ by 1/p.



Figure B.1: Regular polygonal grain with m sides and of radius r.

In the case of a regular polygon with m sides and radius r, as illustrated in Figure B.1, $\mathbf{e_1} + \mathbf{e_2}$ always points in the direction of the center of the polygon. Since each of $\mathbf{e_1}$ and $\mathbf{e_2}$ are unit vectors, we have $||\mathbf{e_1} + \mathbf{e_2}|| = 2\sin(\pi/m)$. The length of the projection of two adjacent edges onto direction normal to $\mathbf{e_1} + \mathbf{e_2}$ is then $2\cos(\pi/m)\sin(\pi/m)$. Therefore, we are left with a vector pointing in the direction of the center of the polygon and with a magnitude $1/(r\cos(\pi/m))$. If we take into account the mobility of the grain boundary M and the surface tension γ , then after one time step of size Δt , the radius changes:

$$\Delta r = -\frac{M\gamma\Delta t}{r\cos(\frac{\pi}{m})} \tag{B.1}$$

for a regular polygon with n sides and of radius r. This equation is used in Section 3.5.

Further information about this program and the algorithms used can be found in [48].

Proposed method

Data structure

We use two basic data structures, one representing a node and one representing a grain. Edges are represented implicitly through these two other data structures. We provide here a pseudo-code representation of a data structure representing a node. The id number, as well as the coordinates

```
class CNode {
   int id;
   double x, y;
   double dx, dy;
   int valence; // classifies all nodes by valence, either 2 or 3
   CGrain* neighbors[3]; // pointers to all neighboring grains
}
```

x, y and velocities dx, dy are straightforward. The valence variable stores how many neighbors (grains or nodes) a particular node has; this will provide information about whether a node is a vertex node or a boundary node. The CGrain vector stores references to all three neighboring grains. The data structure for each grain is somewhat similar.

```
class CGrain {
   int id;
   vector <CNode*> bnodes;
}
```

For each grain with a given id, we store only an ordered list of (references to) CNodes. This information is enough to calculate all properties of interest of a given grain. As a simple example, we can calculate how many triple junctions lay on the boundary of a grain by counting the number of its associated CNodes that have valence 3.

Time step, minimum and maximum edge length

On each time step, we set the size of the longest l_{max} and shortest l_{min} allowable edges in the system. If N is the number of grains in the system, then:

$$l_{max} = \frac{1}{5\sqrt{N}} \tag{B.2}$$

and

$$l_{min} = \frac{1}{100\sqrt{N}}.\tag{B.3}$$

We set the step size Δt based on this:

$$\Delta t = l_{min}^2 / 20. \tag{B.4}$$

On every step, if a grain has an area smaller than $2\pi l_{min}^2$ then we erase it. If an edge grows to be longer than l_{max} , then we refine it on the following step, placing an extra boundary node at its

midpoint. If an edge shrinks to be shorter than l_{min} then we remove one of its nodes, and move the other node to the midpoint of that edge segment.

Node motions

We begin by explaining how we move individual boundary nodes. Figure B.2 illustrates a typical boundary node σ , adjacent to two edges $\mathbf{e_1}$ and $\mathbf{e_2}$. Here we consider the edges as edges and also as



Figure B.2: A typical boundary node σ , adjacent to two edges $\mathbf{e_1}$ and $\mathbf{e_2}$.

vectors, pointing away from the boundary node σ . The area of the shaded parallelogram is $||\mathbf{e_1} \times \mathbf{e_2}||$. For reasons of numerical stability, we choose to move boundary nodes in the direction of $\mathbf{e_1} + \mathbf{e_2}$. If we moved node σ by exactly $\mathbf{e_1} + \mathbf{e_2}$, then we would change the areas of the two adjacent grains by $\pm ||\mathbf{e_1} \times \mathbf{e_2}||$. However, we only want to change the area of the adjacent grains by $\pm \alpha \Delta t$, as described in Chapter 3. Therefore, in one time step of size Δt , we move σ with a displacement vector:

$$\Delta \sigma = \frac{\alpha \Delta t}{||\mathbf{e_1} \times \mathbf{e_2}||} \left(\mathbf{e_1} + \mathbf{e_2}\right). \tag{B.5}$$

Determining the motion of a triple point is very similar. Figure B.3 shows a triple point τ and three neighboring edges and grains. We need to move the triple node τ in such a way that changes



Figure B.3: A triple-node τ where Grains 1, 2, and 3 meet, where \mathbf{e}_i are the vectors from the triple-node to the nearest nodes on each of the three boundaries, and α_i are the "turning angles", or exterior angles with respect to each of the three grains.

the areas of each grain by an amount $\alpha_i - \pi/3$, where α_i is the turning angle of that grain at that triple point. We can make similar calculations to those we made for boundary nodes. We are then left with a system of three equations in two unknown spatial variables:

$$\begin{bmatrix} \mathbf{e_1} - \mathbf{e_2} \\ \mathbf{e_2} - \mathbf{e_3} \\ \mathbf{e_3} - \mathbf{e_1} \end{bmatrix} \mathbf{dv} = 2M\gamma\Delta t \begin{bmatrix} \alpha_1 - \frac{\pi}{3} \\ \alpha_2 - \frac{\pi}{3} \\ \alpha_3 - \frac{\pi}{3} \end{bmatrix}$$
(B.6)

This system is *not* overdetermined because only two equations are independent. This leaves us then:

$$\Delta \tau = 2M\gamma \Delta t \begin{bmatrix} \mathbf{e_1} - \mathbf{e_2} \\ \mathbf{e_3} - \mathbf{e_1} \end{bmatrix}^{-1} \begin{bmatrix} \alpha_1 - \frac{\pi}{3} \\ \alpha_3 - \frac{\pi}{3} \end{bmatrix}$$
(B.7)

Removing small faces

When the area of a face is less than $2\pi l_{min}^2$, then we delete it. Most of the time, this face has only two or three sides. If it has more than three sides, then we remove sides one at a time until only three are left. When there are three sides left to a face, then we remove it by collapsing its three corners to one triple node. This topological change is illustrated in Figure B.4.



Figure B.4: The collapse of a three-sided face to a triple point. First we remove all edge nodes; next, a new point is placed at the geometric average of the three triple-points. Other neighboring boundary nodes are left in place.

Removing a two-sided face is done in a very similar manner. This topological change is illustrated in Figure B.5.



Figure B.5: The collapse of a two-sided face to an edge. We begin by removing the edge nodes and then remove two triple nodes, leaving the other neighboring nodes in place.

Appendix C

Simulation Details for 3D

In this appendix we provide additional detail of the data structure and algorithms used for evolving three dimensional cell structures through mean curvature flow. It is meant to supplement the details provided in Chapter 4 of the thesis.

Data structure

We employed two basic data structures, one representing a node in the system, and one representing an entire grain. Edges and faces are represented only implicitly through these other data structures. We provide here a pseudo-code representation of a data structure representing a node.

```
class CNode {
   int id;
   double x, y, z;
   double dx, dy, dz;
   vector <CGrain*> neighbors;
```

vector <CNode*> corners[4];

}

The id number, as well as the coordinates x, y, z and velocities dx, dy, dz are straightforward. The neighbors vector stores references for all neighboring grains; there may be either 2, 3, or 4 neighboring grains depending on the type of node. The array of corners vectors store all neighboring nodes in a given grain. These vectors are ordered so that the neighbors are a fan of sorts in a particular grain. The data structure for each grain is somewhat similar.

```
class CGrain {
public:
    int id;
    vector <triplet> triplets;
}
```

For each grain with a given id, we store only a list of triplets. Each triplet is merely an ordered set of CNode references. In particular, the first and third references are always to boundary nodes, and the second one is always a face node. Triangles sharing a face node are grouped together.

Local quantities and node motions

In Section 4.3.3 we described how every face, edge, and vertex node should be moved in order to satisfy the MacPherson-Srolovitz relation. Here we provide more detail to explain how the local curvature and mean widths should be calculated. We begin by calculating a local version of the mean width for each node.

Each node is connected by edges to a set of neighboring nodes. Figure C.1 shows a typical face with a center node and boundary nodes around the perimeter of the face. For every node in the



Figure C.1: A typical face with a few edges drawn between the face node and the boundary nodes.

system, for each of its neighboring bodies, we can assemble an ordered list of neighboring nodes. Together, these nodes allow us to study the local surface of each neighboring grain around each particular node. Although Figure C.1 shows the local surface associated with a face node, a similar picture can be drawn for edges nodes and vertex nodes.

The unit normal of any triangular face element can then be calculated from the edges $\mathbf{e_i}$. If a triangular face is bordered by two ordered edges $\mathbf{e_i}$ and $\mathbf{e_j}$, then the unit normal is $(\mathbf{e_i} \times \mathbf{e_j})/|\mathbf{e_i}||\mathbf{e_j}|$. If we label these normal vectors $\mathbf{n_i}$, then turning angles between triangular face elements can be determined by calculating the angle between their unit normals. If two adjacent triangular face

elements have unit normals \mathbf{n}_i and \mathbf{n}_j , then the angle α between them is $\arccos(\mathbf{n}_i \cdot \mathbf{n}_j)$.

The local mean curvature \mathcal{L}_n associated with any node n in a system is given by Equation 4.11:

$$\mathcal{L}_n(\mathbf{D}) = \frac{1}{4\pi} \sum_i e_i \alpha_i,\tag{4.11}$$

Where e_i is the length of an edge *i* and α_i is the oriented angle between the two triangular face elements that meet along that edge. This can be easily calculated using our data structure for any node in the system and with respect to any adjacent grain. The $\frac{1}{4\pi}$ comes from the following. The mean curvature of a surface is the integral of the mean curvature over the surface divided by 2π . For a triangulated surface, we can compute this by taking the sum of $e_i\alpha_i$ over all edges. Because each edge is associated with two adjacent nodes, only one half of this value should be associated with each of the two adjacent nodes. We are thus left with the $\frac{1}{4\pi}$ factor. This same procedure can be used to calculate the localized mean curvature at a vertex node, such as that illustrated in Figure C.2.



Figure C.2: A typical vertex node with a few edges drawn between the vertex node and adjacent nodes.

Calculating the local triple edge length \mathcal{M}_n is very straightforward. For every node, we sum the lengths of all adjacent triple edges and divide by two. We divide by two because each triple edge is associated with two adjacent nodes. The local triple edge length is zero for all face nodes because face nodes are always adjacent to zero triple edges. Vertex nodes in our model are always adjacent to exactly three triple edges; edge nodes are always adjacent to exactly two triple edges. If e_j are the lengths of all triple edges adjacent to a node n, then the local triple edge length is:

$$\mathcal{M}_n(\mathbf{D}) = \frac{1}{2} \sum_j e_j. \tag{4.12}$$
Topological changes

All topological changes involve changes to the geometry of the cell structure as well changes to the topology of the neighboring nodes and cells. We provide here details about the topological changes the occur during grain growth in the code that we use, specifically explaining where we place nodes after topological changes.

In what follows, we use capital letters to represent face, edge, and vertex nodes. We treat nodes as points in space and as vectors from the origin, the meaning being made clear from context. We denote the translation of a point or vector A by a vector B as A+B. Likewise, we denote the scaling of a vector A by a scalar c as cA. We use ||A|| to denote the Euclidean norm of a vector A. Last, we use two conventions to simplify the presentation of geometrical algorithms. We use AB = B - A as a shorthand for a vector that begins at A and ends at B; we use $A \cdot B = (A+B)/2$ to denote either the midpoint between points A and B, or else the average of the two vectors A and B. We write $AB \cdot CD$ as shorthand for $(AB) \cdot (CD)$. Also, because every face contains at most one face node, we occasionally refer to a face using its face node. For example, "a face X" refers to the unique face that contains the face node X.

Three-dimensional cell structures in this thesis are always embedded in the torus $\mathbb{T}^3 = S^1 \times S^1 \times S^1$. In practice, the x, y, and z coordinates of all points lie in the unit cube $[0, 1]^3$, where periodic boundary conditions are implemented by identifying $\{x, y, 0\}$ with $\{x, y, 1\}$ for all x, y, and so forth. Operations on nodes and vectors are then fixed appropriately.

Removing a triple edge

When removing an edge, we need to remove two quadruple points and replace the shrinking edge with a new triangular face. We explain here where we place the nodes associated with the new face.

Figure C.3 shows an edge between vertex nodes A and B that is shrinking and that will be replaced by a new triangular face. We begin by creating a new node F_0 and placing it at $A \cdot B$ (the midpoint of nodes A and B). We create three new vertex nodes for the triangle, F_1 , F_2 , and F_3 and place them as follows:

$$F_{i} = F_{0} + \frac{||AB||}{||AU_{i} \cdot BD_{i}||} (AU_{i} \cdot BD_{i}),$$
(C.1)

where $i \in \{1, 2, 3\}$. At this point, the length of each edge between F_0 and F_i has the same length as the initial edge being removed. To make the structure more numerically stable, we then move F_0 to the midpoint of the three face nodes, $F_0 = (F_1 + F_2 + F_3)/3$. No new edge nodes are added.



Figure C.3: A shrinking triple edge is removed and replaced by a triangular face.

As explained in Section 4.3.5, after an edge is removed, the combinatorial information for all affected data structures is reconfigured. Moreover, the volumes of all adjacent grains are recalculated, as are the velocities of all adjacent nodes.

Removing a triangular face

When a face is shrinking too quickly, as described in Section 4.3.5, we remove it. If the face has more than three sides, then we remove its shortest edge using the procedure described above, and repeat this procedure until no more than three edges are left. We now describe how three- and two-sided faces are removed.

Removing a three-sided face begins by removing all edge nodes from the face, so we are left with only one face node and three vertex nodes. In Figure C.4 we consider a face F_0 (i.e. the unique face with face node F_0) with vertex nodes F_1 , F_2 , and F_3 . In removing this face we create two new



Figure C.4: A shrinking triangular face is removed and replaced by a triple edge.

vertex nodes, A and B. We use p to denote the perimeter of the triangle to be erased, after all edge

nodes have been removed. We then place the two new vertex nodes at the following points:

$$A = F_0 + \frac{p}{10} \sum_{i=1}^{3} \frac{F_0 U_i}{||F_0 U_i||}$$
(C.2)

and

$$B = F_0 + \frac{p}{10} \sum_{i=1}^{3} \frac{F_0 D_i}{||F_0 D_i||}$$
(C.3)

The length of the new edge is then roughly twice that of a side of the deleted triangle. According to this rule, each triangle will be replaced by an edge whose linear size is proportional to the triangle. This seemed like a wiser choice than choosing a more global constant for the new edge length.

After removing a triangular face, the combinatorial information for all affected data structures is reconfigured. Moreover, the volumes of all adjacent grains are recalculated, as are the velocities of all adjacent nodes.

Removing a two-sided face

Removing a two-sided face was the most challenging topological change implemented in our code. We begin with a two-sided face F_0 , illustrated on the left side of Figure C.5. This two-sided face is the intersection of two grains, one above the face and one below it (as illustrated in the figure). The face is also adjacent to two other grains, one in front of it and one behind it. The grains in front and behind the two-sided face meet at two other faces, N_1 and N_2 .



Figure C.5: The disappearance of a face with two sides, shaded dark gray. Two other faces, with face nodes N_1 and N_2 , merge together during this topological change. Thick lines indicate triple edges; thin lines indicate edges between a face node and an edge node.

We begin by removing all edge nodes that are incident with face F_0 . Next we remove face F_0 from the grains above and below the face. Figure C.6 shows the removal of face F_0 from the perspective of the grain above the face and the grain below it.

Next we create a new face node N and attach it to all nodes that had been adjacent to faces N_1



Figure C.6: All edge nodes are removed from face F_0 , and then the face itself is removed; this is shown from the perspective of the grains above and below. Triple edges are shown in solid lines; face edges are shown in dashed lines.

and N_2 ; this can be seen in Figure C.5. We place face node N at the center of mass of the boundaries of faces N_1 and N_2 . At this point, the new face N might have a very unnatural geometry. In general most faces in grain growth structures are quite round. Figure C.7(a) shows a typical face from a



Figure C.7: (a) A typical face in a grain-growth structure; angles between the normals of adjacent triangular facets are very small. (b) A face that results from the merging of two faces such as N_1 and N_2 ; angles between normals of adjacent triangular facets can be very large.

grain-growth structure. The face is relatively round, and the normals between neighboring triangular facets are small. Figure C.7(b) shows a face that results from merging two faces such as N_1 and N_2 . Here, the geometry of the face is quite rough, and angles between adjacent facets can be quite large. This creates numerical instability. To resolve this problem, we smooth out the boundaries of this face. Figure C.8 shows a sketch of what face N might look like after smoothing out the boundary. In particular, we move each node around the boundary to the center of its two neighboring edge nodes. We iterate this process three times, at which point the triple lines around this face are quite straight. The local structure is then substantially more stable.



Figure C.8: Face after edge nodes have been smoothed out.

After removing a two-sided face, the combinatorial information for all affected data structures is reconfigured. Moreover, the volumes of all adjacent grains are recalculated, as are the velocities of all adjacent nodes.

Removing a tetrahedron-shaped grain

We have already mentioned in Section 4.3.6 that when we have decided to remove a particular grain, we begin by counting its number of faces. If the grain has more than four faces, then we use the procedure described above to remove the face with the fewest number of sides and then amongst those, the one with the smallest area. We repeat this procedure until the grain has only three or four faces remaining. If a grain has four faces but two of them have four sides and two of them have two sides, then we remove one of the two-sided faces first. An illustration of this process was shown at the beginning of Section 4.3.5.

When removing a tetrahedron-shaped grain, the first thing we do is remove all edge nodes. Figure C.9 illustrates the way in which a tetrahedron-shaped grain is collapsed to a single vertex node. We



Figure C.9: Removing a tetrahedron. (a) We begin with a tetrahedron-shaped grain with many edge nodes and with vertex nodes C_i . (b) All edge nodes have been removed. (c) The grain has been collapsed to a single vertex node.

begin in Figure C.9 with a tetrahedron-shaped grain with many edges nodes. Our first step is to remove all edge nodes. Then we collapse the entire structure to a single vertex node. The location of the new vertex node N is the average of the vertex nodes:

$$C = \frac{1}{4} \sum_{i=1}^{4} C_i.$$
 (C.4)

We leave the four vertex nodes C_i in the original locations. After removing this grain, the combinatorial information for all affected data structures is reconfigured. Moreover, the volumes of all adjacent grains are recalculated, as are the velocities of all adjacent nodes.

Removing a three-faced grain

The final topological change that we explain is the removal of a grain with three faces, each of each has two sides. Figure C.10 shows the removal of a grain with three faces. We begin by removing all



Figure C.10: Removing a grain with three faces. (a) We begin with a grain with three faces and many edge nodes. (b) All edge nodes have been removed. At this point three triple edges extend from A to B; they are drawn at slightly different positions so as to make them visible in the figure. (c) The grain has been collapsed to a single triple edge segment.

edges nodes. At this point three triple edges extend from A to B, each of the three faces has 0 area and the grain has 0 volume. Next, we remove the three faces from each of the three neighboring grains. Last, we draw a triple edge segment between the two vertex nodes attached to either end of this grain. The combinatorial information for all affected data structures is reconfigured, the volumes of all adjacent grains, and the velocities of all adjacent nodes, are recalculated.

Bibliography

- R. Yoda, T. Yokomaku, and N. Tsuji, "Plastic deformation and creep damage evaluations of type 316 austenitic stainless steels by EBSD," *Materials Characterization*, vol. 61, no. 10, pp. 913–922, 2010.
- [2] C. Reid, "Bubble foam 3." http://www.flickr.com/photos/macrobubbles/4320059890/, January 2010.
- [3] G. Reisch, "Typvs Vniversalis Terrae, Ivxta Modernorvm Distinctionem et Extensionem per Regna et Provincias." http://www.historicmapworks.com/, 1513.
- [4] K. Peters, "Plagiomnium affine laminazellen." http://commons.wikimedia.org/wiki/File: Plagiomnium_affine_laminazellen.jpeg, January 2006.
- [5] E. Lazar, "Fractured glaze on ceramic in Safed, Israel," June 2011.
- [6] M. Nogueira, W. Ferraz, and A. Sabioni, "Diffusion of the 65Zn radiotracer in ZnO polycrystalline ceramics," *Materials Research*, vol. 6, no. 2, pp. 167–171, 2003.
- [7] C. Reid, "Bubble foam 5." http://www.flickr.com/photos/macrobubbles/4319325927/, January 2010.
- [8] J. Waters, "Bubbles/foam detail." http://www.flickr.com/photos/jedwaters/3298617998/, February 2009.
- M. Grayson, "The heat equation shrinks embedded plane curves to round points," Journal of Differential Geometry, vol. 26, no. 2, pp. 285–314, 1987.
- [10] E. Gaba, "Minimal surface curvature planes." http://en.wikipedia.org/wiki/File: Minimal_surface_curvature_planes-en.svg, June 2006.

- [11] M. Gage and R. Hamilton, "The heat equation shrinking convex plane curves," Journal of Differential Geometry, vol. 23, no. 1, pp. 69–96, 1986.
- [12] G. Huisken and A. N. U. C. for Mathematical Analysis, "Flow by mean curvature of convex surfaces into spheres," *Journal of Differential Geometry*, vol. 20, pp. 237–266, 1984.
- [13] M. Grayson, "A short note on the evolution of a surface by its mean curvature," Duke Mathematical Journal, vol. 58, no. 3, pp. 555–558, 1989.
- [14] U. Mayer, "A dumbbell that becomes singular." http://www.math.utah.edu/ mayer/math-/numerics.html, October 2009.
- [15] E. A. Lazar, "Molecular dynamic studies of the fracture of metals." Yeshiva University, 2006.
- [16] E. Lazar, R. MacPherson, and D. Srolovitz, "A more accurate two-dimensional grain growth algorithm," Acta Materialia, vol. 58, pp. 364–372, Jan. 2010.
- [17] R. Vogel, "Über den wechselseitigen auf-und abbau sich berührender metallischer kristallite im konglomerat," Zeitschrift für anorganische und allgemeine Chemie, vol. 126, no. 1, pp. 1–38, 1923.
- [18] T. Sutoki, "On the mechanism of crystal growth by annealing," Scientific Reports of Tohoku Imperial University, vol. 17, no. 2, pp. 857–876, 1928.
- [19] P. A. Beck, *Metal Interfaces*, p. 208. Cleveland, Ohio: American Society for Testing Materials, 1952.
- [20] D. Harker and E. Parker, "Grain shape and grain growth," Transactions of the American Society of Metals, vol. 34, pp. 156–195, 1945.
- [21] C. S. Smith, "Grains, phases and interfaces an interpretation of microstructure," in Transactions, American Institute of Mining and Metallurgical Engineers, pp. 15–51, 1948.
- [22] J. Burke, "Some factors affecting the rate of grain growth in metals," AIME Transactions, vol. 180, pp. 73–91, 1949.
- [23] E. Steinitz, "Polyeder und raumeinteilungen," Encyclopädie der mathematischen Wissenschaften, Band 3, pp. 1–139, 1922.
- [24] S. Jendrol' and E. Jucovič, "On the toroidal analogue of eberhard's theorem," Proceedings of the London Mathematical Society, vol. 3, no. 3, p. 385, 1972.

- [25] G. Huisken and A. Polden, "Geometric evolution equations for hypersurfaces," Calculus of variations and geometric evolution problems: lectures given at the 2nd session of the Centro Internazionale Estivo (CIME) held in Cetraro, Italy, June 15-22, 1996, p. 45, 1999.
- [26] G. Huisken and T. Ilmanen, "The inverse mean curvature flow and the Riemannian Penrose inequality," *Journal of Differential Geometry*, vol. 59, no. 3, pp. 353–437, 1998.
- [27] G. Perelman, "The entropy formula for the ricci flow and its geometric applications," arxiv, November 2002.
- [28] G. Perelman, "Ricci flow with surgery on three-manifolds," arxiv, March 2003.
- [29] G. Perelman, "Finite extinction time for the solutions to the ricci flow on certain threemanifolds," arxiv, July 2003.
- [30] W. Mullins, "Two-dimensional motion of idealized grain boundaries," Journal of Applied Physics, vol. 27, pp. 900–904, March 1956.
- [31] J. von Neumann, Written Discussion. Cleveland, Ohio: American Society for Metals, 1952.
- [32] D. J. Srolovitz, M. P. Anderson, P. S. Sahni, and G. S. Grest, "Computer simulation of grain growth," Acta Metallurgica, vol. 32, pp. 783–802, 1984.
- [33] M. Anderson, G. Grest, and D. Srolovitz, "Computer simulation of normal grain growth in three dimensions," *Philosophical Magazine Part B*, vol. 59, pp. 293–329, March 1989.
- [34] J. Geiger, A. Roosz, and P. Barkoczy, "Simulation of grain coarsening in two dimensions by cellular-automaton," Acta materialia, vol. 49, no. 4, pp. 623–629, 2001.
- [35] D. Raabe, "A 3d probabilistic cellular automaton for the discrete mesoscale simulation of recrystallization and grain growth phenomena," in *Grain Growth in Polycrystalline Materials III* (H. Weiland, B. Adams, and A. Rollett, eds.), (Warrendale, PA), pp. 179–185, The Minerals, Metals and Materials Society, 1998.
- [36] L. Chen and W. Yang, "Computer simulation of the domain dynamics of a quenched system with a large number of nonconserved order parameters: The grain-growth kinetics," *Physical Review B*, vol. 50, pp. 15752–15756, Dec. 1994.
- [37] C. Krill and L. Chen, "Computer simulation of 3-d grain growth using a phase-field model," Acta Materialia, vol. 50, pp. 3057–3073, July 2002.

- [38] K. Kawasaki, T. Nagai, and K. Nakashima, "Vertex models for two-dimensional grain growth," *Philosophical Magazine Part B*, vol. 60, pp. 399–421, 1989.
- [39] K. Fuchizaki, T. Kusaba, and K. Kawasaki, "Computer modelling of three-dimensional cellular pattern growth," *Philosophical Magazine Part B*, vol. 71, pp. 333–357, March 1995.
- [40] H. Frost, C. Thompson, C. Howe, and J. Whang, "A two-dimensional computer simulation of capillarity-driven grain growth: Preliminary results," *Scripta Metallurgica*, vol. 22, pp. 65–70, 1988.
- [41] B. Sun, Z. Suo, and W. Yang, "A finite element method for simulating interface motion," Acta Materialia, vol. 45, pp. 1907–1915, 1997.
- [42] A. Haslam, S. Phillpot, D. Wolf, D. Moldovan, and H. Gleiter, "Mechanisms of grain growth in nanocrystalline fcc metals by molecular-dynamics simulation," *Materials Science and En*gineering A, vol. 318, no. 1-2, pp. 293–312, 2001.
- [43] P. C. Millett, R. P. Selvam, and A. Saxena, "Molecular dynamics simulations of grain size stabilization in nanocrystalline materials by addition of dopants," *Acta Materialia*, vol. 54, no. 2, pp. 297 – 303, 2006.
- [44] B. Merriman, J. Bence, and S. Osher, "Motion of multiple junctions: A level set approach," *Journal of Computational Physics*, vol. 112, no. 2, pp. 334–363, 1994.
- [45] H. Zhao, T. Chan, B. Merriman, and S. Osher, "A variational level set approach to multiphase motion," *Journal of Computational Physics*, vol. 127, no. 1, pp. 179–195, 1996.
- [46] G. Russo and P. Smereka, "A level-set method for the evolution of faceted crystals," SIAM Journal on Scientific Computing, vol. 21, no. 6, pp. 2073–2095, 2000.
- [47] M. Bernacki, Y. Chastel, T. Coupez, and R. Logé, "Level set framework for the numerical modelling of primary recrystallization in polycrystalline materials," *Scripta Materialia*, vol. 58, no. 12, pp. 1129–1132, 2008.
- [48] K. Brakke, "Surface evolver," Experimental Mathematics, vol. 1, pp. 141–165, 1992.
- [49] S. Cox, "A viscous froth model for dry foams in the surface evolver," Colloids and Surfaces A: Physicochemical and Engineering Aspects, vol. 263, pp. 81–89, 2005.

- [50] K. Marthinsen, O. Hunderi, and N. Ryum, "The influence of spatial grain size correlation and topology on normal grain growth in two dimensions," *Acta Materialia*, vol. 44, no. 4, pp. 1681–1689, 1996.
- [51] D. Reinelt and A. Kraynik, "Simple shearing flow of a dry kelvin soap foam," Journal of Fluid Mechanics, vol. 311, no. 1, pp. 327–343, 1996.
- [52] D. Weaire and N. Rivier, "Soap, cells and statistics: Random patterns in two dimensions," *Contemporary Physics*, vol. 25, no. 1, pp. 59–99, 1984.
- [53] D. Weaire and J. Kermode, "Computer simulation of a two-dimensional soap froth," *Philosophical Magazine Part B*, vol. 48, no. 3, pp. 245–259, 1983.
- [54] C. Smith, "Some elementary principles of polycrystalline microstructure," Metallurgical Reviews, vol. 9, no. 1, pp. 1–48, 1964.
- [55] D. Aboav, "The arrangement of cells in a net," *Metallography*, vol. 13, no. 1, pp. 43 58, 1980.
- [56] V. Fradkov, L. Shvindlerman, and D. Udler, "Computer simulation of grain growth in two dimensions," *Scripta metallurgica*, vol. 19, no. 11, pp. 1285–1290, 1985.
- [57] V. Fradkov and D. Udler, "Two-dimensional normal grain growth: topological aspects," Advances in Physics, vol. 43, no. 6, pp. 739–789, 1994.
- [58] D. Weygand, Y. Bréchet, and J. Lépinoux, "A vertex dynamics simulation of grain growth in two dimensions," *Philosophical Magazine B*, vol. 78, no. 4, pp. 329–352, 1998.
- [59] D. Moldovan, D. Wolf, S. Phillpot, and A. Haslam, "Mesoscopic simulation of two-dimensional grain growth with anisotropic grain-boundary properties," *Philosophical Magazine A*, vol. 82, no. 7, pp. 1271–1297, 2002.
- [60] J. Chen, V. Kotta, H. Lu, D. Wang, D. Moldovan, and D. Wolf, "A variational formulation and a double-grid method for meso-scale modeling of stressed grain growth in polycrystalline materials," *Computer Methods in Applied Mechanics and Engineering*, vol. 193, no. 12-14, pp. 1277–1303, 2004.
- [61] T. Lee and J. Chen, "Modeling of grain growth using voronoi discretization and natural neighbor interpolants," *International Journal for Computational Methods in Engineering Science* and Mechanics, vol. 7, no. 6, pp. 475–484, 2006.

- [62] K. Brakke, Surface Evolver Manual, 2005.
- [63] C. Maurice, "Numerical modelling of grain growth: Current status," in Recrystallization and Grain Growth: Proceedings of the First Joint International Conference, pp. 123ñ–134, Springer, 2001.
- [64] L. Barrales-Mora, V. Mohles, P. Konijnenberg, and D. Molodov, "A novel implementation for the simulation of 2-d grain growth with consideration to external energetic fields," *Computational Materials Science*, vol. 39, no. 1, pp. 160 – 165, 2007.
- [65] F. Lewis, "The correlation between cell division and the shapes and sizes of prismatic cells in the epidermis of cucumis," *Anatomical Record*, vol. 38, pp. 341–376, May 1928.
- [66] N. Rivier and A. Lissowski, "On the correlation between sizes and shapes of cells in epithelial mosaics," *Journal of Physics A: Mathematical and General*, vol. 15, p. L143, 1982.
- [67] J. K. Mason, "The invariance of the steady-state grain growth structure to variations in initial conditions." Personal communication, 2011.
- [68] J. K. Mason, "A coarsening model for an ideal dry two-dimensional foam." Personal communication, 2011.
- [69] G. Grest, D. Srolovitz, and M. Anderson, "Computer simulation of grain growth-iv. anisotropic grain boundary energies," Acta Metallurgica, vol. 33, no. 3, pp. 509–520, 1985.
- [70] A. Rollett, D. Srolovitz, and M. Anderson, "Simulation and theory of abnormal grain growthanisotropic grain boundary energies and mobilities," *Acta Metallurgica*, vol. 37, pp. 1227–1240, Apr. 1989.
- [71] E. Holm, G. Hassold, and M. Miodownik, "On misorientation distribution evolution during anisotropic grain growth," Acta Materialia, vol. 49, no. 15, pp. 2981–2991, 2001.
- [72] E. Lazar, J. Mason, R. MacPherson, and D. Srolovitz, "A more accurate three-dimensional grain growth algorithm," Acta Materialia, 2011.
- [73] J. Konrad, S. Zaefferer, and D. Raabe, "Investigation of orientation gradients around a hard Laves particle in a warm-rolled Fe₃Al-based alloy using a 3D EBSD-FIB technique," Acta Materialia, vol. 54, no. 5, pp. 1369–1380, 2006.

- [74] S. Nielsen, E. Lauridsen, D. Jensen, and H. Poulsen, "A three-dimensional X-ray diffraction microscope for deformation studies of polycrystals," *Materials Science and Engineering A-Structural Materials Properties Microstructure and Processing*, vol. 319, pp. 179–181, 2001.
- [75] D. Saylor, D. Mason, and G. Rohrer, "Experimental method for determining surface energy anisotropy and its application to magnesia," *Journal of the American Ceramic Society*, vol. 83, no. 5, pp. 1226–1232, 2000.
- [76] R. Napolitano, S. Liu, and R. Trivedi, "Experimental measurement of anisotropy in crystalmelt interfacial energy," *Interface Science*, vol. 10, no. 2, pp. 217–232, 2002.
- [77] J. Zou, D. Kotchetkov, A. Balandin, D. Florescu, and F. Pollak, "Thermal conductivity of GaN films: Effects of impurities and dislocations," *Journal of Applied Physics*, vol. 92, p. 2534, 2002.
- [78] Y. Liu, S. Primdahl, and M. Mogensen, "Effects of impurities on microstructure in Ni/YSZ-YSZ half-cells for SOFC," *Solid State Ionics*, vol. 161, no. 1-2, pp. 1–10, 2003.
- [79] R. MacPherson and D. Srolovitz, "The von Neumann relation generalized to coarsening of three-dimensional microstructures," *Nature*, vol. 446, pp. 1053–1055, Apr. 2007.
- [80] D. Srolovitz, G. Grest, and M. Anderson, "Computer simulation of grain-growth V. Abnormal grain growth," Acta Metallurgica, vol. 33, no. 12, pp. 2233–2247, 1985.
- [81] P. Mulheran and J. Harding, "A simple statistical model for grain growth in materials," Acta Metallurgica et Materialia, vol. 39, no. 10, pp. 2251 – 2254, 1991.
- [82] Y. Saito, "Monte Carlo simulation of grain growth in three-dimensions," ISIJ International, vol. 38, no. 6, pp. 559–566, 1998.
- [83] S. Xiaoyan and L. Guoquan, "A simple and efficient three-dimensional monte carlo simulation of grain growth," *Scripta Materialia*, vol. 38, no. 11, pp. 1691 – 1696, 1998.
- [84] L. Hui, W. Guanghou, D. Feng, B. Xiufang, and F. Pederiva, "Monte carlo simulation of three-dimensional polycrystalline material," *Materials Science and Engineering A*, vol. 357, no. 1-2, pp. 153 – 158, 2003.
- [85] Q. Yu and S. K. Esche, "Three-dimensional grain growth modeling with a monte carlo algorithm," *Materials Letters*, vol. 57, no. 30, pp. 4622 – 4626, 2003.

- [86] O. M. Ivasishin, S. V. Shevchenko, N. L. Vasiliev, and S. L. Semiatin, "3d monte-carlo simulation of texture-controlled grain growth," *Acta Materialia*, vol. 51, no. 4, pp. 1019 – 1034, 2003.
- [87] Q. Yu, Y. Wu, and S. K. Esche, "Modeling of grain growth characteristics in three-dimensional domains and two-dimensional cross sections," *Metallurgical and Materials Transactions A*, vol. 36, no. 7, pp. 1661–1666, 2005.
- [88] D. Zollner and P. Streitenberger, "Three-dimensional normal grain growth: Monte carlo potts model simulation and analytical mean field theory," *Scripta Materialia*, vol. 54, pp. 1697–1702, May 2006.
- [89] G. Thomas, R. de Almeida, and F. Graner, "Coarsening of three-dimensional grains in crystals, or bubbles in dry foams, tends towards a universal, statistically scale-invariant regime," *Physical Review E*, vol. 74, no. 2, 2006.
- [90] H. Wang, G. quan Liu, and X. ge Qin, "Grain size distribution and topology in 3d grain growth simulation with large-scale monte carlo method," *International Journal of Minerals*, *Metallurgy and Materials*, vol. 16, no. 1, pp. 37 – 42, 2009.
- [91] O. Ivasishin, S. Shevchenko, and S. Semiatin, "Implementation of exact grain-boundary geometry into a 3-D Monte-Carlo (Potts) model for microstructure evolution," Acta Materialia, vol. 57, no. 9, pp. 2834–2844, 2009.
- [92] E. A. Holm, J. A. Glazier, D. J. Srolovitz, and G. S. Grest, "Effects of lattice anisotropy and temperature on domain growth in the two-dimensional potts model," *Physical Review A*, vol. 43, pp. 2662–2668, Mar 1991.
- [93] Y. Kim, S. Hwang, M. Kim, S. Kwun, and S. Chae, "Three-dimensional Monte-Carlo simulation of grain growth using triangular lattice," *Materials Science and Engineering: A*, vol. 408, no. 1-2, pp. 110–120, 2005.
- [94] D. Raabe, "Cellular automata in materials science with particular reference to recrystallization simulation," Annual review of materials research, vol. 32, pp. 53–76, 2002.
- [95] K. G. F. Janssens, "Random grid, three-dimensional, space-time coupled cellular automata for the simulation of recrystallization and grain growth," *Modelling and Simulation in Materials Science and Engineering*, vol. 11, pp. 157—171, 2003.

- [96] H. Ding, Y. He, L. Liu, and W. Ding, "Cellular automata simulation of grain growth in three dimensions based on the lowest-energy principle," *Journal of Crystal Growth*, vol. 293, no. 2, pp. 489–497, 2006.
- [97] S. Kim, D. Kim, W. Kim, and Y. Park, "Computer simulations of two-dimensional and threedimensional ideal grain growth," *Physical Review E*, vol. 74, no. 6, p. 61605, 2006.
- [98] L. Vanherpe, N. Moelans, B. Blanpain, and S. Vandewalle, "Bounding box algorithm for threedimensional phase-field simulations of microstructural evolution in polycrystalline materials," *Physical Review E*, vol. 76, p. 056702, Nov 2007.
- [99] M. Dorr, J. Fattebert, M. Wickett, J. Belak, and P. Turchi, "A numerical algorithm for the solution of a phase-field model of polycrystalline materials," *Journal of Computational Physics*, vol. 229, no. 3, pp. 626–641, 2010.
- [100] V. Ginzburg and L. Landau, "On the theory of superconductivity (in Russian)," Zh. Eksperim. Teor. Fiz., vol. 20, pp. 1064–1082, 1950.
- [101] V. Tikare, E. A. Holm, D. Fan, and L. Q. Chen, "Comparison of phase-field and potts models for coarsening processes," Acta Materialia, vol. 47, no. 1, pp. 363 – 371, 1998.
- [102] L. Chen and J. Shen, "Applications of semi-implicit fourier-spectral method to phase field equations," *Computer Physics Communications*, vol. 108, no. 2-3, pp. 147 – 158, 1998.
- [103] A. Karma and W.-J. Rappel, "Quantitative phase-field modeling of dendritic growth in two and three dimensions," *Physical Review E*, vol. 57, pp. 4323–4349, Apr 1998.
- [104] I. McKenna, M. Gururajan, and P. Voorhees, "Phase field modeling of grain growth: effect of boundary thickness, triple junctions, misorientation, and anisotropy," *Journal of Materials Science*, vol. 44, pp. 2206–2217, 2009.
- [105] N. Moelans, F. Wendler, and B. Nestler, "Comparative study of two phase-field models for grain growth," *Computational Materials Science*, vol. 46, pp. 479–490, August 2009.
- [106] M. Elsey, S. Esedoglu, and P. Smereka, "Diffusion generated motion for grain growth in two and three dimensions," *Journal of Computational Physics*, vol. 228, no. 21, pp. 8015 – 8033, 2009.

- [107] M. Elsey, S. Esedoğlu, and P. Smereka, "Large-scale simulation of normal grain growth via diffusion-generated motion," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*, vol. 467, no. 2126, pp. 381–401, 2011.
- [108] T. Nagai, S. Ohta, K. Kawasaki, and T. Okuzono, "Computer simulation of cellular pattern growth in two and three dimensions," *Phase Transitions*, vol. 28, pp. 177–211, 1990.
- [109] D. Weygand, Y. Brechet, J. Lepinoux, and W. Gust, "Three-dimensional grain growth: a vertex dynamics simulation," *Philosophical Magazine B-Physics of Condensed Matter Statistical Mechanics Electronic Optical and Magnetic Properties*, vol. 79, pp. 703–716, May 1999.
- [110] A. Kuprat, "Modeling microstructure evolution using gradient-weighted moving finite elements," SIAM Journal on Scientific Computing, vol. 22, no. 2, pp. 535–560, 2000.
- [111] F. Wakai, N. Enomoto, and H. Ogawa, "Three-dimensional microstructural evolution in ideal grain growth - general statistics," Acta Materialia, vol. 48, pp. 1297–1311, Apr. 2000.
- [112] A. Kuprat, D. George, G. Straub, and M. Demirel, "Modeling microstructure evolution in three dimensions with Grain3D and LaGriT," *Computational Materials Science*, vol. 28, no. 2, pp. 199–208, 2003.
- [113] L. Barrales-Mora, G. Gottstein, and L. Shvindlerman, "Three-dimensional grain growth: Analytical approaches and computer simulations," *Acta Materialia*, vol. 56, pp. 5915–5926, Dec. 2008.
- [114] M. Syha and D. Weygand, "A generalized vertex dynamics model for grain growth in three dimensions," Modelling and Simulation in Materials Science and Engineering, vol. 18, p. 015010, 2010.
- [115] S. Townsend, H. Frost, and C. Nichols, "Mechanisms of grain growth in bubble-fence-delineated films," *Modeling and Simulation of Thin-Film Processing*, pp. 143–148, 1995.
- [116] M. E. Glicksman, "Analysis of 3-d network structures," *Philosophical Magazine*, vol. 85, pp. 3–31, Jan. 2005.
- [117] W. Williams and C. Smith, "A study of grain shape in an aluminum alloy and other applications of stereoscopic microradiography," *Transactions, American Institute of Mining and Metallurgical Engineers*, vol. 194, pp. 755–765, 1952.

- [118] F. Rhines, K. Craig, and R. DeHoff, "Mechanism of steady-state grain-growth in aluminum," *Metallurgical Transactions*, vol. 5, no. 2, pp. 413–425, 1974.
- [119] D. Rowenhorst, A. Lewis, and G. Spanos, "Three-dimensional analysis of grain topology and interface curvature in a β-titanium alloy," Acta Materialia, vol. 58, pp. 5511–5519, 2010.
- [120] A. Ferro and M. Fortes, "The elimination of grains and grain boundaries in grain growth," *Interface Science*, vol. 5, pp. 263–278, Nov. 1997.
- [121] J. Glazier and D. Weaire, "The kinetics of cellular patterns," Journal of Physics: Condensed Matter, vol. 4, pp. 1867–1894, February 1992.
- [122] R. Kusner, "The number of faces in a minimal foam," Proceedings of the Royal Society of London Series A-Mathematical Physical and Engineering Sciences, vol. 439, pp. 683–686, Dec. 1992.
- [123] R. T. DeHoff, "The spherical image concept applied to grain structures," Acta Metallurgica Et Materialia, vol. 42, pp. 2633–2643, Aug. 1994.
- [124] H. S. M. Coxeter, "Close-packing and froth," Illinois Journal of Mathematics, vol. 2, no. 4B, pp. 746–758, 1958.
- [125] D. Barnette, "On p-vectors of 3-polytopes," Journal of Combinatorial Theory, vol. 7, pp. 99– 103, 1969.
- [126] P. Engel, "On the enumeration of the simple 3-polyhedra," Acta Crystallographica Section A: Foundations of Crystallography, vol. 59, no. 1, pp. 14–17, 2002.
- [127] G. Brinkmann and B. McKay, "Fast generation of planar graphs," MATCH Communications in Mathematical and in Computer Chemistry, vol. 58, no. 2, pp. 323–357, 2007.
- [128] F. Lutz, "Enumeration and random realization of triangulated surfaces," Discrete Differential Geometry, pp. 235–253, 2008.
- [129] "The manifold page." http://page.math.tu-berlin.de/~lutz/stellar/, September 2011.
- [130] L. Weinberg, "A simple and efficient algorithm for determining isomorphism of planar triply connected graphs," *IEEE Transactions on Circuit Theory*, vol. CT13, no. 2, pp. 142–&, 1966.
- [131] L. Weinberg, "On the maximum order of the automorphism group of a planar triply connected graph," SIAM Journal on Applied Mathematics, vol. 14, no. 4, pp. 729–&, 1966.

- [132] D. Johnson, "The np-completeness column," ACM Transactions on Algorithms (TALG), vol. 1, no. 1, pp. 160–176, 2005.
- [133] E. Lucas, *Récréations mathématiques*, vol. 1. Gauthier-Villars, 1882.
- [134] E. Lazar, J. Mason, R. MacPherson, and D. Srolovitz, "Characterizing the steady state microstructure of two- and three-dimensional grain growth," in preparation.
- [135] D. Fan and L.-Q. Chen, "Diffusion-controlled grain growth in two-phase solids," Acta Materialia, vol. 45, no. 8, pp. 3297 – 3310, 1997.
- [136] D. Kinderlehrer, I. Livshits, and S. Ta'asan, "A variational approach to modeling and simulation of grain growth," SIAM Journal on Scientific Computing, vol. 28, no. 5, pp. 1694–1715, 2006.
- [137] S. Hilgenfeldt, A. Kraynik, S. Koehler, and H. Stone, "An accurate von neumann's law for three-dimensional foams," *Physical Review Letters*, vol. 86, no. 12, pp. 2685–2688, 2001.
- [138] P. R. Rios and M. E. Glicksman, "Topological and metrical analysis of normal grain growth in three dimensions," Acta Materialia, vol. 55, pp. 1565–1571, 2007.
- [139] G. Liu, "Applied stereology in materials science and engineering," Journal of Microscopy, vol. 171, no. 1, pp. 57–68, 1993.
- [140] J. Ohser and F. Mücklich, Statistical analysis of microstructures in materials science. Wiley Chichester, UK, and Weinheim, Germany, 2000.
- [141] K. Nakashima, T. Nagai, and K. Kawasaki, "Scaling behavior of two-dimensional domain growth: Computer simulation of vertex models," *Journal of Statistical Physics*, vol. 57, no. 3, pp. 759–787, 1989.
- [142] V. Fradkov, A. Kravchenko, and L. Shvindlerman, "Experimental investigation of normal grain growth in terms of area and topological class," *Scripta metallurgica*, vol. 19, no. 11, pp. 1291–1296, 1985.
- [143] P. A. Beck, "Annealing of cold worked metals," Advances in Physics, vol. 3, no. 11, pp. 245– 324, 1954.
- [144] P. Feltham, "Grain growth in metals," Acta metallurgica, vol. 5, no. 2, pp. 97–105, 1957.

- [145] D. Aboav and T. Langdon, "The shape of grains in a polycrystal," Metallography, vol. 2, no. 2-3, pp. 171 – 178, 1969.
- [146] P. Rios and M. Glicksman, "Polyhedral model for self-similar grain growth," Acta Materialia, vol. 56, pp. 1165–1171, 2008.
- [147] M. Hillert, "On the theory of normal and abnormal grain growth," Acta Metallurgica, vol. 13, no. 3, pp. 227–238, 1965.
- [148] X. Xuea, F. Righettia, H. Telleya, T. Lieblinga, and A. Mocellin, "The Laguerre model for grain growth in three dimensions," *Philosophical Magazine Part B*, vol. 75, no. 4, pp. 567–585, 1997.
- [149] J. Stavans, "The evolution of cellular structures," Reports on Progress in Physics, vol. 56, pp. 733–789, 1993.
- [150] S. Jurinea, S. Cox, and F. Graner, "Dry three-dimensional bubbles: growth-rate, scaling state and correlations," *Colloids and Surfaces A: Physicochemical and Engineering Aspects*, vol. 263, pp. 18–26, 2005.
- [151] D. Aboav, "The arrangement of grains in a polycrystal," *Metallography*, vol. 3, no. 4, pp. 383– 390, 1970.
- [152] D. Weaire, "Some remarks on the arrangement of grains in a polycrystal," Metallography, vol. 7, no. 2, pp. 157–160, 1974.
- [153] H. Federer, "Geometric measure theory." Springer, 1969, 1996.
- [154] D. Klain and G. Rota, Introduction to geometric probability. Cambridge Univ Press, 1997.
- [155] K. A. Brakke, "Grain growth with the surface evolver," in Video Proceedings of the Workshop on Computational Crystal Growing (J. E. Taylor, ed.), (Providence, RI), American Mathematical Society, 1992.
- [156] F. Wakai, Y. Shinoda, S. Ishihara, and A. Dominguez-Rodriguez, "Topological transformation of grains in three-dimensional normal grain growth," *Journal of Materials Research*, vol. 16, no. 7, pp. 2136–2142, 2001.