The Role of Topology and Topological Changes in the Mechanical Properties of Epithelia

by

Meryl Spencer

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Physics)
in The University of Michigan
2019

Doctoral Committee:

Associate Professor David K Lubensky, Chair
Professor Mark Newman
Professor Jennifer Ogilvie
Assistant Professor Kevin Wood


TEACHERS
STUDENTS
RESEARCHERS
WORKERS
JOIN GEO

Meryl Spencer
maspenc@umich.edu
ORCID iD: 0000-0003-1994-4100

ACKNOWLEDGEMENTS

This work would have not have been possible without the mentorship of my adviser David. Much of my thesis work relies on analysis of thousands of images taken by Jesus Lopez-Gay.

I sustained a major spinal cord injury just as I started to write my thesis. I would not have been physically capable of finishing it, if not for the amazing medical care I received from my entire team at the neuro-ICU and 6A. I could not have afforded the care without medical benefits from my grad union GEO. Thank you from the bottom of my heart everyone who cooked me meals, cleaned my house, and gave me moral support throughout my long recovery. Special thanks to Patricia Klein for organizing my care, and my parents for their constant support.

This work probably would have been possible without the support of my fellow graduate students, but it wouldn't have been nearly as fun. I'd like to thank my amazing first-year cohort especially, Paige, Rutu, Karishima, Chrisy and Kevin for working through Jackson with me, as well as Jessie, Joe, Glenn, Peter, Ansel, and Anthony for joining in second-year. John Ware has supported me from start to finish, from working through old qualifying exams through proof-reading thesis chapters.

My entire family has given me moral and finantial support throughout college and graduate school, even if they often didn't understand what I was doing. Thanks for sticking with me.

TABLE OF CONTENTS

LIST OF FIGURES

Figure

LIST OF TABLES

Table

LIST OF APPENDICES

Appendix

ABSTRACT

Epithelia, quasi-two-dimensional sheets of cells, are important in molding organs into their correct final shape and size during animal development. Epithelia are active materials that are capable of both generating and reacting to mechanical forces, in a manner that depends on the organization of their cells. Cells in an epithelium may divide, exchange neighbors, or otherwise remodel their packing topology, thereby creating a complex feedback loop between tissue topology and mechanical forces. A full theory of the interplay between mechanical forces and cellular arrangement has not yet been developed. Here we work towards developing such a theory using a vertex model framework, which represents complex biological processes as an active network of cell-cell interactions. We consider several specific problems:

We carefully derive the forces acting on vertices, places where three or more cells meet, with special attention to fourfold vertices. This work results in a mathematical proof of the criterion for stabilizing fourfold vertices, which places theoretical limits on the types of tissues that can support stable fourfold vertices.

Continuous supra-cellular actomyosin cables are capable of generating large forces to either resist external stress or drive cell motion. These cables have been extensively studied in isolation, but there has been little work on the effect of multiple parallel cables on tissue mechanics. Here we show that these cables prevent cells from becoming elongated or misshapen under large stress anisotropies and can only arise in certain favorable topologies. We develop two measures of the favorability of a disordered packing to forming cables, a quality we call cableness, and show that passive cell flow reduces cableness whereas oriented cell divisions increase cableness.

A large anisotropic stress is applied to the Drosophila pupal notum for a few hours during its development, at which time it develops internal apical actomyosin fibers. We present a toy model incorporating these fibers into the network of cell-cell interactions, based on the assumption that these fibers form in order to resist the applied stress, and validate predictions of the model against experimental data.

We also summarize the computational methods that are the foundation of our scientific results. We present the design philosophy for our highly modular vertex model, as well as the algorithms we developed to correctly implement T1 transitions.

We also discuss our use of automated image analysis techniques in the context of fluorescent imaging, including both morphological operations and machine learning algorithms.

CHAPTER I

Introduction

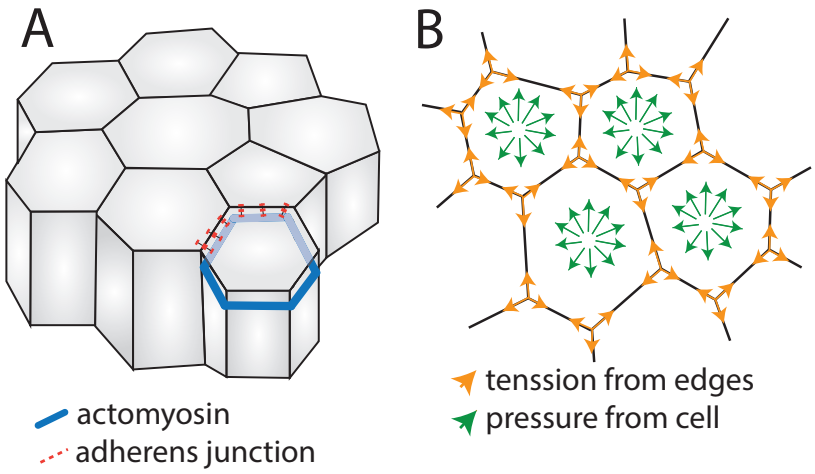

Figure 1.1: **A** cartoon of an epithelium. Adherens junctions keep cells in contact with each other. An apical band of actomyosin provides a contractile force in the cell. **B** Physicists model of the epithelium as a two dimensional network of contractile edge tensions balanced by cell pressures.

Not many physicists think about why they don't have an eyeball growing out of the bottom of their foot, but maybe more of them should. After all, it is quite amazing that from a single initial cell, all the cells in your body are directed to their correct final locations, ensuring that that you grow exactly two eyes placed symmetrically in the front of your head. The whole developmental process would seem quite miraculous if it didn't happen every day. Physicists generally study systems at or near equilibrium, but there is a treasure trove of interesting questions to be asked about systems far from equilibrium, and the biological world provides many examples.

Development is generally thought of as being composed of four major processes: morphogenesis, differentiation, growth, and region specification. This thesis will focus

exclusively on morphogenesis, the process by which tissues develop their shape. We restrict ourselves to epithelial tissues, which form the outer surfaces of many organs, such as the kidney, stomach and intestines, and therefore play a large role in sculpting organs into their proper size and shape [1, 2, 3, 4, 5].

An epithelium is a quasi-two-dimensional sheet of cells whose membranes are bound together by protein complexes at sites called adherens junctions, as shown in figure 1.1 [6, 7, 8]. Each cell has a ring of actin filaments bound together by adhesion molecules, which attach to the ahderens junctions. Myosin and other molecular motors walk along the actin filaments, creating a contractile force that drives cell rearrangement [9, 10]. Epithelia are capable of both generating and reacting to mechanical forces in a manner that depends on the organization of their cells. Therefore, understanding how the structure of the cell-cell network effects the physical properties of epithelia is an important part of understanding morphogenesis. This thesis aims to develop a theory of the interplay between mechanical forces and cellular arrangement in epithelia by representing complex biological processes as an active network of cell-cell interactions.

Three topological changes drive epithelial morphogenesis. The first two processes are inverses of each other: apoptosis and mitosis. In apoptosis, cells are extruded from the epithelial layer. Their neighboring cells move in to fill the space left by the extruded cell, preventing gaps in the tissue. During mitosis, a cell rounds up and divides along its long axis, creating two cells from one [11, 12]. Mitosis often drives an increases in size of a tissue, but oriented divisions can also have more subtle effects on the tissue shape and properties, which we will discuss in chapter IV.

The third process, called a T1 transition, swaps neighboring cells, as shown in figure 1.2. One of the most notable instances of large scale T1 transitions in nature is during the process of germ-band extension in *Drosophila*, in which the initially square germ-band tissue undergoes a series of T1 transitions that dramatically narrow the tissue along the dorsal-ventral axis causing it to wrap around the posterior of the embryo [13, 14]. We will discuss T1 transitions and their intermediary state, a fourfold vertex, in greater detail in chapter II.

There have been two well known attempts to quantify the total tissue shape change in terms of these fundamental topological processes as well as cell shape change by both the Jülicher and Bellïache labs [16, 17]. They quantify the total contribution to the final shape from T1 transitions, mitosis, apoptosis, cell shape changes, and so forth. Both of these systems work on movies of epithelia in which the outline of every cell is tracked through time and space.

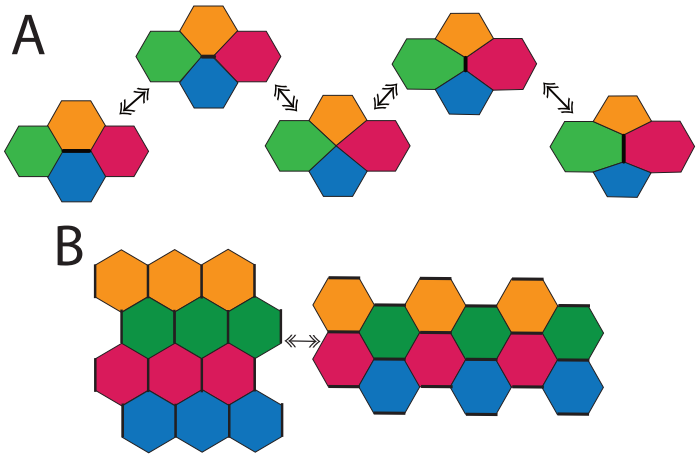

Figure 1.2: **A** Cartoon of a T1 transition. An edge shrinks to form a fourfold vertex and then elongates in the perpendicular direction causing neighbor exchange. **B** Cartoon of convergent extension. Vertical edges undergo T1 neighbor exchange causing the tissue to elongate.

The broader question of morphogenesis fits squarely into the realm of biology. However, physicists have an important role to play in understanding tissue mechanics and more broadly in biological modeling. There is a growing push to incorporate the tools and methods of understanding from physics and engineering to biological problems [18, 19, 20, 21, 22]. This framework is often referred to as systems biology, which began gaining prominence as a field around 20 years ago. Physical modeling allows us to synthesize large amounts of data and make testable predictions about how a system will respond based on you hypothesis of how it works.

We approach morphogenisis at the cellular scale, and for the most part ignore signaling pathways, focusing on mechanical behaviors without regard to what exactly underlies the mechanical properties. Recent advances in computational power have allowed for the computational modeling of epithelia at the cellular scale. Three major categories of computational models are in common use: vertex models, cellular Potts models, and finite element models [20]. Cellular Potts models are generalizations of the Ising model, in which each cell is represented by a different state and the Hamiltonian is structured to include adhesion energies and volume constraints [23, 24]. Although these models can correctly reproduce some cell behavior, they can also be difficult to interpret. Finite element models split each cell into multiple finite elements [25]. In this thesis we focus exclusively on vertex models, in which the epithelium is described by a network structure of edges that represent the cell boundaries. Edges have a contractile force that is offset by pressure in the cells. The model works by integrating the equations of motion of the vertices as they are pushed and pulled on

by edge and cellular forces. Chapters II and III discuss the technical details of these models.

Vertex models are highly flexible, and can also be used to study a number of different phenomena. Much work is focused on creating models that recreate the behavior of a specific experimental system. The goal of these models is to capture the large scale behavior of the system with as few parameters as possible. The model helps synthesize what is known about individual proteins into an understanding of their function. Vertex models are also used to understand the basic properties of cell sheets from a condensed matter perspective. For example, some groups are interested in what parameters govern the transition between liquid-like and solid-like phases [26, 27, 28], while others study the effects of cell division on tissue level stress anisotropies [29].

This thesis is divided into three parts. The first part covers the vertex model of epithelial mechanics and spans chapters II-III. Chapter II discusses the stability of fourfold vertices in the vertex model framework. Vertex models are based on models of dry foams, in which it is well established that fourfold vertices are always unstable. In computational vertex models, it is generally assumed that all of the edges are straight, rather than unconstrained as in the foam model. This simplification introduces direct pressure effects on vertices. The effect of these pressure effects on the stability of fourfold vertices was unknown prior to our work. We proved that in the most basic case, in which all edges are assumed to have the same constant tension, fourfold vertices remain unstable. However, when edges with orientationaly dependent tensions are allowed, fourfold vertices may be stabilized. This stability is important because fourfold vertices are an intermediate step in the T1 process, one of the fundamental topology changing processes in morphogenesis. Although fourfold vertices were once thought to always be unstable, there is mounting evidence of fourfold vertices that are stable over long time scales in various biological systems [30, 31, 32, 33, 34].

We discuss some of the more subtle details of the implementation of the vertex model designed for the Lubensky lab in chapter III. Beyond assisting future generations of graduate students in understanding the large code base we have established, we hope that this chapter will encourage others to write down how their vertex models are implemented, which is unfortunately rare in the field today [35].

The second part of the thesis spans chapters IV-VI and covers the mechanics of tissues under stress, with an emphasis on the collaboration between the Lubensky and Bellïache labs on the role of epithelial stress fibers.

In chapter IV we will define *cableness* as a measure of how agreeable a tissue's topology is to forming supra-cellular actomyosin cables. Such cables are formed when the edges that divide cells align with one another to form a straight line capable of resisting or inducing large tissue stresses. We originally conceived of the concept of cableness in the context of the *Drosophilia* pupal notum. This tissue appears to change topology from an initially disordered state to one in which the cells align into columns. Although we first became aware of the role topology plays in forming cables in the notum, the feature applies more generally to all tissues that form many parallel cables. We take all of chapter IV to develop a quantitative understanding of cables more broadly.

In chapter V, we return our attention to the *Drosophilia* pupal notum. During an approximately eight hour period in pupal development, cells in a specific section of the notum develop internal apical actomyosin fibers. Such fibers are a standard feature of cells in isolation [36, 37, 38, 39], but had not been previously observed in epithelia, where cells are in constant contact with one another. Along with our experimental collaborators in the Belïache lab, we have been working to understand the role that these fibers play in morphogenesis from both a signaling and mechanical perspective.

We developed a simple theoretical model that incorporates internal fibers into the vertex model by representing them as internal edges. The model predicts that larger cells require more fibers to resist elongation, which matches with the experimental data. We also predict that tissues that are in a more cable-friendly orientation require fewer fibers to resist elongation, as the cables formed by the edges already serve the function of resisting stress. Using the cableness measures developed in chapter IV, we show that the tissue becomes more cabley over time, coinciding with the disappearance of the fibers.

The data analysis in chapter V relies on automated image processing of large amounts of data. Chapter VI is intended as a review of the common techniques in modern image processing, including the recent use of machine learning for image categorization. As an illustrative example, we will cover the process we use to automatically identify cells from florescent images, as well as our unsuccessful attempt to automatically identify fibers using machine learning algorithms.

The third part of the thesis stands apart from the first two, focusing on the mathematical properties of a vertex model inspired network. In chapter VII, we calculate the percolation thresholds, a measure of a networks connectivity, of honeycomb lattices in which some edges have been swapped in a T1 process that introduces disorder in

the system. We find that the percolation threshold increases with increasing defects, which is consistent with known lattices of the same class.

Understanding animal development is a huge undertaking. Recent advances in gene editing and imaging have allowed us to collect more data than ever on the movement of various proteins and other biological material through the developmental process. However, it is not enough to simply record and diagram an ever increasing web of protein interactions; we need physical models to help us synthesize our knowledge and produce testable predictions that will move us towards a theory of development. Physical models, like the vertex model, allow us to bypass exact knowledge of molecular motors and filaments, by using physical laws to determine what forces these molecules must exert. Models of epithelia are especially important, because epithelia are so ubiquitous in our bodies and they set the final shape of many of our organs and other tissues. In this thesis we broaden our knowledge of the implications imbedded in vertex models governing the stability of fourfold vertices, and offer algorithms for better implementing the vertex model. We also investigate the role that newly discovered internal stress fibers play in resisting stress and cell elongation in epithelia, by developing a mechanical model that we validate against experimental data. Overall this thesis represents a step along the path of bringing a physical understanding to development.

Bibliography

[1] J. A. Zallen, "Planar polarity and tissue morphogenesis," *Cell*, vol. 129, pp. 1051–1063, Jun 2007.

[2] T. Lecuit and P. F. Lenne, "Cell surface mechanics and the control of cell shape, tissue patterns and morphogenesis," *Nature Reviews Molecular Cell Biology*, vol. 8, pp. 633–644, Aug 2007.

[3] S. Quintin, C. Gally, and M. Labouesse, "Epithelial morphogenesis in embryos: asymmetries, motors and brakes," *Trends in Genetics*, vol. 24, pp. 221–230, May 2008.

[4] M. Osterfield, C. A. Berg, and S. Y. Shvartsman, "Epithelial Patterning, Morphogenesis, and Evolution: Drosophila Eggshell as a Model," *Developmental Cell*, vol. 41, pp. 337–348, May 2017.

[5] C.-P. Heisenberg and Y. Bellaïche, "Forces in tissue morphogenesis and patterning," *Cell*, vol. 153, no. 5, pp. 948 – 962, 2013.

[6] M. Rauzi, P.F. Lenne, and T. Lecuit, "Planar polarized actomyosin contractile flows control epithelial junction remodelling," *Nature*, vol. 468, p. 1110, Dec 2010.

[7] A. M. Greiner, H. Chen, J. P. Spatz, and R. Kemkemer, "Cyclic Tensile Strain Controls Cell Shape and Directs Actin Stress Fiber Formation and Focal Adhesion Alignment in Spreading Cells," *PLOS One*, vol. 8, Oct 2013.

[8] S. K. Wu, S. Budnar, A. S. Yap, and G. A. Gomez, "Pulsatile contractility of actomyosin networks organizes the cellular cortex at lateral cadherin junctions," *Euro. J. of Cell Biology*, vol. 93, pp. 396–404, Oct 2014.

[9] R. Fernandez-Gonzalez, S. d. M. Simoes, J.-C. Roeper, S. Eaton, and J. A. Zallen, "Myosin II Dynamics Are Regulated by Tension in Intercalating Cells," *Developmental Cell*, vol. 17, pp. 736–743, Nov 2009.

[10] S. Curran, C. Strandkvist, J. Bathmann, M. de Gennes, A. Kabla, G. Salbreux, and B. Baum, "Myosin II Controls Junction Fluctuations to Guide Epithelial Tissue Ordering," *Developmental Cell*, vol. 43, Nov 2017.

[11] O. Hertwig, "Das Problem der Befruchtung und der Isotropie des Eies, eine Theorie der Vererbung," *Jenaische Zeitschrift fr Naturwissenschaften*, vol. 18, p. , 1884.

[12] C. Cadart, E. Zlotek-Zlotkiewicz, M. Le Berre, M. Piel, and H. K. Matthews, "Exploring the Function of Cell Shape and Size during Mitosis," *Developmental Cell*, vol. 29, pp. 159–169, Apr 2014.

[13] S. d. M. Simoes, A. Mainieri, and J. A. Zallen, "Rho GTPase and Shroom direct planar polarized actomyosin contractility during convergent extension," *Journal of Cell Biology*, vol. 204, pp. 575–589, Feb 2014.

[14] D. Kong, F. Wolf, and J. Grohans, "Forces directing germ-band extension in drosophila embryos," *Mechanisms of Development*, vol. 144, pp. 11 – 22, 2017.

[15] A. C. Martin, M. Kaschube, and E. F. Wieschaus, "Pulsed contractions of an actin-myosin network drive apical constriction," *Nature*, vol. 457, Jan 2009.

[16] M. Merkel, R. Etournay, M. Popovcfi, G. Salbreux, S. Eaton, and F. Jülicher, "Triangles bridge the scales: Quantifying cellular contributions to tissue deformation," *Phys. Rev. E*, vol. 95, p. 032401, Mar 2017.

[17] B. Guirao, S. U. Rigaud, F. Bosveld, A. Bailles, J. Lopez-Gay, S. Ishihara, K. Sugimura, F. Graner, and Y. Bellaïche, "Unified quantitative characterization of epithelial tissue development," *eLife*, vol. 4, p. e08519, Dec 2015.

[18] U. Sauer, M. Heinemann, and N. Zamboni, "Getting closer to the whole picture," *Science*, vol. 316, no. 5824, pp. 550–551, 2007.

[19] H. Kitano, "Computational systems biology," *Nature*, vol. 420, pp. 206–210, Nov 2002.

[20] Y. Mao and J. B. A. Green, "Systems morphodynamics: understanding the development of tissue hardware," *Philosophical Transactions of the Royal Society B-Biological Sciences*, vol. 372, May 2017.

[21] J. Lewis, "From signals to patterns: Space, time, and mathematics in developmental biology," *Science*, vol. 322, pp. 399–403, Oct 2008.

[22] Y. Lazebnik, "Can a biologist fix a radio? Or, what I learned while studying apoptosis," *Cancer Cell*, vol. 2, pp. 179–182, Sep 2002.

[23] T. Hirashima, E. G. Rens, and R. M. H. Merks, "Cellular Potts modeling of complex multicellular behaviors in tissue morphogenesis," *Development Growth & Differentiation*, vol. 59, pp. 329–339, Jun 2017.

[24] J. A. Glazier and F. Graner, "Simulation of the differential adhesion driven rearrangement of biological cells," *Phys. Rev. E*, vol. 47, pp. 2128–2154, Mar 1993.

[25] H. Chen and G. Brodland, "Cell-level finite element studies of viscous cells in planar aggregates," *J. of Biomechanical Eng.-Transactions Of The ASME*, vol. 122, pp. 394–401, Aug 2000.

[26] D. Bi, X. Yang, M. C. Marchetti, and M. L. Manning, "Motility-Driven Glass and Jamming Transitions in Biological Tissues," *Phys. Rev. X*, vol. 6, Apr 2016.

[27] D. M. Sussman and M. Merkel, "No unjamming transition in a Voronoi model of biological tissue," *Soft Matter*, vol. 14, pp. 3397–3403, May 2018.

[28] D. M. Sussman, M. Paoluzzi, M. C. Marchetti, and M. L. Manning, "Anomalous glassy dynamics in simple models of dense biological tissue," *EPL*, vol. 121, Feb 2018.

[29] M. Wu and M. Mani, "A mechanism for the proliferative control of tissue mechanics in the absence of growth," *bioRxiv*, 2018.

[30] R. P. Simone and S. DiNardo, "Actomyosin contractility and Discs large contribute to junctional conversion in guiding cell alignment within the Drosophila embryonic epithelium," *Development*, vol. 137, pp. 1385–1394, Apr 2010.

[31] P.-L. Bardet, B. Guirao, C. Paoletti, F. Serman, V. Léopold, F. Bosveld, Y. Goya, V. Mirouse, F. Graner, and Y. Bellaïche, "PTEN controls junction lengthening and stability during cell rearrangement in epithelial tissue," *Dev. Cell*, vol. 25, no. 5, pp. 534–546, 2013.

[32] G. Trichas, A. M. Smith, N. White, V. Wilkins, T. Watanabe, A. Moore, B. Joyce, J. Sugnaseelan, T. A. Rodriguez, D. Kay, R. E. Baker, P. K. Maini, and S. Srinivas, "Multi-Cellular rosettes in the mouse visceral endoderm facilitate the ordered migration of anterior visceral endoderm cells," *PLOS Bio.*, vol. 10, Feb 2012.

[33] M. Tamada and J. A. Zallen, "Square cell packing in the Drosophila embryo through spatiotemporally regulated EGF receptor signaling," *Dev. Cell*, vol. 35, no. 2, pp. 151–161, 2015.

[34] M. J. Harding, H. F. McGraw, and A. Nechiporuk, "The roles and regulation of multicellular rosette structures during morphogenesis," *Development*, vol. 141, no. 13, pp. 2549–2558, 2014.

[35] A. G. Fletcher, J. M. Osborne, P. K. Maini, and D. J. Gavaghan, "Implementing vertex dynamics models of cell populations in biology within a consistent computational framework," *Progress in Biophysics and Molecular Biology*, vol. 113, no. 2, pp. 299 – 326, 2013.

[36] R. Kaunas, P. Nguyen, S. Usami, and S. Chien, "Cooperative effects of Rho and mechanical stretch on stress fiber organization," *PNAS*, vol. 102, pp. 15895–15900, Nov 2005.

[37] J. Wang, P. Goldschmidt-Clermont, and F. Yin, "Contractility affects stress fiber remodeling and reorientation of endothelial cells subjected to cyclic mechanical stretching," *Annals Of Biomedical Engineering*, vol. 28, pp. 1165–1171, Oct 2000.

[38] K. Burridge and E. S. Wittchen, "The tension mounts: Stress fibers as force-generating mechanotransducers," *Journal Of Cell Biology*, vol. 200, pp. 9–19, Jan 2013.

[39] T. Ishizaki, Y. Morishima, M. Okamoto, T. Furuyashiki, T. Kato, and S. Narumiya, "Coordination of microtubules and the actin cytoskeleton by the Rho effector mDia1," *Nature Cell Biology*, vol. 3, pp. 8–14, Jan 2001.

CHAPTER II

Vertex stability and topological transitions in vertex models of foams and epithelia

Notes: Adapted from Spencer, Meryl A., Jabeen, Zahera, and Lubensky, David K., "Vertex stability and topological transitions in vertex models of foams and epithelia," *Eur. Phys. J. E*, vol. 40, no. 1, p. 2, 2017.

2.1 Introduction

From the lining of the gut to the surface of the skin, epithelial tissues are one of the essential building blocks of animal organs. The motion of epithelial cells over time correspondingly drives many aspects of animal development and morphogenesis, and understanding this movement is thus a central problem in quantitative biology. Although there has been remarkable progress in identifying and imaging the proteins involved in the development of specific epithelia [1, 2, 3, 4, 5, 6], it remains a major challenge to translate this molecular knowledge into a higher level picture of how the organization of epithelial cells emerges from local mechanical interactions. Computational modeling represents an important tool to address this question, and it is hence essential to have well-understood models available to describe epithelia. In this chapter, we begin to address this need by deriving some general results on the stability of fourfold vertices in a widely-used class of vertex models [7, 8, 9, 10, 11].

A simple epithelium is a quasi-two-dimensional sheet of cells characterized by strong inter-cellular adhesion [12, 13]. This adhesion occurs primarily at a belt of adherens junctions, composed largely of cadherins, which hold the adjacent cell membranes together. Additionally, each cell has a band of contractile cortical acto-myosin running along the inside of the adherens junctions. The combination of adherens junction and contractile actin ring leads to an effective line tension along cell-cell

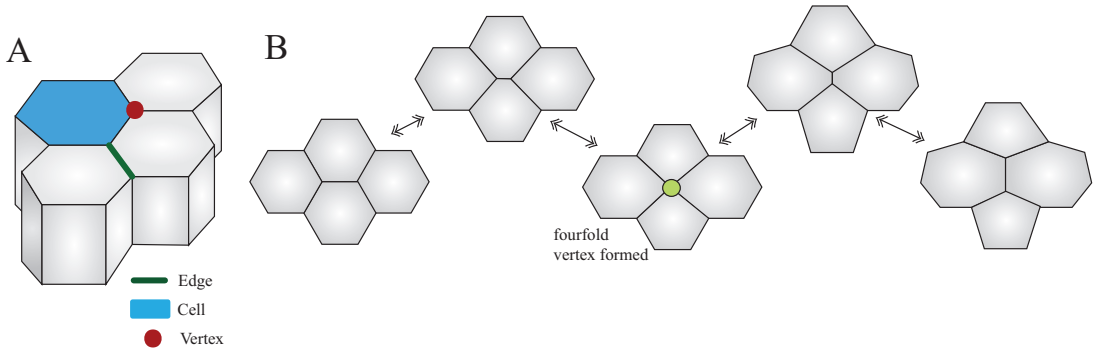

Figure 2.1: **A:** Cartoon of cells in an epithelial sheet. A single cell is shaded blue. The interface between two cells forms an edge (one edge is highlighted by the bold green line). The red dot indicates a vertex, defined as a point at which three or more cells touch. We treat the epithelium as a two-dimensional sheet, focusing on the level of the adherens junctions near the apical (top) surface. **B:** Cartoon of epithelial cells undergoing a T1 topological transition (viewed from above). An edge shrinks down until a fourfold vertex is formed, then a new edge elongates in a roughly perpendicular direction. As a result, the cells exchange neighbors, altering the topology of the cell packing. The middle panel shows the moment at which a fourfold vertex (light green dot) appears. The fourfold vertex has four neighboring cells and four neighboring edges and could in principle either be stable or resolve into either of the two different topologies shown to the left and right.

junctions which the cell can modulate by targeting adhesion molecules or myosin and their regulators. Thus, for example, the tension can be made to vary as a function of junctional orientation as a result of regulation by the planar cell polarity pathway [1, 2, 3, 4, 5, 6] (as seen, e.g., in *Drosophila* germ band extension [12]).

When viewed as a two dimensional sheet of tightly packed cells, the epithelium strongly resembles a dry soap film. Indeed, interfacial tension plays a central role in the physics of both systems, and foam-inspired models are thus frequently used to describe epithelia. The standard model for the mechanics of a dry foam, which we here refer to as the *Plateau model*, goes back to the work of Plateau in the 1800s [14]. It posits that the final shape of a group of bubbles is determined by minimizing a surface tension energy proportional to the total bubble surface area (in 3 dimensions) or the total length of the interfaces between bubbles (in two dimensions).

Many recent computational descriptions of epithelia have been based on so-called vertex models [7, 8, 9, 10, 11], a class of simplified variants of the Plateau model that have been applied to systems including epithelia, foams, and metal grains [7, 15, 16, 17, 18]. The two models share the basic feature of an energy that grows with the total length of the bubble-bubble or cell-cell interfaces. They differ in that, whereas

the Plateau model allows the interfaces between bubbles or cells to take arbitrary shapes, vertex models impose that (in two dimensions) these interfaces must always be straight lines, which we refer to as *edges*. (Extensions which allow for curved edges [19, 20, 21, 22] and for three-dimensional cells [23] have been proposed but are beyond the scope of this work.) The major degrees of freedom in vertex models are then the positions of the *vertices* where three or more cells meet and which are joined by edges to form polygonal cells (fig. 7.1A).

A fourfold vertex occurs whenever a vertex has four neighboring cells and edges as opposed to the much more common three. Fourfold vertices generally resolve into two threefold vertices by pairing the edges of the fourfold vertex and growing a new edge between them. There are two different ways to pair the edges, resulting in two different final cell arrangements (fig. 7.1B). Cellular rearrangements that switch between these two topologies, through the intermediate of a fourfold vertex, allow the epithelial sheet to change shape and enable cells of specific types to find their correct location and morphology. Indeed, this process, known as a T1 transition, has been shown to play a central role in morphogenetic movements like tissue elongation [24, 25, 26]. Though fourfold vertices usually break up, tissues in which fourfold vertices remain stable over a relatively long timescale have also recently been observed [27, 28, 29, 30, 31].

Although vertex models (proposed by Honda in the 1980s [15, 32, 33]) clearly ignore many features of real cell shape, they are thought to capture the essential physics when cells are close to polygonal, and they have been applied successfully to study many features of epithelial morphogenesis [7, 10, 34, 35]. Moreover, they have the advantage of being both simple and straightforwardly extensible to include effects ranging from the dynamics of proteins localized at the edges to buckling into the third dimension [20, 35, 36]. Despite their increasing popularity, however, some of these models' fundamental theoretical properties are poorly understood [37, 38]. Most notably, in the Plateau model of dry foams which inspired vertex models, fourfold vertices (fig. 7.1B) are always unstable, breaking up into two threefold vertices [14]. Because vertex models demand that cell-cell junctions remain straight, cell pressure plays a somewhat different role in them from their role in the Plateau model (where edges can take on any shape), and the standard arguments leading to this instability cannot be taken over directly from the Plateau model. It is thus unclear whether the instability is likewise always present in vertex models. Here, we show that the vertex model does not allow for stable fourfold vertices at mechanical equilibrium when all edges have the same tension. In contrast, we find that introducing a simple dependence of tensions on edge orientation is sufficient to stabilize fourfold vertices.

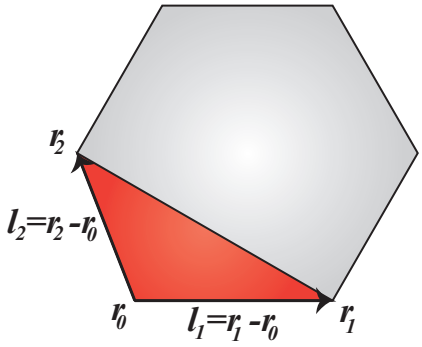

Figure 2.2: Cartoon of a cell with vertices at positions $\boldsymbol{r}_0$, $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$. Movement of vertex $\boldsymbol{r}_0$ affects the lengths of the adjacent edges $\boldsymbol{l}_1$ and $\boldsymbol{l}_2$ and the area of the shaded triangle bounded by these edges. We assume that the face of the cell is in the x-y plane of a standard right-handed coordinate system with the $\hat{\boldsymbol{z}}$ axis projecting out of the plane . The area of the shaded region is then $\frac{1}{2}\hat{\boldsymbol{z}} \cdot (\boldsymbol{l}_1 \times \boldsymbol{l}_2)$.

This result may help to explain the observation of long-lived fourfold vertices in some biological systems [16, 24, 27, 28, 29, 30, 31, 39]. Moreover, our examination of the dynamics of fourfold vertices suggests an improved algorithm for treating T1 transitions in simulations which removes the potential for spurious oscillations and incorrect resolutions present in some prior *ad hoc* approaches. This procedure will be especially useful as we develop more complex models of epithelia that couple cell shape and the dynamics of junctional proteins [20].

In the remainder of this chapter, we investigate the stability of fourfold vertices and dynamics near topological transitions in vertex models. We begin with a full description of the model, and we then proceed to develop equations describing the dynamics near fourfold vertices. In section 2.3, we state the conditions under which a fourfold vertex is stable. The subsequent two sections then show that it is impossible to satisfy all of the stability conditions simultaneously for stationary vertices with equal tensions, demonstrating that the model does not admit stable fourfold states in this case. In section 2.6, we argue that stable fourfold vertices do become possible when the assumptions of mechanical equilibrium or of equal tensions are relaxed, potentially shedding light on why fourfold vertices are observed in some biological systems. We conclude by touching on the implications of our results for the design of algorithms to simulate vertex models.

2.2 The vertex model

2.2.1 Definition

Although we will eventually consider generalizations where the forces on vertices cannot be derived from an energy, the vertex model is most commonly stated in terms of an effective energy that is a function of the vertex positions; the final tissue shape is then given by a minimum of the energy function. The form of this energy differs slightly between authors, but a basic version is [33],

$$E = \sum_i \Gamma_i l_i + \frac{K}{2} \sum_\alpha (A_\alpha - A_{0\alpha})^2. \tag{2.1}$$

The first term describes the interfacial tension along the edges and combines the effects of both cell-cell adhesion and actomyosin contractility in the adherens band [12, 13]. The sum over i runs over all edges, with edge i having tension Γ_i and length l_i. The second term describes the energy cost of deforming cells from their preferred area. The sum over α runs over all of the cells in the tissue. $A_{0\alpha}$ is the preferred area of cell α, A_α is the cell's actual area, and the constant K parameterizes cells' resistance to area changes. A common further simplification is to assume that all edges have the same properties so that $\Gamma_i = \Gamma$ for all i; we will call this assumption the *equal tension vertex model*.

From eq. 2.1 we can immediately find the force on a vertex by taking the derivative with respect to the vertex position,

$$\boldsymbol{F}_{\boldsymbol{r}_0} = -\frac{\partial E}{\partial \boldsymbol{r}_0}, \tag{2.2}$$

where $\boldsymbol{r}_0$ is the position of the vertex in the two-dimensional plane of the epithelium. We evaluate this force using eq. 2.1:

$$\frac{\partial E}{\partial \boldsymbol{r}_0} = K \sum_{[\alpha]} (A_\alpha - A_{0\alpha}) \frac{\partial A_\alpha}{\partial \boldsymbol{r}_0} + \sum_{[i]} \Gamma_i \frac{\partial l_i}{\partial \boldsymbol{r}_0}. \tag{2.3}$$

The movement of a single vertex only effects the lengths and areas of its neighboring cells and edges, so the sums over all edges i and cells α become sums over neighboring edges $[i]$ and cells $[\alpha]$. In order to work out the derivatives it is helpful to introduce some new notation. Let $\boldsymbol{l}_i = \boldsymbol{r}_i - \boldsymbol{r}_0$ be the edge between the vertex at $\boldsymbol{r}_0$ and the adjacent vertex at position $\boldsymbol{r}_i$, as shown in fig. 2.2. The cell is taken to be in the two-dimensional x-y plane, with z normal to its surface. The change in the edge

length is

$$\frac{\partial l_i}{\partial \boldsymbol{r}_0} = -\hat{\boldsymbol{l}}_i, \tag{2.4}$$

where $\hat{\boldsymbol{l}}_i$ is a unit vector which points out from $\boldsymbol{r_0}$ along edge $\boldsymbol{l}_i$. The only change to the area of the adjacent cells comes from the triangle made by the two edges adjacent to the vertex (shown as the shaded region in fig. 2.2). The change in the area of this triangle is given by

$$\begin{aligned}\frac{\partial A}{\partial \boldsymbol{r}_0} &= \frac{\partial}{\partial \boldsymbol{r}_0}\left[\frac{1}{2}\hat{\boldsymbol{z}}\cdot(\boldsymbol{l}_1\times\boldsymbol{l}_2)\right] \\ &= \frac{1}{2}\hat{\boldsymbol{z}}\times(\boldsymbol{l}_2-\boldsymbol{l}_1).\end{aligned} \tag{2.5}$$

The change in the energy from a small movement of one vertex is then given by:

$$\frac{\partial E}{\partial \boldsymbol{r}_0} = \frac{K}{2}\sum_{[\alpha]}(A_\alpha - A_{0\alpha})\big[\hat{\boldsymbol{z}}\times(\boldsymbol{l}_{\alpha 2}-\boldsymbol{l}_{\alpha 1})\big] + \sum_{[i]} -\Gamma_i\hat{\boldsymbol{l}}_i, \tag{2.6}$$

where $\boldsymbol{l}_{\alpha 1}$ and $\boldsymbol{l}_{\alpha 2}$ are the two edges which are neighbors of both vertex $\boldsymbol{r_0}$ and cell α, ordered such that $\hat{\boldsymbol{z}}\cdot(\boldsymbol{l}_{\alpha 1}\times\boldsymbol{l}_{\alpha 2})>0$. Let the pressure in cell α be given by P_α. By definition,

$$P_\alpha = -\frac{\partial E}{\partial A_\alpha} = -K(A_\alpha - A_{0\alpha}). \tag{2.7}$$

Therefore the total force on any vertex $\boldsymbol{r}_0$ is given by

$$\boldsymbol{F}_{\boldsymbol{r}_0} = \sum_{[\alpha]}\frac{P_\alpha}{2}\big[\hat{\boldsymbol{z}}\times(\boldsymbol{l}_{\alpha 2}-\boldsymbol{l}_{\alpha 1})\big] + \sum_{[i]}\Gamma_i\hat{\boldsymbol{l}}_i. \tag{2.8}$$

Note that the direction of the pressure force from a given cell on a given vertex depends on the lengths of the two edges that the cell and vertex share; the force vector does not in general bisect the angle between the two edges. The second term gives the force from the tension on the neighboring edges.

Although we have derived eq. 2.8 from a particular energy function, its physical interpretation, in which each vertex is directly affected by the pressures of the surrounding cells and the tensions of the surrounding edges, suggests a wider validity. In fact, we can take eq. 2.8 to *define* a broader class of vertex models in which the pressure P_α in cell α and the tension Γ_i on edge i are given functions of variables that could include edge length and orientation, cell shape, cell types, protein concentrations, and so on. This class includes as a special case models that posit variants of the energy of eq. 2.1, like those that include a term quadratic in cell perimeter [10];

the pressures and tensions are then given as $P_\alpha = -\partial E/\partial A_\alpha$ and $\Gamma_i = \partial E/\partial l_i$. A vertex model defined directly in terms of the force on a vertex, however, also encompasses models that cannot be derived from any underlying global energy, including examples in which tensions depend on the protein concentration on an edge [20] or on edge orientation. In the remainder of this section and in section 2.3, we will formulate vertex dynamics and the conditions for local stability of fourfold vertices in terms of arbitrary pressures P_α and tensions Γ_i; starting in section 2.4, we will then turn to consider what our stability conditions imply for some specific, simple choices of the Γ_i.

2.2.2 Dynamics

To determine the motion of the vertices we make the common assumption that the vertices experience a drag force proportional to their velocity, so that

$$\boldsymbol{F}_{\boldsymbol{r}_0} = \mu \dot{\boldsymbol{r}}_0 \tag{2.9}$$

where μ is the drag constant. (Other assumptions about the form of dissipation have also been proposed [40] but will not be considered here.) An arbitrary vertex then moves according to

$$\frac{d\boldsymbol{r}_0}{dt} = \frac{1}{\mu}\left(\sum_{[\alpha]} \frac{P_\alpha}{2}\left[\hat{\boldsymbol{z}} \times (\boldsymbol{l}_{\alpha 2} - \boldsymbol{l}_{\alpha 1})\right] + \sum_{[i]} \Gamma_i \hat{\boldsymbol{l}}_i\right). \tag{2.10}$$

The sum over the neighboring cells $[\alpha]$ includes taking the difference between neighboring edges, which also appear in the second sum over the neighboring edges $[i]$. By expressing the forces in terms of the pressure difference across an edge, we can combine these sums into a single sum over neighboring edges [41]. To illustrate how the sums are merged let us consider an arbitrary vertex, which happens to be fourfold, with cells L, M, N, O, and edges 1, 2, 3, 4 as shown in fig. 2.3A. Explicitly writing out the force on the vertex from eq. 2.8 gives

$$\begin{aligned}\boldsymbol{F}_{\boldsymbol{r}_0} = {} & \Gamma_1 \hat{\boldsymbol{l}}_1 + \Gamma_2 \hat{\boldsymbol{l}}_2 + \Gamma_3 \hat{\boldsymbol{l}}_3 + \Gamma_4 \hat{\boldsymbol{l}}_4 \\ & + \frac{P_L}{2}\hat{\boldsymbol{z}} \times (\boldsymbol{l}_4 - \boldsymbol{l}_1) + \frac{P_M}{2}\hat{\boldsymbol{z}} \times (\boldsymbol{l}_1 - \boldsymbol{l}_2) \\ & + \frac{P_N}{2}\hat{\boldsymbol{z}} \times (\boldsymbol{l}_2 - \boldsymbol{l}_3) + \frac{P_O}{2}\hat{\boldsymbol{z}} \times (\boldsymbol{l}_3 - \boldsymbol{l}_4).\end{aligned} \tag{2.11}$$

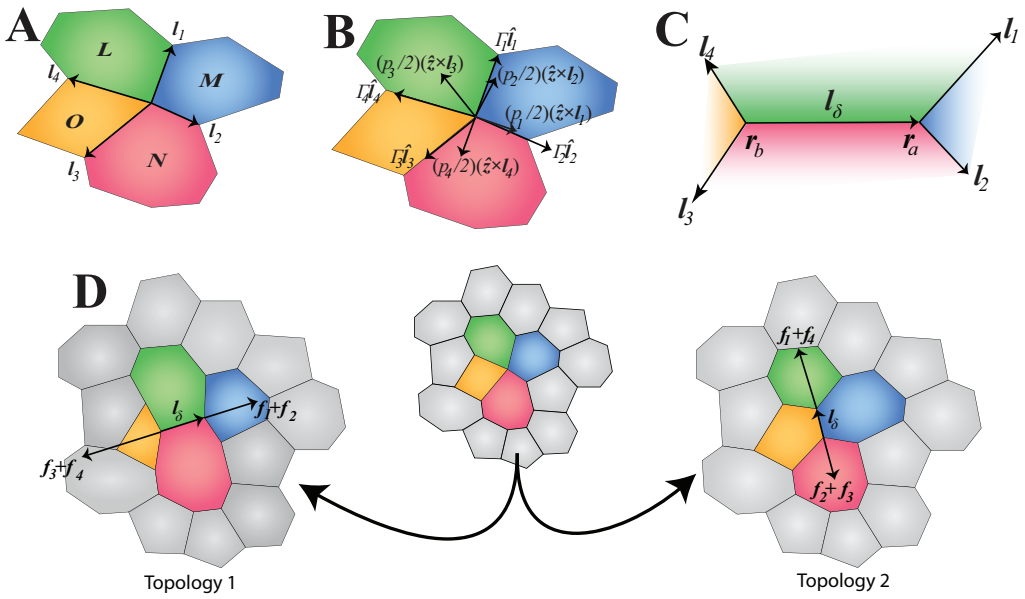

Figure 2.3: **A** : Cartoon of a fourfold vertex with neighboring cells L, M, N, O and edges $\boldsymbol{l}_1$, $\boldsymbol{l}_2$, $\boldsymbol{l}_3$, $\boldsymbol{l}_4$. Note that the direction of the edges is outward from the vertex, and the edges are numbered clockwise. **B** : Cartoon showing eight different forces acting on the fourfold vertex, two associated with each edge (eq. 2.15). The four edges produce a tension $\Gamma_i \hat{\boldsymbol{l}}_i$. The effect of the pressures from the four cells can be written in terms of the pressure differences across the edges; if p_i is the pressure difference across edge i, we can view the pressures as exerting a force $\frac{p_i}{2}(\hat{\boldsymbol{z}} \times \boldsymbol{l}_i)$ perpendicular to each edge. **C** : Cartoon of two threefold vertices which share an edge $\boldsymbol{l}_\delta$. As its length l_δ shrinks to zero, the vertices $\boldsymbol{r}_a$ and $\boldsymbol{r}_b$ will merge to form a single fourfold vertex. **D** : Cartoon of the resolution of a fourfold vertex. The fourfold vertex (center) can break apart into two threefold vertices in either of two topologies (left, right). In each case, we can associate a told force $\boldsymbol{f}_i$ with each edge i that includes tension and pressure jump contributions. In Topology 1 (left), forces $\boldsymbol{f}_3$ and $\boldsymbol{f}_4$ act one of the new vertices and forces $\boldsymbol{f}_1$ and $\boldsymbol{f}_2$ act on the other new vertex, so that the net force trying to extend the new edge $\boldsymbol{l}_\delta$ is $\boldsymbol{f}_1 + \boldsymbol{f}_2 - \boldsymbol{f}_3 - \boldsymbol{f}_4$; this is counteracted by the tension Γ_δ on the new edge (eq. 2.18). The situation is the same in Topology 2 (right), but with the edges paired differently.

We regroup the terms so that each term contains only one edge.

$$\begin{aligned}\boldsymbol{F}_{\boldsymbol{r}_0} &= \Gamma_1\hat{\boldsymbol{l}}_1 + \Gamma_2\hat{\boldsymbol{l}}_2 + \Gamma_3\hat{\boldsymbol{l}}_3 + \Gamma_4\hat{\boldsymbol{l}}_4 \\ &+ \frac{P_L - P_O}{2}(\hat{\boldsymbol{z}} \times \boldsymbol{l}_4) + \frac{P_M - P_L}{2}(\hat{\boldsymbol{z}} \times \boldsymbol{l}_1) \\ &+ \frac{P_N - P_M}{2}(\hat{\boldsymbol{z}} \times \boldsymbol{l}_2) + \frac{P_O - P_N}{2}(\hat{\boldsymbol{z}} \times \boldsymbol{l}_3)\end{aligned} \tag{2.12}$$

We can further simplify this expression by introducing the notation

$$p_i = P_\alpha - P_{\alpha'}, \tag{2.13}$$

where α and α' are the cells on either side of edge i, so that p_i represents the difference in pressure across an edge taken counterclockwise around the vertex. For example in the configuration show in fig. 2.3B, $p_1 = P_M - P_L$. In this simplified notation the force on our fourfold vertex is

$$\begin{aligned}\boldsymbol{F}_{\boldsymbol{r}_0} &= \Gamma_1\hat{\boldsymbol{l}}_1 + \frac{p_1}{2}(\hat{\boldsymbol{z}} \times \boldsymbol{l}_1) + \Gamma_2\hat{\boldsymbol{l}}_2 + \frac{p_2}{2}(\hat{\boldsymbol{z}} \times \boldsymbol{l}_2) \\ &+ \Gamma_3\hat{\boldsymbol{l}}_3 + \frac{p_3}{2}(\hat{\boldsymbol{z}} \times \boldsymbol{l}_3) + \Gamma_4\hat{\boldsymbol{l}}_4 + \frac{p_4}{2}(\hat{\boldsymbol{z}} \times \boldsymbol{l}_4).\end{aligned} \tag{2.14}$$

In general we can write the force on any vertex r_0 as

$$\boldsymbol{F}_{\boldsymbol{r}_0} = \sum_{[i]} \left[\Gamma_i\hat{\boldsymbol{l}}_i + \frac{p_i}{2}(\hat{\boldsymbol{z}} \times \boldsymbol{l}_i)\right]. \tag{2.15}$$

2.3 Fourfold vertex stability

In this section we will work out the criteria which a fourfold vertex must satisfy in order to be stable. As preparation, we first in section 2.3.1 examine the dynamics of neighboring threefold vertices as the length of their shared edge approaches zero. When the edge length reaches zero, a fourfold vertex can be formed; once formed, it can either persist as a fourfold vertex, or it can resolve into threefold vertices in one of two possible topologies (fig. 2.3D). We call a fourfold vertex stable if, when it is broken apart into two threefold vertices separated by a small shared edge $\boldsymbol{l}_\delta$, the forces on the two threefold vertices push them back together, causing the edge $\boldsymbol{l}_\delta$ to shrink to zero; this condition must hold for *both* possible resolution topologies. Section 2.3.2 makes this notion of stability more precise and addresses some technical questions that it raises. Finally, in section 2.3.3 we work out the criterion for a fourfold vertex to be stable against resolving in one topology. The criterion for the other topology

then follows immediately, and combining the two gives us our final set of stability conditions.

2.3.1 Dynamics of a small edge

Consider the dynamics of a pair of threefold vertices $\boldsymbol{r}_a$ and $\boldsymbol{r}_b$ which share an edge as shown in fig. 2.3C. Define the shared edge $\boldsymbol{l}_\delta$ as $\boldsymbol{l}_\delta = \boldsymbol{r}_a - \boldsymbol{r}_b$. This edge evolves according to

$$\dot{\boldsymbol{l}}_\delta = \dot{\boldsymbol{r}}_a - \dot{\boldsymbol{r}}_b = \frac{1}{\mu}\left(\boldsymbol{F}_{\boldsymbol{r}_a} - \boldsymbol{F}_{\boldsymbol{r}_b}\right), \tag{2.16}$$

where $\boldsymbol{F}_{\boldsymbol{r}_a}$ and $\boldsymbol{F}_{\boldsymbol{r}_b}$ are the forces on the two vertices given by eq. 2.15. This equation uses the conventions that the direction of $\boldsymbol{l}_i$ is taken outward from the vertex and that the pressures p_i are taken counterclockwise around the vertex. As we are now dealing with two vertices, we modify these conventions slightly to (arbitrarily) take vertex a as the reference vertex, so that the contribution from the tension on $\boldsymbol{l}_\delta$ is positive in the force on vertex a and negative in the force on vertex b. Similarly, we define the pressure difference p_δ across $\boldsymbol{l}_\delta$ to be taken counterclockwise around $\boldsymbol{r}_a$; because both p_δ and $\boldsymbol{l}_\delta$ then flip signs when forces on the vertex $\boldsymbol{r}_b$ are considered, the pressure difference across the shared edge contributes with the same sign to the forces on both vertices. Substituting,

$$\begin{aligned}
\dot{\boldsymbol{l}}_\delta = \frac{1}{\mu}\Bigg[&\Gamma_1\hat{\boldsymbol{l}}_1 + \frac{p_1}{2}(\hat{\boldsymbol{z}}\times\boldsymbol{l}_1) \\
&+ \Gamma_2\hat{\boldsymbol{l}}_2 + \frac{p_2}{2}(\hat{\boldsymbol{z}}\times\boldsymbol{l}_2) - \Gamma_\delta\hat{\boldsymbol{l}}_\delta + \frac{p_\delta}{2}(\hat{\boldsymbol{z}}\times\boldsymbol{l}_\delta)\Bigg] \\
-\frac{1}{\mu}\Bigg[&\Gamma_3\hat{\boldsymbol{l}}_3 + \frac{p_3}{2}(\hat{\boldsymbol{z}}\times\boldsymbol{l}_3) \\
&+ \Gamma_4\hat{\boldsymbol{l}}_4 + \frac{p_4}{2}(\hat{\boldsymbol{z}}\times\boldsymbol{l}_4) + \Gamma_\delta\hat{\boldsymbol{l}}_\delta + \frac{p_\delta}{2}(\hat{\boldsymbol{z}}\times\boldsymbol{l}_\delta)\Bigg],
\end{aligned} \tag{2.17}$$

where Γ_δ is the tension of the shared edge.

Define $\boldsymbol{f}_i$ as the contribution to the force associated with edge i, $\boldsymbol{f}_i = \Gamma_i\hat{\boldsymbol{l}}_i + \frac{p_i}{2}(\hat{\boldsymbol{z}}\times\boldsymbol{l}_i)$. Then the shared edge follows the equation of motion

$$\mu\dot{\boldsymbol{l}}_\delta = \boldsymbol{f}_1 + \boldsymbol{f}_2 - \boldsymbol{f}_3 - \boldsymbol{f}_4 - 2\Gamma_\delta\hat{\boldsymbol{l}}_\delta. \tag{2.18}$$

The forces $\boldsymbol{f}_1$ through $\boldsymbol{f}_4$ can in general depend on $\boldsymbol{l}_\delta$, and indeed on the positions of all the other vertices. Importantly, however, all four forces generically approach a finite, nonzero limit as $l_\delta \to 0$. (This contrasts with the situation in a standard linear

stability problem in which forces would go to zero linearly with l_δ.) As we discuss in more detail in the next section, when looking at vertex stability we will always be interested in the limit of small l_δ. To leading order in this limit, $\boldsymbol{f}_1$ through $\boldsymbol{f}_4$ can thus be evaluated at $\boldsymbol{l}_\delta = 0$ and treated as constants. Our subsequent development always assumes that this limit has been taken.

2.3.2 Defining fourfold vertex stability

To think about vertex stability, we would like to imagine, informally, that the fourfold vertex is constantly subject to noise or other small perturbations and that, from time to time, these perturbations cause it to break up into a pair of barely separated threefold vertices, with more or less random topology and orientation. If, for small enough perturbations, the fourfold vertex always re-forms, then we should call it stable. On the other hand, if the vertex dynamics ever tend to move the two newly formed threefold vertices apart, we would like to call the fourfold vertex unstable. Thus, to define stability more carefully, we ask what happens if, at some instant, a fourfold vertex is replaced by two threefold vertices whose separation $\boldsymbol{l}_\delta$ is infinitesmally small (and whose average position is at most infinitesmally different from the position of the original fourfold vertex). The separation $\boldsymbol{l}_\delta$ is then allowed to evolve according to eq. 2.18. If, when the magnitude $l_\delta = |\boldsymbol{l}_\delta|$ of the separation between the two vertices is small enough, its time derivative $\mathrm{d}l_\delta/\mathrm{d}t$ is always negative, for both possible resolution topologies and for any choice of orientation $\hat{\boldsymbol{l}}_\delta$, then the fourfold vertex is stable. If there is any choice of separation orientation $\hat{\boldsymbol{l}}_\delta$ and topology for which $\mathrm{d}l_\delta/\mathrm{d}t$ remains positive for arbitrarily small l_δ, then the vertex is unstable. Finally, if, as l_δ goes to zero, $\mathrm{d}l_\delta/\mathrm{d}t$ approaches zero for one or more choices of $\hat{\boldsymbol{l}}_\delta$ but is otherwise negative, then the fourfold vertex is either marginally stable or marginally unstable, and the calculation must be pursued to higher order in l_δ than we consider here.

Several aspects of this definition of stability deserve further comment. First, the positions of vertices, and hence the forces in eq. 2.18 and the stability of a given fourfold vertex, may change with time. Thus, vertex stability is an instantaneous notion, and we should really talk about the stability or instability of a vertex at some time t_0; in particular, in the most general case it is possible for a fourfold vertex to be stable at time t_0 but then, because of the natural time evolution of the cell packing and without any change in system parameters, to go unstable at some later time $t_1 > t_0$. (Of course, often we will be interested in cell packings that have reached a local mechanical equilibrium and are no longer changing with time, in which case these

concerns do not apply.) Second, to determine stability at some time t_0 in practice, we evaluate the forces in eq. 2.18 as if both new vertices were located exactly at the position of the fourfold vertex in question and all other vertices were frozen at their positions at time t_0. This is appropriate because of the observation that all of the force terms in eq. 2.18 generically have finite, nonzero limits as l_δ approaches zero at fixed orientation $\hat{\boldsymbol{l}}_\delta$. Except in the marginal case described in the preceding paragraph, for small enough l_δ these finite terms must dominate any corrections due to infinitesmal deviations of vertices from their positions at time t_0. Finally, the same reasoning explains why we can focus exclusively on the dynamics of the separation $\boldsymbol{l}_\delta$ and can ignore the possibility of collective instability modes that involve the motion of many vertices: As long as $\mathrm{d}l_\delta/\mathrm{d}t$ is finite and nonzero as $l_\delta \to 0$, infinitesmal perturbations to other vertex positions can change its magnitude infinitesmally, but cannot affect its sign.

2.3.3 Stability conditions

In accordance with the notion of stability described in the previous section, we now imagine that the fourfold vertex momentary splits into two infinitesimally close threefold vertices as shown in fig. 2.3D. In order for the vertex to be stable we want the vertex dynamics to force the two vertices back together. We know that in general the vertices' shared edge evolves according to eq. 2.18. Let $\boldsymbol{\mathcal{F}} = \boldsymbol{f}_1 + \boldsymbol{f}_2 - \boldsymbol{f}_3 - \boldsymbol{f}_4$, and let θ be the angle between $\boldsymbol{\mathcal{F}}$ and the edge $\boldsymbol{l}_\delta$. The time derivatives of the length l_δ and direction θ of the newly formed edge are given by

$$l_\delta \dot{\theta} = -\frac{\mathcal{F}}{\mu} \sin\theta \tag{2.19}$$

$$\dot{l}_\delta = \frac{\mathcal{F}}{\mu} \cos\theta - \frac{2\Gamma_\delta}{\mu},, \tag{2.20}$$

where $\mathcal{F} = |\boldsymbol{\mathcal{F}}|$. From eq. 2.20, we conclude that the shared edge grows the fastest when $\theta = 0$. Therefore it is sufficient to look at new edges which form along the line of force $\boldsymbol{\mathcal{F}}$ to prove stability. In this case eq. 2.20 reduces to

$$\dot{l}_\delta = \frac{\mathcal{F} - 2\Gamma_\delta}{\mu}. \tag{2.21}$$

The edge shrinks whenever

$$\Gamma_\delta > \frac{\mathcal{F}}{2}. \tag{2.22}$$

We note in passing that when the forces are derived from an energy, we can also

see that eq. 2.22 must be the stability criterion by looking at the energy. The change in energy to lowest order in $\boldsymbol{l}_\delta$ is $\delta E = -\boldsymbol{\mathcal{F}} \cdot \boldsymbol{l}_\delta + 2\Gamma_\delta |\boldsymbol{l}_\delta|$, so that the change in energy is positive whenever $\Gamma_\delta > \frac{\boldsymbol{\mathcal{F}} \cdot \boldsymbol{l}_\delta}{2 l_\delta}$. The right hand side is minimized when $\boldsymbol{\mathcal{F}}$ is in the direction of the new edge, and so the vertex is stable whenever $\Gamma_\delta > \frac{\mathcal{F}}{2}$.

It is important to remember that the vertex can resolve in two different topologies (fig. 2.3D) which have different $\boldsymbol{\mathcal{F}}$. Therefore, for the vertex to be stable, both of the following conditions must be met:

$$\Gamma_\delta \geq \frac{|\boldsymbol{f}_1 + \boldsymbol{f}_2 - \boldsymbol{f}_3 - \boldsymbol{f}_4|}{2} \tag{2.23}$$

$$\Gamma_\delta \geq \frac{|\boldsymbol{f}_2 + \boldsymbol{f}_3 - \boldsymbol{f}_1 - \boldsymbol{f}_4|}{2} \tag{2.24}$$

The condition in eq. 2.23 ensures that the fourfold vertex is stable against resolution into two threefold vertices in topology 1 (fig. 2.3D), by enforcing the stability criterion derived in eq. 2.22. Similarly, condition 2.24 ensures that the vertex is stable against resolution in topology 2 (fig. 2.3D).

Except in section 2.6.1, we will primarily be interested in what follows in the stability of fourfold vertices that are in mechanical equilibrium—that is, on which the net force is zero. (Because of our assumption of local dissipation at the vertex, eq. 2.9, mechanical equilibrium of a vertex is equivalent to its being stationary.) If this additional condition holds, then $\boldsymbol{f}_1 + \boldsymbol{f}_2 + \boldsymbol{f}_3 + \boldsymbol{f}_4 = \boldsymbol{0}$, and one can replace $-\boldsymbol{f}_3 - \boldsymbol{f}_4$ by $\boldsymbol{f}_1 + \boldsymbol{f}_2$ and $-\boldsymbol{f}_1 - \boldsymbol{f}_4$ by $\boldsymbol{f}_2 + \boldsymbol{f}_3$ in eqs. 2.23–2.24 (thereby removing all dependence on $\boldsymbol{f}_4$ in both inequalities). The two inequalities can then be rewritten explicitly in terms of the p_i and Γ_i as

$$\Gamma_\delta \geq |\Gamma_1 \hat{\boldsymbol{l}}_1 + \frac{p_1 l_1}{2}(\hat{\boldsymbol{z}} \times \hat{\boldsymbol{l}}_1) + \Gamma_2 \hat{\boldsymbol{l}}_2 + \frac{p_2 l_2}{2}(\hat{\boldsymbol{z}} \times \hat{\boldsymbol{l}}_2)| \tag{2.25}$$

$$\Gamma_\delta \geq |\Gamma_3 \hat{\boldsymbol{l}}_3 + \frac{p_3 l_3}{2}(\hat{\boldsymbol{z}} \times \hat{\boldsymbol{l}}_3) + \Gamma_2 \hat{\boldsymbol{l}}_2 + \frac{p_2 l_2}{2}(\hat{\boldsymbol{z}} \times \hat{\boldsymbol{l}}_2)|. \tag{2.26}$$

Similarly, the equation of mechanical equilibrium takes the form

$$\begin{aligned} 0 = {} & \Gamma_1 \hat{\boldsymbol{l}}_1 + \frac{p_1 l_1}{2}(\hat{\boldsymbol{z}} \times \hat{\boldsymbol{l}}_1) + \Gamma_2 \hat{\boldsymbol{l}}_2 + \frac{p_2 l_2}{2}(\hat{\boldsymbol{z}} \times \hat{\boldsymbol{l}}_2) \\ & + \Gamma_3 \hat{\boldsymbol{l}}_3 + \frac{p_3 l_3}{2}(\hat{\boldsymbol{z}} \times \hat{\boldsymbol{l}}_3) + \Gamma_4 \hat{\boldsymbol{l}}_4 + \frac{p_4 l_4}{2}(\hat{\boldsymbol{z}} \times \hat{\boldsymbol{l}}_4). \end{aligned} \tag{2.27}$$

A physical interpretation of these stability conditions is that eq. 2.25 and eq. 2.26 require that the tension on the new edge is high enough that it counteracts the forces from the tensions and pressure differences across the edges. This stops the vertex from

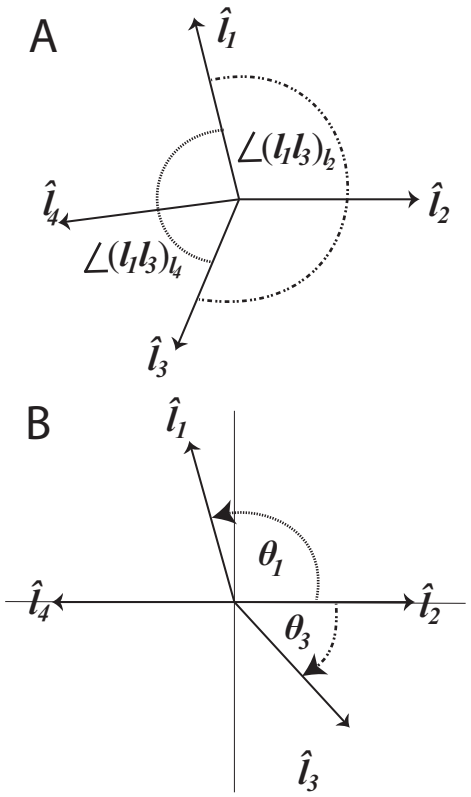

Figure 2.4: **A** : The two angles $\angle(l_1 l_3)_{l_2}$ and $\angle(l_1 l_3)_{l_4}$ between the non-adjacent edges $\boldsymbol{l}_1$ and $\boldsymbol{l}_3$ are shown. The quantities $\angle(l_i l_j)_{l_k}$ are defined as the unsigned magnitudes of the angles, so $\angle(l_i l_j)_{l_k} = \angle(l_j l_i)_{l_k}$. The angles $\angle(l_1 l_3)_{l_2}$ and $\angle(l_1 l_3)_{l_4}$ together make a full circle, implying $\angle(l_1 l_3)_{l_2} + \angle(l_1 l_3)_{l_4} = 2\pi$. **B** : The angles θ_1 and θ_3 are defined in the usual manner as the signed angles between the positive x axis (which here coincides with $\boldsymbol{l}_2$) and, respectively, $\boldsymbol{l}_1$ and $\boldsymbol{l}_3$. Hence, as drawn, $\theta_1 > 0$ and $\theta_3 < 0$.

resolving in either of the possible topologies. Equation 2.27 constrains the vertex to be in mechanical equilibrium.

2.4 No stable, stationary fourfold vertices exist in Plateau's model

In section 2.5 we will show that there are no stable, stationary fourfold states under the condition that all of the edges have the same tension, and in section 2.6 we will give some examples of stable fourfold vertices that arise when we lift this requirement. In this section, we first work through the simplest special case of eqs. 2.25-2.27 to give the reader some intuition about the main proof presented in section 2.5. We will use the same structure for our proof in both sections.

The simplest possible situation is one in which $p_i = 0$ and $\Gamma_i = \Gamma$. This is

equivalent to Plateau's model of a dry foam, because in Plateau's model the pressure does not affect the motion of the vertices directly but instead changes the angle of vertices' neighboring edges. (Equivalently, changes to cell areas when a new edge is created are higher order in δ than are changes in edge lengths and thus can be neglected in calculations of vertex stability in Plateau's model [14].)

To begin the proof that stationary vertices cannot be stable in this case we first rewrite the criteria for stability from eqs. 2.25-2.27. After dividing by Γ, we have

$$1 \geq |\hat{\boldsymbol{l}}_1 + \hat{\boldsymbol{l}}_2| \tag{2.28}$$

$$1 \geq |\hat{\boldsymbol{l}}_3 + \hat{\boldsymbol{l}}_2| \tag{2.29}$$

$$0 = \hat{\boldsymbol{l}}_1 + \hat{\boldsymbol{l}}_2 + \hat{\boldsymbol{l}}_3 + \hat{\boldsymbol{l}}_4. \tag{2.30}$$

As in fig. 2.3A, we label the edges in the clockwise direction from 1 through 4, and we assume that each pair of successively numbered edges bounds a single cell: cell M lies between $\boldsymbol{l}_1$ and $\boldsymbol{l}_2$, cell N lies between $\boldsymbol{l}_2$ and $\boldsymbol{l}_3$, and so on. For our model to be physically reasonable we cannot have two or more cells occupying the same space, so we must reject any configurations in which edge 1 moves through edge 2 in such a way that cell M inverts and partially overlaps cell N. In order to avoid such unphysical overlap, we require that the ordering of the edges around the vertex remain fixed, and thus in particular that the labels 1 through 4 always appear in increasing order in the clockwise direction.

As shown in fig. 2.4A, the non-adjacent edges 1 and 3 are separated by two angles, one encompassing edge 2 and the other encompassing edge 4, which together make up a full circle. We call the (necessarily positive) magnitudes of these two angles $\angle(l_1 l_3)_{l_2}$ and $\angle(l_1 l_3)_{l_4}$; more generally, we refer to the magnitude of the angle between non-adjacent edges $\boldsymbol{l}_i$ and $\boldsymbol{l}_j$ that encompasses $\boldsymbol{l}_k$ as $\angle(l_i l_j)_{l_k}$.

To show that a fourfold vertex cannot be stable in the Plateau model, begin by taking an arbitrary pair of non-adjacent edges $\boldsymbol{l}_i$ and $\boldsymbol{l}_j$. Either $\angle(l_i l_j)_{l_k} \leq \pi$ or $\angle(l_i l_j)_{l_m} \leq \pi$ (where $\boldsymbol{l}_k$ and $\boldsymbol{l}_m$ are the other two edges at the vertex); we choose without loss of generality to label the edges so that $\angle(l_i l_j)_{l_k} \leq \pi$. We may then apply a rotation followed by (if needed) a reflection to the fourfold vertex and relabel the edges so that $\angle(l_i l_j)_{l_k}$ becomes $\angle(l_1 l_3)_{l_2}$ and $\hat{\boldsymbol{l}}_2 = \hat{\boldsymbol{x}}$ (fig. 2.5 and section 2.5.2). Let θ_i be the signed angle between edge i and the x axis, as shown in fig 2.4B. Then $\theta_1 > 0$ and $\theta_3 < 0$. (Note also that because $\angle(l_1 l_3)_{l_2} \leq \pi$ by assumption, neither θ_1 nor θ_3 can have magnitude larger than π.) We will continue to use this convention in section 2.5.

The next step is to convert eqs. 2.28 and 2.29 to polar coordinates,

$$1 \geq (\cos\theta_1 + 1)^2 + (\sin\theta_1)^2 \tag{2.31}$$

$$1 \geq (\cos\theta_3 + 1)^2 + (\sin\theta_3)^2\,, \tag{2.32}$$

and to solve the system of inequalities. In this case we can immediately deduce that, for these conditions to hold and the vertex to be stable, we must have $\theta_1 \geq 2\pi/3$ and $\theta_3 \leq -2\pi/3$. It follows that $\angle(l_1 l_3)_{l_2} = \theta_1 - \theta_3 \geq 4\pi/3$, which contradicts our initial assumption that $\angle(l_1 l_3)_{l_2} \leq \pi$. Hence, the vertex must be unstable.

Our proof that there are no stable states in the equal tension vertex model will follow the same basic structure. First we will express the conditions 2.25-2.26 in polar coordinates. We will solve the resulting system of inequalities to get bounds on the angle between any two non-adjacent edges $\angle(l_i l_j)_{l_k}$. We will then show that the given bounds lead to a contradiction.

2.5 No stable, stationary fourfold vertices exist in the equal tension vertex model

Throughout section 2.5 we will work with a special case of the vertex model, which we call the equal tension vertex model, which shares important features with the Plateau model of foams. In the equal tension vertex model, as in the Plateau model, every edge has the same tension $\Gamma_i = \Gamma$; unlike the Plateau model, however, the equal tension vertex model does not put any restrictions on the cell pressures P_α. In this section, we consider only fourfold vertices that are stationary and in mechanical equilibrium.

Our argument that such fourfold vertices can never be stable in the equal tension model proceeds as follows: In section 2.5.1 we introduce the variables ρ_i, which are dimensionless ratios of an edge's length, tension, and pressure difference. This reduces the number of variables in the problem to eight (four edge directions and four ρ_i). In section 2.5.2, we express the stability conditions 2.25-2.26 in polar coordinates and use the symmetries of the problem to reduce the number of free variables to seven. In section 2.5.3 we analyze the resulting system of inequalities, concluding that fourfold vertices are unstable unless $\angle(l_i l_j)_{l_k} = \pi$ for any choices of non-adjacent edges $\boldsymbol{l}_i$ and $\boldsymbol{l}_j$ and intervening edge $\boldsymbol{l}_k$. Finally, in section 2.5.4 we show that if $\angle(l_i l_j)_{l_k} = \pi$ for all pairs of non-adjacent edges, it is impossible to satisfy all three stability and equilibrium conditions 2.25-2.27. Thus, no stable, stationary fourfold vertices are

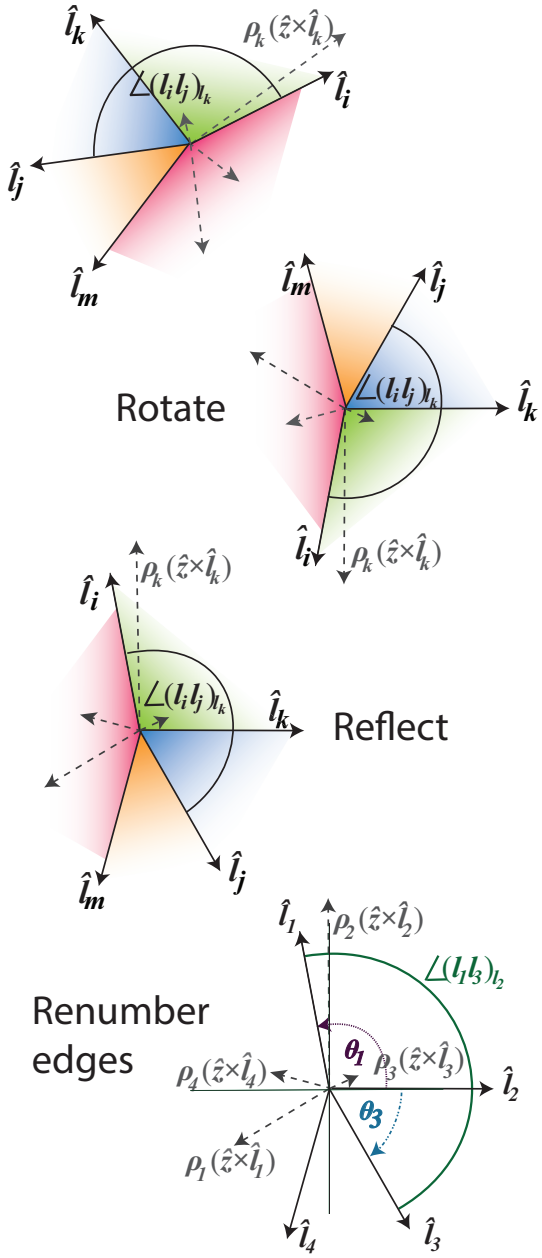

Figure 2.5: Cartoon of the procedure in section 2.5.2 to exploit the symmetries of the problem in order to reduce the number of free variables. An arbitrary angle $\angle(l_i l_j)_{l_k}$ between non adjacent edges can be transformed under rotations such that $\boldsymbol{l}_k$ lies on the positive x axis; if needed, a reflection through the x axis ensures that $\rho_k(\hat{\boldsymbol{z}} \times \hat{\boldsymbol{l}}_k)$ lies on the positive y axis and thus that $\rho_k \geq 0$. (Note that ρ_k changes sign under reflections.) Finally, the edges can be renumbered from 1 to 4 in the clockwise direction. θ_1 and θ_3 are the signed angles that $\hat{\boldsymbol{l}}_1$ and $\hat{\boldsymbol{l}}_3$ make with the positive x axis, as shown, so that the magnitude of the angle between $\hat{\boldsymbol{l}}_1$ and $\hat{\boldsymbol{l}}_3$ is $\angle(l_1 l_3)_{l_2} = \theta_1 - \theta_3$.

possible in the equal tension model.

2.5.1 Streamlining notation

We begin by writing a more compact version of the general stability conditions given in eqs. 2.25 - 2.26. Let

$$\rho_i = \frac{p_i |\boldsymbol{l}_i|}{2\Gamma}, \tag{2.33}$$

be a scalar which is proportional to the force exerted by the pressure difference across edge i. Recall that the pressure difference is taken counterclockwise around vertex $\boldsymbol{r}_a$ (fig. 2.3C), so the sign of ρ depends on which neighboring cell has the higher pressure. The stability criteria can then be expressed as

$$1 \geq |\hat{\boldsymbol{l}}_1 + \rho_1(\hat{\boldsymbol{z}} \times \hat{\boldsymbol{l}}_1) + \hat{\boldsymbol{l}}_2 + \rho_2(\hat{\boldsymbol{z}} \times \hat{\boldsymbol{l}}_2)| \tag{2.34}$$

$$1 \geq |\hat{\boldsymbol{l}}_3 + \rho_3(\hat{\boldsymbol{z}} \times \hat{\boldsymbol{l}}_3) + \hat{\boldsymbol{l}}_2 + \rho_2(\hat{\boldsymbol{z}} \times \hat{\boldsymbol{l}}_2)| \tag{2.35}$$

$$\begin{aligned} 0 = {} & \hat{\boldsymbol{l}}_1 + \rho_1(\hat{\boldsymbol{z}} \times \hat{\boldsymbol{l}}_1) + \hat{\boldsymbol{l}}_2 + \rho_2(\hat{\boldsymbol{z}} \times \hat{\boldsymbol{l}}_2) \\ & + \hat{\boldsymbol{l}}_3 + \rho_3(\hat{\boldsymbol{z}} \times \hat{\boldsymbol{l}}_3) + \hat{\boldsymbol{l}}_4 + \rho_4(\hat{\boldsymbol{z}} \times \hat{\boldsymbol{l}}_4). \end{aligned} \tag{2.36}$$

By absorbing the lengths of the edges into the coefficients ρ_i, the problem is now poised entirely in terms of unit vectors. The problem is reduced to eight variables: the four angles of the edges with respect to the x-axis $\theta_1, \theta_2, \theta_3, \theta_4$, and the four ρ coefficients.

As an aside, if we further assume the pressures have the simple form of eq. 2.7 and express the ρ_i in terms of the areas of the cells,

$$\begin{aligned} \rho_i &= \frac{p_i l_i}{2\Gamma} \\ &= \frac{(P_\alpha - P_{\alpha'})\, l_i}{2\Gamma} \\ &= \frac{K l_i \left[(A_{\alpha'} - A_\alpha) + (A_{0\alpha} - A_{0\alpha'})\right]}{2\Gamma}, \end{aligned} \tag{2.37}$$

it becomes clear that the preferred area $A_{0\alpha}$ of the cells does not affect the stability in the common case in which $A_{0\alpha}$ is the same for all cells.

2.5.2 Exploiting symmetries

The stability criteria 2.34 and 2.35 both contain terms with edge two. We would like to use the symmetries of the problem to fix this shared edge and reduce the

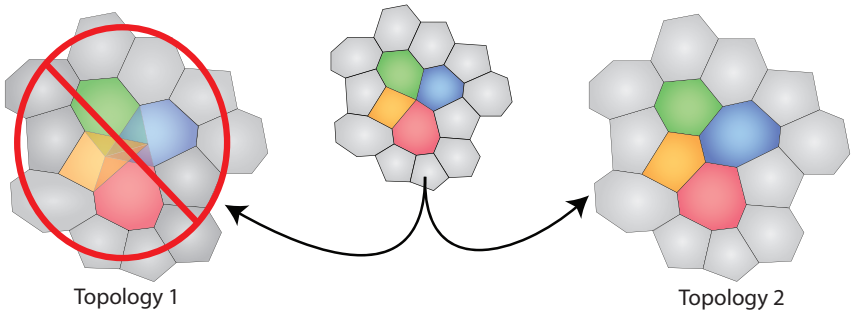

Figure 2.6: Cartoon of the unphysical resolution of a fourfold vertex due to large pressure effects. In the left topology the two resulting threefold vertices are pushed through each other by the pressure of the neighboring cells. This creates a physically impossible state in which cells overlap.

number of free variables. The problem has rotation and reflection symmetry as well as arbitrary edge labels.

Let us look at an arbitrary pair of non-adjacent edges $\boldsymbol{l}_i$ and $\boldsymbol{l}_j$ (fig. 2.5, top). Either $\angle(l_i l_j)_{l_k} \leq \pi$ or $\angle(l_i l_j)_{l_m} \leq \pi$, since the two angles together make up a full circle. Without loss of generality, label the edges so that $\angle(l_i l_j)_{l_k} \leq \pi$. We can then use the problem's rotational symmetry to impose $\hat{\boldsymbol{l}}_k = \hat{\boldsymbol{x}}$. It will be useful later in the proof to place restrictions on the sign of ρ_k. If ρ_k is initially negative we can reflect the system about the x axis, as shown in fig. 2.5. This reflection has the effect of changing the sign of ρ_k, so that we can impose that $\rho_k \geq 0$. We are free to relabel $\boldsymbol{l}_k$ as $\boldsymbol{l}_2$ and to relabel the rest of the edges in order clockwise from 1 to 4. Since we can perform this procedure starting from any pair of non-adjacent edges $\boldsymbol{l}_i$ and $\boldsymbol{l}_j$, our arguments in the remainder of this section hold for all pairs of non-adjacent edges.

2.5.3 Bounds on the angle between non-adjacent edges

We next turn to the central problem of determining the implications of the stability criteria of eqs. 2.34 and 2.35 for the angles between edges. Let θ_i be the signed angle between an edge and the x-axis, where θ_1 is positive and θ_3 is negative due to the clockwise labeling of edges as shown in fig. 2.5. The stability conditions can be written in terms of the θ_i and ρ_i as

$$1 \geq 2 + \rho_1^2 + \rho_2^2 + 2(1 + \rho_1\rho_2)\cos\theta_1 + 2(\rho_2 - \rho_1)\sin\theta_1 \tag{2.38}$$

$$1 \geq 2 + \rho_3^2 + \rho_2^2 + 2(1 + \rho_3\rho_2)\cos\theta_3 + 2(\rho_2 - \rho_3)\sin\theta_3. \tag{2.39}$$

Our goal is to put a lower bound on the angle $\angle(l_1 l_3)_{l_2} = \theta_1 - \theta_3$. An important property of our system of inequalities is that the conditions on θ_1 are completely independent of the value of θ_3 and vice versa. Neither variable depends on the other, but they both depend on ρ_2. This allows us to break the overall optimization problem of finding the minimum value of $\theta_1 - \theta_3$ into two separate sub-problems: finding the minimum value of θ_1 as a function of ρ_2 and finding the maximum value of θ_3 as a function of ρ_2.

For our first optimization problem, we would like to find the minimum value of θ_1 that can be obtained by varying ρ_1 for an arbitrary, fixed value of ρ_2 and subject to the constraint of eq. 2.38. Due to the inequality constraint we cannot use the method of Lagrange multipliers to solve this optimization problem. Instead, we use its generalization to the case where the optimum can occur either on the boundary of a region or within that region, the Karush-Kuhn-Tucker conditions [42, 43]. Let the function to be maximized be $h(\theta_1, \rho_1) = -\theta_1$ and the constraining function be $g(\theta_1, \rho_1) = 1 + \rho_1^2 + \rho_2^2 + 2(1 + \rho_1\rho_2)\cos\theta_1 + 2(\rho_2 - \rho_1)\sin\theta_1 \leq 0$. The optimality conditions are then

$$\nabla h(\theta_1, \rho_1) - \lambda \nabla g(\theta_1, \rho_1) = 0 \tag{2.40}$$

$$\lambda[g(\theta_1, \rho_1) - 0] = 0 \tag{2.41}$$

$$g(\theta_1, \rho_1) \leq 0 \tag{2.42}$$

$$\lambda \geq 0, \tag{2.43}$$

where the gradient is taken with respect to the variables θ_1 and ρ_1, and $\lambda \in \mathbb{R}$ is the the Karush-Kuhn-Tucker multiplier. This produces the system of equations

$$0 = -1 + 2\lambda\left[(1 + \rho_1\rho_2)\sin\theta_1 + (\rho_1 - \rho_2)\cos\theta_1)\right] \tag{2.44}$$

$$0 = -2\lambda\left(\rho_1 + \rho_2\cos\theta_1 - \sin\theta_1\right) \tag{2.45}$$

$$0 = \lambda g(\theta_1, \rho_1) \tag{2.46}$$

$$0 \geq g(\theta_1, \rho_1) \tag{2.47}$$

$$0 \leq \lambda \tag{2.48}$$

In section 2.5.2, we showed that we can use symmetry operations to make ρ_2 positive without loss of generality. We also chose to focus on the smaller of the two angles between a pair of non-adjacent edges, so that $\angle(l_1 l_3)_{l_2} \leq \pi$, and we numbered the edges clockwise as show in fig. 2.5 (bottom). This gives additional constraints on the

solution:

$$0 \leq \rho_2, \tag{2.49}$$

$$0 < \theta_1 \leq \pi, \tag{2.50}$$

where θ_1 cannot be zero because edges must be separated by cells of non-zero area. The solution to the full system of equations is

$$\theta_1 = \arctan\left[-\rho_2, 1\right]. \tag{2.51}$$

where $\arctan\left[x, y\right]$ is the angle whose tangent is y/x and that lies the quadrant is given by the signs of x and y.

We may now independently optimize θ_3 for an arbitrary value of ρ_2. Let the function to be maximized be $h(\theta_3, \rho_3) = \theta_3$ and the constraining function be $g(\theta_3, \rho_3) = 1 + \rho_3^2 + \rho_2^2 + 2(1 + \rho_3\rho_2)\cos\theta_3 + 2(\rho_2 - \rho_3)\sin\theta_3 \leq 0$. The optimality conditions are the same as eqs. 2.40-2.43, which produces the system of equations:

$$0 = 1 + 2\lambda\left[(1 + \rho_3\rho_2)\sin\theta_3 + (\rho_3 - \rho_2)\cos\theta_3\right] \tag{2.52}$$

$$0 = -2\lambda\left(\rho_3 + \rho_2\cos\theta_3 - \sin\theta_3\right) \tag{2.53}$$

$$0 = \lambda g(\theta_3, \rho_3) \tag{2.54}$$

$$0 \geq g(\theta_3, \rho_3) \tag{2.55}$$

$$0 \leq \lambda \tag{2.56}$$

We have an additional two constraints given by the way we set up the problem:

$$0 \leq \rho_2, \tag{2.57}$$

$$0 > \theta_3 \geq -\pi. \tag{2.58}$$

This system of equations has two real solutions:

$$\theta_3 = \arctan\left[\frac{-2 - \rho_2\sqrt{\rho_2^2 - 3}}{1 + \rho_2^2}, \frac{-2\rho_2 + \sqrt{\rho_2^2 - 3}}{1 + \rho_2^2}\right], \tag{2.59}$$

$$\theta_3 = \arctan\left[\rho_2, -1\right]. \tag{2.60}$$

For all $\rho_2 \in [0, \sqrt{3}]$, the solution of eq. 2.60 is greater than that of eq. 2.59, and elsewhere the solution given by eq. 2.59 has a non-zero imaginary part, so the true maximum θ_3 is given by the solution in eq. 2.60. Subtracting our two independently

optimized solutions we have that the minimum possible value of the angle $\angle(l_1 l_3)_{l_2}$ is

$$\begin{aligned}\angle(l_1 l_3)_{l_2} &= \theta_1 - \theta_3 \\ &\geq \arctan\left[-\rho_2, 1\right] - \arctan\left[\rho_2, -1\right] \\ &= \pi,\end{aligned} \tag{2.61}$$

where the last identity holds for all nonnegative ρ_2. As we began by choosing $\angle(l_1 l_3)_{l_2} \leq \pi$, either $\angle(l_1 l_3)_{l_2} = \pi$ or the fourfold vertex is unstable. Moreover, because $\angle(l_1 l_3)_{l_2} + \angle(l_1 l_3)_{l_4} = 2\pi$, stability then also implies that $\angle(l_1 l_3)_{l_4} = \pi$. The same holds for any pair of nonadjacent edges, by the argument in section 2.5.2. In other words, the fourfold vertex is unstable unless $\hat{\boldsymbol{l}}_1 = -\hat{\boldsymbol{l}}_3$ and $\hat{\boldsymbol{l}}_2 = -\hat{\boldsymbol{l}}_4$. In the next section, we show that under these assumptions it is impossible to satisfy all three stability conditions 2.34-2.36.

2.5.4 Finding a contradiction when non-adjacent edges have 180° separation

Suppose that $\hat{\boldsymbol{l}}_1 = -\hat{\boldsymbol{l}}_3$ and $\hat{\boldsymbol{l}}_2 = -\hat{\boldsymbol{l}}_4$. It is easy to show that condition 2.36 (mechanical equilibrium) is then only satisfied when $\rho_1 = \rho_3$ and $\rho_2 = \rho_4$. Since $\angle(l_1 l_3)_{l_2} = \theta_1 - \theta_3 = \pi$, $\theta_3 = \theta_1 - \pi$. Substituting this equality and $\rho_3 = \rho_1$ into eqs. 2.38 and 2.39 yields

$$0 \geq 1 + \rho_1^2 + \rho_2^2 + 2(1 + \rho_1\rho_2)\cos\theta_1 + 2(\rho_2 - \rho_1)\sin\theta_1 \tag{2.62}$$

$$0 \geq 1 + \rho_1^2 + \rho_2^2 - 2(1 + \rho_1\rho_2)\cos\theta_1 - 2(\rho_2 - \rho_1)\sin\theta_1. \tag{2.63}$$

Together these two conditions imply

$$0 \geq 1 + \rho_1^2 + \rho_2^2, \tag{2.64}$$

which is a contradiction because the right-hand side is always greater than one. Thus, there can be no stable, stationary fourfold vertices in the equal tension vertex model.

Before moving on from the stationary, equal tension case, we should finally note that, strictly speaking, our proof of instability applies to a vertex model that allows cell overlap. Although such a situation is not common in practice, it can occur that pressure differences between cells are large enough that they overwhelm the tensions and cause the fourfold vertex to try to resolve by pushing the cells through each other as shown in fig. 2.6. If such resolution with overlap is forbidden, the vertex's stability

increases, and we cannot at the moment rigorously rule out the possibility that in this case fourfold vertices could become stable in the equal tension model. In reality, of course, if cell overlap is a concern then there is a good chance the model is being studied in a pathological parameter regime.

2.6 Examples of modifications that allow for stable fourfold vertices

It was already known that Plateau's model of soap foams, on which the vertex model is based, does not allow for stable fourfold vertices. In the last section we gave a proof that, even with the addition of pressure effects which arise in vertex models with straight edges, there are still no stable fourfold vertices. Given that fourfold vertices are seen in various epithelial tissues [16, 27, 28, 29, 30], one might naturally wonder what extensions of the model would allow stable fourfold vertices to form. One well-studied example occurs in the avian oviduct epithelium, where two different types of cells are arranged in a checkerboard pattern with edges between like cell types having higher tension [16]. In this section we will give two examples of modifications which allow for stable fourfold vertices in epithelia even when only a single cell type is present. This gives us some insight into what additional biological mechanisms might exist in epithelia which are not present in simple foams and which could lead to higher order vertices.

2.6.1 Vertices not in mechanical equilibrium

So far we have only considered fourfold vertices which are in mechanical equilibrium. If the vertex is moving relative to the epithelial tissue, eq. 2.27 no longer holds, and the forces associated with the four edges can become very unbalanced. It turns out that the vertex model then does admit stable fourfold vertices. An example of such a stable state is given in fig. 5.3. The observation that moving fourfold vertices tend to be more stable than their stationary counterparts might explain why they have been observed to persist in tissues undergoing rapid morphogenetic movements [44, 29].

2.6.2 Anisotropic tension

In previous sections, we investigated vertex stability in a model in which all edges have the same tension. Unlike foams, however, cells can regulate their tensions so

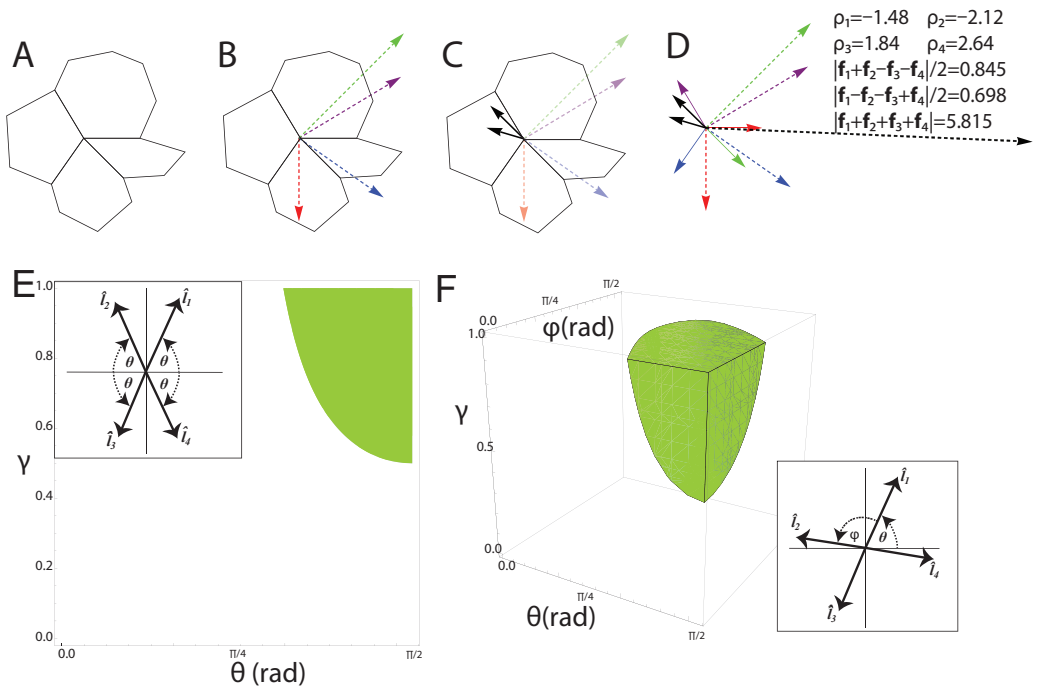

Figure 2.7: Situations in which fourfold vertices can become stable. **A**–**B** : Example of a fourfold vertex stabilized through movement. All $\Gamma_i = \Gamma = 1$, so that the tension force from each edge is 1. The vertex is then stable for the quoted values of the pressure differences **A** : The fourfold vertex and adjacent cells. **B** : The force from the pressure differences across each edge $\rho_i(\hat{\boldsymbol{z}} \times \hat{\boldsymbol{l}}_i)$ is shown as a dashed line. The magnitudes are to scale. **C** : Solid black arrows represent the two values of $\mathcal{F}$ corresponding to the two possible resolution topologies. **D** : All of the forces on the vertex are shown. Solid colored arrows represent the edges, which contribute a force of $\hat{\boldsymbol{l}}_i$. The dashed colored arrows represent the force from the pressure across each edge $\rho_i(\hat{\boldsymbol{z}} \times \hat{\boldsymbol{l}}_i)$. Solid black arrows are the two values of $\mathcal{F}$, and the dashed black arrow is the velocity vector. Values of ρ_i, $\mathcal{F}$ and the total force are given on the right. Note that both solid black arrows are shorter than the four arrows giving the edge tensions, indicating that $|\mathcal{F}| < \Gamma < 2\Gamma$, amply satisfying the stability conditions of eqs. 2.22–2.24. **E** − **F** : Parameter space in which fourfold vertices with anisotropic edge tensions are stable. **E** : Stability for symmetric vertices; γ gives the strength of the anisotropy in the tension and θ gives the angle of the edges with respect to the x-axis (*inset*). The region of parameter space in which fourfold vertices are stable is shown in green. **F** : Stability for asymmetric vertices with paired edges; γ gives the strength of the anisotropy in the tension, θ gives the angle of $\hat{\boldsymbol{l}}_1$ with respect to the x axis, and φ gives the angle between $\hat{\boldsymbol{l}}_1$ and $\hat{\boldsymbol{l}}_2$ (*inset*). The region of parameter space in which fourfold vertices are stable is shown in green.

that these differ from one edge to the next. One example of this is the anisotropic edge tensions produced through the planar cell polarity pathway [1, 2, 3, 4, 5, 6, 30] (which breaks rotational symmetry by defining a preferred direction in the plane of the epithelium).

A very simple model of planar cell polarity is to assume that tension regulating proteins (such as myosin) are recruited to edges based on the edges' angle with the overall polarity orientation, so that edges have an anisotropic tension given (in appropriate dimensionless units) by $\Gamma_i = 1 + \gamma \cos 2\theta_i$, where $\gamma \in [0, 1]$ gives the strength of the anisotropy, and θ_i is the angle between the edge and the planar polarity axis (which we will always take to be the x axis) [5, 45, 46]. We will make the further assumption that there is some time lag for proteins to move onto the newly forming edge, so that the new edge tension will not depend on the angle, but instead will simply be the unit tension $\Gamma_\delta = 1$. In order to further simplify the model we will also assume that effects from pressure are negligible. The force on a fourfold vertex is then described by five variables: γ and the four angles θ_i between the edges and the polarity axis.

With the additional effects of polarization some stable fourfold states exist. We begin our examination of the stable states by looking only at states which are symmetric about both the x and y axes (fig. 5.3E, inset). Let θ be the angle between the high tension x-axis and the edges. From the conditions given in eqs. 2.25-2.26 it is easy to show that the vertex is stable if it satisfies both:

$$1 > 2(1 + \gamma \cos 2\theta) \cos \theta \tag{2.65}$$

$$1 > 2(1 + \gamma \cos 2\theta) \sin \theta. \tag{2.66}$$

The solutions to this series of inequalities are shown in fig. 5.3E. In general we have stable fourfold vertices when the strength of the polarization is fairly high and θ is near $\frac{\pi}{2}$. This makes intuitive sense because this represents all of the edges being near the low tension axis and the strength of the tension being relatively low.

We now lift the restriction of symmetry in order to look for more general instances of stability. We will assume that the edges come in equal and opposite pairs ($\hat{\boldsymbol{l}}_1 = -\hat{\boldsymbol{l}}_3$ and $\hat{\boldsymbol{l}}_2 = -\hat{\boldsymbol{l}}_4$), so that mechanical equilibrium is ensured and the number of free parameters is still low. (With this restriction, we still cannot explore all possible states of the model, but the variety of available vertex geometries is large enough to clearly demonstrate how polarized tensions can lead to stability.) We now have three free parameters. Let θ be the angle between the high tension axis (the x-axis)

and the first edge, φ be the angle between the first and second edges, and γ be the strength of the polarization. We then have that $\theta_1 = \theta$, $\theta_2 = \theta + \varphi$, $\theta_3 = \theta + \pi$, and $\theta_4 = \theta + \varphi + \pi$. In order to have stability the following two inequalities must hold.

$$\begin{aligned} 1 > &\Big((1 + \gamma \cos 2\theta) \cos\theta \\ &+ \big[1 + \gamma \cos 2(\theta + \phi)\big] \cos(\theta + \phi) \Big)^2 \\ &+ \Big((1 + \gamma \cos 2\theta) \sin\theta \\ &+ \big[1 + \gamma \cos 2(\theta + \phi)\big] \sin(\theta + \phi) \Big)^2 \end{aligned} \tag{2.67}$$

$$\begin{aligned} 1 > &\Big((1 + \gamma \cos 2\theta) \cos\theta \\ &+ \big[1 + \gamma \cos 2(\theta + \phi - \pi)\big] \cos(\theta + \phi - \pi) \Big)^2 \\ &+ \Big((1 + \gamma \cos 2\theta) \sin\theta \\ &+ \big[1 + \gamma \cos 2(\theta + \phi - \pi)\big] \sin(\theta + \phi - \pi) \Big)^2 \end{aligned} \tag{2.68}$$

The solution to this series of inequalities is shown in fig. 5.3F. This is reasonable because more angles are stable as the amount of polarization increases and once again these angles represent the edges placed near the low tension axis.

Stable fourfold vertices are seen in some systems with planar cell polarity [27, 28, 30]. The stability of these vertices may be due to the decreased tension on edges along the low tension axis.

2.7 Implications for computational models

Although vertex models are widely used to simulate epithelial dynamics, there is currently no standard procedure for dealing with T1 transitions in such simulations. Some naive implementations can resolve fourfold vertices in ways that produce unphysical behavior. For example, approaches that automatically perform a T1 transition whenever an edge becomes too small, or more generally that assume that a fourfold vertex must always break up into two threefold vertices, can lead to spurious oscillations when the fourfold vertex should in fact be stable; importantly, as we showed in the preceding section, moving vertices can become stable even when

all tensions are equal, so this issue can in principle arise in almost all vertex model simulations. Something similar can occur when a fourfold vertex is resolved into two threefold vertices with a separation $\boldsymbol{l}_\delta$ that is not parallel to $\boldsymbol{\mathcal{F}}$ (though this phenomenon can be avoided—see below—if $|\boldsymbol{l}_\delta|$ is chosen small enough). In this section we briefly describe a method, based on the theoretical developments of the previous sections, that carries out T1 transitions in a consistent fashion and so avoids these and similar difficulties. Complete pseudo-code for this algorithm appears in the appendix.

The essential idea of our algorithm is that T1 transitions must be dealt with in two steps: First, an edge whose length is below a chosen cutoff is removed and the two threefold vertices joined by that edge are merged into a single fourfold vertex. Then, one checks the stability of the fourfold vertex against breaking in both allowed topologies (recognizing, as shown in Sec. 2.6, that the fourfold vertex could be stable). This requires creating temporary threefold vertices, with zero separation, and corresponding edges, so that the forces on the two new vertices can be calculated in each topology. Depending on the stability of the fourfold vertex, three outcomes are possible: 1) The fourfold vertex is found to be stable and allowed to persist. (In this case, the vertex could still become unstable at some later time, so one must continue to monitor its stability as the simulation progresses.) 2) The fourfold vertex resolves into two threefold vertices in the same topology as the original threefold vertices. One thus effectively rejects the proposed T1 transition even though the initial edge length is less than the cutoff. If one observes a series of such events involving the same edge, one can conclude that the dynamics is trying to drive the edge towards a nonzero length less than the cutoff length, and thus that the cutoff has been chosen too large for the system being studied. (One can readily imagine schemes to dynamically update the cutoff length in such circumstances, or even to assign different cutoff lengths to different edges, but for simplicity we do not include them in our pseudocode.) 3) The fourfold vertex resolves into two threefold vertices in the new topology, and a T1 transition occurs.

Once it has been determined that a fourfold vertex is unstable one needs to make a new edge of finite length, which raises the question of the most appropriate orientation for the new edge. From eq. 2.19 the new edge rotates at a rate

$$\dot{\theta} = -\frac{\mathcal{F}}{\mu l_\delta} \sin\theta, \tag{2.69}$$

and its length changes according to

$$\dot{l}_\delta = \frac{\mathcal{F}}{\mu}\cos\theta - \frac{2\Gamma_\delta}{\mu}. \tag{2.70}$$

The edge orientation must clearly relax to $\theta = 0$ as long as $\boldsymbol{\mathcal{F}}$ remains approximately constant over the relaxation timescale. Because $\dot{\theta}$ diverges like $1/l_\delta$, it is reasonable to guess that this will be the case if the initial edge length $l_{0\delta}$ is chosen small enough. More precisely, one can estimate that the edge relaxes to $\theta = 0$ on a timescale $\frac{\mu l_{0\delta}}{\mathcal{F}}$. Over that time, the change in edge length will be of order $l_{0\delta}$. Thus, the fractional change in the new edge's length during the relaxation process is of order one. Nonetheless, if $l_{0\delta}$ is small compared to the scale, typically of order a cell size, over which $\boldsymbol{\mathcal{F}}$ changes appreciably, then the variation in $\boldsymbol{\mathcal{F}}$ over the time it takes θ to rotate to zero can still be neglected. We thus conclude that if they are short enough, new edges will always quickly rotate to become parallel with $\boldsymbol{\mathcal{F}}$, whatever their initial orientation. It is then reasonable in simulations simply always to create new edges with $\theta = 0$.

2.8 Discussion

Vertex models are important tools to study the interplay between local cell mechanics and global tissue shape and motion. One aspect of this interaction during tissue remodeling and development is the T1 transition, in which a fourfold vertex is formed as an intermediary stage. More generally, the local behavior of fourfold vertices affects cell shape and mechanics, and thereby morphogenesis at larger scales.

Here, we have introduced a formulation of the stability of fourfold vertices in vertex models with straight edges that holds for arbitrary edge tensions and cell pressures (whether or not derived from an underlying energy function). Using this formulation, we have given the first proof that, in the simplest case of equal edge tensions and vertices in mechanical equilibrium—analogous to the conditions in a dry foam—fourfold vertices are never stable in vertex models, just as they are not in the Plateau model of foams.

We have also shown that if either of the assumptions of equal edge tensions or mechanical equilibrium is relaxed, fourfold vertices can become stable. Interestingly, long-lived fourfold and higher order vertices have been observed in epithelia moving relative to the surrounding fluid [29, 44] and in tissues where junctional tensions are influenced by planar cell polarity [27, 28, 30], suggesting that both stabilization

scenarios may have biological relevance.

Lastly, our treatment of vertex stability has clear implications for the simulation of vertex models and especially for the implementation of T1 transitions in computational modeling (see appendix A). Moreover, whereas our analytic results apply to models that in principle allow for cell overlap, in computational formulations this problem can be addressed by checking for overlap after T1 transitions. Disallowing overlap may stabilize some fourfold vertices in the limit where the force on the vertex from the cell pressure dominates over the tension on the edges (though such parameter regimes are not those thought to be physically relevant in most studies of vertex models, and in particular one could question whether it is a good approximation to force edges to remain straight when pressures are high enough). Our discussion in this chapter has been limited to fourfold vertices, but higher order vertices, like the rosettes seen during *Drosophila* germband extension [25], can be investigated in an entirely analogous manner, by checking whether the vertex is stable against breaking up into every possible combination of two lower order vertices; of course, the number of stability conditions will increase rapidly with the order of the vertex.

Although the relatively simple models for determining pressures and edge tensions that we have adopted here capture many aspects of the behavior of real epithelia, certain systems clearly require more sophisticated descriptions. For example, in the pupal dorsal notum of *Drosophila pten* mutants, vertices are seen to undergo oscillatory T1 transitions that appear to be driven by disparities in the timescales for transport of different proteins to newly formed edges [28, 47]. Our description of vertex stability can readily be extended to include many effects along these lines. In particular, as long as the new edge is much shorter than the existing edges, the stability problem can still be expressed in terms of the dynamics of the new edge $\boldsymbol{l}_\delta$, which in turn are determined by the—now possibly time-dependent—tensions and pressures of the surrounding edges and cells. Similarly, our formalism can encompass buckling of the epithelial sheet into the third dimension [35, 36] without any significant modifications, because even a bent epithelium appears locally flat when l_δ is much less than the sheet's radius of curvature, as it must be immediately after a fourfold vertex has broken up.

On the other hand, our formalism assumes that vertex stability is solely a consequence of local edge tensions and cell pressures; it does not include the effects of other phenomena that might be relevant in some biological systems and that would require more substantial changes to our basic model. For example, it is possible that in some circumstances cells could recruit proteins specifically to fourfold vertices to stabilize

or destabilize them. Similarly, the models studied here neglect effects associated with the fluid dynamics of molecular transport to and from vertices [48] and include interactions between the epithelium and its substrate only in the coarsest fashion, as one of the sources of the local friction force on vertices. Our calculations thus represent only an initial step towards understanding the rich physics of topology changes and vertex stability in epithelia and planar foams.

Bibliography

[1] J. D. Axelrod, "Progress and challenges in understanding planar cell polarity signaling," *Seminars in Cell & Developmental Biology*, vol. 20, no. 8, pp. 964 – 971, 2009.

[2] T. Lecuit, P.-F. Lenne, and E. Munro, "Force Generation, Transmission, and Integration during Cell and Tissue Morphogenesis," *Annual Review of Cell and Developmental Biology*, vol. 27, pp. 157–184, 2011.

[3] F. Bosveld, I. Bonnet, B. Guirao, S. Tlili, Z. Wang, A. Petitalot, R. Marchand, P.-L. Bardet, P. Marcq, F. Graner, and Y. Bellaïche, "Mechanical Control of Morphogenesis by Fat/Dachsous/Four-Jointed Planar Cell Polarity Pathway," *Science*, vol. 336, pp. 724–727, May 2012.

[4] A. Classen, K. Anderson, E. Marois, and S. Eaton, "Hexagonal packing of Drosophila wing epithelial cells by the planar cell polarity pathway," *Dev. Cell*, vol. 9, pp. 805–817, Dec 2005.

[5] Y. Mao, A. L. Tournier, P. A. Bates, J. E. Gale, N. Tapon, and B. J. Thompson, "Planar polarization of the atypical myosin Dachs orients cell divisions in Drosophila," *Genes & Dev.*, vol. 25, pp. 131–136, Jan 2011.

[6] E. Assémat, E. Bazelliéres, E. Pallesi-Pocachard, A. L. Bivic, and D. Massey-Harroche, "Polarity complex proteins," *(BBA) - Biomembranes*, vol. 1778, no. 3, pp. 614–630, 2008.

[7] A. G. Fletcher, M. Osterfield, R. E. Baker, and S. Y. Shvartsman, "Vertex Models of Epithelial Morphogenesis," *Biophys. J.*, vol. 106, pp. 2291–2304, Jun 2014.

[8] A. G. Fletcher, J. M. Osborne, P. K. Maini, and D. J. Gavaghan, "Implementing vertex dynamics models of cell populations in biology within a consistent computational framework," *Progress in Biophysics & Molecular Biology*, vol. 113, pp. 299–326, Nov 2013.

[9] S. Schilling, M. Willecke, T. Aegerter-Wilmsen, O. A. Cirpka, K. Basler, and C. von Mering, "Cell-Sorting at the A/P boundary in the Drosophila wing pri-

mordium: a computational model to consolidate observed non-local effects of Hh signaling," *PLOS Comp. Bio.*, vol. 7, Apr 2011.

[10] R. Farhadifar, J.-C. Röper, B. Algouy, S. Eaton, and F. Jülicher, "The influence of cell mechanics, cell-cell interactions, and proliferation on epithelial packing," *Current Biology*, vol. 17, pp. 2095–2104, Dec 2007.

[11] P. A. Raymond, S. M. Colvin, Z. Jabeen, M. Nagashima, L. K. Barthel, J. Hadidjojo, L. Popova, V. R. Pejaver, and D. K. Lubensky, "Patterning the Cone Mosaic Array in Zebrafish Retina Requires Specification of Ultraviolet-Sensitive Cones," *PLOS ONE*, vol. 9, Jan 2014.

[12] M. Rauzi, P.-F. Lenne, and T. Lecuit, "Planar polarized actomyosin contractile flows control epithelial junction remodelling," *Nature*, vol. 468, p. 1110, Dec 2010.

[13] A. M. Greiner, H. Chen, J. P. Spatz, and R. Kemkemer, "Cyclic Tensile Strain Controls Cell Shape and Directs Actin Stress Fiber Formation and Focal Adhesion Alignment in Spreading Cells," *PLOS ONE*, vol. 8, Oct 2013.

[14] D. Weaire and S. Hutzler, *The Physics of Foams*. Clarendon Press, 1999.

[15] H. Honda, H. Yamanaka, and M. Dan-Sohkawa, "A computer-simulation of geometrical configurations during cell-division," *J. Theory. Bio.*, vol. 106, no. 3, pp. 423–435, 1984.

[16] H. Honda, H. Yamanaka, and G. Eguchi, "Transformation of a polygonal cellular-pattern during sexual-maturation of the avian oviduct epithelium - computer-simulation," *J. of Embryology and Experimental Morphology*, vol. 98, pp. 1–19, Nov 1986.

[17] T. Okuzono and K. Kawasaki, "intermittent flow behavior of random foams - a computer experiment on foam rheology," *Phys. Rev. E*, vol. 51, pp. 1246–1253, Feb 1995.

[18] H. Frost, C. Thompson, C. Howe, and J. Whang, "A two-dimensional computer simulation of capillarity-driven grain growth: Preliminary results," *Scripta Metall*, vol. 22, 1988.

[19] Y. Ishimoto and Y. Morishita, "Bubbly vertex dynamics: A dynamical and geometrical model for epithelial tissues with curved cell shapes," *Phys. Rev. E*, vol. 90, p. 052711, Nov 2014.

[20] G. Salbreux, L. K. Barthel, P. A. Raymond, and D. K. Lubensky, "Coupling Mechanical Deformations and Planar Cell Polarity to Create Regular Patterns in the Zebrafish Retina," *PLOS Comp. Bio.*, vol. 8, Aug 2012.

[21] K. Sherrard, F. Robin, P. Lemaire, and E. Munro, "Sequential Activation of Apical and Basolateral Contractility Drives Ascidian Endoderm Invagination," *Current Bio.*, vol. 20, pp. 1499–1510, Sep 2010.

[22] H. Chen and G. Brodland, "Cell-level finite element studies of viscous cells in planar aggregates," *J. OF Biomech. Engineering-Transactions of the ASME*, vol. 122, pp. 394–401, Aug 2000.

[23] E. Hannezo, J. Prost, and J.-F. Joanny, "Theory of epithelial sheet morphology in three dimensions," *PNAS*, vol. 111, pp. 27–32, Jan 2014.

[24] B. Aigouy, R. Farhadifar, D. B. Staple, A. Sagner, J.-C. Röper, F. Jülicher, and S. Eaton, "Cell Flow Reorients the Axis of Planar Polarity in the Wing Epithelium of Drosophila," *Cell*, vol. 142, no. 5, pp. 773–786, 2010.

[25] J. T. Blankenship, S. T. Backovic, J. S. P. Sanny, O. Weitz, and J. A. Zallen, "Multicellular rosette formation links planar cell polarity to tissue morphogenesis," *Dev. Cell*, vol. 11, pp. 459–470, Oct 2006.

[26] M. Rauzi, P. Verant, T. Lecuit, and P.-F. Lenne, "Nature and anisotropy of cortical forces orienting Drosophila tissue morphogenesis," *Nat. Cell Bio.*, vol. 10, p. 1401, Dec 2008.

[27] R. P. Simone and S. DiNardo, "Actomyosin contractility and Discs large contribute to junctional conversion in guiding cell alignment within the Drosophila embryonic epithelium," *Development*, vol. 137, pp. 1385–1394, Apr 2010.

[28] P.-L. Bardet, B. Guirao, C. Paoletti, F. Serman, V. Léopold, F. Bosveld, Y. Goya, V. Mirouse, F. Graner, and Y. Bellaïche, "PTEN controls junction lengthening and stability during cell rearrangement in epithelial tissue," *Dev. Cell*, vol. 25, no. 5, pp. 534–546, 2013.

[29] G. Trichas, A. M. Smith, N. White, V. Wilkins, T. Watanabe, A. Moore, B. Joyce, J. Sugnaseelan, T. A. Rodriguez, D. Kay, R. E. Baker, P. K. Maini, and S. Srinivas, "Multi-Cellular rosettes in the mouse visceral endoderm facilitate the ordered migration of anterior visceral endoderm cells," *PLOS Bio.*, vol. 10, Feb 2012.

[30] M. Tamada and J. A. Zallen, "Square cell packing in the Drosophila embryo through spatiotemporally regulated {EGF} receptor signaling," *Dev. Cell*, vol. 35, no. 2, pp. 151–161, 2015.

[31] M. J. Harding, H. F. McGraw, and A. Nechiporuk, "The roles and regulation of multicellular rosette structures during morphogenesis," *Development*, vol. 141, no. 13, pp. 2549–2558, 2014.

[32] H. Honda, "Geometrical Models for Cells in Tissues," *Int. Rev. Cytol.*, vol. 81, pp. 191–248, 1983.

[33] T. Nagai and H. Honda, "A dynamic cell model for the formation of epithelial tissues," *Phil. Mag. B*, vol. 81, no. 7, pp. 699–719, 2001.

[34] S. Ishihara, K. Sugimura, S. J. Cox, I. Bonnet, Y. Bellaïche, and F. Graner, "Comparative study of non-invasive force and stress inference methods in tissue," *Eur. Phys. J. E*, vol. 36, p. 9859, Apr 2013.

[35] P. Spahn and R. Reuter, "A Vertex Model of Drosophila Ventral Furrow Formation," *PLOS ONE*, vol. 8, Sep 2013.

[36] X. Du, M. Osterfield, and S. Y. Shvartsman, "Computational analysis of three-dimensional epithelial morphogenesis using vertex models," *Phys. Bio.*, vol. 11, Dec 2014.

[37] D. B. Staple, R. Farhadifar, J. Röper, B. Aigouy, S. Eaton, and F. Jülicher, "Mechanics and remodelling of cell packings in epithelia," *Eur. Phys. J. E*, vol. 33, pp. 117–127, Oct 2010.

[38] D. Bi, J. H. Lopez, J. M. Schwarz, and M. L. Manning, "A density-independent rigidity transition in biological tissues," *Nature Physics*, vol. 11, p. 1074, Dec 2015.

[39] K. T. Sakurai, T. Kojima, T. Aigaki, and S. Hayashi, "Differential control of cell affinity required for progression and refinement of cell boundary during Drosophila leg segmentation," *Dev. Bio.*, vol. 309, pp. 126–136, Sep 2007.

[40] K. Kawasaki, T. Nagai, and K. Nakashima, "Vertex models for two-dimensional grain growth," *Phil. Mag. B*, vol. 60, no. 3, pp. 399–421, 1989.

[41] G. Odell, G. Oster, P. Alberch, and B. Burnside, "The mechanical basis of morphogenesis 1. epithelial folding and invagination," *Dev. Bio.*, vol. 85, no. 2, pp. 446–462, 1981.

[42] W. Karush, *Minima of Functions of Several Variables with Inequalities as Side Constraints*. PhD thesis, Univ. of Chicago, 1939.

[43] H. Kuhn and A. Tucker, "Nonlinear programing," *Proc. Second Berkeley Symp. Math. Stat. Probab*, pp. 481–492, 1951.

[44] P. Campinho, M. Behrndt, J. Ranft, T. Risler, N. Minc, and C.-P. Heisenberg, "Tension-oriented cell divisions limit anisotropic tissue tension in epithelial spreading during zebrafish epiboly," *Nat. Cell Bio.*, vol. 15, pp. 65–70, Dec 2013.

[45] M. Zajac, G. Jones, and J. Glazier, "Model of convergent extension in animal morphogenesis," *Phys. Rev. Letters*, vol. 85, pp. 2022–2025, Aug 2000.

[46] M. Rauzi, P. Verant, T. Lecuit, and P.-F. Lenne, "Nature and anisotropy of cortical forces orienting Drosophila tissue morphogenesis," *Nat. Cell Bio.*, vol. 10, p. 1401, Dec 2008.

[47] Y. Mao, A. L. Tournier, A. Hoppe, L. Kester, B. J. Thompson, and N. Tapon, "Differential proliferation rates generate patterns of mechanical tension that orient tissue growth," *The EMBO Journal*, vol. 32, no. 21, pp. 2790–2803, 2013.

[48] M. Durand and H. A. Stone, "Relaxation time of the topological T1 process in a two-dimensional foam," *Phys. Rev. Letters*, vol. 97, Dec 2006.

CHAPTER III

Implementing the vertex model

Notes: I wrote the majority of the vertex model code with contributions from Jeremy Hadidjojo (primarily the visual representation in Matlab and input/output handling) and Hayden Nunley (primarily the numerical integrator and mitosis implementation).

3.1 Introduction

In every chapter in this thesis, software implementing the vertex model is hiding in the background. The code used to implement the vertex model is often glossed over in favor of highlighting the science produced with it. Although the code is just a tool to assist us in our scientific endeavors, its development was nontrivial, and some of the problems we encountered while developing the code are interesting in and of themselves.

This chapter highlights some of the important choices that we made to ensure that the algorithms driving our computational model correctly reflected the theoretical model itself. We also hope that this chapter will be useful to other scientists as they build upon the robust central core of the code for their own projects. For this reason, we will spend some time discussing practical aspects of implementing the code for one's own work. We assume that the reader is already familiar with the theory behind computational vertex models. For those unfamiliar, we recommend the recent reviews [1, 2, 3]. The notation used here is consistent with the notation defined in chapter III which also defines the model. Because the vertex model allows for topological changes in the the network structure implementing the model is not as simple as integrating a large number of ODEs. It is necessary to keep track of the connections between edges, cells, and vertices, and update the equations of motion as these connections change.

3.2 Overview

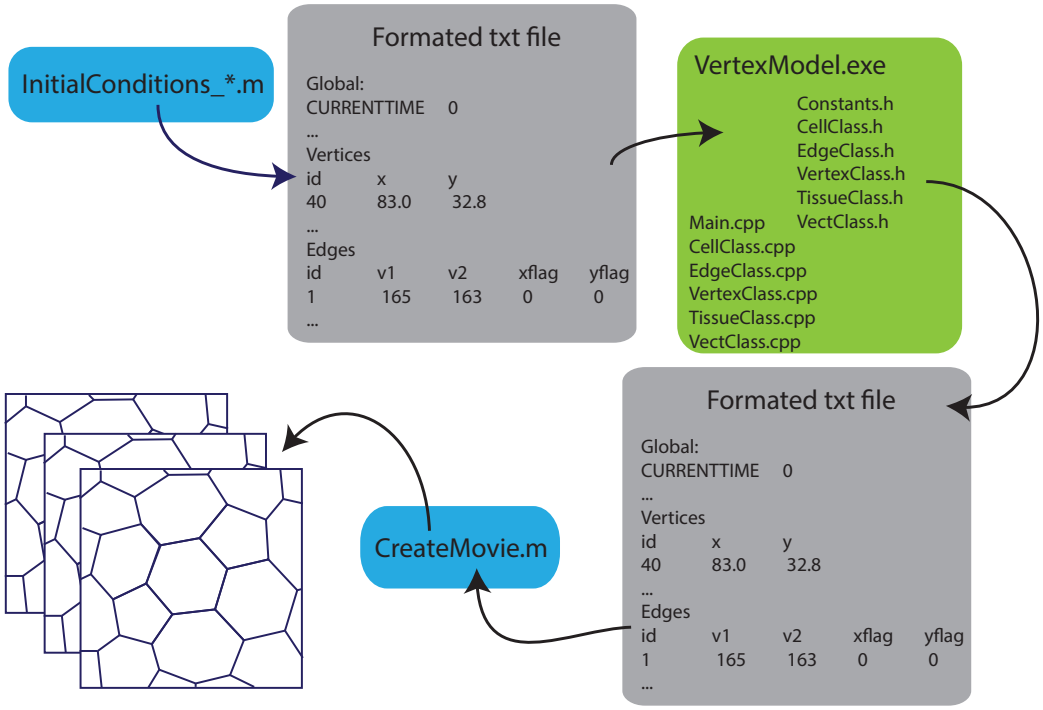

Figure 3.1: Cartoon of the work flow of the vertex model simulation. Initial packings can either be regular hexagons by using `InitialConditions_Honeycomb.m`, or Voronoi tessellation using `InitialConditions_Voronoi.m` .

The vertex model code is split into three sections which are run in different programs. A cartoon of the workflow is given in figure 3.1. First the user creates an initial packing (generally a Voronoi tessellation or a regular hexagonal packing) and chooses all parameter values. This is done in a single Matlab script. For more information on choosing parameter values, see section 3.3. The output of the script is a text file containing all the parameter values and the locations and connectivity of all of the cells, edges, and vertices. For further information on input/output handling, see section 3.4.

The second stage is a C++ executable which numerically integrates the equations of motion. This stage contains the core functionality. It is made up of six header files which contain the definitions of the cell, edge, vertex, tissue and vector classes as well as global constants. There are six associated source files, one for each class plus a main file.

It should be noted that C++11 has a built in class called `vector` which is *not* a mathematical vector but instead an array class with built in memory handling.

We have defined our own class `Vect` which performs the standard operations on two dimensional vectors. The `Cell`, `Edge`, and `Vertex` classes all contain the data and functions that define their basic properties as well as functions to retrieve pointers to their neighbors. For a more in depth discussion of these class structures, see section 3.5. The `Tissue` class contains all of the functions governing the tissue evolution as well as the master lists of all cells, edges, and vertices. The main file handles input/output and initialization of the class objects.

The final stage is a set of Matlab scripts which produce movies of the evolution of the tissue in time.

3.3 Standard assumptions and parameter set

The base class of the vertex model contains a number of assumptions regarding the form of the model. These assumptions can be changed by implementing subclasses of any or all of the objects, which override the standard behavior (see section 3.8). The base class assumes that the tissue lives in a box of dimensions $L_x \times L_y$ with periodic boundary conditions. The box may or may not be under external stress. The form of the cell pressure is given by

$$P = K\big(A - A_0\big). \tag{3.1}$$

All edges have the same constant contractile force γ_0.

The user must specify the number of cells and the values of γ_0, K, A_0, L_x, L_y, α_m, and α_T as defined in chapter II. When determining the value of the parameters, it is important to note that there are three independent units of measure in the vertex model equations: length, time, and force. For historical reasons we define our standard units as:

$$l_0 = 25, \tag{3.2}$$

$$\gamma_0 = 10, \tag{3.3}$$

$$t_0 = 1, \tag{3.4}$$

where l_0 is the average length of an edge, γ_0 is the magnitude of the edge tension, and t_0 is an arbitrary unit of time. The value of l_0 and the number of initial cells set the initial box size. In these units, a tissue relaxes to its equilibrium state in about 10 time units. Table 6.3.2 gives the standard parameter values in terms of these base

parameter	value	notes
L_x	$1.5l_0\sqrt{N_{cells}}$	Length of the tissue
L_y	$1.5l_0\sqrt{N_{cells}}$	Height of the tissue
A_0	L_xL_y/N_{cells}	Cells' preferred area
K	3.3 γ_0/l_0	Sets the pressure effects to 1/3 of the tension
α_m	$\gamma_0 t_0/l_0$	Drag parameter; sets the timescale
α_T	$10\alpha_m$	Box evolution timescale
dt	$t_0/500$	initial stepsize

Table 3.1: Standard set of parameter values used.

units. Any parameter set can be used in our vertex model, but we recommend using these parameters as a starting guide. Parameter values can be changed in the Matlab initialization file or in the main file according to user preference. Any unassigned global variables are set to their default values as specified in the main file.

3.4 Input-output

In order to save the network of cells, edge, and vertices for future use, we created a standard input-output text file format that interfaces with both the C++ code and the Matlab visualization scripts. An example of this standard formatting appears in figure 3.1. Input file has contain only the state of the network at the initial timepoint. Output files contain many timepoints which are concatenated in one file and read in sequence to produce movies of the tissue evolution.

The programs begin reading the files at the line `Global:`. This allows the user to write un-formatted notes above this line. Between the lines `Global:` and `endGlobal`, every global variable is defined. Any new global variables should be added here.

The line `Vertices` denotes the beginning of the list of vertices. Immediately following this line is a list of the recorded vertex characteristics. By default we record the vertex id and x and y location. Each vertex is listed on its own line until the line `Edges` is reached. Edges and cells are recorded in the same way. The default edge characteristics are the id, vertices, xflag and yflag, where the flags denote an edge that passes through the boundary. The default cell characteristics are the id, A0, cell type, number of edges, and list of edges. More characteristics can be recorded by adding the name of the characteristic to the first line after the object name and adding the value of the characteristic to each line representing an object of the appropriate type

separated from the other characteristics by a tab. The function `ReadInputData` is responsible for parsing the input file and reconstructing the network as C++ objects with pointers to their neighbors.

3.5 Defining the cell, edge, and vertex classes

In this section, we will discuss the `Cell`, `Edge`, and `Vertex` objects. Before we get into a detailed discussion of the object classes, we will take a moment to discuss the paradigm we used to guide our object creation. In computer science one often speaks of optimization in terms of scaling laws of the runtime of algorithms. However, there are other axes to optimize over, such as clarity of code or ease of adapting existing code to new situations. The driving force in developing our code base was to write clear, highly modular code that was easy to debug, often at the expense of optimal runtime. For this project development and testing of new code is more time-consuming than the total run time of a simulation (which is often less than a day). We also intend for the code to be accessible to undergraduate and first-year graduate students, many of whom have little to no coding experience.

All objects in C++ are made of two components, data members and object functions. Data members are any variables, such as scalars, lists, or strings. Object functions are any functions that describe operations on the object, such as finding the center of a cell. In general, most information about cells, vertices, or edges can be defined either in terms of data members or object functions. For example, we might define a double `length` which stores the length of an edge, or a function `GetLength` which calculates the length of the edge given its vertex positions.

In designing our objects, we chose to keep only the minimal set of data which is needed to reconstruct the tissue, where as others have chosen to keep multiple copies of all data in order to optimize compute time using multiple cores [4]. We avoid having two copies of the same information in multiple places to avoid potentially storing conflicting copies. This choice was made to support the ideal that the code should be easy to debug. As an example, take a list of cells neighboring cell C. One way to record this data is to create a data member of the Cell object which is a list of its neighboring cells. When we write the T1 function we must explicitly remember to update the list of neighboring cells for each cell involved in the T1. If we forget to do so, the code will not crash when it reaches a T1 function, but continue to run. Problems caused by this mistake may only manifest many time steps later, perhaps when one of the affected cells undergoes mitosis. We might now reasonably believe

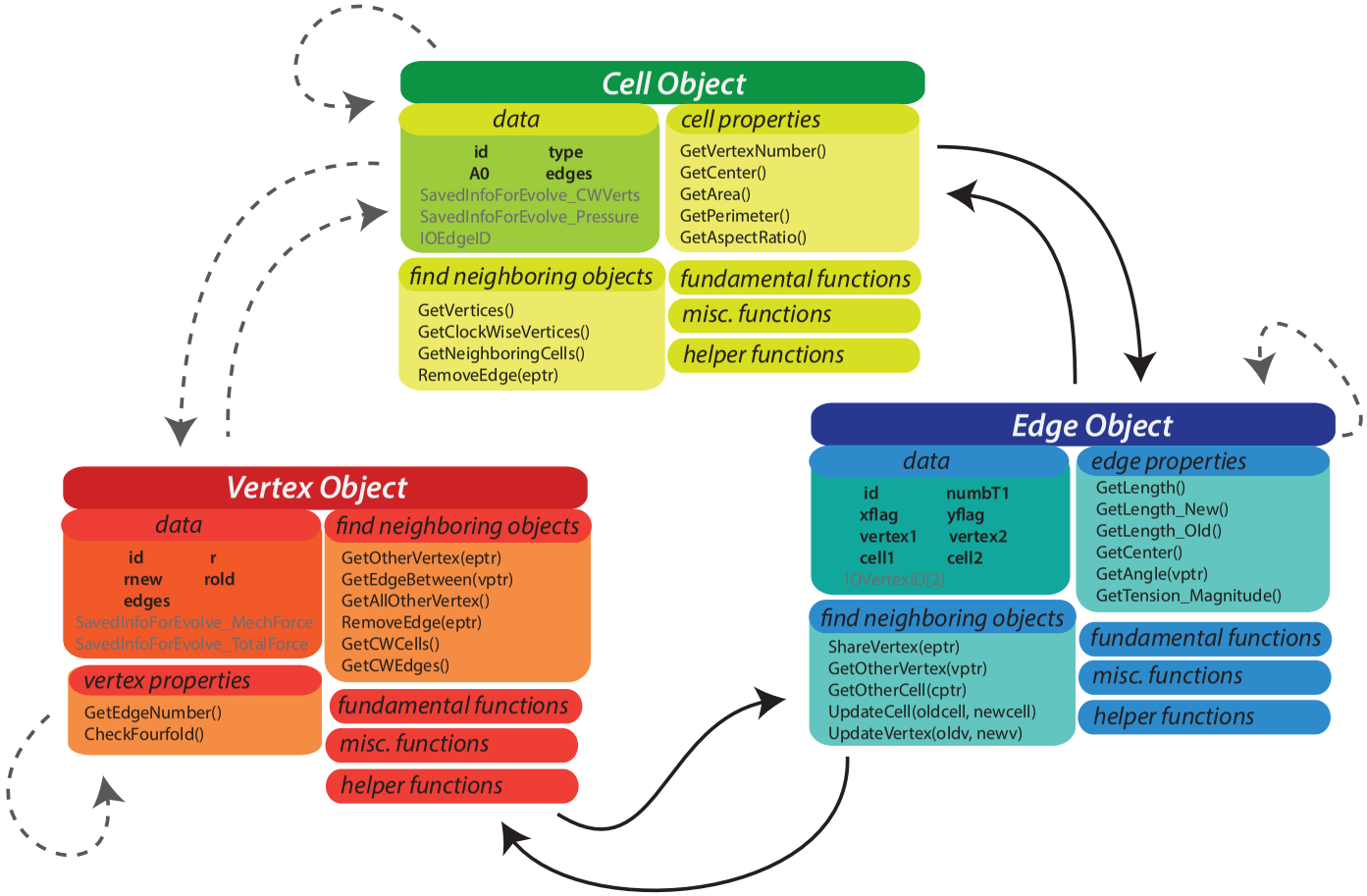

Figure 3.2: Cartoon of the organization of the `Cell`, `Edge`, and `Vertex` objects. Each object contains functions whose purpose is to find neighboring objects in the tissue. Solid arrows represent neighboring objects that are stored explicitly, whereas dashed arrows represent function calls that find neighbors on demand.

that there is some problem in the mitosis function, leading to significant delay in finding the actual bug.

Instead of recording the neighboring cells of cell C as a data member, we write an object function `GetNeighboringCells` which fetches this information anytime it is needed. This allows us to avoid the previous problem. By keeping only the minimum data set, it is more difficult to introduce errors later on, and easier to fix them when they happen. Each object in our code contains only the minimal data needed to reconstruct the tissue, with the exception of data named `SavedInfoForEvolve*`, which we will discuss in section 3.7.

A cartoon showing the organization of the objects, and their object functions is given in fig 3.2. The first category of functions are necessary fundamental functions for defining objects in C++, such as the constructor and destructor. The second category of functions finds properties of the objects, such as the length of an edge or the perimeter of the cell. We have listed all of these functions in figure 3.2. The third category functions fetch pointers to neighboring objects and are listed in figure 3.2. The arrows in figure 3.2 show the connectivity of objects by either member data or object functions. Solid arrows represent lists of neighboring objects stored as member data, whereas dashed arrows represent the existence of functions that retrieve neighboring objects when necessary. Many of these functions contain the same guts such as determining the clockwise ordering of vertices. All of the shared guts are defined in helper functions, which new users of the code should not need to update. The fifth class of functions is a small miscellaneous class.

3.6 Anatomy of the tissue class

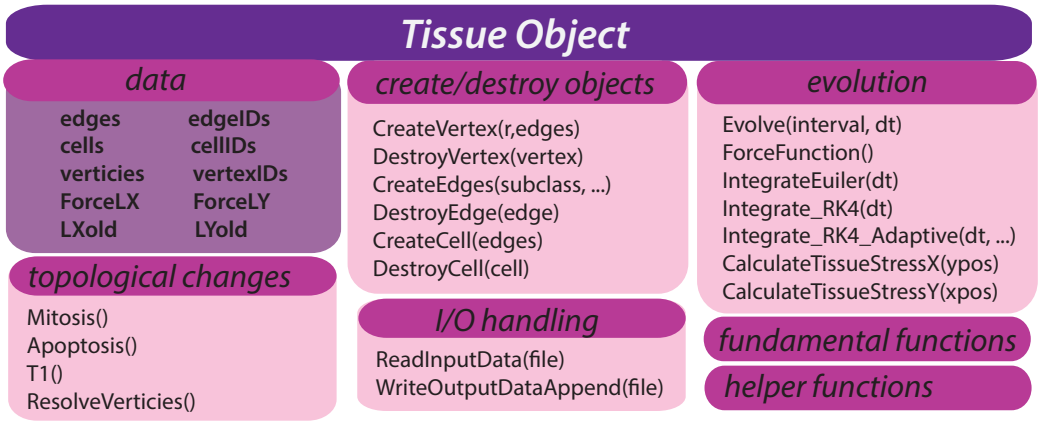

Figure 3.3: Cartoon of the organization of the tissue object.

The final class of user-defined objects is the tissue class. The tissue class serves as a container which holds all the other objects and which performs the numerical integration (see section 3.7). Figure 3.3 shows a cartoon of the tissue object. The tissue class contains linked lists of every edge, cell, and vertex in the tissue. We use linked lists because they can efficiently remove elements from their interior. It also contains a list of every ID used. When objects are removed from the tissue, they are removed from the linked list, but their IDs remain in the list of IDs so that IDs are not reused when new objects are created. This allows us to track individual cells, or edges in time. IDs are used exclusivly for I/O purposes, within the code objects are identified by pointers. There are four main types of functions in the tissue class in addition to the fundamental functions and helper functions. The first type of function creates or destroys objects from the tissue. These functions ensure that when an object is deleted, it is removed from both the linked list and from any other objects' data. The second type of function deals with I/O handling; for more information, see section 3.4. The third type function numerically integrates the vertex dynamics. A more thorough discussion of these functions is given in section 3.7. The final type of function implements topological changes in the tissue. A detailed description of the mathematics and algorithms used for the T1 transition is given in chapter 1.

3.7 Evolution in time

The dynamics of the vertex model are implemented within the tissue class. Pseudocode for the major high level functions is given in figure 3.4. The evolve function integrates the differential equations forward in time by dt, including making any necessary topological changes. Various integrators have been implemented by Hayden Nunley including fixed and variable step-size Euler and forth order RungeKutta integration schemes. The name of the integration scheme to be used is a global variable set in the input file. For more information, see Nunley's forthcoming thesis.

The form of the force function used is specified in the function `ForceFunction`. In the base class, the force on every vertex is a combination of the force from pressure in the neighboring cells and the tensions on the edges as described in chapter II. To use a different force function, for example one with noise in the vertex position, the user should define a subclass tissue with its own virtual implementation of the `ForceFunction` function.

At every timestep, the force function loops over all cells to compute their pressure. In order to compute the pressure the cell must retrieve all of its vertices and put them

Evolve(interval, dt)

```
endtime=currenttime+interval
reinitialize=true
while(currenttime<endtime)
        if(reinitialize)
                for all cells: record clockwise verticies
        for all verticies: v.rold=v.r
        LXold=LX
        LYold=LY
        Integrate_Euiler(dt)
        currenttime += dt
        UpdateVertexPos()
        LX=LXnew
        LY=LYnew
        f1=Apoptosis()
        f2=T1()
        f3=ResolveVerticies()
        f4=Mitosis()
        reinitialize=false
        if( f1 || f2 || f3 || f4 ) reinitialize=true
```

Integrate_Euler(dt)

```
ForceFunction()
for all verticies:
        v.rnew=v.TotalForce * (dt / alphaM) + v.rold
LXnew = (dt / alphaT) * ForceLX + LXold;
LYnew = (dt / alphaT) * ForceLY + LYold;
```

ForceFunction(dt)

```
for all cells:
        c.Pressure=c.GetPressure(c.cw_verticies)
for all verticies:
        for all neighboring edges:
                FT=e.GetTension_ForceOnVertex(v)
                FP=force from pressure across edge
                v.MechForce+=FT+FP
SigmaX=CalculateTissueStressX+ExternalStressX
SigmaY=CalculateTissueStressY+ExternalStressY
for all verticies:
        v.TotalForce.x=v.MechForce.x+(alphaM/alphaT)*v.r.x*SigmaX
        v.TotalForce.y=v.MechForce.y+(alphaM/alphaT)*v.r.y*SigmaY
ForceLX=LX*SigmaX
ForceLX=LY*SigmaY
```

Figure 3.4: Pseudo-code for the high level functions implementing the time evolution of the vertex model.

in clockwise order since the pressure relies on calculating the cell area which in turn requires a clockwise list of vertex positions. This is a time consuming operation and it scales linearly with both the number of cells and the number of integration steps taken. This created a major bottleneck in the run time of the code. For this reason we chose to add a data member to the cell class which stores a list of its clockwise vertices; an exception to our design principals discussed in section 1.5. The only time a cell can change the clockwise ordering of it's vertices is when the tissue changes topology. We set a reinitialize flag any time a topological change occurs and only recalculate the list of clockwise vertices when the flag has been triggered.

3.7.1 Implementing external stress

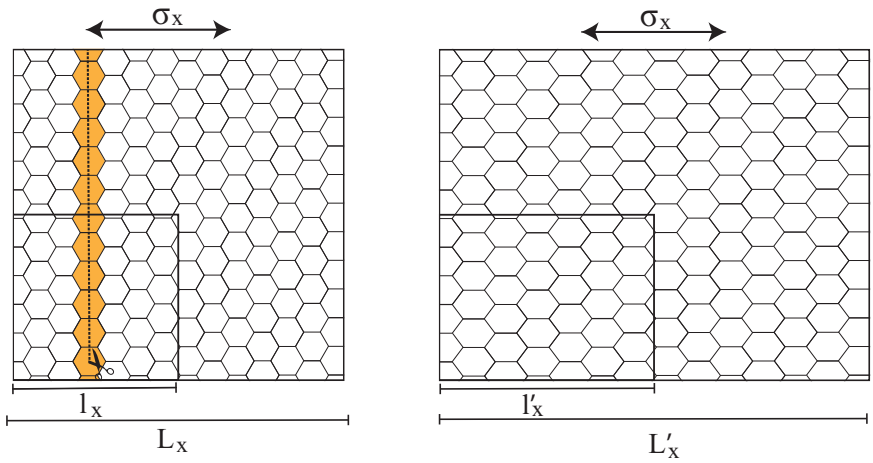

Figure 3.5: Physical constraints on the evolution of the box size. Given the same external stress σ_{xx} we want the smaller box to evolve as if it was embedded in a larger tissue.

The addition of an external stress is not given in the energy equation that defines the vertex model. There are a number of different approaches to implementing stress (see [3, 5]). Unfortunately, the way in which stress and box size changes are implemented is often omitted from the literature, which leads to difficulty replicating previous results. Here we will derive the implementation of external stress that we use in our vertex model starting from the energy equation:

$$E = \sum_i (\gamma_i l_i) + \frac{K}{2} \sum_j (A_j - A_0)^2. \tag{3.5}$$

By definition, the total stress in the tissue in the x dimension is the average of the force along a vertical cut through the tissue for all such cuts. In a tissue in

perfect mechanical equilibrium, the force along any vertical cut must be independent of the line of the cut; otherwise there would be a stress imbalance. Therefore we can approximate the stress everywhere by looking at only one cut through the tissue at tech time point, as long as we randomize the position of the cut and use a small enough step size. The force per unit length along a single cut is given by

$$\sigma^I_{xx} = \frac{1}{L_y}\bigg(\sum_i F_{Ti} + \sum_j F_{Pj}\bigg), \tag{3.6}$$

where σ^I_{xx} denotes the internal stress of the tissue along the x-axis, and F_T is the force along the cut from the tension on the edges, and F_P is the force from the pressure in the cells. The factor of L_y comes from the fact that a vertical cut has length L_y.

The force from the edge tensions is is simply the magnitude of the tension perpendicular to the cut,

$$\sum_i F_{Ti} = \sum_i \gamma_i \cos(\theta_i), \tag{3.7}$$

where θ_i is the angle that the i^{th} edge makes with the horizontal. The force exerted by the cell pressure is equal to the pressure times the length of the cut,

$$\sum_j F_{Pj} = \sum_j h_j P_j, \tag{3.8}$$

where h_j is the length of the cut through cell j and P_j is the cell pressure. Therefor, the internal stress in the x dimension is given by

$$\sigma^I_{xx} = \frac{1}{L_y}\bigg(\sum_i \gamma_i \cos(\theta_i) + \sum_j h_j P_j\bigg). \tag{3.9}$$

Likewise, the internal stress in the y dimension is

$$\sigma^I_{yy} = \frac{1}{L_x}\bigg(\sum_i \gamma_i \sin(\theta_i) + \sum_j h_j P_j\bigg). \tag{3.10}$$

Having determined how to calculate the internal stress of the tissue, we require a unified way of handling the changes in both the vertex position and box size due to the effects of both the mechanical force and the stress. Let R be the collection of vertex positions $R = \{\vec{r}_1, \vec{r}_2, ...\vec{r}_n\}$ and Γ be the collection of all the edge tensions.

Ultimately we want a set of four differential equations

$$\alpha_T \dot{L}_x = f_1(L_x, L_y, F_x^E, F_y^E, R, \Gamma) \tag{3.11}$$

$$\alpha_T \dot{L}_y = f_2(L_x, L_y, F_x^E, F_y^E, R, \Gamma) \tag{3.12}$$

$$\alpha_m \dot{x}_i = f_3(L_x, L_y, F_x^E, F_y^E, R, \Gamma) \tag{3.13}$$

$$\alpha_m \dot{y}_i = f_4(L_x, L_y, F_x^E, F_y^E, R, \Gamma) \tag{3.14}$$

which completely and consistently describe the evolution of the tissue over time. Here F_x^E and F_y^E are external forces applied to the box. The functions governing the motion in the x and y dimension are equivalent, so we will drop the equations of motion in the y dimension from here on. The value of f has units of force, so α has units of (time $\cdot$ force)/length.

We begin by looking at two illuminating special cases; a tissue where their is no net force on the box, and a tissue with in mechanical equilibrium. In a tissue with no net force on the box, the box size remains constant in time and the vertices move inside the box according to the local forces. Let F_m be the mechanical force on a vertex given by the pressure and tension. The equations of motion for the box and vertices are given by

$$\alpha_T \dot{L}_x = 0 \tag{3.15}$$

$$\alpha_m \dot{x} = F_{mx} \tag{3.16}$$

In a tissue in mechanical equilibrium, changes in vertex position are solely based on changes in box size. To have a physically reasonable system, we require that two edges that are initially the same length and direction before a change in box size to remain the same length and direction after a change in box size, regardless of their positions in the box. Stated mathematically, we require

$$\vec{r}_1 - \vec{r}_2 = \vec{r_3} - \vec{r}_4 \implies \vec{r'}_1 - \vec{r'}_2 = \vec{r'_3} - \vec{r'}_4 \tag{3.17}$$

where the primes are the vertex positions under a transformation of the box from $\{L_x, L_y\}$ to $\{L'_x, L'_y\}$. Let $\bar{x} = \frac{x}{L_x}$ be the relative x position of a vertex. If the tissue transforms by holding the relative positions constant so that $\bar{x} = \bar{x'}$ then

$$x \frac{L'_x}{L_x} = x', \tag{3.18}$$

and it is easy to see that 3.17 holds. This implies that $\frac{dx}{dL_x} = \bar{x}$ and x evolves in time according to

$$\dot{x} = \frac{dx}{dL_x}\frac{dL_x}{dt} = \bar{x}\dot{L}_x. \tag{3.19}$$

We are now left with determining how the box size should scale. Physically, we want to apply the same stress to two boxes, one imbedded inside the other, and have them evolve consistently together as shown in fig. 3.5. The internal stress felt by the two tissues is the same. In the larger tissue twice as many cells and edges contribute to the stress, but the tissue is twice as long, so the effect cancels out. Let $l_x = \frac{1}{2}L_x$. We require that

$$l_x + dl_x = \frac{1}{2}(L_x + dL_x), \tag{3.20}$$

which implies that

$$dl_x = \frac{1}{2}dL_x. \tag{3.21}$$

Therefore, for the definition of $\dot{L}_x$ to make physical sense it must scale with L_x. For simplicity we take $\alpha_T\dot{L}_x = (\frac{F_x^E}{L_y} - \sigma_{xx}^I)L_x$. The full set of equations of motion in the state of no net mechanical force are given by

$$\alpha_T\dot{L}_x = \left(\frac{F_x^E}{L_y} - \sigma_{xx}^I\right)L_x \tag{3.22}$$

$$\alpha_m\dot{x}_i = \frac{\alpha_m}{\alpha_T}x_i\left(\frac{F_x^E}{L_y} - \sigma_{xx}^I\right) \tag{3.23}$$

The most straightforward way to create a set of equations of motion that satisfy our two special cases is through a simple linear combination

$$\alpha_T\dot{L}_x = \left(\frac{F_x^E}{L_y} - \sigma_{xx}^I\right)L_x, \tag{3.24}$$

$$\alpha_m\dot{x}_i = F_{mx} + \frac{\alpha_m}{\alpha_T}x_i\left(\frac{F_x^E}{L_y} - \sigma_{xx}^I\right). \tag{3.25}$$

The ratio $\frac{\alpha_m}{\alpha_T}$ determines the speed at which the box moves relative to the vertices. Unfortunately, there is no particularly obvious value for this ratio. If $\frac{\alpha_T}{\alpha_m} = 1$ the box moves too fast, violating our assumption that vertices generally reorganize on a timescale much faster than the tissue changes shape. On the other hand, set the ratio too high and the simulation may take too long to run to be of any practical use. In our experience ratios of 10-100 tend to work well.

3.8 Extending the base code through the use of polymorphism

There are many different variants of the vertex model and many different problems of interest. Our code is built to be a flexible base on which to build different and more complex vertex models. We can have many different vertex models operating at once by creating different subclasses that implement problem-specific behavior. In C++ a subclass inherits all of the object data and object functions from its parent class, and can be given additional data and functions and can override the behavior of parent class functions. As an example, the energy function that defines the vertex model is sometimes written with an additional perimeter term which gives more rounded cells [6]. In this case the energy is given by

$$E = \sum_i (\gamma_i l_i) + \frac{K}{2} \sum_j (A_j - A_0)^2 + \frac{\Lambda}{2} \sum_j (P_j - P_0)^2, \tag{3.26}$$

where P_j is the perimeter of cell j, P_0 is the cells' prefered perimeter and Λ is the relative strength of the force induced by changing the cell perimeter. The tension on a vertex from an edge l has additional terms

$$F = \big[\gamma + \Lambda\,(P_1 - P_0) + \Lambda\,(P_2 - P_0)\,\big]\hat{l}, \tag{3.27}$$

where P_1 and P_2 are the perimeters of it's adjacent cells, corresponding to the perimeter deformations of its adjacent cells.

We might wish to create a spin-off model that uses this energy, while keeping our standard model intact. We can do this easily by creating sub-classes of the cell and edge objects which have additional or modified behavior. We define a subclass of the cell `Cell_Perim` which contains one additional data member P_0, the preferred perimeter of the cell. We also define an new edge subclass `Edge_Perim` which overites the standard `GetTension_Magnitude()` function with one that returns the value $\gamma_0 + \Lambda\,(P_1 - P_0) + \Lambda\,(P_2 - P_0)$.

Using the subclass objects allows us to quickly modify the behavior of the model without overriding previous behavior or making copies of the code, which can introduce errors and bugs. The many different variations of the vertex model used in chapter IV were produced using this method.

Bibliography

[1] A. G. Fletcher, J. M. Osborne, P. K. Maini, and D. J. Gavaghan, "Implementing vertex dynamics models of cell populations in biology within a consistent computational framework," *Progress in Biophysics and Molecular Biology*, vol. 113, no. 2, pp. 299 – 326, 2013.

[2] S. Alt, P. Ganguly, and G. Salbreux, "Vertex models: from cell mechanics to tissue morphogenesis," *Philosophical Transactions Of The Royal Society B-Biological Sciences*, vol. 372, May 2017.

[3] D. L. Barton, S. Henkes, C. J. Weijer, and R. Sknepnek, "Active Vertex Model for cell-resolution description of epithelial tissue mechanics," *PLOS Computational Biology*, vol. 13, Jun 2017.

[4] D. M. Sussman, "cellGPU: Massively parallel simulations of dynamic vertex models," *arXiv*, 2017.

[5] S. Curran, C. Strandkvist, J. Bathmann, M. de Gennes, A. Kabla, G. Salbreux, and B. Baum, "Myosin II Controls Junction Fluctuations to Guide Epithelial Tissue Ordering," *Developmental Cell*, vol. 43, pp. 480+, Nov 2017.

[6] D. Bi, X. Yang, M. C. Marchetti, and M. L. Manning, "Motility-Driven Glass and Jamming Transitions in Biological Tissues," *Phys. Rev. X*, vol. 6, Apr 2016.

CHAPTER IV

Multicellular actomyosin cables in epithelia under external anisotropic stress

Notes: Adapted from M.A. Spencer, J. Lopez-Gay, H. Nunley, Y. Bellache, and D.K. Lubensky, "Multicellular actomyosin cables in epithelia under external anisotropic stress," *arXiv:1809.04569* [q-bio.TO]. Sep 2018.

4.1 Introduction

We have spent the last two chapters developing the physical framework that we use to understand the mechanics of epithelia. With this framework fully established, we now turn our attention to biological systems under anisotropic mechanical stress. In the next two chapters, we will focus on multicellular actomyosin cables formed either along the junctions and associated cortex of cells, or through internal apical fibers.

The long range organization and patterning of cells in an epithelium helps drive the morphogenesis of tissues [1, 2, 3, 4]. One such pattern is the alignment of the junctions and associated cortex of multiple cells to form a continuous assembly of actomyosin, which we refer to as a cable (Fig 4.1A). The appearance and role of isolated cables, for example, at compartment boundaries, and during wound healing and dorsal closure, has been well studied [4, 5, 6, 7, 8]. However, the role of multiple parallel cables within a single tissue in morphogenesis remains less well understood. A number of different tissues contain such parallel actomyosin cables, including the the *Drosophila* wing imaginal disk [9, 10], pupal wing blade [11, 12], and ventral epidermis [13], as well as the mouse heart [14]. These cables are presumed to have large contractile tensions, and thus to maintain anisotropic mechanical stresses. Although there is some knowledge of the molecular events driving the formation of these cables in

some cases, little attention has been paid to the role of cell packing in aligning these junctions. Here we begin the work of understanding what role cell packing topology plays in cable formation.

An actomyosin cable forms when the junctional actomyosin across multiple cells aligns and assembles into a supra-cellular structure. It is currently unclear exactly how this continuity is achieved, although it is assumed that modified adherens junctions could link the ends of cables between adjacent cells, allowing forces to be transmitted along the cable [15]. Actomyosin cables were first identified in wound healing where they were called 'actin purse-strings' [7]. They have since been discovered in a number of different systems [5]. There appear to be different pathways involved in cable formation that are system dependent. For example, the planar cell polarity proteins Frizzled and Flamingo are upstream activators of actin and myosin on the cable at the leading edge of dorsal closure in *Drosophila*, whereas Notch signaling is the upstream activator of cable formation at the dorsal-ventral compartment boundary [16, 17, 18]. Independently of the nature of the signaling pathway contributing to their formation, cables are defined by the up-regulation of myosin leading to increased tension along the affected junctions [5, 19, 10]. If the increase in tension is in *response* to an externally imposed mechanical stress, the cables are relatively stable, as in the pupal wing blade [11]. If the stress anisotropy is instead internally generated, the cables are short lived, collapsing to form multicellular rosettes and driving tissue flow, as in convergent extension and neural tube closure [20, 21].

There are many interesting questions to ask about the relationship between mechanical stress and tissue topology [22, 23, 24, 25]. When an external stress is applied to an epithelium it may react by selectively increasing its internal tension at the junctions and associated cortex, by up-regulating myosin along the axis of stress [26, 27, 28, 29, 30]. Here we show that this up-regulation will lead to the formation of parallel cables *only* when the initial cell arrangement is favorable. This is easy to see in the cartoons in Fig. 4.1 where we show two perfectly ordered tissues in the 'cable forming orientation' (CFO) and the 'non-cable forming orientation' (NCFO). These two orientations behave differently under stress. As the stress anisotropy increases, cells in the CFO become brick shaped and cables from along the columns of cells. Because the brick-like packing contains lines of perfectly vertical junctions, cells in the CFO can in principal support any vertical stress without collapsing (provided, of course, that they can upregulate the tension on the cables sufficiently). Cells in the NCFO, in contrast, become highly elongated under an imposed stress anisotropy; they eventually collapse down to zero area as the applied stress increases. These cells

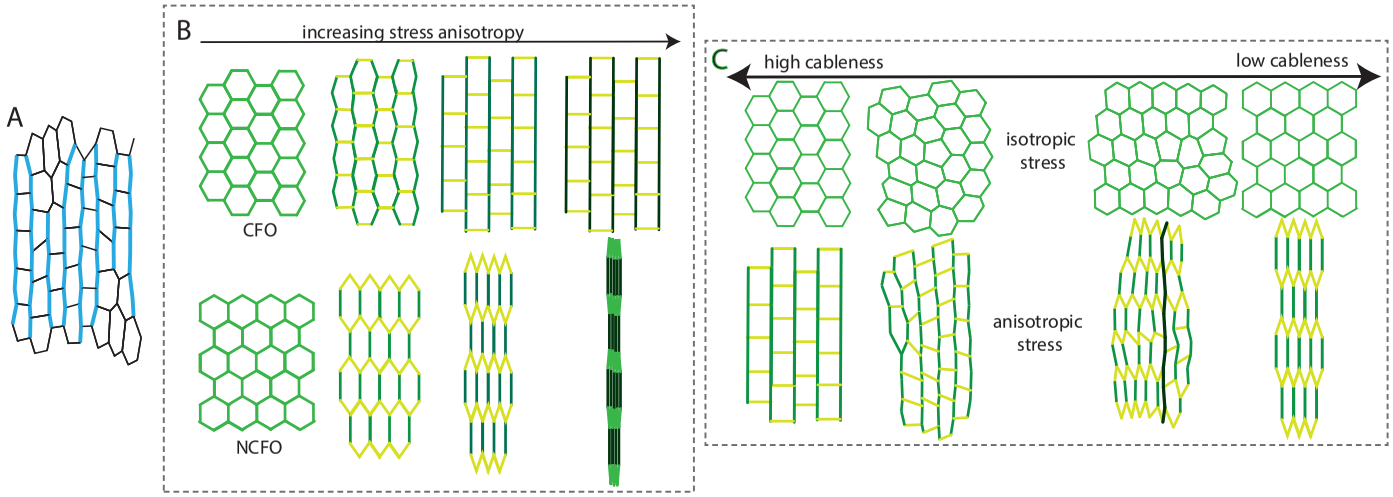

Figure 4.1: **A:** Cartoon of an epithelium with many cables highlighted in blue. We define a cable as a continuous, approximately straight line of myosin enriched junctions. **B:** Cartoon of tissues in the cable-forming orientation (CFO) and non-cable-forming orientation (NCFO). Edge color represents the tension on the edge; darker indicates higher tension. Cells in the CFO form brick shapes as they are stressed and reach a point at which their shape does not change further with increasing stress anisotropy. Cells in the NCFO collapse as they are stressed; the high tension edges are not neighbors, and do not form cables. **C:** Cartoon of epithelia with a range of cableness in the vertical direction. The packings with high cableness form many parallel cables when they are stressed, whereas the packings with low cableness form few to no cables. We define a *cably* packing to be one with high cableness.

are unable to form cables, because their high tension vertical edges cannot align unless the cells shrink to nothing. Previous studies have used the number of neighboring high tension vertical edges to identify cables [26].

Importantly, although the CFO and NCFO are related by an overall 30^o rotation, most epithelia are constrained at their boundaries such that they cannot rigidly rotate. Thus, shifting between the CFO and NCFO requires changing the cellular packing topology. In this chapter we focus on the formation of multiple parallel cables, as opposed to single isolated cables, such as those that form at compartment boundaries [5, 31]. For the duration of the thesis we always define the vertical axis to have higher stress, such that cables only form parallel to the y-axis.

Although the difference between the CFO and the NCFO can easily be understood from a few pictures, it is far less obvious how this distinction translates to a more realistic, disordered cell packing that is intermediate between the two limiting cases. Our ultimate aim is to understand how cell packing topology affects cable formation in just such disordered tissue. As a first step towards this larger goal, here we devise ways of quantifying how favorable a tissue topology is to forming cables in a given direction, a quality which we will refer to as *cableness*. Fig. 4.1C shows cell packings we would intuitively like to define as having different cableness in both their stress free and anisotropically stressed states. Those with high cableness form many parallel cables under anisotropic stress, whereas those with low cableness form few to no cables. Additionally, the cells in tissues with low cableness are more likely to become highly deformed under stress than cells in tissues with high cableness. In tissues with high cableness, the high tension edges align to form continuous cables, so that the force is evenly distributed through the tissue. In tissues with low cableness, most high tension edges are not connected to one another. Few cables form and lines of force frequently branch. Below we translate these qualitative ideas into quantitative measures of cableness. We define cableness to be a measure of a cell packing topology's *potential* to form cables under applied stress, so that a tissue has the same cableness before and after an applied stress (assuming it does not change topology). Therefore, the cableness of a tissue is a function solely of its topology and not of the presence or absence of cables at any given time (Fig. 4.1C).

As always, we will base our understanding of mechanical forces in an epithelium on the vertex model framework [32, 33, 34]. We will begin with a brief description of our computational implementation of the vertex model, including some relevant structural choices we made. We will then define a *stretching procedure* to computationally add anisotropic stress to tissues, which we will use to determine their cableness. Next

we define two different cableness measures based on the geometry and distribution of tension in stretched tissues. Using these two measures we will investigate how the cableness of a tissue changes as cells undergo both oriented T1 topological changes [32] and oriented divisions. Finally, we will apply our measures to data taken from the *Drosophila* pupa notum, where we will find that the cell arrangement changes over time to increase the tissue's cableness in response to an applied stress.

4.2 Materials and methods

In this section we will give a description of the vertex model, which is the theoretical and computational model we use to understand the physics of an epithelium. There are many different variations of the vertex model, each of which has different properties and is best used in different situations. In this section we define the variants of the vertex model used in this chapter; subsequent sections explain why we've chosen one or the other. We also define the stretching procedure which we use when calculating a tissue's cableness.

4.2.1 Theoretical framework: vertex model

Vertex models are a common way of understanding the physics of simple epithelia at the level of cellular scale mechanical forces [2, 32, 35, 33, 34]. They describe an epithelium as a quasi-two-dimentional sheet composed of cells, edges, and vertices. Edges describe the cortex at the level of the adherens junctions. A vertex is defined as any place three or more edges meet. Both cells and edges push and pull on vertices through mechanical forces. The mechanical force on a vertex is given by

$$\vec{F} = \sum_{\text{edges } i} \gamma_i \hat{l}_i + \sum_{\text{cells } j} \frac{P_j}{2} \hat{z} \times \left(\vec{l}_{j1} - \vec{l}_{j2}\right), \tag{4.1}$$

where $\vec{l}$ represents the length and orientation of an edge pointing out from the vertex, and $\hat{z}$ is perpendicular to the plane of the epithelium [32]. The first term describes the force from the edges on the vertex. The sum runs over all edges connected to the vertex. The strength of the force on each edge is given by its tension γ_i. The second term describes the force from the vertex's neighboring cells. Each cell has a pressure P_j determined by its deformation from its preferred area and acting in the direction of $\hat{z} \times \left(\vec{l}_{j1} - \vec{l}_{j2}\right)$ where $\vec{l}_{j1}$ and $\vec{l}_{j2}$ are the edges of cell j adjacent to the vertex taken clockwise. Vertex motion is assumed to be over-damped so that the velocity $\vec{v}_k$ of a

vertex k is proportional to the force acting on it.

$$\alpha \vec{v_k} = \vec{F}_k, \tag{4.2}$$

where F_k is the force on vertex k given by Eq 4.1. From some initial placement of cells, edges, and vertices the equation of motion for every vertex is integrated forward in time to model the dynamics of the tissue.

4.2.1.1 Choices in the vertex model

There are a number of choices to be made when implementing a vertex model. Because we use many different forms of the vertex model, we will briefly cover the choices that need to be made and their implications. These choices are summarized in Fig.4.2.

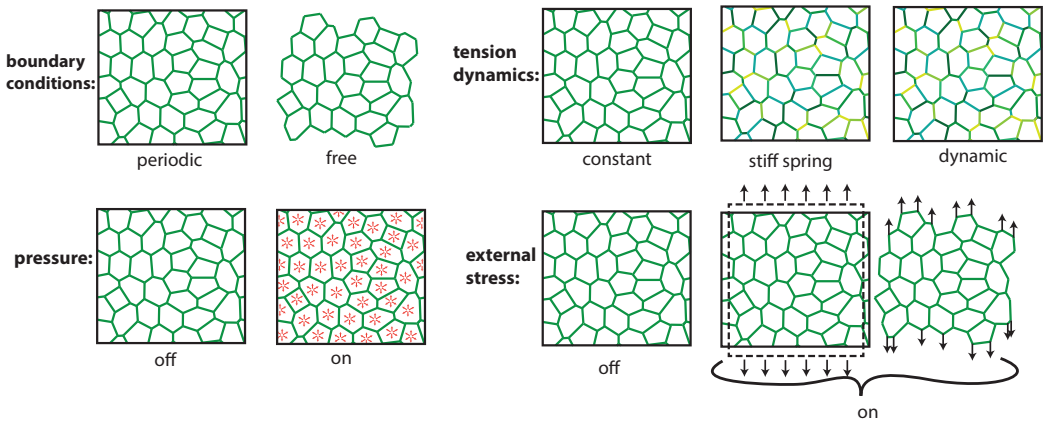

Figure 4.2: Cartoon of some of the options available to customize a vertex model. The tissue can have free or periodic boundary conditions. Cells may or may not have pressure, and edge tensions may be constant and identical for all edges or evolve according to different rules. Different colored edges in the cartoon represent edges with different tensions. The tissue can experience anisotropic stresses or not. As discussed in the text, anisotropic stress can be implemented in different ways. Each of these choices can be made independently of the others.

Choice 1: boundary conditions We use two different boundary conditions, free and periodic. When using free boundary conditions the equations of motion of the vertices are given by Eq 4.2. The major upside to using free boundary conditions is that the vertex model can be seeded using skeletonizations of experimental images. The downside is that one must specify the behavior of boundary cells. When using periodic boundary conditions the cells live in a rectangular box of size L_x by L_y. One could hold L_x and L_y fixed, but we generally want to be in a constant tension

ensemble so we let the box size evolve in time according to the algorithms in S1 Appendix. In the limit of a large tissue we expect the boundary conditions to be irrelevant.

Choice 2: cell pressure Vertex models may or may not include pressure forces. Including cell pressures gives a more realistic simulation, and pressures are essential when edge tensions are contractile. However, pressures can be dispensed with in other contexts, which generally decreases simulation runtime.

Choice 3: form of edge tension Every edge has some tension γ_i. We often assume that every edge has the same tension γ, which corresponds to the case in which edge tension is slow to respond to imposed stress. We will also use length dependent edge tensions so that edges resist compression and expansion while rotating freely. When edges respond quickly to stress we choose a simple form of

$$\beta_\gamma \dot{\gamma}_i = \left(\vec{F}_a - \vec{F}_b\right) \cdot \hat{l}_i, \tag{4.3}$$

where $\vec{F}_a$ and $\vec{F}_b$ are the forces on the vertices of edge i, as in [36]. We also sometimes allow the tension on every edge to act like a stiff spring, so that

$$\gamma = \kappa(l - l_0). \tag{4.4}$$

Choice 4: external stresses. We generally want to set stresses in the simulations and let the tissue size vary based on this applied stress. When including external stress on a tissue with periodic boundary conditions the box size changes according to the difference between internal and external stress, as described in S1 Appendix. When including external stresses in tissues with free boundary conditions a constant force is applied to every boundary vertex, which we take to be any vertex with only two edges. The same total force is applied to the top and bottom row of vertices. We distribute this force evenly amongst all the vertices on one boundary to prevent large torques. We choose this procedure, rather than introducing a rigid boundary to which we apply a force, because we want to ensure that stress is reasonably evenly distributed across the tissue.

4.2.2 Stretching procedure

In order to develop measures of cableness we will look at the behavior of disordered cell packings under highly anisotropic stress. We thus want a standardized way to impose anisotropic stress that captures whether a given packing can align its

edges into cables. To this end, we introduce a *stretching procedure* that allows the edges in any cell packing to rotate and increase their tension in a simple way in response to a strong, uniaxial stress while maintaining a constant topology. We expect the behavior of cells in this model to qualitatively but not quantitatively reflect behavior under more involved rules more directly inspired by specific biological systems. Fig. 4.3 shows the variants of the vertex model we used. First an input topology is created through a variety of methods which we will discuss later. Then the stretching procedure is applied to the input topology. Free boundary conditions were chosen so that the method can be easily applied to skeletonizations of experimental images of epithelia. Cells do not exert any pressure on their surrounding vertices; this cuts down on the number of free parameters in the model and helps to exaggerate cell deformations. We expect that the pressure does not play a large role in the real behavior of cells highly anisotropic under stress, as isotropic cell pressures can never counteract anisotropic stresses. The tension on every edge acts like a stiff spring:

$$\gamma = \kappa(l - l_0), \tag{4.5}$$

where κ represents the spring constant and l_0 is the initial length of the edge. We make this choice because it is the simplest version of feedback on the edge tension in response to mechanical stretching. A fixed, external stress is applied and the tissue relaxes to its equilibrium state which we refer to as the *stretched* state. We do not allow any topological changes since we are looking to measure the cableness of the initial topology. When looking at properties of the stretched state we always restrict measures to the middle 50% of cells to avoid any boundary effects. A network of threefold-coordinated vertices without any cell pressures is generically under-constrained, and we expect it to be able to undergo a finite deformation without generating any internal stress by rotating its edges at constant length. At some point as it is stretched, it will undergo a transition from this floppy state to a stiff state where the vertices are aligned so as to support a nonzero tension in the edges [37, 38, 39]. The final state of mechanical equilibrium in our stretching procedure is above this stiffening transition. In general, this state could depend on the magnitude of the stress we impose in a way that reflects the nonlinear network elasticity beyond the transition. In practice, however, we use a large value of κ (S3 Table) so that edge length depends very weakly on edge tension. In this limit, the spatial arrangement of vertices in the final state of self stress is essentially independent of the imposed stress, which serves only to set an overall scale for the edge tensions.

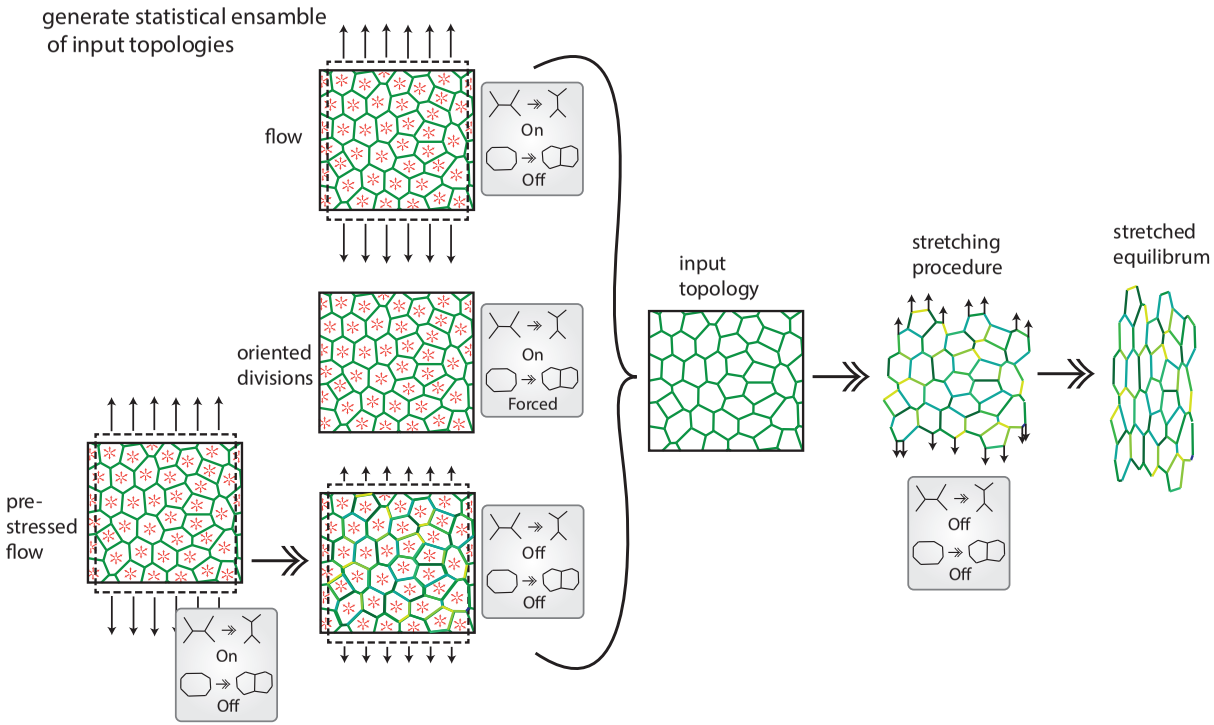

Figure 4.3: Cartoon of the major vertex model options used to investigate cableness. Statistical ensembles of input topologies are created in three different ways. Flow topologies are created by exerting a large anisotropic stress on packings with pressure effects and constant edge tensions. No cell divisions are allowed and T1s happen when energetically favorable. Oriented division topologies are created by dividing each cell exactly once in a tissue with pressure effects and constant edge tensions. T1s happen when energetically favorable. Pre-stressed flow topologies are seeded with the topology created by the flow procedure. A small anisotropic stress is applied to the tissue, which has pressure effects and dynamically changing edge tensions. Neither divisions nor T1s are allowed. The *stretching procedure* is a vertex model where anisotropic stress is applied to a tissue with free boundary conditions, no pressure effects, and stiff-spring edge tensions. We refer to this final state of mechanical equilibrium after stretching as the *stretched equilibrium*. All topological changes are suppressed. The resulting stretched equilibrium packing is used to measure cableness.

4.3 Results

We have split our results into three sections. In the first section we will describe how we measured cableness in disordered tissues. In the second section we will use our new cableness measures to show that oriented cell divisions promote cableness. In the third section we will discuss data from the *Drosophila* pupa notum which suggests that the notum becomes more cably after a round of oriented cell divisions.

4.3.1 Defining measures of cableness

We would like to produce a measure of cableness that corresponds to our intuitive notion of cableness (e.g., Fig. 4.1C) and that can be applied not only to simulation results but also directly to skeletonized images of real cell packings in order to determine the cableness of real tissues. This means the measure should depend solely on edge orientation and length which can be determined from images. Additionally the measure should work on pre-stressed as well as unstressed cell packings, as tissues observed in experiments will not typically be stress-free. Recall that our definition of cableness depends on a tissue's potential to form cables and is therefore a function solely of its topology, not of the presence or absence of cables at any given time.

Before we look for a measure that will work on disordered tissues we will look for insights from our toy model tissues the CFO and NCFO (Fig. 4.1B). An obvious difference between the CFO and NCFO is the edge orientation as measured by the average of $\cos(6\theta)$, where θ is the angle of the edge to the horizontal axis, which is 1 for the CFO and -1 for the NCFO. In fact this and closely related measures have been used before to quantify orientational order[11, 40, 41]. However, this measure only works in stress free tissues, and does not correctly distinguish cableness in geometries under stress. For example $\langle\cos(6\theta)\rangle$ for the CFO under high stress anisotropy is $-1/3$. Moreover, it's not clear that $\langle\cos(6\theta)\rangle$ necessarily predicts with cableness in disordered packings. Below we define alternate metrics that correlate well with $\langle\cos(6\theta)\rangle$ in unstressed tissues but that behave better under stress and correspond more directly to intuitive notions of cableness.

4.3.1.1 Creating control packings through cell flow

As we move from our toy models to disordered tissues we need to generate a statistical ensemble of disordered packings that we have independent reason to believe are more or less cably. It is well established that flowing tissues experience oriented T1 topological transitions, in which an edge shrinks to a fourfold vertex and then grows

in the perpendicular direction causing cells to exchange neighbors [32] (S1 Figure). When edge tensions are constant tissues under large external stress anisotropy will flow [42, 43, 44]. These oriented T1 transitions change the value of $\langle \cos(6\theta) \rangle$ in the tissue, and thus we expect the cableness to change as well (Fig. 4.4A). It is notable that vertical flow induces oriented T1 transitions from horizontal to vertical edges which pushes the tissue into a less cably direction. Therefore a tissue's passive response to stress inhibits the tissue's ability to form cables.

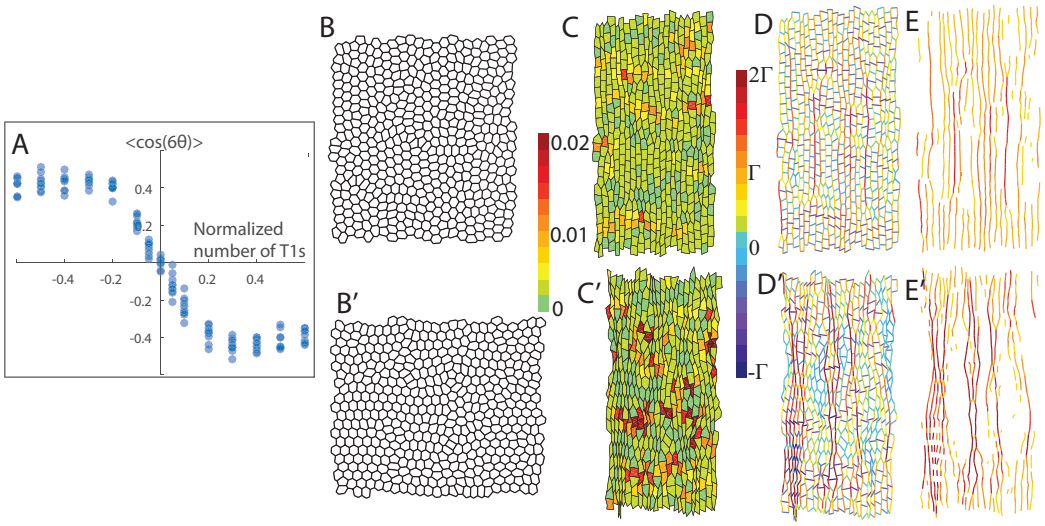

Figure 4.4: **A** The average edge orientation of tissues as measured by the average of $\cos 6\theta$ (where θ is the angle of the edge from the horizontal) is shown for our statistical ensemble of flow based packings. Each point represents data from the central region of one packing of 1000 cells (approximately 500 cells in the central region). Negative values on the x-axis indicate tissues that flowed perpendicular to the axis of cableness. **B (B')** Resulting packing from flowing cells perpendicular (parallel) to the cableness axis with 60 percent of edges undergoing a T1 transition. **C (C')** Stretched equilibrium packing, color indicates cell convexity as given by $(P - H)/H$, where P is the perimeter and H is the convex hull. **D (D')** Stretched equilibrium packing, color indicates edge tension, where Γ is the average force applied to boundary vertices. **E (E')** Same as D (D') with only high tension edges shown.

To generate packings with varying cableness we thus induced flow in a vertex model. We measured the extent of the flow, and thus the expected cableness, by counting the number of T1 transitions. A cartoon of the vertex model we used to generate these packings is given in Fig. 4.3 in the panel labeled 'flow'. The simulation is seeded with a Voronoi tessellation of randomly placed points in the plane and relaxes from this initial condition under isotropic stress. Once it has relaxed an external anisotropic stress of $\sigma_{yy} = 2\gamma\sqrt{N_c}/A$, $\sigma_{xx} = \gamma\sqrt{N_c}/(2A)$ (where N_c is the number of cells and A is the total area of the tissue) is applied and the tissue is allowed to flow

until a specified number of T1 transitions is reached. Once the desired number of T1s has been reached the tissue again relaxes under isotropic stress. Periodic boundary conditions are used for convenience. Pressure effects from cells are turned on to give a more realistic simulation. Edge tensions are constant to allow for cells to flow. We created tissues of 1000 cells each by inducing various levels of flow in both the vertical and horizontal directions. We then applied the stretching procedure.

4.3.1.2 Initial cableness measures

In order to define measures of cableness that correspond to physical properties of tissues under stress, we began by looking at the qualitative behavior of our statistical ensemble of flowing tissues in their stretched state (Fig. 4.4B-E). The primed panels represent a tissue which flowed parallel to the axis of cableness, expected to be less cably, whereas unprimed frames correspond to a tissue that flowed perpendicular to the axis of cableness, expected to be more cably.

Fig. 4.4C shows the convexity of cells in the stretched equilibrium as measured by the normalized difference between the perimeter and convex hull. There is a clear difference in the number of highly concave cells in the two tissues. Our first measure of cableness $\mathcal{C}$ is defined as

$$\mathcal{C} = \frac{1}{N_{cells}} \sum_{cells} \Theta\left(\frac{P - H}{H} - \epsilon\right), \tag{4.6}$$

where Θ is the Heaviside step function, P is the perimeter, H is the convex hull and ϵ is a small, positive cutoff. This essentially measures the fraction of concave cells in the tissue, disregarding cells that are only barely concave. Thus, small values of $\mathcal{C}$ correspond to high cableness. We require the cutoff ϵ because the brick shaped cells in the highly cably packings are often slightly very slightly concave, to an extent that is neither apparent to the naked eye nor likely to be biologically meaningful. We use an ϵ of 0.01 because it is roughly an order of magnitude larger than the numerical error of our simulation.

A second measure of cableness comes from the distribution of tensions in the equilibrium state after pulling. Fig. 4.4E shows the difference in behavior of the load bearing edges between a cably and non-cably tissue. The tissue with higher cableness has more evenly spaced cables with moderate tensions, whereas the tissue with low cableness has few unevenly spaced cables with high tensions. We create a quantitative scalar measure $\mathcal{T}$ that describes the difference in the way the tension is distributed between cably and non-cably packings.

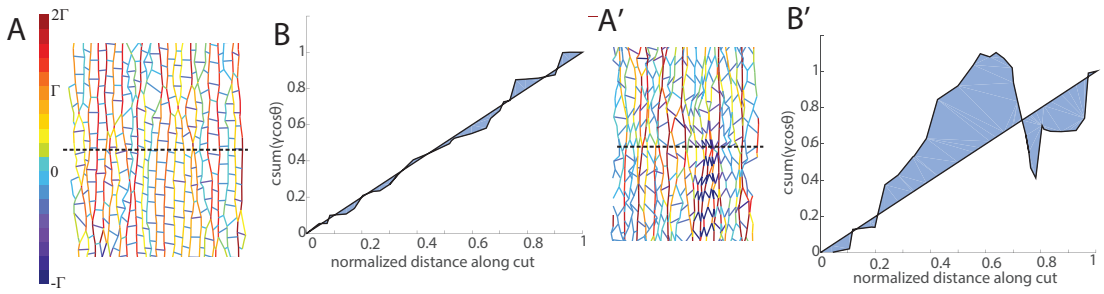

Figure 4.5: **A, A'** A more and less cably tissue after applying the stretching procedure. Color indicates edge tension, where Γ is the average force applied to boundary vertices. **B, B'** Plot of the difference between the unit line and the normalized cumulative sum $h(x)$ as a function of normalized distance x across the horizontal cut shown in A, A' (dashed line). The value of $\mathcal{T}$ is given by the area of the shaded region divided by the number of edges cut.

To get $\mathcal{T}$ we look at a horizontal cut through the tissue as shown in Fig. 4.5A(A′). Let

$$h(x) = \frac{\sum_0^x \gamma_i \cos\theta_i}{\sum_0^1 \gamma_i \cos\theta_i} \tag{4.7}$$

be the normalized cumulative sum of the vertical component of the edge tension of every edge which intersects the cut up to normalized distance x along the cut. In a cably tissue, where there are many lines of high tension roughly evenly spaced, $h(x)$ will be approximately linear. In the non-cably tissue, where there are only a few lines of high tension in clusters throughout the tissue, $h(x)$ will be step-like. The integral of the difference between the unit linear function and $h(x)$ (blue shaded reigon in 4.5B(B′)) should be low for cably packings and high for non-cably packings. We define $\mathcal{T}_h$ as

$$\mathcal{T}_h = \frac{1}{N}\int_0^1 dx\sqrt{\left[x - h(x)\right]^2}, \tag{4.8}$$

where N is the number of edges intersecting the cut. In order to remove boundary effects from starting the cumulative sum $h(x)$ at the left edge of the tissue, we calculate $\mathcal{T}_h$ starting at every edge and wrapping around the tissue and always take the minimum value $\mathcal{T}_{h,\min}$. The cableness measure $\mathcal{T}$ is the median of $\mathcal{T}_{h,\min}$ over 50 evenly spaced horizontal cuts through the tissue. Fig. 4.6 shows the result of applying our two cableness measures to the topologies generated by flow. Both measures decrease with increasing $\langle\cos(6\theta)\rangle$, and are in good agreement with one another. However, the values of cableness at no flow are higher than we would expect from the overall trend. We will investigate the reason for this bump in the next section.

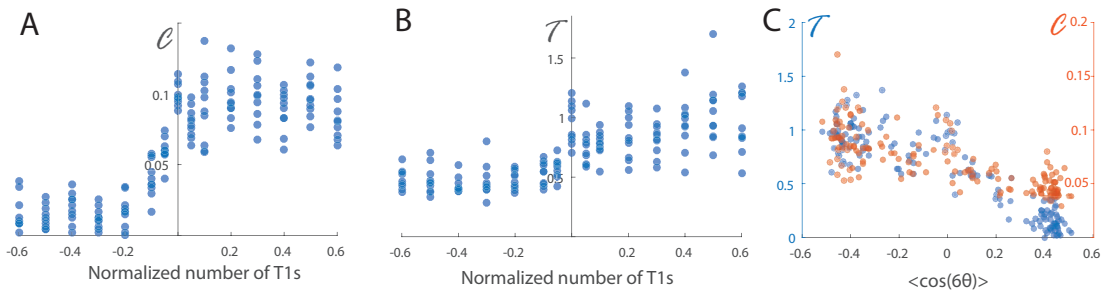

Figure 4.6: **A** Result of applying the cell convexity based cableness measure $\mathcal{C}$ to the packings generated by flow. Each point represents data from the central region of one packing of 1000 cells (approximately 500 cells in the central region). Negative values on the x-axis indicate flow parallel to the cableness axis. **B** Result of applying the tension based cableness measure $\mathcal{T}$ to the packings generated by flow. Each point represents data from the central region of one packing of 1000 cells (approximately 500 cells in the central region). Negative values on the x-axis indicate flow parallel to the cableness axis. **C** Cableness measures $\mathcal{C}$ (red) and $\mathcal{T}$ (blue) as a function of the edge orientation. Smaller values of $\mathcal{C}$ and $\mathcal{T}$ correspond to higher cableness.

Cableness along a single axis depends on the level of tissue disorder

We want to understand why there is a bump in both of our cableness measures at no flow. One difference between the tissues which have minimal flow and tissues which do not flow is the level of disorder as measured by the standard deviation in the number of edges per cell, which has been used as a measure of topological order in tissues [45]. We created three ensembles of packings with$\langle\cos(6\theta)\rangle \simeq 0$, which we might naively expect to have the same cableness. The first was identical to the ensemble used as the initial conditions for the flow simulations (see previous section). The second ensemble was created by inducing random cell divisions in an initially isotropic packing. From an initially isotropic packing cells were chosen at random to divide along a random axis until every cell divided exactly once. To create the third ensemble we began with a hexagonal packing and randomly selected half of the vertical edges to undergo T1 transitions.

We measured the cableness in these tissues and found that the values of $\mathcal{C}$ and $\mathcal{T}$ are linearly correlated with the edge number disorder (Fig. 4.7). Increasing disorder decreases the cableness of tissues along a given axis. However, when we take the differences $\mathcal{C}_y - \mathcal{C}_x$ or $\mathcal{T}_y - \mathcal{T}_x$ in the cableness along the y and x axis we find that the difference does not depend on the disorder. Taking the difference in cableness along perpendicular axis has the additional benefit of centering the cableness of isotropic tissues at zero. We will use these modified measures as our definitions of cableness for the rest of the thesis.

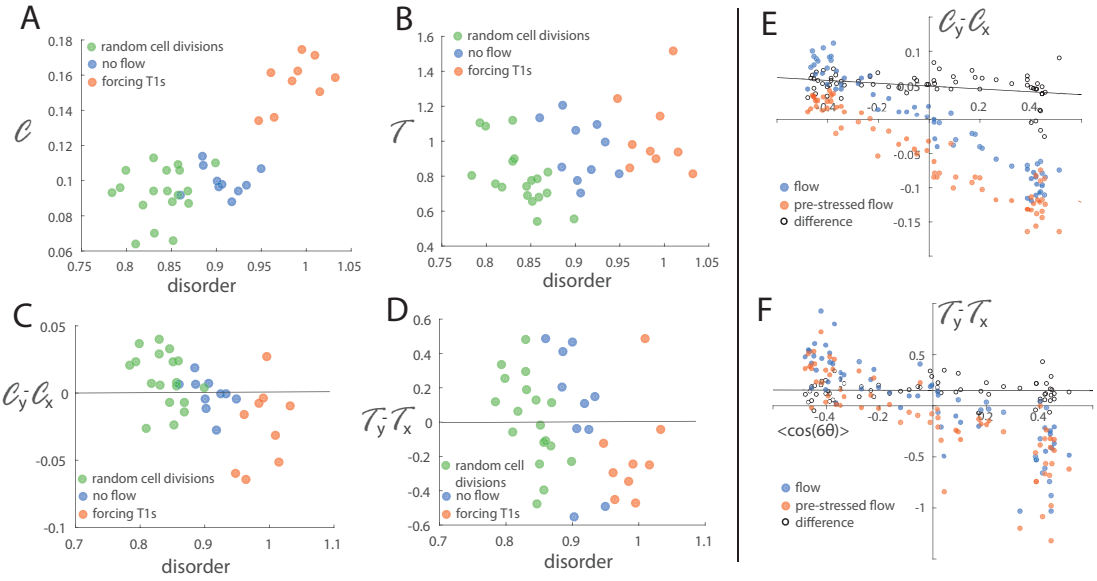

Figure 4.7: Input topologies with $\langle \cos(6\theta) \rangle \simeq 0$ were created through three different methods. In the first method cells underwent divisions at a random orientation. In the second method half of the vertical edges in an initially hexagonal packing were forced to undergo T1 transitions. In the final method the set of tissues with no flow (i.e. zero normalized T1 transitions) from Fig. 6 was used. The disorder on the horizontal axis (**A**–**D**) is defined as the standard deviation in edge number per cell throughout the tissue. Each point corresponds to one independently generated packing of 1000 cells. **A** $\mathcal{C}$ is strongly correlated with disorder. **B** $\mathcal{T}$ is more weakly correlated with disorder. **C, D** Taking the difference in cableness between the x and y axis centers the data at 0 and removes most of the dependence on disorder. **E** The difference measure $\mathcal{C}_y - \mathcal{C}_x$ has the same trend between relaxed and pre-stressed tissues, with a vertical offset. **F** The difference measure $\mathcal{T}_y - \mathcal{T}_x$ is decreased slightly by pre-stressing the tissue.

4.3.1.3 Validating cableness measures on pre-stressed tissues

Up to this point we have always applied our cableness measures to stress free tissues. However, we would like to apply our measures to experimental data, which is frequently from tissues subject to applied stress. In order to verify that our measures hold on pre-stressed data we applied a stress anisotropy to the input topology packings generated by flow. We let the tensions on the edges evolve according to

$$\beta_\gamma \dot{\gamma}_i = \left(\vec{F}_a - \vec{F}_b\right) \cdot \hat{l}_i, \tag{4.9}$$

where $\vec{F}_a$ and $\vec{F}_b$ are the forces on the vertices of edge i and β_γ is a characteristic relaxation time. We did not allow any topological changes to occur. This process more closely resembles the behavior of cells under anisotropic stress than the stretching

procedure which we use to determine cableness. Fig. 4.7E,F shows the results of applying our two cableness measures to both the pre-stressed and relaxed packings. Pre-stressing the tissue decreases the value of $\mathcal{C}_y - \mathcal{C}_x$ slightly without changing the overall trend, and had no noticeable effect on the tension based cableness measure. In introducing the idea of cableness we said that it is a property of a cell packing topology independent of its state of stress. These results show that our cableness measures satisfy this criterion to a good approximation. (It is worth underlining that this is not true of all conceivable measures of cableness; for example, an intuitively appealing measure based on cell elongation which fails this test is described in S2 Appendix.)

4.3.2 Oriented cell divisions promote cableness

Given that passive cell flow in the direction of applied stress decreases the cableness of a tissue in that direction, we would like to find a fundamental topology-changing process that increases cableness. Here we show that oriented divisions are one such process. Elongated cells are known to divide preferentially perpendicular to their long axis which in turn tends to align with the applied stress [10, 46, 12, 14, 47, 48, 49]. A cartoon of the vertex model used to create a statistical ensemble of packings derived from oriented divisions is given in Fig. 4.3 panel labeled 'oriented divisions'. The simulation has periodic boundary conditions, constant edge tensions, and pressure forces from cells. The initial packing is a Voronoi tessellation of random points in the plane which has been isotropically relaxed allowing for topological changes. Every cell divides exactly once in a random order. For simplicity and consistency we directly impose the orientation of the cleavage plane. The cleavage plane is either horizontal, vertical, or at a random unbiased angle. It should be noted that this increases the internal stress in the tissue in agreement with [50]. The cableness of each tissue in this ensemble of packings was determined and the results are shown in Fig. 4.8. Tissues in which the cells divide with a horizontal cleavage plane (as would be expected for a vertical applied stress) are more cably than tissues in which the cells divide vertically or in a random orientation.

4.3.3 Example: *Drosophila* epithelium

In the previous two sections we have established several measures of cableness and shown that cableness increases in the direction of applied tension when cells divide perpendicular to the applied tension. In this section we will give an example of how we

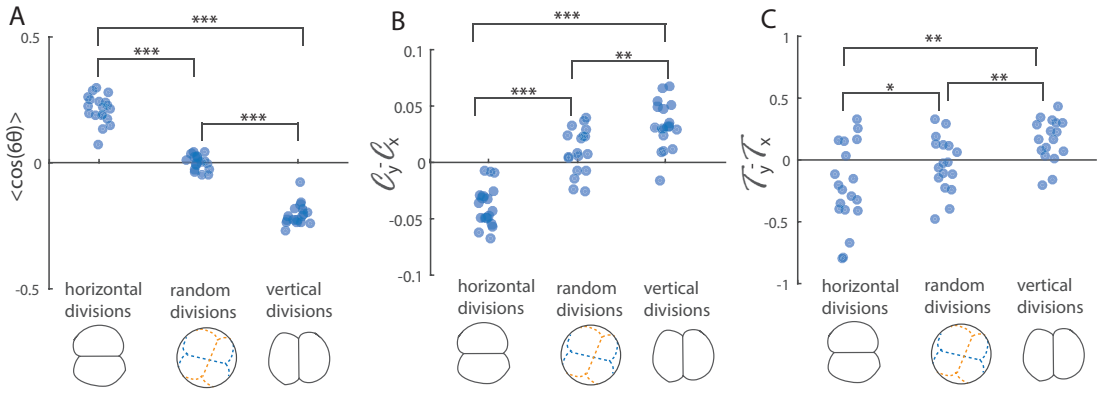

Figure 4.8: Each point corresponds to one packing topology with 1000 cells generated through cell divisions. Stars represent significance at the 0.05, 0.01, and 0.001 levels according to a t-test. **A** Edge orientation of packings generated by oriented cell divisions. (Horizontal divisions are defined to be those with a horizontal cleavage plane, which would tend to be produced by a mechanical stress in the vertical direction.) **B** Result of applying the cell convexity based cableness measure $\mathcal{C}_y - \mathcal{C}_x$ to the packings generated by oriented cell divisions. **C** Result of applying the tension based cableness measure $\mathcal{T}_y - \mathcal{T}_x$ to the packings generated by oriented cell divisions.

can apply our measure to biological data. We looked at the *Drosophila* pupa notum at 18 and 32 hours after pupa formation (APF). This system is of interest because it is known to undergo a wave of oriented divisions along with an increase in stress anisotropy over this time period, leading us to hypothesize that the tissue should increase its cableness between 18 and 32 hAPF [51]. Fig. 4.9A,B gives an example tissue at 18 and 32 hAPF. The images are skeletonized and used as the input topology for the stretching procedure. Image acquisition and segmentation were carried out as described in [51].

Fig. 4.9C,D show the qualitative results of the stretching procedure applied both perpendicular and parallel to the midline. At 18 hAPF the tissue's response to the stretching procedure is the same along both the vertical and horizontal axis. By 32 hAPF the tissue's response to the stretching procedure is highly dependent on the axis of stretch. When stretched perpendicular to the midline most cells remain convex and edges form many cables which rarely branch. In contrast when stretched parallel to the AP axis the cells are forced into highly irregular shapes, and the lines of force through the tissue frequently branch. The quantitative cableness measures agree with our prediction that the tissue becomes more cably in the vertical direction in time. Thus in this system mechanical stress orients cell divisions and these divisions allow the system to become more cably in the direction of stress. This more cably tissue can then form actomyosin cables to oppose the applied stress and prevent further

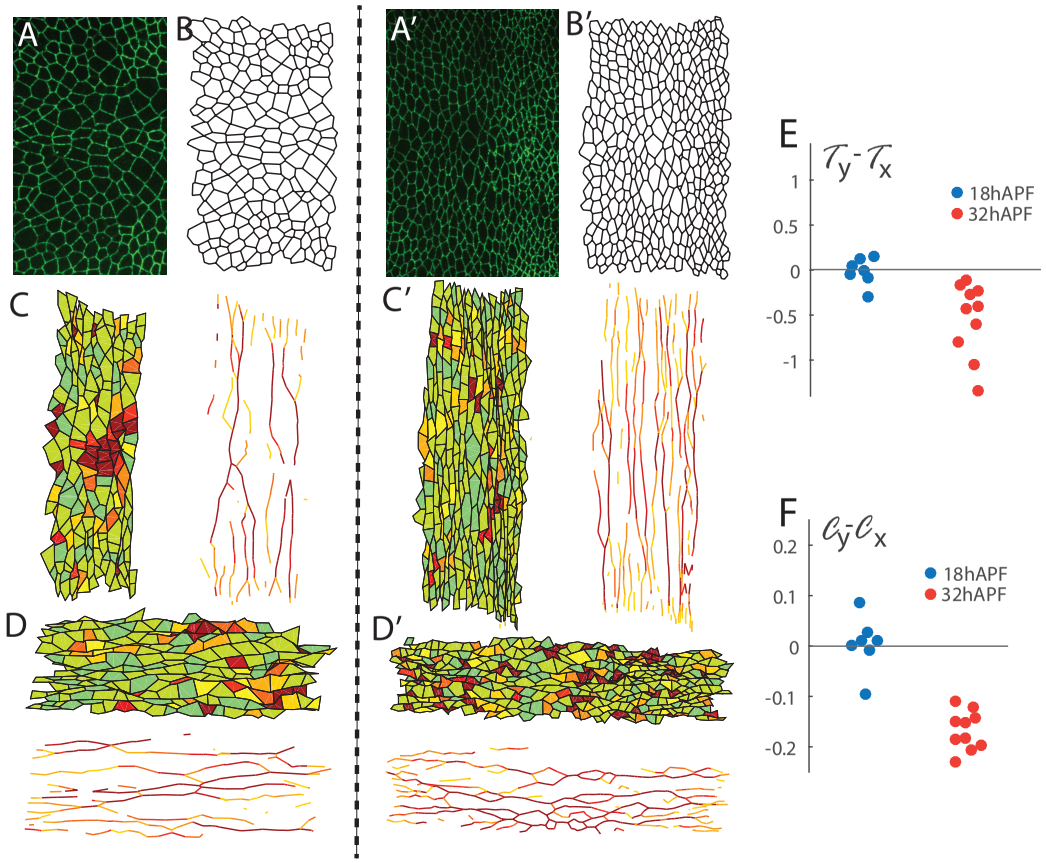

Figure 4.9: Unprimed frames are taken at 18 hours after pupa formation. Primed frames are from a pupa imaged at 32 hours after pupa formation after the tissue has undergone a round of oriented cell divisions with (on average) a horizontal cleavage plane and is under significantly higher stress in the vertical direction. **A (A'):** Image of the apical cell outlines of the wild type tissue labeled by ECad-GFP. The midline runs horizontally through the center of the image, and posterior is to the left. **B (B'):** Vertex model seeded from the pupa data. **C (C'):** Results of the pulling procedure along the vertical axis. **D (D'):** Results of the pulling procedure along the horizontal axis. **E:** Tension-based cableness measure. Each point represents one pupa. Tissues at 32 hours APF are more cably than tissues at 18 hours APF. **F:** Concavity based cableness measure. Each point represents one pupa. Tissues at 32 hours APF are more cably than tissues at 18 hours APF.

tissue elongation.

4.4 Conclusion

Multiple parallel actomyosin cables can only form in tissues if they have a favorable topology, which we call cableness. In a cartoon model of a tissue, cableness corresponds to the average edge orientation as measured by $\langle\cos(6\theta)\rangle$. However, it is unclear whether $\langle\cos(6\theta)\rangle$ remains a good predictor of cableness in realistic, disordered tissues, and it is obviously of less use in tissues with few hexagonal cells, or in tissues in which the cells are elongated. Here we defined two cableness measures $\mathcal{C}_y - \mathcal{C}_x$ and $\mathcal{T}_y - \mathcal{T}_x$, which quantify intuitive notions of whether a tissue does a good job of forming cables. We find that these measures correlate well with $\langle\cos(6\theta)\rangle$ in unstressed, statistically isotropic tissues with low topological disorder, but also reflect a physically and biologically meaningful idea of cableness in more disordered or pre-stressed tissues.

The convexity measure $\mathcal{C}_y - \mathcal{C}_x$ describes the geometry of cably tissues. When a tissue has high cableness its cells form a brick-like structure, whereas tissues with low cableness tend to have highly elongated cells which become concave and overlap when disorder is introduced to the system. These properties may also shed light on the role of cableness in morphogenesis. We expect that tissues with high cableness will react better to anisotropic stresses, distributing the force evenly throughout the tissue while maintaining relatively round cells.

The tension based measure $\mathcal{T}_y - \mathcal{T}_x$ describes the extent to which forces are spread evenly through the tissue. In disordered tissues that lack cableness only one or a few cables form in the tissue, whereas in tissues with high cableness many parallel cables form when the tissue is stressed. We expect that one could also define cableness measures based on the network properties of the stretched packings along the lines of network measures of force chains, and we hope that this work inspires additional work along these lines [52, 53, 54].

In order to maintain high cableness along one axis tissues must give up the ability to form cables along the perpendicular axis. However, tissues with a greater level of disorder also have lower cableness along a single axis. This suggests that cableness along a single axis is a function both of the orientational order of the cells and the total level of disorder in the tissue.

Passive cell flow along the axis of higher stress in an epithelium has the effect of reducing the cableness in that direction. Unlike the well know alignment of liquid

crystals due to flow, in this case the natural direction of the flow serves to disrupt the alignment of cells into a cably orientation [42, 43, 44]. Sugimura and Ishihara made note of this in the *Drosophila* pupal wing from around 24 to 32 hours after pupa formation. During this time there is cell flow along the proximal-distal (PD) axis and the cells become less cably in that direction as measured by $\langle \cos(6\theta) \rangle$ of hexagonal cells [11]. Therefore, the only way for cells to align in the CFO through T1 transitions is if the T1 transitions shrink edges parallel to the high stress axis. This requires some amount of overshoot in the tension on these edges, so that they constrict under stress rather than elongating. This appears to happen in the *Drosophila* pupal wing prior to cell flow [12] and in the *Drosophila* notum [51]. When cell flow is driven by internal stress we predict that cables will collapse into multi-cellular rosettes as seen in *Drosophila* intercalating cells [26].

A second process that increases the cableness of a tissue is oriented cell divisions. It has been well established for centuries that cells tend to divide along their long axis [49, 47]. Therefore, in a tissue under high vertical stress, cells will elongate vertically and tend to divide with a horizontal cleavage plane, thus increasing the cableness of the tissue in the vertical direction. It has previously been argued that oriented divisions can relax stress by elongating tissue in the direction of the imposed pulling [48, 55]. Here, we report a complementary role: they also cause packing topology to change so that cells are better able to form oriented cables and resist deformation by external stress. Both multicellular myosin cables and oriented divisions along the high stress axis are seen in tissues including the *Drosophila* larval wing and the mouse heart [9, 10, 14].

Studies of multicellular actomyosin cables have to date focused largely on isolated cables that form at well-defined boundaries, for example between two different tissues or compartments. Here we hope to spark interest in the role of in morphogenesis of collections of parallel cables within a single tissue. Whereas a single cable can form in almost any arrangement of cells, we have argued here that arrays of parallel cables are feasible only in cell packings that are sufficiently cably. Moreover, because the natural cell flow in the direction of applied stress tends to 'undo' cableness, anywhere that multiple cables are seen one must ask the question: how did the cells arrange themselves to allow cables to form? Here we have shown that one natural mechanism for increasing a tissue's ability to form cables is oriented cell divisions.

We have proposed two initial ways of quantifying a tissue's ability to form cables. While these methods have many promising features, we believe that further study of cables, perhaps from a network standpoint, may yield even better ways of quantifying

cableness. We have argued that any measure of cableness must work on both relaxed and stressed tissues since cableness, as we define it, is a property of a cell packing topology, not a measure of edge alignment.

Bibliography

[1] T. Lecuit and P. F. Lenne, "Cell surface mechanics and the control of cell shape, tissue patterns and morphogenesis," *Nature Reviews Molecular Cell Biology*, vol. 8, pp. 633–644, Aug 2007.

[2] R. Farhadifar, J.-C. Roeper, B. Algouy, S. Eaton, and F. Jüelicher, "The influence of cell mechanics, cell-cell interactions, and proliferation on epithelial packing," *Current Biology*, vol. 17, pp. 2095–2104, Dec 2007.

[3] F. Bosveld, I. Bonnet, B. Guirao, S. Tlili, Z. Wang, A. Petitalot, R. Marchand, P.-L. Bardet, P. Marcq, F. Graner, and Y. Bellaïche, "Mechanical control of morphogenesis by fat/dachsous/four-jointed planar cell polarity pathway," *Science*, vol. 336, no. 6082, pp. 724–727, 2012.

[4] M. Osterfield, T. Schpbach, E. Wieschaus, and S. Y. Shvartsman, "Diversity of epithelial morphogenesis during eggshell formation in drosophilids," *Development*, vol. 142, no. 11, pp. 1971–1977, 2015.

[5] K. Röper, "Supracellular actomyosin assemblies during development," *BioArchitecture*, vol. 3, no. 2, pp. 45–49, 2013.

[6] M. Michel and C. Dahmann, "Regulating mechanical tension at compartment boundaries in drosophila," *Fly*, vol. 10, no. 4, pp. 204–209, 2016.

[7] P. Martin and J. Lewis, "Actin cables and epidermal movement in embryonic wound-healing," *Nature*, vol. 360, pp. 179–183, Nov 1992.

[8] J. J. West, T. Zulueta-Coarasa, J. A. Maier, D. M. Lee, A. E. E. Bruce, R. Fernandez-Gonzalez, and T. J. C. Harris, "An actomyosin-arf-gef negative feedback loop for tissue elongation under stress," *Curr Biol*, vol. 27, no. 15, pp. 2260–2270 e5, 2017.

[9] Y. Mao, A. L. Tournier, A. Hoppe, L. Kester, B. J. Thompson, and N. Tapon, "Differential proliferation rates generate patterns of mechanical tension that orient tissue growth," *EMBO Journal*, vol. 32, pp. 2790–2803, Oct 2013.

[10] L. LeGoff, H. Rouault, and T. Lecuit, "A global pattern of mechanical stress polarizes cell divisions and cell shape in the growing drosophila wing disc," *Development*, vol. 140, no. 19, pp. 4051–4059, 2013.

[11] K. Sugimura and S. Ishihara, "The mechanical anisotropy in a tissue promotes ordering in hexagonal cell packing," *Development*, vol. 140, no. 19, pp. 4091–4101, 2013.

[12] F. Jüelicher and S. Eaton, "Emergence of tissue shape changes from collective cell behaviours," *Seminars in Cell & Developmental Biology*, vol. 67, pp. 103–112, Jul 2017.

[13] R. P. Simone and S. DiNardo, "Actomyosin contractility and Discs large contribute to junctional conversion in guiding cell alignment within the Drosophila embryonic epithelium," *Development*, vol. 137, pp. 1385–1394, Apr 2010.

[14] A. Francou, C. De Bono, and R. G. Kelly, "Epithelial tension in the second heart field promotes mouse heart tube elongation," *Nature Communications*, vol. 8, Mar 2017.

[15] K. Röper, "Integration of cell-cell adhesion and contractile actomyosin activity during morphogenesis," *Curr Top Dev Biol*, vol. 112, pp. 103–27, 2015.

[16] L.-H. Chang, P. Chen, M.-T. Lien, Y.-H. Ho, C.-M. Lin, Y.-T. Pan, S.-Y. Wei, and J.-C. Hsu, "Differential adhesion and actomyosin cable collaborate to drive Echinoid-mediated cell sorting," *Development*, vol. 138, pp. 3803–3812, Sep 2011.

[17] J. Kaltschmidt, N. Lawrence, V. Morel, T. Balayo, B. Fernandez, A. Pelissier, A. Jacinto, and A. Arias, "Planar polarity and actin dynamics in the epidermis of Drosophila," *Nature Cell Biology*, vol. 4, pp. 937–944, DEC 2002.

[18] R. J. Major and K. D. Irvine, "Localization and requirement for Myosin II at the dorsal-ventral compartment boundary of the Drosophila wing," *Developmental Dynamics*, vol. 235, pp. 3051–3058, Nov 2006.

[19] K. P. Landsberg, R. Farhadifar, J. Ranft, D. Umetsu, T. J. Widmann, T. Bittig, A. Said, F. Jüelicher, and C. Dahmann, "Increased cell bond tension governs cell sorting at the Drosophila anteroposterior compartment boundary," *European Journal Of Cell Biology*, vol. 89, p. 39, Mar 2010.

[20] S. d. M. Simoes, A. Mainieri, and J. A. Zallen, "Rho GTPase and Shroom direct planar polarized actomyosin contractility during convergent extension," *Journal of Cell Biology*, vol. 204, pp. 575–589, Feb 2014.

[21] T. Nishimura, H. Honda, and M. Takeichi, "Planar cell polarity links axes of spatial dynamics in neural-tube closure," *Cell*, vol. 149, no. 5, pp. 1084–1097, 2012.

[22] C. G. Vasquez and A. C. Martin, "Force transmission in epithelial tissues," *Dev Dyn*, vol. 245, no. 3, pp. 361–71, 2016.

[23] T. Wyatt, B. Baum, and G. Charras, "A question of time: tissue adaptation to mechanical forces," *Current Opinion in Cell Biology*, vol. 38, pp. 68–73, 2016.

[24] N. Noll, M. Mani, I. Heemskerk, S. J. Streichan, and B. I. Shraiman, "Active tension network model suggests an exotic mechanical state realized in epithelial tissues," *Nat Phys*, advance online publication, 2017.

[25] A. Nestor-Bergmann, E. Johns, S. Woolner, and O. E. Jensen, "Mechanical characterization of disordered and anisotropic cellular monolayers," *Physical Review E*, vol. 97, no. 5, p. 052409, 2018.

[26] R. Fernandez-Gonzalez, S. d. M. Simoes, J.-C. Roeper, S. Eaton, and J. A. Zallen, "Myosin II Dynamics Are Regulated by Tension in Intercalating Cells," *Developmental Cell*, vol. 17, pp. 736–743, Nov 17 2009.

[27] G. Salbreux, G. Charras, and E. Paluch, "Actin cortex mechanics and cellular morphogenesis," *Trends in Cell Biology*, vol. 22, no. 10, pp. 536 – 545, 2012.

[28] R. Levayer and T. Lecuit, "Biomechanical regulation of contractility: spatial control and dynamics," *Trends in Cell Biology*, vol. 22, no. 2, pp. 61 – 81, 2012.

[29] C. G. Vasquez, M. Tworoger, and A. C. Martin, "Dynamic myosin phosphorylation regulates contractile pulses and tissue integrity during epithelial morphogenesis," *The Journal of Cell Biology*, vol. 206, no. 3, pp. 435–450, 2014.

[30] W. Choi, B. R. Acharya, G. Peyret, M.-A. Fardin, R.-M. Mége, B. Ladoux, A. S. Yap, A. S. Fanning, and M. Peifer, "Remodeling the zonula adherens in response to tension and the role of afadin in this response," *The Journal of Cell Biology*, vol. 213, no. 2, pp. 243–260, 2016.

[31] D. M. Sussman, J. M. Schwarz, M. C. Marchetti, and M. L. Manning, "Soft yet sharp interfaces in a vertex model of confluent tissue," *Physical Review Letters*, vol. 120, no. 5, p. 6, 2018.

[32] Spencer, Meryl A., Jabeen, Zahera, and Lubensky, David K., "Vertex stability and topological transitions in vertex models of foams and epithelia," *Eur. Phys. J. E*, vol. 40, no. 1, p. 2, 2017.

[33] A. G. Fletcher, J. M. Osborne, P. K. Maini, and D. J. Gavaghan, "Implementing vertex dynamics models of cell populations in biology within a consistent computational framework," *Progress In Biophysics & Molecular Biology*, vol. 113, pp. 299–326, Nov 2013.

[34] A. G. Fletcher, M. Osterfield, R. E. Baker, and S. Y. Shvartsman, "Vertex Models of Epithelial Morphogenesis," *Biophysical Journal*, vol. 106, pp. 2291–2304, Jun 2014.

[35] S. Schilling, M. Willecke, T. Aegerter-Wilmsen, O. A. Cirpka, K. Basler, and C. von Mering, "Cell-Sorting at the A/P Boundary in the Drosophila Wing Primordium: A Computational Model to Consolidate Observed Non-Local Effects of Hh Signaling," *PLOS Computational Biology*, vol. 7, Apr 2011.

[36] N. Noll, M. Mani, and S. J. S. B. I. Heemskerk, Idseand Streichan, "Active tension network model suggests an exotic mechanical state realized in epithelial tissues," *Nature Physics*, 2017.

[37] J. C. Feng, H. Levine, X. M. Mao, and L. M. Sander, "Nonlinear elasticity of disordered fiber networks," *Soft Matter*, vol. 12, no. 5, pp. 1419–1424, 2016.

[38] A. Sharma, A. J. Licup, K. A. Jansen, R. Rens, M. Sheinman, G. H. Koenderink, and F. C. MacKintosh, "Strain-controlled criticality governs the nonlinear mechanics of fibre networks," *Nature Physics*, vol. 12, no. 6, 2016.

[39] M. F. J. Vermeulen, A. Bose, C. Storm, and W. G. Ellenbroek, "Geometry and the onset of rigidity in a disordered network," *Physical Review E*, vol. 96, no. 5, p. 053003, 2017.

[40] G. K. Xu, Y. Liu, and B. Li, "How do changes at the cell level affect the mechanical properties of epithelial monolayers?," *Soft Matter*, vol. 11, no. 45, pp. 8782–8, 2015.

[41] D. Y. Chen, W. Y. Aw, D. Devenport, and S. Torquato, "Structural characterization and statistical-mechanical model of epidermal patterns," *Biophysical Journal*, vol. 111, no. 11, pp. 2534–2545, 2016.

[42] L. Archer and R. Larson, "A molecular theory of flow-alignment and tumbling in sheared nematic liquid-crystals," *Journal Of Chemical Physics*, vol. 103, pp. 3108–3111, Aug 1995.

[43] M. Lukaschek, D. Grabowski, and C. Schmidt, "Shear-induced alignment of a hexagonal lyotropic liquid-crystal as studied by rheo-NMR," *Langmuir*, vol. 11, pp. 3590–3594, Sep 1995.

[44] G. Rienäcker and S. Hess, "Orientational dynamics of nematic liquid crystals under shear flow," *Physica A: Statistical Mechanics and its Applications*, vol. 267, no. 3, pp. 294 – 321, 1999.

[45] S. Curran, C. Strandkvist, J. Bathmann, M. de Gennes, A. Kabla, G. Salbreux, and B. Baum, "Myosin II Controls Junction Fluctuations to Guide Epithelial Tissue Ordering," *Developmental Cell*, vol. 43, pp. 480+, Nov 2017.

[46] F. Bosveld, O. Markova, B. Guirao, C. Martin, Z. Wang, A. Pierre, M. Balakireva, I. Gaugue, A. Ainslie, N. Christophorou, D. K. Lubensky, N. Minc, and Y. Bellaïche, "Epithelial tricellular junctions act as interphase cell shape sensors to orient mitosis," *Nature*, vol. 530, Feb 2016.

[47] C. Cadart, E. Zlotek-Zlotkiewicz, M. Le Berre, M. Piel, and H. K. Matthews, "Exploring the Function of Cell Shape and Size during Mitosis," *Developmental Cell*, vol. 29, pp. 159–169, Apr 2014.

[48] T. P. J. Wyatt, A. R. Harris, M. Lam, Q. Cheng, J. Bellis, A. Dimitracopoulos, A. J. Kabla, G. T. Charras, and B. Baum, "Emergence of homeostatic epithelial packing and stress dissipation through divisions oriented along the long cell axis," *Proceedings of the National Academy of Sciences*, vol. 112, no. 18, pp. 5726–5731, 2015.

[49] O. Hertwig, "Das Problem der Befruchtung und der Isotropie des Eies, eine Theorie der Vererbung," *Jenaische Zeitschrift fr Naturwissenschaften*, vol. 18, p. , 1884.

[50] M. Wu and M. Mani, "A mechanism for the proliferative control of tissue mechanics in the absence of growth," *bioRxiv*, 2018.

[51] B. Guirao, S. U. Rigaud, F. Bosveld, A. Bailles, J. Lopez-Gay, S. Ishihara, K. Sugimura, F. Graner, and Y. Bellaiche, "Unified quantitative characterization of epithelial tissue development," *ELife*, vol. 4, Dec 2015.

[52] M. Kramár, A. Goullet, L. Kondic, and K. Mischaikow, "Quantifying force networks in particulate systems," *Physica D: Nonlinear Phenomena*, vol. 283, pp. 37–55, 2014.

[53] C. Giusti, L. Papadopoulos, E. T. Owens, K. E. Daniels, and D. S. Bassett, "Topological and geometric measurements of force-chain structure," *Physical Review E*, vol. 94, no. 3, p. 032909, 2016.

[54] D. S. Bassett, E. T. Owens, M. A. Porter, M. L. Manning, and K. E. Daniels, "Extraction of force-chain network architecture in granular materials using community detection," *Soft Matter*, vol. 11, no. 14, pp. 2731–44, 2015.

[55] P. Campinho, M. Behrndt, J. Ranft, T. Risler, N. Minc, and C.-P. Heisenberg, "Tension-oriented cell divisions limit anisotropic tissue tension in epithelial spreading during zebrafish epiboly," *Nature Cell Biology*, vol. 15, Dec 2013.

CHAPTER V

Mechanics of actomyosin fibers in the *Drosophila* pupal notum

Notes: This chapter will be incorporated into a forthcoming paper combining experimental and theoretical results on the appearance of fibers in the *Drosophila* pupal notum with lead author Jesus Lopez-Gay.

5.1 Introduction

In the last chapter we discussed the formation of cables along the cell junctions and cortex. In this chapter, we will use this understanding to give us insight into the role of apical actomyosin fibers in the *Drosophila* pupal notum, which we believe play a similar role.

As we previously discussed, actomyosin networks localized to the cortex of epithelia are known to play a role in force generation and mechanosensing [1, 2, 3, 4]. Internal actomyosin chains called stress fibers have also be extensively studied in single cells [5, 6, 7, 8, 9]. However, apical internal actomyosin fibers, which peel off from the cortex have not previously been observed in epithelia. The epithelium of the *Drosophila* pupal notum is an exciting system due to the presence of internal apical actomyosin fibers in cells in the region between the macrochaetae during the period of 20-35 hours after pupa formation (APF), see figure 5.1. During this time the tissue is elongates slightly as it is pulled perpendicular to the midline, and their is some cell flow twards the anterior near the midline [10].

Figure 5.2 shows the evolution of our region of interest over this time period. The central bright region in the images is the midline. Florescent cadherin is used to mark the cell cortex. Myosin is also florescently tagged, reveling both cortex-bound and internal myosin. Figure 5.2B',C' give an enlarged view of cells with multiple

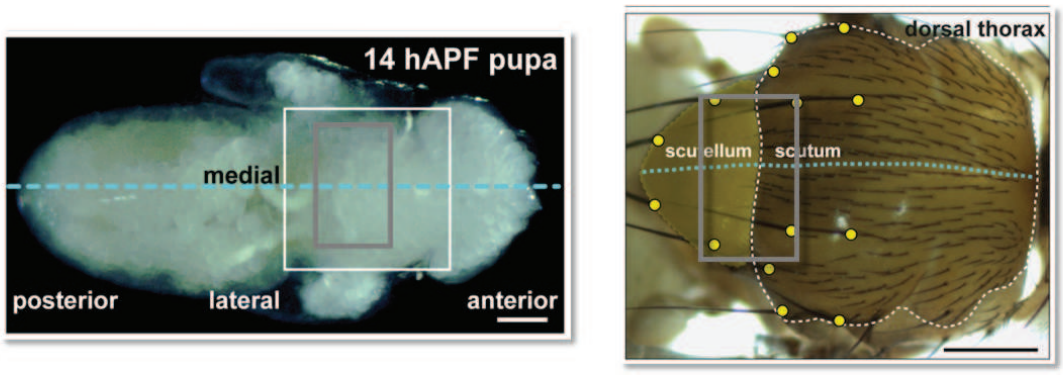

Figure 5.1: Region of interest of the *Drosophila* pupal notum. Left: full pupa. Right: region of interest is defined by the gray box. In all following images the posterior is on the left. *Image provided by Jesus Lopez-Gay* .

actomyosin fibers.

An anisotropic stress perpendicular to the midline grows during 22-26 hours APF, after which the stress remains constant. The cells undergo a wave of divisions which begins at 20h APF and ends at 28h APF. Figure 5.2 shows the total number of fibers within the region of interest between 18-36h APF. The total number of fibers in the tissue spikes to around 600 fibers at 26 hAFP after which it decreases until only approximately 100 fibers remain by 36h APF.

There are many questions we would like to ask about the role of fibers in morphogenesis; we will answer a few basic questions here, but due to the novel nature of this work there is much we still do not understand. This chapter represents only the beginning of what we expect will be a long term project. Since the appearance of fibers corresponds to an increasing vertical stress in the tissue, we hypothesize that the fibers are a response to the stress that help cells resist elongation. Our first goal is to devise a simple mechanical model that is consistent with this hypothesis, and see if it generates any testable predictions. Our second goal is to develop a theory that explains the loss of fibers around 32 hAPF despite the continued high stress anisotropy.

In section 5.2, we develop a toy model that captures the essential physics of fibers in epithelia. We describe the predictions of the model in section 5.4 and test the fiber scaling predictions against the experimental data. Finally, in sections 5.5 - 5.4 we will investigate the role that cell orientation plays in the need for fibers and develop

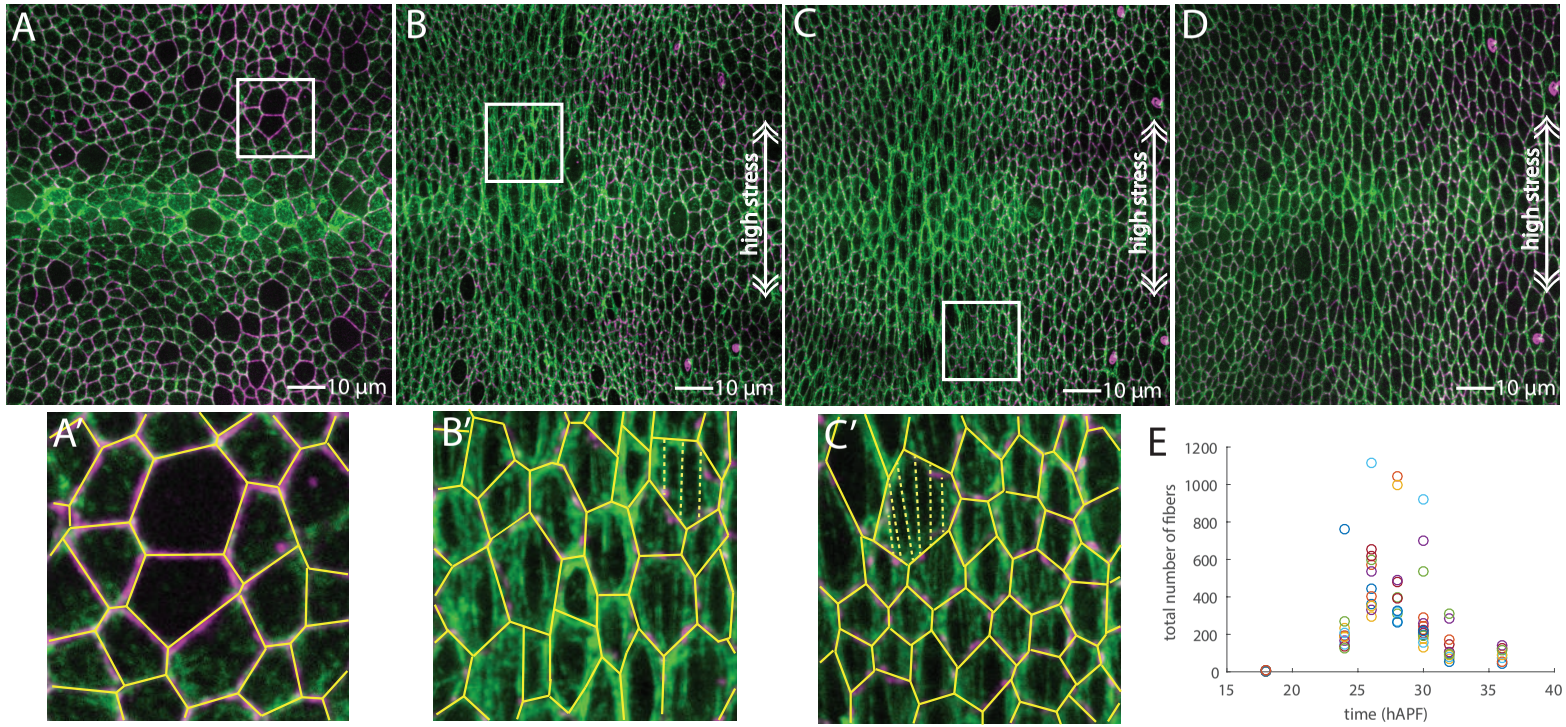

Figure 5.2: Images of the region of interest of the *Drosophila* pupal notum. Cadherin is labeled in magenta and myosin is labeled in green. Images provided by the Bellaïche lab. Images are from separate pupa imaged at (A) 18 hours APF, (B) 26 hours APF, (C) 30 hours APF, (D) 36 hours APF. Primed frames are enlarged views of the box in the corresponding unprimed frame. Cells are outlines in yellow. Dashed yellow lines identify fibers in a representitive cell. (E) plot of the total number of fibers over time in the region of interest. Each circle represents one pupa.

a hypothesis about why fibers disappear.

5.2 The toy model

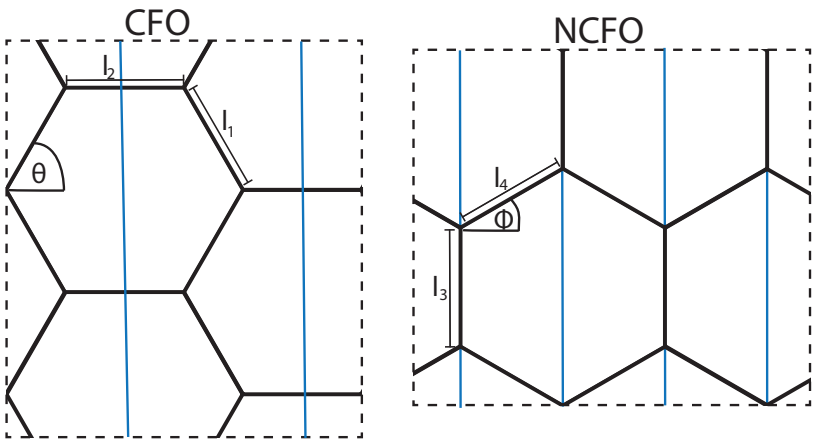

Figure 5.3: Two toy models of a regular hexagonal epithelium with fibers in blue. We assume that each tissue is under anisotropic external stress which is greater in the vertical direction. Each model is defined by six variables: two edge lengths l_i, two edge tensions γ_i, the tension on the fiber γ_f and the angle between the edges θ or ϕ.

In order to understand the mechanical implications of actomyosin fibers in cells under anisotropic stress, we created a simple mathematical model. We assume that the physics of the epithelium is well described by simple vertex model. Each edge has some contractile tension γ, which provides resistance against external stresses. For simplicity, we assume that every cell is an identical hexagon, so that the geometry of the tissue is completely described by two edge lengths and the angle of an edge with the horizontal, as shown in figure 5.3. In a tissue under vertical stress, there are two different fundamental cell orientations, which correspond to the cable forming and non-cable forming orientations described in chapter IV. We examine the behavior of cells in both orientations.

As long as fibers are vertically aligned with another fiber or edge to form a continuous line, we believe that there is no mechanical difference between a cell with three fibers and the cell with one fiber with three times the tension. For this reason, we place only a single fiber in every cell and assume that a single fiber with a large tension corresponds to the presence of a few fibers in the biological system. Although our model simplifies the complex dynamics of fibers, we believe that it captures the essential physics of the system.

For each cell orientation we can describe the tissue in terms of six variables: two

edge lengths l_i, two edge tensions γ_i, the tension on the fiber γ_f and the angle between the edges θ or ϕ. There are four physical constraints on the system. We assume that the cells are incompressible with a constant area

$$\begin{aligned} A &= 2l_1 \sin\theta \, (l_2 + l_1 \cos\theta) \qquad \text{(CFO)} \\ A &= 2l_4 \cos\phi \, (l_3 + l_4 \sin\phi) \, . \qquad \text{(NCFO)} \end{aligned} \tag{5.1}$$

In order for the tissue to be in mechanical equilibrium, we require that

$$\begin{aligned} \cos\theta &= \frac{\gamma_2}{2\gamma_1} \qquad \text{(CFO)} \\ \sin\phi &= \frac{\gamma_3 - \gamma_f}{2\gamma_4}. \qquad \text{(NCFO)} \end{aligned} \tag{5.2}$$

We assume that the tension on each edge is proportional to the concentration of myosin and that the total amount of myosin λ is fixed. This gives the constraint

$$\begin{aligned} \lambda &= 2\gamma_1 l_1 + \gamma_2 l_2 + \gamma_f (2l_1 \sin\theta) \qquad \text{(CFO)} \\ \lambda &= 2\gamma_4 l_4 + \gamma_3 l_3 + \gamma_f \, (l_3 + 2l_4 \sin\phi) \, . \qquad \text{(NCFO)} \end{aligned} \tag{5.3}$$

Finally, the difference between the vertical and horizontal stress $\sigma_D = \sigma_{yy} - \sigma_{xx}$ is given by

$$\begin{aligned} \sigma_D &= \frac{\gamma_1 \sin\theta + \gamma_f}{l_2 + l_1 \cos\theta} - \frac{\gamma_2}{2l_1 \sin\theta} \qquad \text{(CFO)} \\ \sigma_D &= \frac{\gamma_3 + \gamma_f}{2l_4 \cos\phi} - \frac{\gamma_4 \cos\phi}{l_3 + l_4 \sin\phi}. \qquad \text{(NCFO)} \end{aligned} \tag{5.4}$$

Given our six parameters and four constraints, we have two free parameters in the model, which we generally take to be the tension on the fiber and the tension on one edge.

The value of $S = P/\sqrt{A}$, where P is the perimeter is a measure of elongation. Larger values of S imply that a polygon is more elongated. Using our four physical constraints, we can solve for the value of S as a function of the tension on the fiber and the tension on one of the edges. For cells in the cable forming orientation,

$$\begin{aligned} S(\gamma_2, \gamma_f, \sigma_D, A, \lambda) &= \frac{\lambda - A\sigma_D}{\sqrt{A}\gamma_2} + \frac{\lambda + A\sigma_D}{\sqrt{A}(\tan\theta + 2\gamma_f)} \left(\frac{2 - \cos\theta}{\sin\theta} \right) \\ \text{where } \tan\theta &= \frac{\lambda^2 - A^2\sigma_D^2}{2\gamma_2^2 A} - 2\frac{\gamma_f}{\gamma_2}. \end{aligned} \tag{5.5}$$

For cells in the non-cable forming orientation,

$$S(\gamma_3, \gamma_f, \sigma_D, A, \lambda) = \left(\frac{2}{\cos\phi} - \tan\phi\right)\left(\frac{(\lambda - A\sigma_D)\tan\phi}{(\gamma_3 - \gamma_f)\sqrt{A}}\right) + \frac{\lambda + A\sigma_D}{(\gamma_3 + \gamma_f)\sqrt{A}} \quad \text{where } \tan\phi = \frac{2A(\gamma_3^2 - \gamma_f^2)}{\lambda^2 - A^2\sigma_D^2}. \tag{5.6}$$

The maximum value of the stress anisotropy allowed in our model is $\sigma_{Dmax} = \lambda/A$. This is a consequence of having restricted both the area and the total amount of myosin, which effectively restricts the tension per unit length allowed in the system.

5.3 Model predictions

By minimizing $S(\gamma_i, \gamma_f, \sigma_D, A)$ with respect to the value of the tension on the high tension edge γ_i, we can determine the minimum cell elongation as a function of the stress anisotropy and the tension on the fiber. Figure 5.4A gives the minimum elongation of a cell as a function of the stress anisotropy and the tension on the fiber for cells in either orientation. For cells in the CFO, the cell elongation is always small no matter the strength of the fiber tension. Cells in the NCFO become highly elongated at large stress anisotropies when the fiber tension is low. Figure 5.4B compares the elongation of a cell in the CFO to a cell in the NCFO with varying fiber tensions. Figure 5.4C,D gives cartoons of cells under different stresses and fiber tensions. The width of an edge represents the tension on the edge. Increasing fiber tensions are represented by the presence of more fibers in the cell. Given that cells prefer to be round, cells in the NCFO benefit more from redistributing cortex bound myosin to the fiber than cells in the CFO.

In both orientations and for any allowable stress anisotropy, there is always some distribution of myosin between the edges and fiber such that the cell is a regular hexagon. The tension on the fiber in this state is given by the white dashed line in figure 5.4A. At the minimum cell elongation we can exploit the additional geometric constraints

$$\begin{gathered} l_1 = l_2 \text{ and } \theta = \pi/3 \\ \text{or} \\ l_3 = l_4 \text{ and } \phi = \pi/6 \end{gathered} \tag{5.7}$$

to find an analytic solution to the distribution of tension between the edges and the

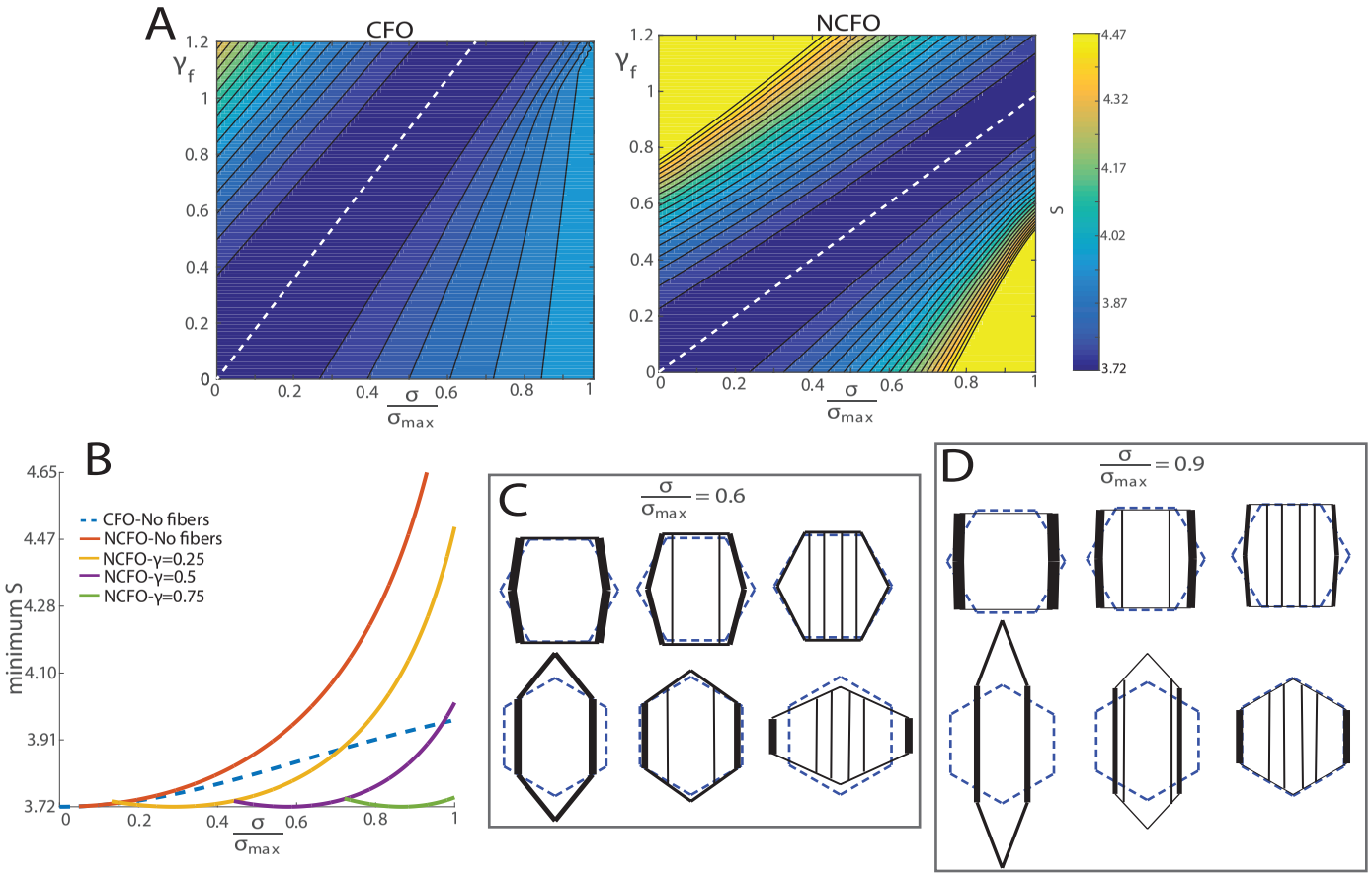

Figure 5.4: Cell elongation as a function of fiber tension and stress anisotropy. **A:** Value of the elongation as a function of the fiber tension for cells in the CFO and NCFO. Tension is scaled such that $\gamma_f = 1$ corresponds to the edge tension in the case of isotropic stress without any fibers. Fiber tensions above the white dashed line correspond to cells which are more elongated in the x-axis than the y-axis, as S does not distinguish between elongation along the x verses y-axis. **B:** Comparison of the minimum elongation of a cell in the CFO with out fibers to a cell in the NCFO as a function of the fiber tension. **C, D:** Cell shapes and edge tensions for a given stress anisotropy. Increased fiber tension is represented by more fibers. Edge and fiber widths correspond to the tension.

fiber. In the NCFO

$$\gamma_3 = \left(\sqrt{\frac{\sqrt{3}}{6}}\right)\frac{\lambda}{\sqrt{A}} \qquad \gamma_{fNCFO} = \left(\sqrt{\frac{\sqrt{3}}{6}}\right)\sqrt{A}\sigma_D \qquad \gamma_4 = \gamma_3 - \gamma_{fNCFO}. \quad (5.8)$$

and in the CFO

$$\gamma_1 = \gamma_2 = \left(\sqrt{\frac{\sqrt{3}}{6}}\right)\frac{\lambda - A\sigma_D}{\sqrt{A}} \qquad \gamma_{fCFO} = \left(\sqrt{\frac{\sqrt{3}}{2}}\right)\sqrt{A}\sigma_D \quad (5.9)$$

In both cases the tension on the fiber scales like $\sqrt{A}\sigma_D$, so that at the same stress anisotropy we expect larger cells to have more fibers.

In summery our toy model gives three predictions:

1. At constant area and stress, cells with no fibers will be less elongated in the CFO than in the NCFO.

2. At constant area, stress, and orientation, cells with more fibers are less elongated.

3. At constant stress and minimum cell elongation, larger cells will have more fibers.

5.4 Model predictions: relations between area elongation and fiber number

Since our tissue of interest has some amount of disorder, we can't apply the predictions from the toy model directly to experimental data. However, we can check some of the broader claims made by the model. Redistributing myosin from the edges to the fiber results in less elongated cells in both orientations and over all stresses. Therefore, we expect that cells with fewer fibers will be more elongated on average. Figure 5.5 gives the elongation density for all cells with the same number of fibers at each time point. Cells with more fibers are on average less elongated than cells with fewer fibers. This holds true if cell elongations are binned pupa by pupa as well (data not shown).

In order to remain minimally elongated, the tension on the fiber must scale with the square root of the area. This is physically intuitive because when cells are perfect hexagons the edges must contribute isotropically to the stress, and so any anisotropic

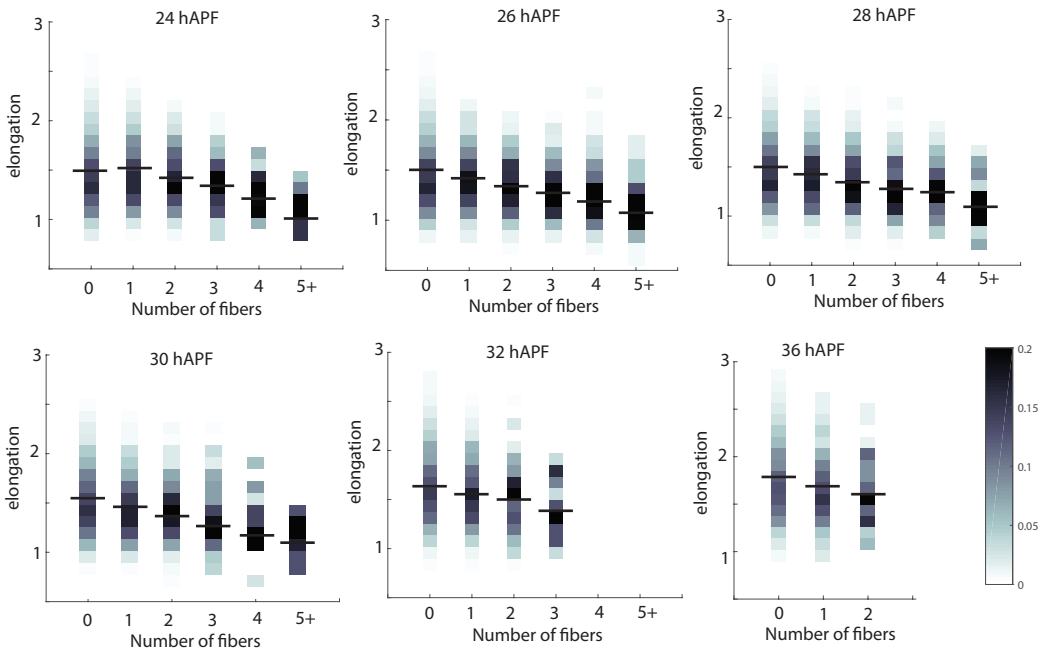

Figure 5.5: Histograms of the cell elongation by time and number of fibers. Cells with more fibers are less elongated. Color represents histogram density; black line is the mean.

stress must come from the fibers. Cells which are twice as large, are wider by a factor of $\sqrt{2}$. Since stress is a measure of force per length, the large cells must compensate for the greater distance between their edges by increasing the tension on the fiber by a factor of $\sqrt{2}$. Even though the real tissue is disordered, we expect the general principal to hold and for larger cells to have more fibers. Figure 5.6 confirms that this is the case.

5.5 Model prediction: fiber alignment and cell orientation

So far we have tested predictions 2 and 3 from our toy model without examining the cell orientation. In this section we will use physical intuition gleaned from the toy model to make predictions about fiber placement in the data, and present a possible reason for the disappearance of fibers after 31 hours.

5.5.1 Physical intuition: why are fibers more helpful in the NCFO?

From figure 5.4C,D we can see that cells in the CFO respond to the stress anisotropy by placing almost all of their myosin on their four near-vertical edges, which creates a brick like structure with relatively non-elongated cells. There is less

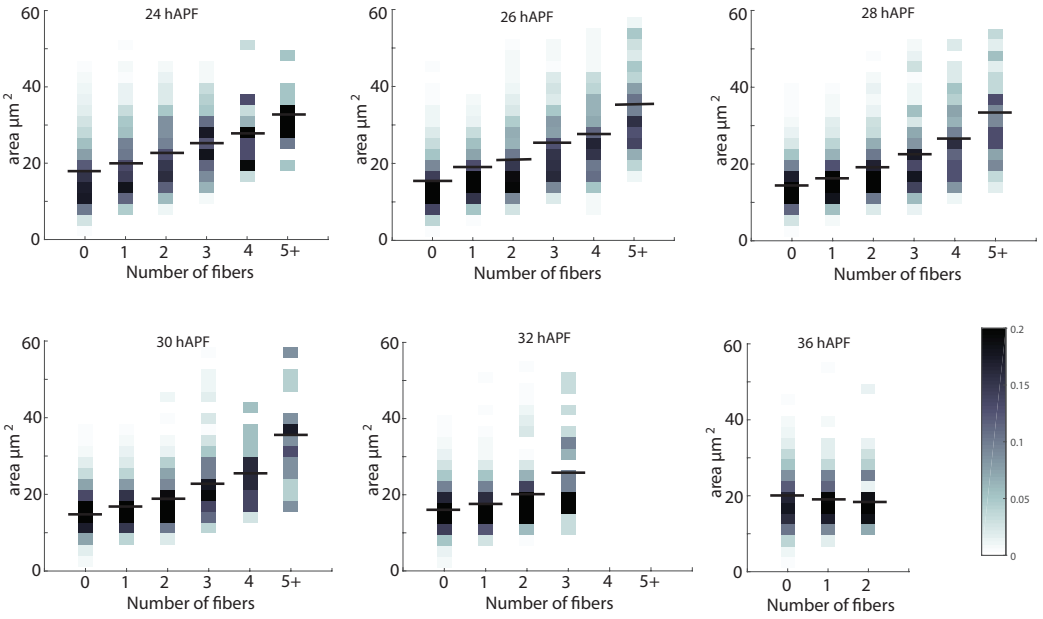

Figure 5.6: Histograms of apical cell area by time and number of fibers. Larger cells have more fibers. Color represents histogram density; black line is the mean.

need for fibers because the cell edges are able to form straight lines of tension on their own as described in chapter IV.

Cells in the NCFO are not able to use this strategy because, as they put more and more tension on their two vertical edges, the angle between the high and low tension edges must increase to preserve force balance. Cells in the NCFO lack the ability to form strait multicellular myosin chains along their edges and therefore have no way to support large stress anisotropies without becoming highly elongated. The addition of fibers keeps the cells round by creating vertical lines of tension running from fiber to cell edge to fiber. Adding fibers to the NCFO allows the tissue to form multicellular actomyosin cables where they would otherwise be prohibited. If our model is correct, we expect that, in a disordered tissue, fibers would assist in the formation of multicellular cables by aligning with vertical edges or other fibers in order to create vertical cables. We will test this hypothesis in the next section.

5.5.2 Data analysis: fiber alignment

We measured fiber alignment by examining the effect of fiber location on the minimum path through the network of cell edges and fibers. We assume that all fibers to be vertically oriented. The tissue can be represented as a digraph in which each edge represents a single cell-cell junction and associated cortex or fiber in the tissue. All edges are directed downward, as shown in figure 5.7. Each edge of the

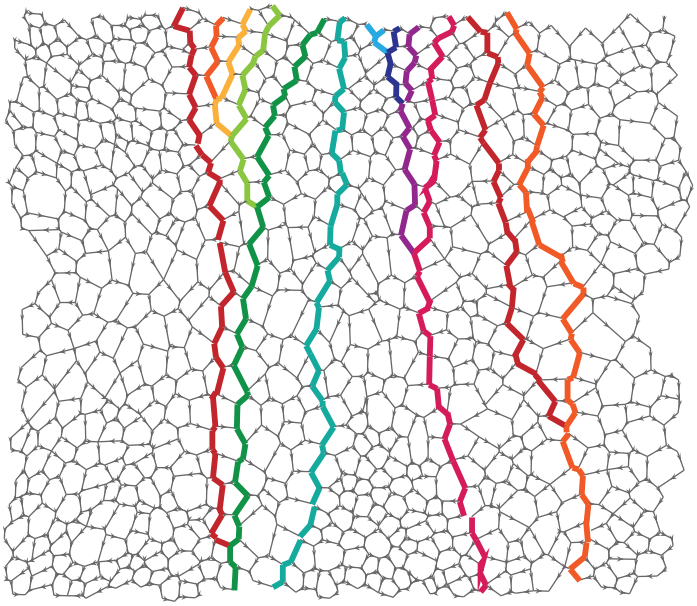

Figure 5.7: Minimum paths from each starting node to any ending node are highlighted. No fibers are present.

Table 5.1: Probability of having fiber alignment better than real alignment. Fibers are either randomly placed throughout the tissue or within their original cell. Time is measures in hours after pupa formation. For pupa where the normalized mean minimum path was lower than any value in the 100 trials a p-value of 0.01 was used.

time	p_{tissue}													combined p
24	0.02	0.01	0.01	0.01	0.35	0.01	0.01	0.01	0.02	0.01				1.722 E-09
26	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01		4.611 E-13
28	0.02	0.08	0.01	0.02	0.01	0.01	0.01	0.01	0.02	0.01	0.01			1.052 E-10
30	0.08	0.18	0.01	0.01	0.02	0.01	0.01	0.01	0.01	0.03	0.12	0.01	0.12	5.250 E-10
32	0.57	0.13	0.01	0.06	0.67	0.06	0.85	0.02	0.26	0.88	0.43			1.300 E-02

time	p_{cell}													combined p
24	0.30	0.33	0.05	0.16	0.45	0.01	0.12	0.04	0.02	0.02				1.439 E-04
26	0.03	0.01	0.01	0.01	0.07	0.04	0.06	0.06	0.04	0.05	0.02	0.05		1.437 E-08
28	0.01	0.67	0.32	0.26	0.03	0.01	0.01	0.05	0.11	0.01	0.01			8.833 E-07
30	0.38	0.29	0.06	0.01	0.19	0.10	0.27	0.02	0.01	0.02	0.03	0.01	0.37	2.480 E-06
32	0.85	0.26	0.12	0.49	0.56	0.01	0.36	0.23	0.14	0.62	0.47			1.091 E-01

digraph is given a weight proportional to its length in the tissue. Because every edge is oriented from a vertex closer to the top of the tissue to a vertex closer to the bottom, our digraph has a clear set of starting and ending nodes (nodes with either zero in or out degree). We can calculate the minimum path length from every starting node to any ending node as shown in figure 5.7. We calculate the normalized mean minimum path (NMMP) by taking the mean of the minimum paths and normalizing by the height of the tissue. Nodes near the edge of the tissue are disregarded in order to prevent boundary effects. Lower values of the normalized minimum path (NMMP) imply that many vertical edges are aligned with each other. The minimum value of the NMMP is one.

For each pupa, we calculate the NMMP of the digraph representing the true

placement of fibers in the tissue. In order to determine how well the fibers are positioned to form vertical cables, we then compare the true NMMP with the NMMP of 100 test tissues with the same edge locations but with randomized fiber placements. Fibers are either placed randomly throughout the tissue or randomly within their original cell. This allows us to calculate the probability that a fiber is placed with the same or better vertical alignment would happen by chance. It is important to use the same number of fibers in our test tissues because the deletion of an edge from a digraph will either increase the NMMP or leave it unchanged. The results are given in table 5.1, further results are given in appendix C. A p-value of 0.01 was assigned to any pupa in which the NMMP was lower than any value in the test tissues. A combined p-value for each time point was calculated using a Fisher's combined probability test, which is commonly used to combine p-values in mata-analysis studies [11, 12, 13]. The test combines multiple p-values into a single combined value P by

$$\begin{aligned} X &= -2\sum_{i=1}^{k} \ln(p_i) \\ P &= 1 - \chi^2(X, 2k). \end{aligned} \tag{5.10}$$

The test is valid as long as the original p values are independent and equally trustworthy.

The results show that the fibers are much more aligned than can be explained by chance at all times except 32 hAPF, when their are few enough fibers that their placement has very little effect on the vertical alignment of the tissue. This result supports our hypothesis that fibers play a role in resisting stress by creating multicellular actomyosin cables.

5.5.3 Data analysis: cell orientation

Given that the fibers do indeed align to form vertical multicellular cables, a potential explanation for the reduction in fibers over time despite the sustained stress is that the tissue may rearrange to become more cabley. Using the metrics developed in chapter IV, we determined the cableness of the tissues. The tissues show a mild increase in cableness between 24 and 32 hours APF, see 5.8. This increase in cableness could help explain the disappearance of the fibers.

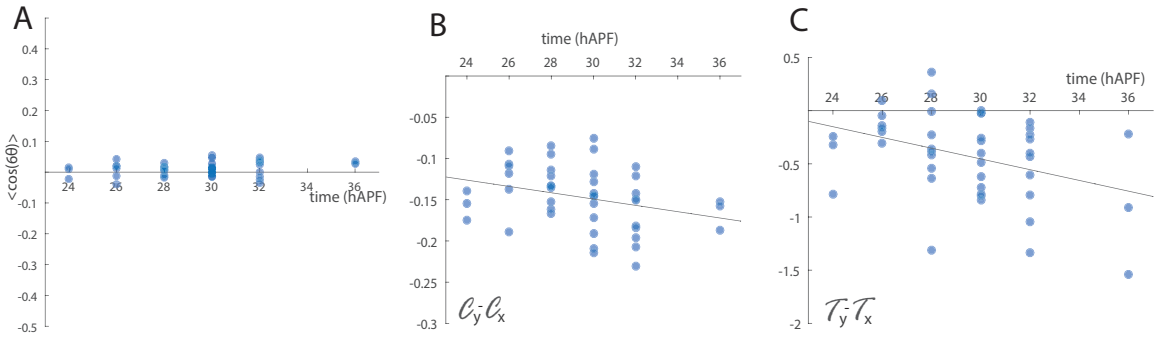

Figure 5.8: Cableness measures applied to the *Drosophila* pupal notum. **A** Cells have no orientational order in $\langle \cos(6\theta) \rangle$. **B, C** Both cableness measures decrease over time implying that the tissue rearranges to form cables.

5.6 Conclusion

The appearance of stress fibers in epithelia is a new discovery by the Bellïache lab, and we do not fully understand how and why they form. We hypothesize that fibers form in response to an applied stress anisotropy and serve to keep cells from elongating. We developed a simple model that describes how redistributing myosin from the cortex to a central fiber can decrease the elongation of cells.

This model predicts that larger cells require more fibers to stay round, which agrees with our experimental data. It also implies that fibers keep cells round by creating multicellular cables where they would otherwise be prohibited by the cell packing topology. We confirm that fibers align with each other and with vertical sections of the cortex to create multicellular cables.

Although the reasons for the disappearance of fibers after 34 hAPF are still poorly understood, the increase in cableness of the tissue and the decrease in the average cell area are likely to contribute to fiber loss.

Bibliography

[1] S. Chanet, C. J. Miller, E. D. Vaishnav, B. Ermentrout, L. A. Davidson, and A. C. Martin, "Actomyosin meshwork mechanosensing enables tissue shape to orient cell force," *Nature Communications*, vol. 8, May 15 2017.

[2] C.-P. Heisenberg and Y. Bellaïche, "Forces in tissue morphogenesis and patterning," *Cell*, vol. 153, no. 5, pp. 948 – 962, 2013.

[3] N. I. Petridou, Z. Spiro, and C.-P. Heisenberg, "Multiscale force sensing in development," *Nature Cell Biology*, vol. 19, pp. 581–588, Jun 2017.

[4] P. Chugh, A. G. Clark, M. B. Smith, D. A. D. Cassani, K. Dierkes, A. Ragab, P. P. Roux, G. Charras, G. Salbreux, and E. K. Paluch, "Actin cortex architecture regulates cell surface tension," *Nature Cell Biology*, vol. 19, Jun 2017.

[5] K. Burridge and E. S. Wittchen, "The tension mounts: Stress fibers as force-generating mechanotransducers," *Journal Of Cell Biology*, vol. 200, pp. 9–19, Jan 7 2013.

[6] J. Wang, P. Goldschmidt-Clermont, and F. Yin, "Contractility affects stress fiber remodeling and reorientation of endothelial cells subjected to cyclic mechanical stretching," *Annals Of Biomedical Engineering*, vol. 28, pp. 1165–1171, Oct 2000.

[7] R. Kaunas, P. Nguyen, S. Usami, and S. Chien, "Cooperative effects of Rho and mechanical stretch on stress fiber organization," *PNAS*, vol. 102, Nov 2005.

[8] N. Elkhatib, M. B. Neu, C. Zensen, K. M. Schmoller, D. Louvard, A. R. Bausch, T. Betz, and D. M. Vignjevic, "Fascin Plays a Role in Stress Fiber Organization and Focal Adhesion Disassembly," *Current Biology*, vol. 24, pp. 1492–1499, Jul 2014.

[9] T. Ishizaki, Y. Morishima, M. Okamoto, T. Furuyashiki, T. Kato, and S. Narumiya, "Coordination of microtubules and the actin cytoskeleton by the Rho effector mDia1," *Nature Cell Biology*, vol. 3, pp. 8–14, Jan 2001.

[10] B. Guirao, S. U. Rigaud, F. Bosveld, A. Bailles, J. Lopez-Gay, S. Ishihara, K. Sugimura, F. Graner, and Y. Bellaïche, "Unified quantitative characterization of epithelial tissue development," *ELife*, vol. 4, Dec 2015.

[11] J. Lasky-Su, "Chapter 19 - statistical techniques for genetic analysis," in *Clinical and Translational Science (Second Edition)* (D. Robertson and G. H. Williams, eds.), pp. 347 – 362, Academic Press, second edition ed., 2017.

[12] H. Dai, J. S. Leeder, and Y. Cui, "A modified generalized Fisher method for combining probabilities from dependent tests," *Frontiers In Genetics*, vol. 5, Feb 2014.

[13] F. R. A, *Statistical Methods for Research Workers*. Oliver Boyd, 1932.

CHAPTER VI

Image analysis and machine learning

6.1 Introduction

In chapter V, we developed a model for the mechanical role of fibers in epithelia. We based our model on observations of fluorescent images of the *Drosophila* pupal notum taken by the Bellaïche lab. In this chapter, we will discuss the technical aspects of working with florescent image data. Fluorescence microscopy is frequently used to image sub-cellular processes [1, 2, 3, 4, 5]. Modern techniques make it easy to collect more images than can be analyzed by hand, which creates a need for tools that can analyze florescent images in an automated or partially-automated manner. In this chapter we give an overview of some standard techniques in image processing as well as the progress we made in creating an automated fiber detection program.

We begin the chapter with an overview of classical image segmentation techniques, followed by a discussion of modern machine learning algorithms for feature classification. We will then discuss the our attempt to develop an algorithm to automatically segment fibers, including the progress we made and the barriers to our success.

6.2 Background: image segmentation

In this section we give a review of the current state of image classification, covering both traditional algorithms and applications of modern machine learning algorithms. Most of the algorithms we discuss come built in to the currently available scientific image editing software. The program ImageJ is widely used in the biological community because it is designed to work with multichannel three-dimensional images, making it especially easy to navigate through thee dimensional movies [6, 7]. It has an intuitive GUI that automatically updates images, and it allows for simple Java

based scripting. Both Matlab and Python have standard machine learning libraries; Python's skimage, and sklearn packages are particularly powerful and flexible [8, 9].

A multi-slice florescent image is composed of three-dimensional pixels called voxels. Each voxel is assigned a number corresponding to the level of florescence at that point in space. For the rest of this chapter, we will assume that the 3D image has been flattened to a two-dimensional 8-bit image. A common flattening procedure is to take the maximal z-projection of the stack where each pixel in the transformed image I has the values of the maximum voxel in the original image I' for all voxels with the same x and y coordinates,

$$I(x, y) = \max_z \big[I'(x, y, z) \big]. \tag{6.1}$$

Maximal z-projection is easy to understand and computationally efficient. However, it has drawbacks for use in florescent imaging. Any bright spots that are located above or below the pane of interest will be captured by a maximal z-projection. If the reigon of interest is located in only a few z levels this can be avoided by first cropping the image in the z-dimention. However, if the tissue was mounted at a tilt more advanced methods must be used to remove unwanted signal from above and below the level of interest. For the fiber data used in this chapter a more complicated algorithm developed in the Bellïache lab was used to flatten the stack that attempts to find the z-level with the best signal for each pixel [Guirao and Rigaud unpublished]. This allows us to avoid noise from bright spots from above or below our region of interest.

6.2.1 Classical algorithms

There are numerous filters and convolutions that can be applied to images (just look at Instagram). Here we will focus on some of the most used filters and algorithms for creating labeled images in which every pixel is assigned to some group (such as cell1, cell2, or background).

Morphological operations apply some non-linear function of a neighborhood of pixels (called the structuring element) to each pixel of the original image [10]. The most basic operations are erosion and dilation of binary images, in which the center pixel is replaced by the largest or smallest value of all pixels in the neighborhood. When applied to grayscale images, this process is called minimum or maximum filtering. The application of a maximum filter followed by a minimum filter (called morphological closing) closes holes in images as shown in figure 6.1A. Morphological

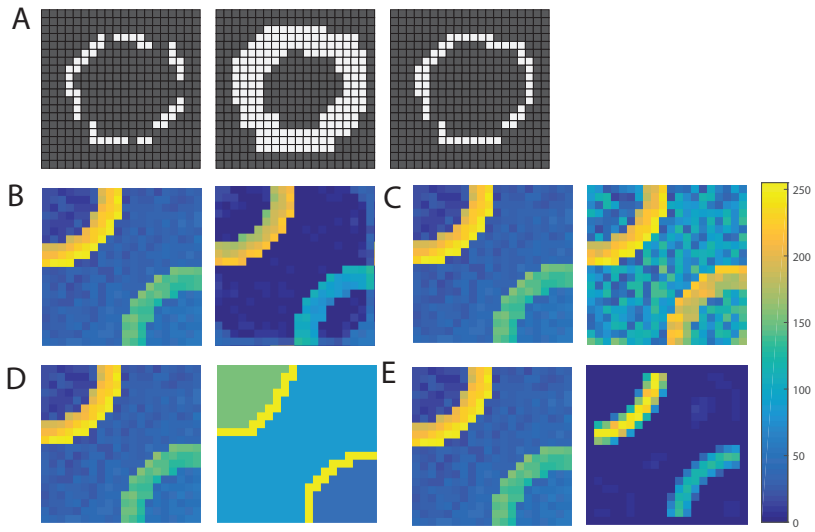

Figure 6.1: Image processing tools. Original images are on the left with processed images on the right. **A:** Demonstration of the effects of morphological opening on an image of a circle with gaps in the signal. First a maximum filter is applied, followed by a minimum filter. Each filter uses a 3×3 square structuring element. **B:** Demonstration of the effect of background subtraction. **C:** Demonstration of the effect of CLAHE local contrast enhancement. **D:** Segmentation by the watershed method is shown. **E:** The hessian transform of the original image is shown.

closing is especially useful for filling in gaps in the signal when imaging the cortex of epithelia.

Florescent images often have significant amounts of background noise, and different levels of contrast over different parts of the image. Background subtraction evens out the average intensity, and removes low levels of noise. In the most basic form of background subtraction, the mean value of all pixels in a structuring element is subtracted from the central pixel,

$$I(x,y) = I(x,y) - \text{mean}\big[I'(x',y')\big], \qquad (x',y') \in \text{strelm}(x,y). \tag{6.2}$$

If there is a large enough contrast between the foreground and background, background subtraction causes all low valued pixels to be set to zero, reducing the noise in the image. More complicated background subtraction schemes weight pixels according to their distance from the center of the structuring element [11].

For images in which the contrast differs across the image, an algorithm called Contrast Limited Adaptive Histogram Equalization (CLAHE) can be used to equalize the contrast by changing the pixel intensity based on the local histogram [12]. In simple histogram equalization image contrast is improved by flattening the histogram

of pixel values such that the cumulative distribution function of the histogram is linear. In images where their are different levels of contrast in different parts of the image simple histogram equalization will not improve contrast in riegons with lovw contrast that are either much lighter or much darker than the rest of the image. CLAHE fixes this problem by calculating the equalized histogram for pixels within a structuring element smaller than the original image. This will produce good results in the low contrast reigons of high or low intensity, but will also amplify noise. CLAHE limits the extent of the noise amplification by limiting the difference between a pixels original and transformed intensity.

Background subtraction and CLAHE are especially important tools when processing movie frames, as florescent proteins degrade as a result of the imaging process, leading to decreasing image contrast, in a process called *photobleaching* [13, 14]. It is important to consider photobleaching when designing florescent imaging experiments, as it leads to a trade-off between the number of images taken and the image quality.

The ultimate goal of automated image processing is to identify features in an image, called *image segmentation*. A common method of image segmentation is the creation of a new image, with the same dimensions as the original, in which the value of each pixel represents a label for the feature. For example, we might give all background pixels a label of 0 and then label each individual cell with a different positive integer.

The watershed algorithm produces a labeled image in which each region contains one local minimum [15, 16]. We use this algorithm to label cells in images where the cell-cell boundaries are marked. The algorithm works by first identifying all local minima of the image within some noise tolerance. Each minimum represents the source of a different region. Each region is 'flooded' by adding neighboring pixels to the region, starting with pixels with the lowest intensity, until the regions collide. The watershed algorithm works well for finding the barrier between regions of low signal; however, it does not perform well when the local minima are shallow and the barrier between the regions contains gaps.

Another method of identifying edges between two regions in an image is through the hessian transformation,

$$H = \begin{bmatrix} \frac{\partial^2 I}{\partial x^2} & \frac{\partial^2 I}{\partial x \partial y} \\ \frac{\partial^2 I}{\partial y \partial x} & \frac{\partial^2 I}{\partial y^2} \end{bmatrix}. \tag{6.3}$$

In order to approximate the derivative in a discreet system Gaussian smoothing with a specifies length scale is first applied, and then the dirivitive is calculated as a finite difference. Let the largest eigenvalue of $\mathbf{H}$ evaluated at (x, y) be λ_1. The transformed

image is given by

$$T(x,y) = \begin{cases} -\lambda_1 & \lambda_1 < 0 \\ 0 & \lambda_1 \geq 0 \end{cases} \tag{6.4}$$

Regions of high intensity in the hessian transformed image correspond to regions in the original image in which there is high signal along one axis that decays like a Gaussian in the perpendicular axis. The hessian transformation is useful because it is able to identify edges, even when they contain gaps.

6.2.2 Machine learning based classification

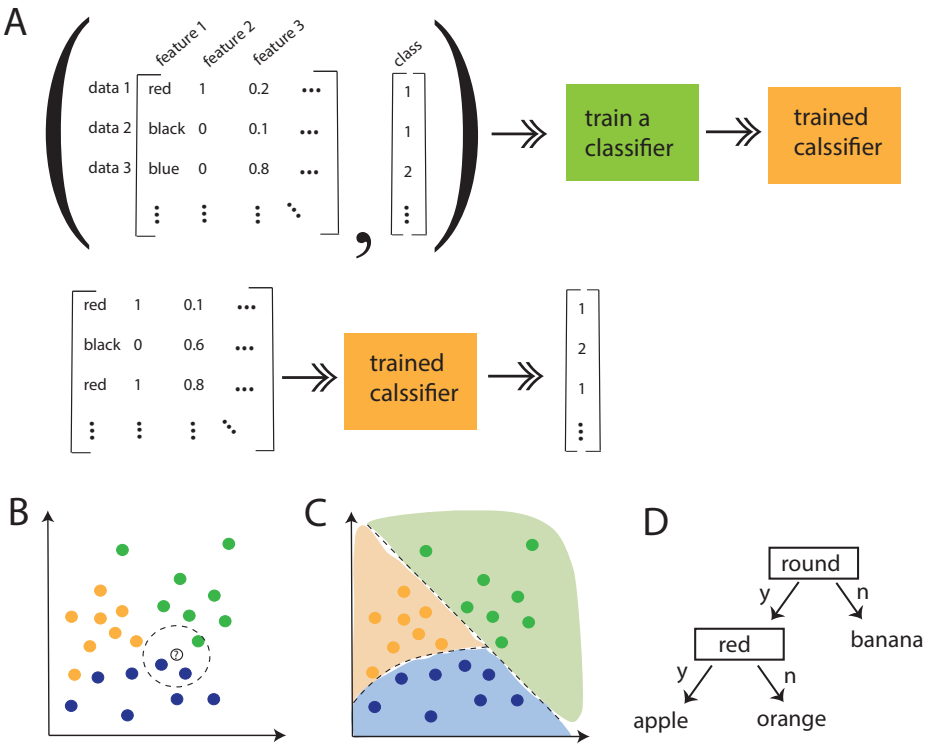

Figure 6.2: **A:** Work flow of a machine learning classifier. The feature matrix X and the classifications Y for a known set of data are used to train the classifier. Once a classifier has been trained, it can be applied to new data to produce classifications. **B:** Cartoon of k-nearest neighbor classification. The data is plotted in a k-dimensional feature space. New data is classified according to the majority classification of existing data within a k-sphere of some radius. **C:** Cartoon of support vector machine (SVM) classification. The classifier attempts to separate the data in phase space. SMV can be linear, quadratic, or based on a more complex partitioning of feature space. **D:** Cartoon of a decision tree classifier. Decision trees attempt to split data one feature at a time.

A recent development in automated feature detection is the use of machine learning for classification of features. Most classifiers follow the same broad work flow (the most notable exception being neural networks which are beyond the scope of this chapter). A cartoon of the work flow is show in figure 6.2D. First, a human identifies features of the data likely to be useful in classifying it. For example, a researcher looking to classify different fruits might choose the color, shape, weight, and number of seeds as features. For each data point, the values of the features are recorded in a matrix X. Additionally, the classification of all of the data is recorded in a vector Y. Together, X and Y are used to train the chosen classifier. Once the classifier has been trained, it can be used to classify new data points as long as their features have been recorded. The accuracy of the classifier can be determined by using it to classify a set of known data and examining the error rate.

There are numerous machine learning classifiers, each of which has many variants. We will briefly cover three different types of classifiers: k-nearest neighbor, support vector, and decision trees. Each of these classification algorithms requires initial human intervention to segment the image and determine an appropriate feature set. Each data point corresponds to one region of the segmented image. Common features for image analysis include the: size and shape of the region as well as the pixel intensity.

Both k-nearest neighbor and support vector machine (SVM) classification use the location of the data in the k-dimensional feature space to classify the data, where k is the number of features. In k-nearest neighbor classification, new data is assigned a class based on the majority classification of all previously classified data within a k-sphere of a given radius. The radius of the sphere is a *hyperparameter* - a parameter of the classification algorithm itself. Classifiers are governed by a number of hyper-parameters. The best way to set these hyperparameters is a wide open question, in which randomly sampling from the parameter space remains one of the best options [17].

SVM classification attempts to partition feature space into compact regions each of which contains one class of data, see figure 6.2. The partitions can be linear, quadratic, or use a more complex partitioning of feature space [10].

Decision tree classifiers generate a tree structure which partitions the data one feature at a time, see figure 6.2D. Decision trees have the advantage of being easy to interpret, with the drawback that they are not generally as accurate. The accuracy of decision trees can be increased by using an ensemble of trees called a random forest [10].

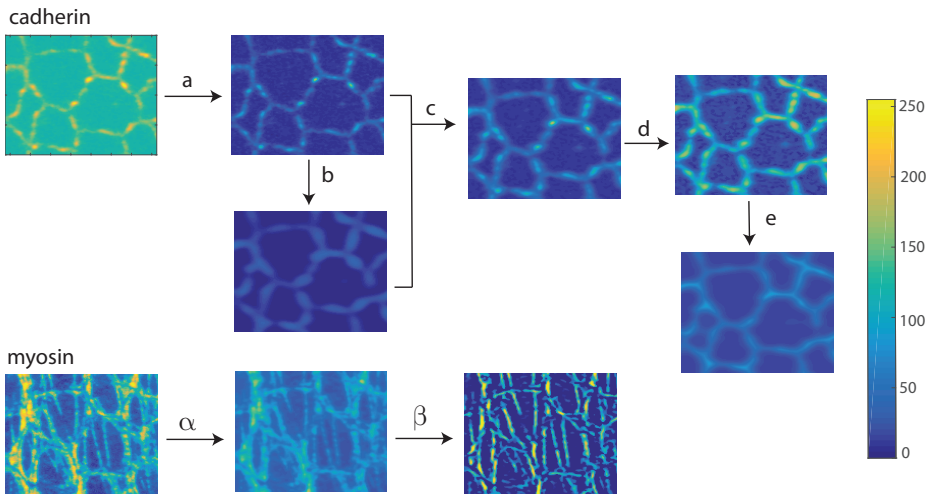

Figure 6.3: Images showing different stages in the pre-processing of the cadherin and myosin signal. **a:** Subtract the background from the original image. **b:** Calculate the hessian and and apply CLAHE to the hessian image. **c:** Add the hessian back into the processed image. Apply another round of background subtraction, and blur the image to close gaps in the signal. **d:** Increase the contrast by CLAHE. **e:** Apply morphological closing to the image. α**:** Subtract the background from the original image. Additionally multiply the image by a highly blurred version of the image to remove areas with low signal. β**:** Calculate the hessian of the processed image.

6.3 Automated fiber detection

Our goal was to segment and label all of the fibers from the data we collected. This segmentation is especially challenging as we do not know of any protein that is only localized on the fibers. Therefore any time we image fibers, we will pick up unwanted signal from the actomyosin cortex which we must remove. The fiber-cortex system was marked with triple tagged gfp myosin, a molecular motor. We took multichannel images in which we marked both the adherens junctions and the cortex-fiber system with the goal of using the adherens junction signal to remove unwanted actomyosin cortex signal from the myosin channel. The cells were marked by cadherin-mKate, a cell adhesion molecule that localizes at the adherens junctions.

We developed a two stage pipeline for fiber segmentation. The first stage segments both the myosin and cadherin channels and subtracts them to obtain an image with segmented fibers. The second stage we uses machine learning algorithms to remove any misidentified fibers from the result of the first stage. In the next two sections, we will describe this procedure in more detail.

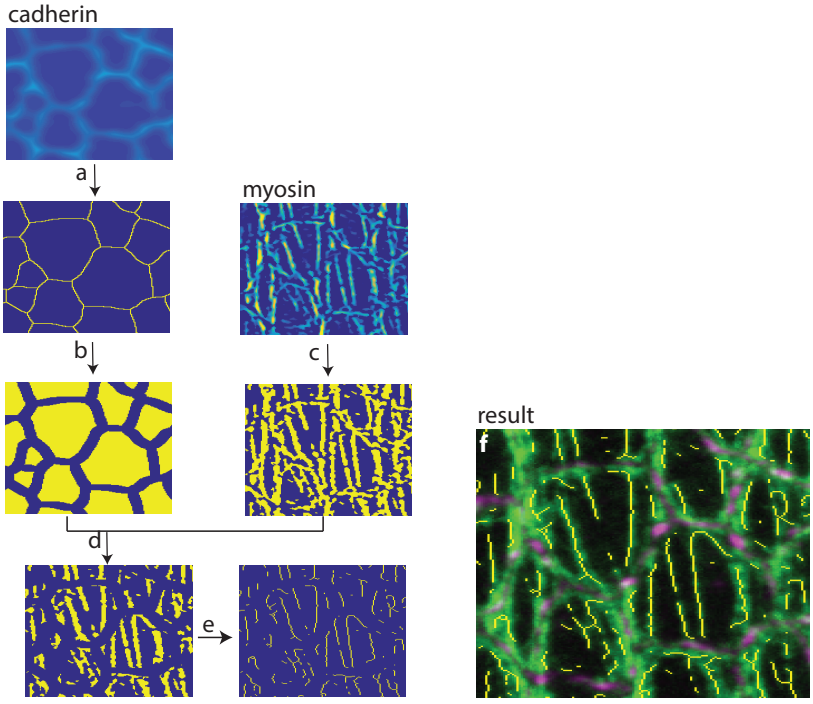

Figure 6.4: Images of different stages in the post-processing of the cadherin and myosin signal. **a:** Cadherin signal is segmented by watershed algorithm. **b:** Dilation of the image to match the thickness of the cortex. **c:** The myosin signal is thresholded. **d:** The myosin and cadherin signal are multiplied to remove unwanted signal from cortex bound myosin. **e:** Image is skeletonized. **f:** Overlay: cadherin is magenta, myosin is green, and potential fibers are yellow.

feature	mean ± std (true fibers)	mean ± std (false fibers)
length	32.3 ± 21.8	18.6 ± 9.9
extended length	52.2 ± 53.9	49.2 ± 71.9
orientation	0.13 ± 0.11	0.18 ± 0.18
curvature	5.5 ± 12.5	2.8 ± 3.7
mean signal	1.23 ± 0.11	1.14 ± 0.12
std signal	0.15 ± 0.05	0.13 ± 0.06
extended mean signal	1.2 ± 0.1	1.1 ± 0.1
extended std signal	0.16 ± 0.06	0.16 ± 0.08
distance to cortex	6.9 ± 3.8	4.4 ± 4.4
cell size	2.1 ± 1.1	2.1 ± 1.3

Table 6.1: Features used in machine learning based classification. All values are in arbitrary units. True fibers represents the set of fibers that were true-positives and identified by the classical segmentation.

6.3.1 Stage one: segment potential fibers

Before segmenting the images, we first preprocess them to remove the effects of noise and incomplete signal. Figure 6.3 shows the major steps in the pre-processing procedure. Noise in the cadherin signal is removed by background subtraction. Gaps in the edges between cells are filled in by adding the hessian of the image to itself. Since the hessian is brightest along tube-like structures this helps smooth out the cell-cell boundaries. We then increase the contrast by CLAHE and apply a morphological closing to further fill in gaps. For the myosin signal we apply a background subtraction and then take the hessian of the image.

Once the images have been preprocessed, they are segmented. This process is shown in figure 6.4. The cadherin signal is segmented using a watershed algorithm and then dilated by a structuring element with the same width as the cortex. The myosin signal is thresholded to produce a binary image. The cadherin and myosin binary images are multiplied to remove the cortex from the myosin signal. Finally, the image is skeletonized; each connected component is replaced by a line. Figure 6.4f shows the result of this procedure overlaid on the original signal. At this point in the process, our goal is to identify every fiber even if we also identify many false positives, since we will remove the false positives with our machine learning classifier.

6.3.2 Stage two: remove misidentified fibers through machine learning classification

Table 6.3.2 lists the ten features we used in our classifiers. We used both the initial segmentation of the fiber and a version of the segmentation in which we extended all fibers to the cortex. The following results are from one pupa with 759 fibers as identified by hand. Our classical segmentation yielded 1150 potential fibers 532 of which corresponded to true fibers. The classical segmentation failed to find 227 fibers, some of which were too close to the cell-cell boundary and cut off by the mask, and some of which had poor signal. The number of unidentified fibers could be decreased by decreasing the threshold with the trade off that more false positives would be identified.

Figure 6.5 gives the confusion matrix for the best classifier that we found within each type of major classifier. The classifiers were trained in Matlab's classification learner application. Results did not differ significantly for classifiers trained in python's sklearn environment.

The initial results were not promising. No single classifier preformed better than 80% accuracy for any metric. It is possible that increasing the volume of the data would increase the classifier accuracy. However, their is greater variation between pupa than within pupa, so it is plausible that additional data would decrease the classifier accuracy. It is highly time consuming to create labeled data, both in terms of human and computational time. Because we were not confident of achieving any significant increase in classifier accuracy with more data, we decided to focus our efforts on classifying fibers by hand.

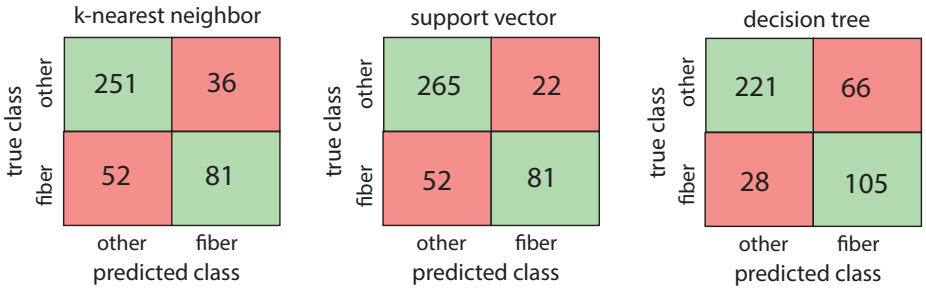

Figure 6.5: Confusion matrices for the best hyper-parameters we found for each type of classifier. Classifiers were trained on 75% of the data and validated on the 25% holdout. The support vector classifier has the best false positive rate at 21%. The (RUS boosted) decision tree has the best false negative rate at 21%. All classifiers had unsatisfactory performance.

6.4 Discussion

There is a growing use of image processing software to assist in data analysis of biological images [6, 18, 19]. Classical morphological operations and segmentation algorithms are frequently used to identify objects in images. More recently, machine learning algorithms have been adapted to help classify biological data that is too large to be classified by hand [20]. Here we described our work in developing an algorithm for automated fiber detection that was ultimately unsuccessful. We were very successful at segmenting the cell-cell boundaries through classical algorithms. We were able to identify a set of potential fibers by using the cortex signal to mask out cortex bound myosin. However, we were unable to produce a machine learning classifier capable of removing the false positives from our set of potential fibers with a high enough accuracy to be useful.

A possible explanation for the poor performance of our classifier is that fibers are difficult to classify in isolation, even for humans. When we looked at cropped images of potential fibers in isolation, the scientists in our lab were not able to determine the true fibers with any accuracy. This suggests that humans identify fibers on the basis of larger structural properties of the image, which were not included in the feature set passed to the classifiers. Neural networks are good at identifying structural properties of images and are especially useful when an appropriate feature set is unclear as they identify the appropriate features autonomously. Greater accuracy could be achievable with a neural network classifier, although these classifiers do not remove the need for initial segmentation of the image as they determine only whether a feature is present, not the location of the feature.

Bibliography

[1] P. van Roessel and A. Brand, "Imaging into the future: visualizing gene expression and protein interactions with fluorescent proteins," *Nature Cell Biology*, vol. 4, pp. E15–E20, Jan 2002.

[2] J. Zhang, R. Campbell, A. Ting, and R. Tsien, "Creating new fluorescent probes for cell biology," *Nature Reviews Molecular Cell Biology*, vol. 3, pp. 906–918, Dec 2002.

[3] J. Lippincott-Schwartz and G. H. Patterson, "Development and use of fluorescent protein markers in living cells," *Science*, vol. 300, no. 5616, pp. 87–91, 2003.

[4] D. M. Chudakov, S. Lukyanov, and K. A. Lukyanov, "Fluorescent proteins as a

toolkit for in vivo imaging," *Trends in Biotechnology*, vol. 23, no. 12, pp. 605 – 613, 2005.

[5] R. Yuste, "Fluorescence microscopy today," *Nature Methods*, vol. 2, pp. 902–904, Dec 2005.

[6] C. A. Schneider, W. S. Rasband, and K. W. Eliceiri, "NIH Image to ImageJ: 25 years of image analysis," *Nature Methods*, vol. 9, pp. 671–675, Jul 2012.

[7] J. Schindelin, I. Arganda-Carreras, E. Frise, V. Kaynig, M. Longair, T. Pietzsch, S. Preibisch, C. Rueden, S. Saalfeld, B. Schmid, J.-Y. Tinevez, D. J. White, V. Hartenstein, K. Eliceiri, P. Tomancak, and A. Cardona, "Fiji: an open-source platform for biological-image analysis," *Nature Methods*, vol. 9, pp. 676–682, Jul 2012.

[8] S. van der Walt, J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart, and T. Yu, "scikit-image: image processing in python," *PeerJ*, vol. 2, p. e453, June 2014.

[9] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[10] P. Soille, *Morphological Image Analysis: Principles and Applications*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004.

[11] S. R. Sternberg, "Biomedical image processing," *Computer*, vol. 16, pp. 22–34, Jan 1983.

[12] S. Pizer, E. Amburn, J. Austin, R. Cromartie, A. Geselowitz, T. Greer, B. ter Haar Romeny, J. Zimmerman, and K. Zuiderveld, "Adaptive histogram equalization and its variations.," *Computer Vision, Graphics, and Image Processing*, vol. 39, pp. 355–368, 1 1987.

[13] J. Lichtman and J. Conchello, "Fluorescence microscopy," *Nature Methods*, vol. 2, pp. 910–919, Dec 2005.

[14] H. C. Ishikawa-Ankerhold, R. Ankerhold, and G. P. C. Drummen, "Advanced Fluorescence Microscopy Techniques-FRAP, FLIP, FLAP, FRET and FLIM," *Molecules*, vol. 17, pp. 4047–4132, Apr 2012.

[15] L. Vincent and P. Soille, "Watersheds in digital spaces: An efficient algorithm based on immersion simulations," *IEEE PAMI, 1991*, vol. 13, no. 6, pp. 583–598, 1991.

[16] Roerdink and Meijster, "The watershed transform: Definitions, algorithms and parallelization strategies," *FUNDINF: Fundamenta Informatica*, vol. 41, 2000.

[17] J. Bergstra and Y. Bengio, "Random Search for Hyper-Parameter Optimization," *Journal of Machine Learning Research*, vol. 13, pp. 281–305, Feb 2012.

[18] A. H. K. Roeder, A. Cunha, M. C. Burl, and E. M. Meyerowitz, "A computational image analysis glossary for biologists," *Development*, vol. 139, no. 17, pp. 3071–3080, 2012.

[19] S. Uchida, "Image processing and recognition for biological images," *Development Growth & Differentiation*, vol. 55, pp. 523–549, May 2013.

[20] C. Sommer and D. W. Gerlich, "Machine learning in cell biology - teaching computers to recognize phenotypes," *Journal Of Cell Science*, vol. 126, pp. 5529–5539, Dec 15 2013.

CHAPTER VII

Honeycomb Lattices with Defects

Notes: Adapted from Spencer, Meryl A., Ziff, Robert M., "Honeycomb lattices with defects," *Phys. Rev. E*, **93**, 042132, 2016.

7.1 Introduction

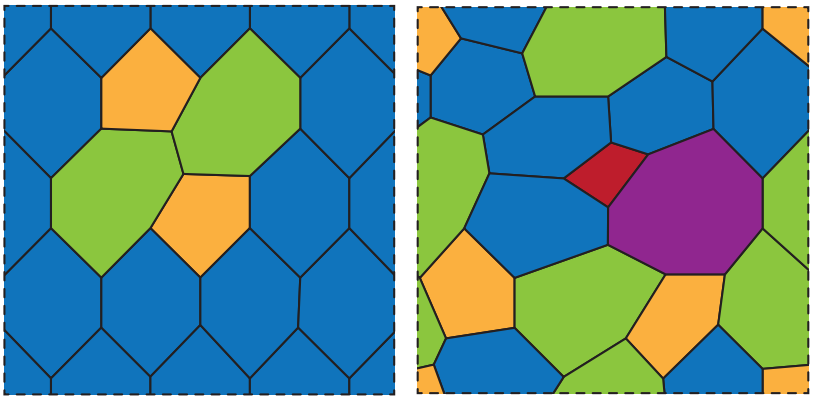

Figure 7.1: Two examples of defective honeycomb (DHC) lattices. Each lattice has 16 tiles and periodic boundary conditions. Red tiles are rectangles, yellow tiles are pentagons, blue tiles are hexagons, green tiles are heptagons, and purple tiles are octagons. **Left:** Example of a DHC lattice with standard deviation in degree of the dual $\sigma = 0.5$. **Right:** Example of a DHC lattice with $\sigma = 1.658$.

In the final chapter of this thesis, we will turn our attention away from biophysical questions and study the mathematical structure of epithelial networks. In the lowest energy state every cell in a vertex model is a hexagon, the generic term for this lattice structure is *honeycomb*. Honeycomb lattices are extremely widespread, forming the structure of many natural and artificial objects. Atoms in graphene and other crystals are arranged in a honeycomb pattern, as are the bubbles that form in thin foams

[1]. Consequently condensed matter and materials science researchers have a great interest in the properties of this lattice. Biologists study cells in epithelial sheets, which arrange themselves into honeycomb lattices [2], not to mention the lattice's namesake, the beehive. The lattice is common in art and architecture too; you might even be sitting in a room with a honeycomb-tiled floor at this very moment!

The ideal honeycomb lattice used in art, architecture and math has been well studied (i.e., [3, 4, 5]), but naturally occurring systems are rarely perfect. Physical systems will almost always contain some defects which cause the systems to vary from the ideal lattice to some degree. Some random lattices already exist to model these situations. One of the most well known is the Poisson-Voronoi tessellation, which is generated by randomly placing a points on a two-dimensional surface in a Poisson distribution and then placing bonds equidistant between each neighboring point. Though this lattice is a good general approximation for many three-coordinated physical systems, the variance of the number of faces per tile is always approximately 1.314 for large graphs [6] , though physical systems may have a different variance. Our goal is to create a graph with the same general features of the Poisson-Voronoi, but whose variance in the number of faces per tile is a parameter, which can be tuned to more closely match the variance found in the natural system being studied.

We present a new lattice network which we call the Defective Honeycomb lattice (DHC) formed by systematically swapping bonds in the honeycomb lattice, as shown in figure 7.1. The purpose of this lattice to model planar three-coordinated systems with a range of distributions of faces per tile. An example of such a system is the bonds between atoms in a material in an amorphous glass solid such as SiO_2. The atoms form a regular hexagonal pattern in the crystalline lattice and a more disordered pattern in the glassy phase [7]. Another example of a system with a changing level of defects comes from biological networks. The planar cells in epithelia can be modeled as tiles on a lattice. As tissues undergo development the cells more relative to one another which changes the properties of the network. Eaton and Julicher [8] have shown that the average number of neighbors of the cells in the Drosophila wing epithelium changes significantly between different phases of development. Percolation is relevant to understanding transport of morphogens though the tissue. Our lattice is also useful in describing natural systems with fixed defects whose variance in number of faces per tile is much higher or lower than the variance of the Poisson-Voronoi.

In this chapter we will describe how the basic properties of the lattice change as more bonds are flipped, as well as giving an algorithm for computationally creating such lattices. The percolation threshold describes the connectivity of a lattice. If

sites or bonds are occupied with some probability p, the percolation threshold is the minimum value of p at which there is a connected path from one side of the lattice to the other, in the limit of an infinite system [9]. The percolation thresholds are known for many lattices, including the honeycomb lattice for bond percolation. In this chapter we will compare the percolation thresholds of several known two-dimensional three-coordinated lattices with our DHC lattices, which are also three coordinated. We will show that there is a strong relationship between the percolation threshold and the variance in the number of faces in the polygons, equivalent to the degree of vertices on the lattice duals.

7.2 Defective Honeycomb (DHC) Lattice

Many naturally occurring biological systems, including cells in epithelial tissues, are approximate honeycomb lattices [8, 10, 11], but there is little understanding of the formal properties of these lattices. Understanding how defects in these lattices affects their properties will give us a more accurate picture of how naturally occurring honeycomb networks behave.

7.2.1 Definition

The inspiration for the DHC lattice is the T1 topological process in foams [1]. In the T1 process one edge separating two bubbles shrinks down to a point and then regrows in a perpendicular direction. This has the effect of swapping which bubbles are neighbors, and the number of neighbors of the four bubbles involved. When viewed as a lattice this process is equivalent to rearranging the five bonds between six neighboring vertices connected in an H shape as shown in figure 7.2.

The DHC lattice is a honeycomb lattice in which a certain number of the lattice bonds have been flipped. The DHC lattice is defined by two parameters: n, the number of hexagonal tiles per row in the original $n \times n$ hexagonal lattice as arranged in figure 7.3 (top left), and F, the number of bonds which are flipped. To generate the lattice F bonds are chosen uniformly at random; the same bond may be chosen more than once. The order in which the bonds are flipped matters, as flipping bond a–b and then b–c is not equivalent to flipping bond b–c and then a–b.

In order to consistently describe lattices of different size we will specify $f = F/(3n^2)$, the fraction of edges flipped, instead of F, to characterize the number of flipped bonds in the lattice.

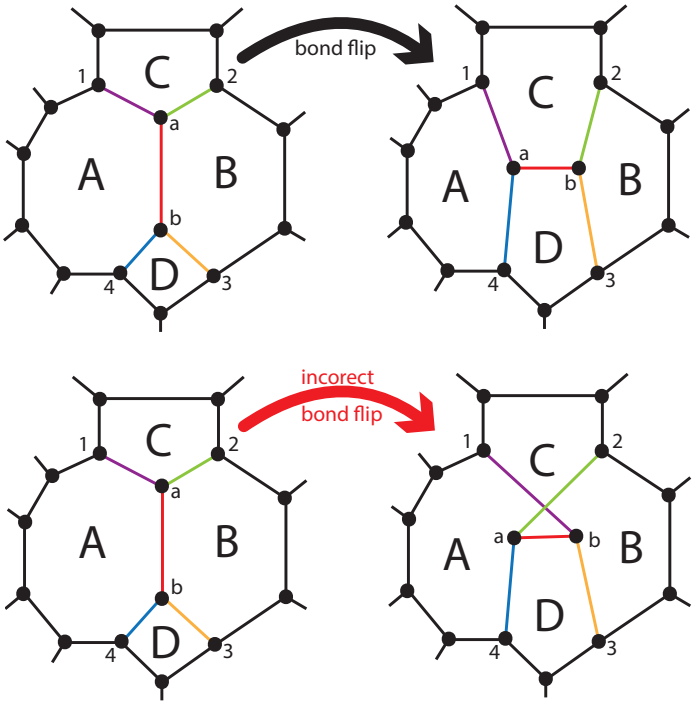

Figure 7.2: Pictorial representation of the flip of bond a–b in the DHC lattice. The bonds between vertices 1, 2, 3, 4, a, and b, are rearranged such that tiles C and D become neighbors instead of A and B. **Top:** Correct flip. The bond 2–a becomes 2–b and the bond 4–b becomes 4–a. The lattice remains planar and the bond a–b separates two new tiles. **Bottom:** Incorrect flip. There is no way to arrange sites a and b in the plane such that none of the bonds cross.

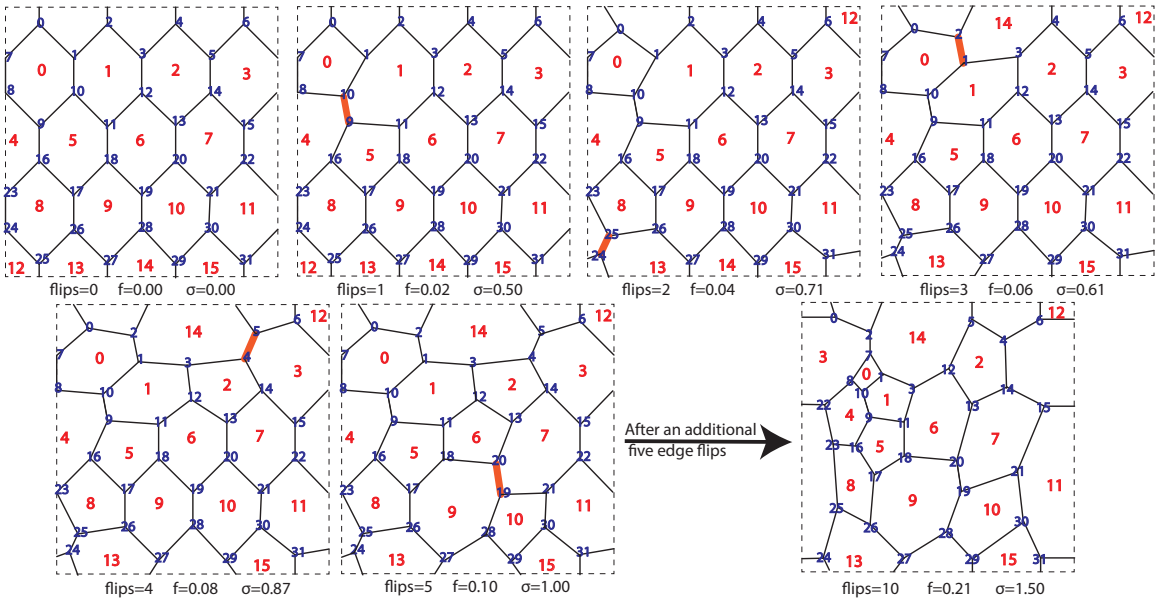

Figure 7.3: Pictures of the generation of an $n = 4$, $f = 10/48 = 0.21$ lattice. Ten distinct edges are flipped one at a time to generate the final lattice. Each panel shows the state of the lattice after each edge flip. The flipped edge is highlighted in red. The sites and tiles are labeled in blue and red respectively as described in apendix D.

7.2.2 Example

Figure 7.3 shows the generation of a 4×4 lattice with 10 flips, so $f = 10/48 \approx 0.21$. In the first panel none of the edges have been flipped and the system is the original honeycomb configuration. In the second panel the first bond chosen, bond 9–10 shown in red, has been flipped. This has the effect of creating two pentagonal tiles and two heptagonal tiles in the otherwise unperturbed honeycomb lattice, increasing the standard deviation in the number of edges per tile, σ. As the edges are flipped the lattice becomes more and more deformed and σ increases to 1.50. The quantity σ, which is the same as the standard deviation in the coordination number of sites on the dual lattice, is important since it gives a quantitative measure of the extent of the difference between the new lattice and the original honeycomb. It is defined as

$$\sigma = \sqrt{\frac{1}{n^2}\sum_i (z_i - 6)^2} \tag{7.1}$$

where z_i is the number of edges of the i-th polygon face, equal to the number of neighbors of the i-th vertex in the dual lattice. The number 6 represents the mean value of z_i, which follows from Euler's formula, vertices$-$edges$+$faces $= 2$, assuming

n is large, and also from the fact that with each flip, the total number of bonds, tiles, and vertices is unchanged from the original honeycomb values. We also consider the variance $v = \sigma^2$.

7.2.3 Properties

There are several important properties of the DHC lattice. The lattice is two dimensional, so the percolation critical exponents of the system are known [9]. When a bond is flipped, four tiles are affected. Two of them (tiles A and B in figure 7.2) lose a bond and go from n-gons to $(n-1)$-gons, while the other two (tiles C and D in figure 7.2) go from n-gons to $(n+1)$-gons. Therefore the average number of edges per tile (equivalently the average degree on the dual lattice) remains constant at 6. As bonds are flipped the standard deviation in the number of edges per tile changes. The flipping of a bond changes the connection between sites, but each site continues to have exactly 3 neighbors. Equivalently the dual lattice remains fully triangulated. We will use these properties to define a class of planar lattices, the honeycomb variant class, where each lattice is two-dimensional, three-coordinated and has an average dual degree of six. We will compare the percolation thresholds of the DHC to the rest of the honeycomb variant class of lattices in section 7.5.

7.3 Generation of Lattices

In the remainder of this chapter we will determine additional properties of the DHC lattices through computational simulation. The process of generating the lattices is mostly straightforward, however there are a few non-trivial components of the algorithm which we will address here.

For physical reasons, we would like our lattice to remain planar. Furthermore, for percolation, the critical exponents have different values in different dimensions, and may change if the lattice is non-planar. As we flip bonds we need a way to guarantee that our lattice stays planar. The specific coordinates of the sites are not relevant, but for an acceptable bond flip there must exist some arrangement of the sites on the plane for which none of the bonds cross. It turns out that the need for a planar representation of the lattice specifies a unique exchange of bonds for any flip trial. Figure 7.2 shows the two topologically distinct ways to flip bond a–b. In both resulting lattices tiles A and B lose bond a–b and tiles C and D gain bond a–b; however they differ in the reconnection of the bonds between sites 1, 2, 3 and 4, and the flipped sites a and b. In the top picture bond 2–a is broken and replaced by bond 2–b and

bond 4–b is broken and replaced by bond 4–a. In the bottom picture bond 1–a is replaced by bond 1–b and bond 4–b is replaced by 4–a. The first set of reconnections are correct because the lattice has a planar representation which is shown. In the bottom case there is no planar representation on the lattice—no matter where sites a and b are located there will be some bonds that cross each other—therefore we need to be careful when reconnecting the bonds after a flip to preserve the planar nature of the lattice.

Lattices are often stored on computers as simple neighbor lists, however here a neighbor list would not be able to determine which of the two distinct ways to flip bond a–b would result in a planar representation. In each case sites a and b exchange one neighbor with each other, but without knowledge of which neighbors belong to which tiles it is impossible to determine if sites 2 and 4 should be exchanged or sites 1 and 3. In order to solve this problem we stored our lattices as a list of all of the sites that make up each tile. The sites were stored in clockwise order to further simplify the flipping algorithm. For example $\texttt{tilelist}[3] = [1, 5, 2, 18, 7]$ denotes that the tile with ID of 3 is a pentagon made of the sites 1, 2, 5, 7, 18, and bonds 1–5, 5–2, 2–18, 18–7 and 7–1. Storing the lattice in this way makes the algorithm given in algorithm 1 for flipping bond a–b simple. It is also easy to translate into the standard neighbor list as the three neighbors of site 2 are all of the IDs immediately following 2 in the entire tile list.

In order to use algorithm 1 to flip bonds in the lattice we must start with the lattice as lists of clockwise sites in every cell. We used the labeling method shown in figure 7.3 to determine tile and site IDs. Tiles are initially labeled 0 to $n^2 - 1$ in order by row and then column. The top site of each hexagon initially has an ID of $\mathrm{ID}_{\mathrm{site}} = 2(\mathrm{ID}_{\mathrm{tile}})$ and the next clockwise site has $\mathrm{ID}_{\mathrm{site}} = 2(\mathrm{ID}_{\mathrm{tile}}) + 1$. This produces a label for every site and tile in the lattice. The lattice is then initialized according to the algorithm given in appendix D.

7.4 Relationship between flips and defects

The DHC lattice is characterized by the number of bonds flipped, and we would like to know how the lattice transforms in the process. We flip the bonds randomly, so we will look at the average behavior over many lattices. To quantify the 'defectiveness' of an individual lattice we calculate the σ of the degree of the dual lattice (7.1). We created a set of independent DHC lattices of three different sizes, $n = 128, 512$ and 1500 and compared the number of flips the lattice had undergone to the standard

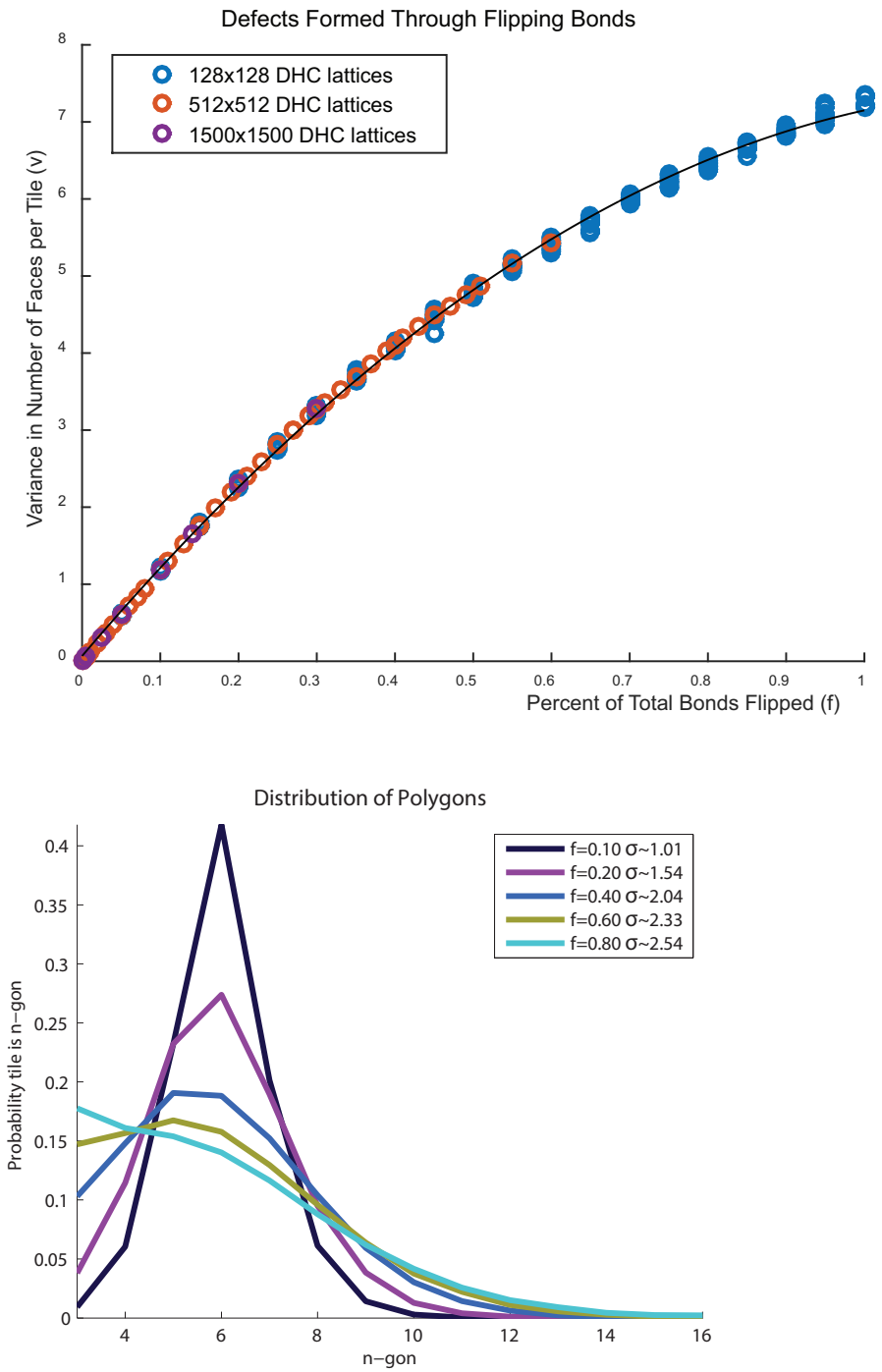

Figure 7.4: Relationship between the variance in the number of edges per tile and the fraction of bonds flipped f. **Top:** Blue circles represent the variance found in 200 independently generated 128×128 tile lattices. Orange circles represent the variance found in 50 independently generated 512×512 tile lattices. Purple circles represent the 8 1500×1500 tile lattices used to find the percolation thresholds in section 7.5. The data can be fit by the function $v = -4.835f^2 + 11.92f + 0.0681$. **Bottom:** Average distribution of polygons over ten independent 128×128 lattices at different flipping fractions. As more bonds are flipped a greater percentage of the tiles are triangles and many-sided polygons.

Algorithm 1: Flip Bond a-b

```
1 A=tile with site list ...a,b,...        ▷ Find the four tiles involved in the
  flip
2 B=tile with site list ...b,a,...
3 C=tile with site list ...a,...
4 D=tile with site list ...b,...
5 A → delete(b)                                  ▷ Change the bonds
6 B → delete(a)
7 C → insert(b) before a
8 D → insert(a) before b
```

deviation of defects σ. The results are shown in figure 7.4. There is a very strong relationship between the extent of defects as characterized by the variance v and the normalized number of bonds flipped, described approximately by $v = -4.835f^2 + 11.92f + 0.0681$. This equation allows us to generate lattices for systems of a given variance.

We also want to characterize the qualitative behavior of the system as the number of flipped bonds increases. We measured the probability of a tile having n sides as a function of f. At small f the tiles are almost all hexagons, with a few pentagons and heptagons. As more of the bonds are flipped the percent of hexagonal tiles in the lattice decreases as shown in figure 7.4 (bottom). The long-term behavior of the system is to create numerous triangular tiles and a few many-sided polygons. The reason for this long-time behavior is that every time a bond is flipped two tiles lose an edge and two tiles gain an edge, so as more bonds are flipped the number of bonds per tile is forced away from the mean of six.

There is an ambiguity in the definition of the lattice when f gets large. If the next bond to be flipped is part of a triangle it cannot be flipped or a tile with only two sites would be created and the lattice would become a multi-graph (where two sites are connected by more than one edge) which is not the behavior we were looking for. Bonds that are part of triangular tiles should not be flipped, but it is up to us to define whether choosing an unflippable bond counts as a flip or not. We in fact chose to count those bonds towards the flip trial count of the lattice.

We were not able to fully determine the long-time behavior of the infinite DHC system due to a specific finite-size effect. We used finite latices with periodic boundary conditions to approximate the infinite system. As the bonds in the system are flipped the tiles which neighbor each other change, and eventually one tile will gain enough bonds that it will wrap around the finite system and neighbor itself, dividing the

Table 7.1: Site and bond thresholds for the DHC lattices from simulations on 1500 x 1500 systems

σ	p_c^{site}	p_c^{bond}
0.24	0.69770 ± 0.00025	0.65320 ± 0.00025
0.55	0.69995 ± 0.00015	0.65497 ± 0.00025
0.77	0.70200 ± 0.00025	0.65715 ± 0.00025
1.09	0.7078 ± 0.00004	0.66130 ± 0.00025
1.28	0.71075 ± 0.00025	0.66440 ± 0.00025
1.52	0.71659 ± 0.00004	0.66860 ± 0.00025
1.81	0.72380 ± 0.00004	0.67470 ± 0.00025
2.10	0.73147 ± 0.00012	0.68140 ± 0.00025

Table 7.2: Honeycomb variant (three-coordinated) lattices

Lattice	Common Name	σ	p_c^{site}	p_c^{bond}
(6^3)	honeycomb	0.000	0.697040 [3]	0.652704 [12]
-	Poisson-Voronoi	1.314 [6]	0.71410 [13]	0.66693 [13]
$(4, 8^2)$	bathroom tile	2.061	0.729723 [14]	0.676803 [14]
$(4, 6, 12)$	cross	2.828	0.747801 [14]	0.693731 [14]
$\frac{3}{4}(3, 9^2) + \frac{1}{4}(9^3)$	martini	3.000	0.764826 [15]	0.707107[16]
$(3, 12^2)$	three-twelve	4.235	0.807901 [17]	0.740421 [14]

lattice into two. When this occurred we stopped the simulation as the lattice was no longer properly defined.

7.5 Determination of p_c

We found the percolation thresholds of the DHC computationally using a method described fully in Appendix D. At the percolation threshold the ratio

$$y(s, p) = \frac{sP_{\geq s}}{\langle s' \leq s \rangle} \tag{7.2}$$

reaches a constant value of $\delta - 1$, for large s, where s is the size of a cluster, $\langle s' \leq s \rangle$ is the expectation value of s for clusters up to size s, $P_{\geq s}$ is the probability that a vertex is in a cluster of at least size s, and $\delta = 1/(\tau - 2) = 91/5$ (in two dimensions) is a critical exponent of percolation. By graphing $y(s, p)$ for different values of the occupation probability p and finding when it reached 17.2 for large s, we were able to determine p_c with high precision.

To find the thresholds, we carried out 100,000 simulations on lattices of size $n =$

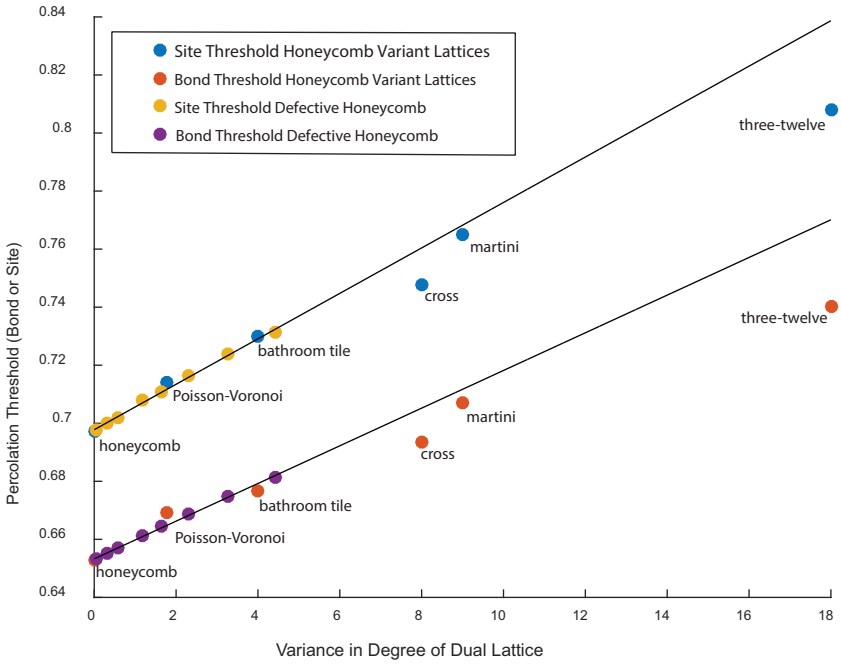

Figure 7.5: Graph of the relationship between the percolation threshold and dual-lattice variance $v = \sigma^2$. The bond (lower) and site (upper) thresholds for the DHC lattice and the previously studied honeycomb variant lattices listed in table 7.2 are shown. The linear regression for the bond thresholds is $p_c = 0.00649v + 0.6533$ and for the site threshold is $p_c = 0.00783v + 0.6978$.

1500, for values of σ equal to 0.00, 0.24, 0.55, 0.77, 1.09, 1.28, 1.52, 1.81, and 2.10 for both site and bond percolation. The details are shown in figure 7.5.

The final results are shown in Figure 7.5, which gives the relationship between the p_c and σ of the lattices. We also include in that plot the known thresholds of the honeycomb variant lattices (other lattices with coordination number 3). As σ increases both the site and bond percolation thresholds increase in a fairly predictable manner. The honeycomb variant lattices we used, which were all the three-coordinated ones where thresholds are known, are summarized in table 7.2.

7.6 Discussion

We introduced the concept of the DHC lattice as a way to better model real-world lattices which are not perfect honeycombs. The lattices are generated probabilistically by choosing edges uniformly at random from a honeycomb lattice to 'flip'. This process creates a new lattice with increasing number of defects as the bonds are flipped. We can quantify the severity of the defects by measuring σ, the standard deviation, or $v = \sigma^2$, the variance in the number of edges per tile. The variance is linearly proportional to the fraction of edges flipped f for small f. As more edges are flipped the distribution of polygons in the lattice changes such that most tiles are triangles or have a large number of edges.

There is a series of well-studied three-coordinated lattices which have many of the same basic properties of the DHC lattices. We compared the percolation threshold of these lattices and found that the percolation threshold for all of the lattices increases linearly in v. This means that the variance is a useful quantity to use to determine the percolation threshold of this class of lattice. The increase of thresholds as v increases means that ideal honeycomb lattice is more robust to connecting paths through the network than its defective counterparts. Patterning in biological systems takes energy and so we would expect that patterns like the honeycomb should provide some advantage over disordered lattices in order to be evolutionarily favorable. The fact that the honeycomb pattern is more robust against site or bond failures than similar networks with defects may be part of the reason it is so common in the natural world.

Bibliography

[1] D. Weaire and S. Hutzler, *The Physics of Foams*. Clarendon Press, 1999.

[2] A. G. Fletcher, M. Osterfield, R. E. Baker, and S. Y. Shvartsman, "Vertex models of epithelial morphogenesis," *Biophysical J.*, vol. 106, no. 11, pp. 2291 – 2304, 2014.

[3] J. L. Jacobsen, "High-precision percolation thresholds and Potts-model critical manifolds from graph polynomials," *J. Phys. A: Math. Th.*, vol. 47, APR 4 2014.

[4] T. Hales, "The honeycomb conjecture," *Discrete & Computational Geometry*, vol. 25, pp. 1–22, Jan 2001.

[5] H. Duminil-Copin and S. Smirnov, "The connective constant of the honeycomb lattice equals root 2+root 2," *Annals of Mathematics*, vol. 175, pp. 1653–1665, May 2012.

[6] H. J. Hilhorst, "New Monte Carlo method for planar Poisson-Voronoi cells," *J. Phys. A: Math. Th.*, vol. 40, no. 11, p. 2615, 2007.

[7] R. Zallen, *The Physics of Amorphous Solids*. John Wiley Sons, 1998.

[8] B. Aigouy, R. Farhadifar, D. B. Staple, A. Sagner, J.-C. Roeper, F. Juelicher, and S. Eaton, "Cell Flow Reorients the Axis of Planar Polarity in the Wing Epithelium of Drosophila," *Cell*, vol. 142, pp. 773–786, Sep 3 2010.

[9] D. Stauffer and A. Aharony, *Introduction to Percolation Theory, 2nd ed.* CRC Press, 1994.

[10] P.-L. Bardet, B. Guirao, C. Paoletti, F. Serman, V. Leopold, F. Bosveld, Y. Goya, V. Mirouse, F. Graner, and Y. Bellaiche, "PTEN controls junction lengthening and stability during cell rearrangement in epithelial tissue," *Developmental Cell*, vol. 25, pp. 534–546, Jun 10 2013.

[11] P. A. Raymond, S. M. Colvin, Z. Jabeen, M. Nagashima, L. K. Barthel, J. Hadidjojo, L. Popova, V. R. Pejaver, and D. K. Lubensky, "Patterning the cone mosaic array in zebrafish retina requires specification of ultraviolet-sensitive cones," *PLOS ONE*, vol. 9, Jan 21 2014.

[12] M. F. Sykes and J. W. Essam, "Exact critical percolation probabilities for site and bond problems in two dimensions," *J. Math. Phys.*, vol. 5, no. 8, pp. 1117–1127, 1964.

[13] A. M. Becker and R. M. Ziff, "Percolation thresholds on two-dimensional Voronoi networks and Delaunay triangulations," *Phys. Rev. E*, vol. 80, p. 041101, Oct 2009.

[14] J. L. Jacobsen, "High-precision percolation thresholds and Potts-model critical manifolds from graph polynomials," *J. Phys. A: Math. Th.*, vol. 47, p. 135001, Apr. 2014.

[15] C. R. Scullard, "Exact site percolation thresholds using a site-to-bond transformation and the star-triangle transformation," *Phys. Rev. E*, vol. 73, p. 016107, Jan 2006.

[16] R. M. Ziff, "Generalized cell–dual-cell transformation and exact thresholds for percolation," *Phys. Rev. E*, vol. 73, p. 016134, Jan 2006.

[17] P. N. Suding and R. M. Ziff, "Site percolation thresholds for Archimedean lattices," *Phys. Rev. E*, vol. 60, pp. 275–283, Jul 1999.

[18] P. L. Leath, "Cluster size and boundary distrabution near percolation threshold," *Phys. Rev. B*, 1976.

[19] M. E. J. Newman and R. M. Ziff, "Efficient Monte Carlo algorithm and high-precision results for percolation," *Phys. Rev. Lett.*, vol. 85, pp. 4104–4107, Nov 2000.

[20] M. E. J. Newman and R. M. Ziff, "Fast Monte Carlo algorithm for site or bond percolation," *Phys. Rev. E*, vol. 64, p. 016706, Jun 2001.

CHAPTER VIII

Conclusion

In this thesis we have explored ideas about disordered active systems through the lens of epithelial tissues. Throughout the process of development, epithelial tissues undergo a series of changes to their morphology. These tissue scale changes are driven by local cell-cell mechanical interactions. Here we have discussed specific questions relating to the broader theme of how individual cell-cell interactions drive the self-assembly of cells into specific structures and patterns.

Throughout this thesis, we have used the vertex model as a mathematical descriptor of epithelial tissues. Although this model has a long history of use in the field of quantitative biology, some of its most basic features remained unproven until this work. In chapter II we made a full and careful definition of an energy derived vertex model, and proved that this model does not support stable fourfold vertices under the standard assumptions. We also gave a biologically relevant adaptation of the model, based on planar cell polarity, that supports the formation of stable fourfold vertices. Additionally, we presented algorithms which correctly implement vertex resolution for computational models. In chapter III we covered the algorithms and data structures used to write the computational model used throughout the thesis. We present this work in the spirit of computational reproducibility, and to highlight solutions to common problems encountered when designing vertex model code. Thus, chapters II and III flesh out the details of the theoretical and computational framework that we then used to answer specific questions about the interplay between cellular packing and tissue stress.

Chapters IV and V deal with the question of the formation of multicellular actomyosin cables in tissues under applied stress. I was initially drawn to the question of cable formation after seeing the experimental data showing fibers in the Drosophila pupa notum. These fibers appeared to be aligned with one-another and the cortex bound actomyosin to create stress-resistant cables. Indeed analysis of the data shows

that the fibers are more aligned than if they were placed randomly and that cells with more fibers are less elongated. This raises the question of why fibers disappear in the notum around 30 hours APF, given that the applied tension remains high. We offer two explanations for this apparent behavior. First, a toy model of the fiber mechanics shows that smaller cells require fewer fibers to resist elongation. Secondly, cells in the cable forming orientation do not require fibers to resist elongation. Analysis of cellular data reveals that cells in the notum undergo a round of division resulting in smaller apical areas coinciding with fiber loss. Using the methods developed in chapter IV, we can also say that the tissue rearranges into a more cable-friendly topology before the loss of fibers. We hypothesize that the combination of these two events leads to the loss of fibers.

Understanding the role of fibers in the notum involved analyzing data from thousands of cells. In chapter VI, we discuss the technical aspects of working with this large amount of data. We came to the conclusion that classical image processing algorithms were sufficient for determining cell outlines. However, even with cutting edge machine learning algorithms, we lacked a large enough data set to train a classifier to automatically detect fibers in images. Untimely, all fiber data in this thesis was curated by hand.

Aspects of the work presented in this thesis are applicable to a wide range of fields from cellular biology to non-linear dynamics. The vertex model has its roots in the mathematics of graph theory, which describes networks of connected nodes. In the final chapter we step away from biological questions and study the more abstract properties of vertex model inspired networks. All vertex models belong to a larger mathematical class of networks. In chapter VII we show that the number of T1 transitions preformed on an initially honeycomb lattice directly corresponds to the percolation threshold, and that this relationship holds for the other known networks in the same class.

The process of development is highly complex, occurring at many different length scales, and involving chemistry, biology and physics. Physicals models, like the vertex model, provide a way for us to understand aspects of the general behavior of cell sheets, even though we lack an understanding of many of the specific small scale components. In this work, we used the vertex model framework to understand new experimental data of the appearance and subsequent disappearance of fibers in the the *Drosophila* pupal notum. This furthers our understanding of the ways in which epithelia can react to mechanical stresses in development.

APPENDICES

APPENDIX A

Pseudo-code for implementing T1 transitions

Algorithms 1-4 give pseudo-code implementing T1 transitions as described in section 2.7. The code assumes an object oriented language (such as C++ or Java) with cell, edge, and vertex objects already defined. We will assume that the edges store data on their neighboring vertices and cells. The cells and vertices only store data on their neighboring edges, and functions have been written to get the other neighboring objects if needed. Objects are referred to in C++ style so that someobject.somedata refers to the data *somedata* stored by the object *someobject*.

The function T1 takes a small edge and replaces it with a new fourfold vertex, and then calls the function ResolveFourfoldVertex on the new vertex.

The function ResolveFourfoldVertex takes as input a fourfold vertex. It calls CheckStability on each of the possible resolution topologies to determine their stability. Once the correct resolution topology has been established the function calls BreakFourfoldVertex to update the edges, cells, and vertices involved in the T1 transition.

The function CheckStability takes as input a fourfold vertex and its associated edges and cells. The cells and edges must be given in clockwise order. The function will create temporary objects representing breaking the fourfold vertex such that edges e_1 and e_2 share a common vertex. The force $\mathcal{F}$ is calculated and returned, and the temporary objects are deleted.

The function BreakFourfoldVertex takes a resolution topology for a fourfold vertex as input and creates the new edge and correctly resigns the neighboring edges, vertices, and cells in the new topology.

Including both the CheckStability and BreakFourfoldVertex functions may seem redundant, but it is vital to have both to deal with the rare but possible case in which

a fourfold vertex is unstable to breaking up in both topologies. In this case the vertex should break in the topology in which it is most unstable.

Algorithm 2: T1

Input: e_0 the edge to undergo T1

Output: None. The function will update the effected edges cell and vertices so they are in the correct locations and have the correct neighbors following a T1 transition.

```
1  v1 ← e0.vertex1
2  v2 ← e0.vertex2
3  c1 ← e0.cell1
4  c2 ← e0.cell2
   ▷ Do not T1 edges which neighbor triangles.  This would produce
     cells with only two sides.
5  if c1.EdgeNumber ≤ 3 or c2.EdgeNumber ≤ 3 then
6      EXIT
   ▷ Get the edges which will make the fourfold vertex
7  forall the e ∈ {v1.edges or v2.edges} do
8      if e ≠ e0 then
9          list4foldedges ← e
   ▷ set the next position of the two vertices to the center of the
     edge
10 v1.xnext ← e0.center
11 v2.xnext ← e0.center
   ▷ move vertices updating any periodic boundary flags if nessacary
12 MoveVertex(v1)
13 MoveVertex(v2)
   ▷ make the new vertex
14 vnew ← e0.center
15 forall the e ∈ list4edges do
16     vnew ← e
   ▷ delete two old vertices
17 Delete(v1)
18 Delete(v2)
   ▷ Remove e0 from the list of edges in its two neighboring cells
19 for c ∈ {c1, c2} do
20     forall the  e ∈ c do
21         if e = e0 then
22             remove e
23 Delete(e0)                          ▷ delete the central edge e0
24 ResolveFourfoldVertex(vnew)         ▷ Resolve the fourfold vertex
```

Algorithm 3: ResolveFourfoldVertex

Input: $v0$ the fourfold vertex to resolve
Output: None

▷ Make lists of the edges and cells in clockwise order
$e \leftarrow v.CWEdges$
$c \leftarrow v.CWCells$
▷ find the stability of each configuration
$f1 \leftarrow$ CheckStability(v, e[0], e[1], e[2], e[3], c[0], c[1], c[2], c[3])
$f2 \leftarrow$ CheckStability(v, e[3], e[0], e[1], e[2], c[3], c[0], c[1], c[2])
case *f1=0* **and** *f2=0*
    EXIT ▷ The vertex is stable so exit
case $f1 \geq f2$
    ▷ The vertex is unstable and should resolve in the first topology
    BreakFourfoldVertex(v, e[0], e[1], e[2], e[3], c[0], c[1], c[2], c[3])
case $f2 > f1$
    ▷ The vertex is unstable and should resolve in the second topology
    BreakFourfoldVertex(v, e[3], e[0], e[1], e[2], c[3], c[0], c[1], c[2])

Algorithm 4: CHECKSTABILITY

Input: v, e1, e2, e3, e4, c1, c2, c3, c4
v: the fourfold vertex
e1, e2, e3, e4: the four edges of v in clockwise order such that (e1,e2) will be neighbors and (e3,e4) will be neighbors when the vertex is split
c1, c2, c3, c4: The four cells of v in clockwise order such that c1 has edges e1, and e2.

Output: Creates temporary objects representing the vertex splitting such that edges (e1,e2) and (e3,e4) are paired and cells c2 and c4 are neighbors. It returns the magnitude of the force pulling the vertices apart. If the vertex is stable against breaking in this topology it returns 0.

```
1  CoppyAll v' ← v, e1' ← e1, c1' ← c1, ...          ▷ Make temporary objects
   ▷ Make the new edge (enew) and vertices (v12, and v34) resulting
     from the split into two threefold vertices
2  v12 ← v.x                                  ▷ the vertex on edges e1' and e2'
3  v12 ← {enew, e1', e2'}
4  v34 ← v.x                                  ▷ the vertex on edges e3' and e4'
5  v34 ← {enew, e3', e4'}
6  enew.length ← 0
7  enew ← {v12, v34}
8  enew ← {c2', c4'}
9  for e ∈ {e1', e2', e3', e4'} do
10 |  e delete v                                    ▷ Update the four edges
11 |  if e ∈ {e1', e2'} then
12 |  |  e ← v12
13 |  else
14 |  |  e ← v34
15 c2' ← enew                                        ▷ Update the cells
16 c4' ← enew
   ▷ Calculate F as given in Sect3A. Let e.FindForce(v) return the
     force on vertex v from edge e given by Γ_e l̂_e + (p_e/2)(ẑ × l_e).
17 F ← (e1.FindForce(v12) + e2.FindForce(v12) + e3.FindForce(v34) +
   e4.FindForce(v34))/2
18 if F > enew.tension then
19 |  return magnitude(F)
20 else
21 |  return 0;
```

Algorithm 5: BreakFourfoldVertex

Input: v, e1, e2, e3, e4, c1, c2, c3, c4
v: the fourfold vertex
e1, e2, e3, e4: the four edges of v in clockwise order such that (e1,e2) will be neighbors and (e3,e4) will be neighbors when the vertex is split
c1, c2, c3, c4: The four cells of v in clockwise order such that c1 has edges e1, and e2.
Output: None

▷ Make the new edge (enew) and vertices (v12, and v34) resulting from the split into two threefold vertices

$v12 \leftarrow v.x + (\frac{L}{2}\hat{\mathcal{F}})$ ▷ Where L specifies new edge lengths
$v12 \leftarrow \{enew, e1', e2'\}$
$v34 \leftarrow v.x - (\frac{L}{2}\hat{\mathcal{F}})$ ▷ Where L specifies new edge lengths
$v34 \leftarrow \{enew, e3', e4'\}$
$enew.length \leftarrow L$
$enew \leftarrow \{v12, v34\}$
$enew \leftarrow \{c2, c4\}$
for $e \in \{e1, e2, e3, e4\}$ **do**
    e delete v ▷ Update the four edges
    if $e \in \{e1, e2\}$ **then**
        $e \leftarrow v12$
    else
        $e \leftarrow v34$
$c2 \leftarrow enew$ ▷ Update the cells
$c4 \leftarrow enew$
Delete v

APPENDIX B

Supplemental information for chapter IV

We created an ensemble of disordered tissues with a range of cableness by inducing different levels of flow in different tissues. The flow forces a series of oriented T1 neighbor exchanges that change the cableness of the tissue. We also tried a more controlled version of this process in which we forced oriented T1s to occur even though they were not energetically favorable, we call this ensemble 'forced T1' packings. To create the packings we began with a ordered hexagonal lattice in the CFO. We then randomly selected horizontal edges and forced them to undergo the T1 process. Figure B.1A-E gives representative examples of tissues with increasing percentage of their edges flipped. This flipping leads to a decrease in the value of the orientational order parameter $\langle \cos(6\theta) \rangle$, as shown in figure B.1F.

Figure B.2 gives the results of applying our cableness measures to the forced T1 packings. The tension measure preforms well. However, the convexity measure preforms very poorly. We believe that this is a result of the underlying crystalline structure of the forced T1 packings. The cells do not become concave when they are less cabley because they are so highly ordered. This degree of ordering is not representative of epithelia, and therefor does not pose an issue for the convexity measure.

Also included in figure B.2 is the result of applying a third cableness measure $\mathcal{E}$ based on cell elongation. Figure B.3A,B shows the elongation of cells in the stretched state for both a cabely and non-cabely packing. The tissue with low cableness has many more highly elongated cells. Figure B.3 gives the histogram of cell elongation as measured by the ratio of the perimeter to the square root of the area for the forced T1 packings at different percentages of flipped edges. The less cabely packings have a much broader histogram. The results are qualitatively the same for the flowing

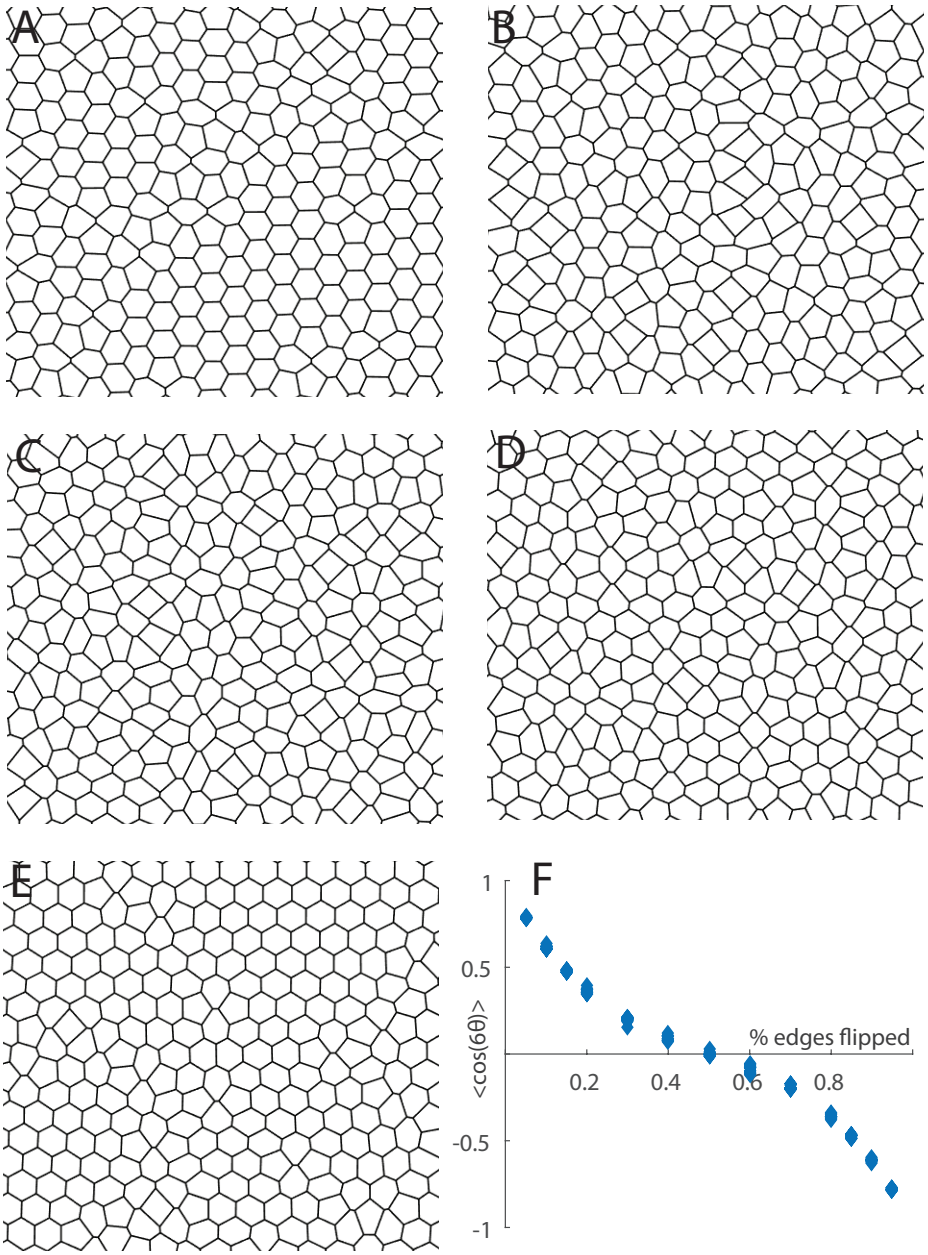

Figure B.1: **A-E** Images of forced T1 packings with 10, 30, 50, 70 and 90 percent of the edges flipped. **F** Relationship between the percentage of edges flipped and the orientational order as measured by $\langle \cos(6t) \rangle$.

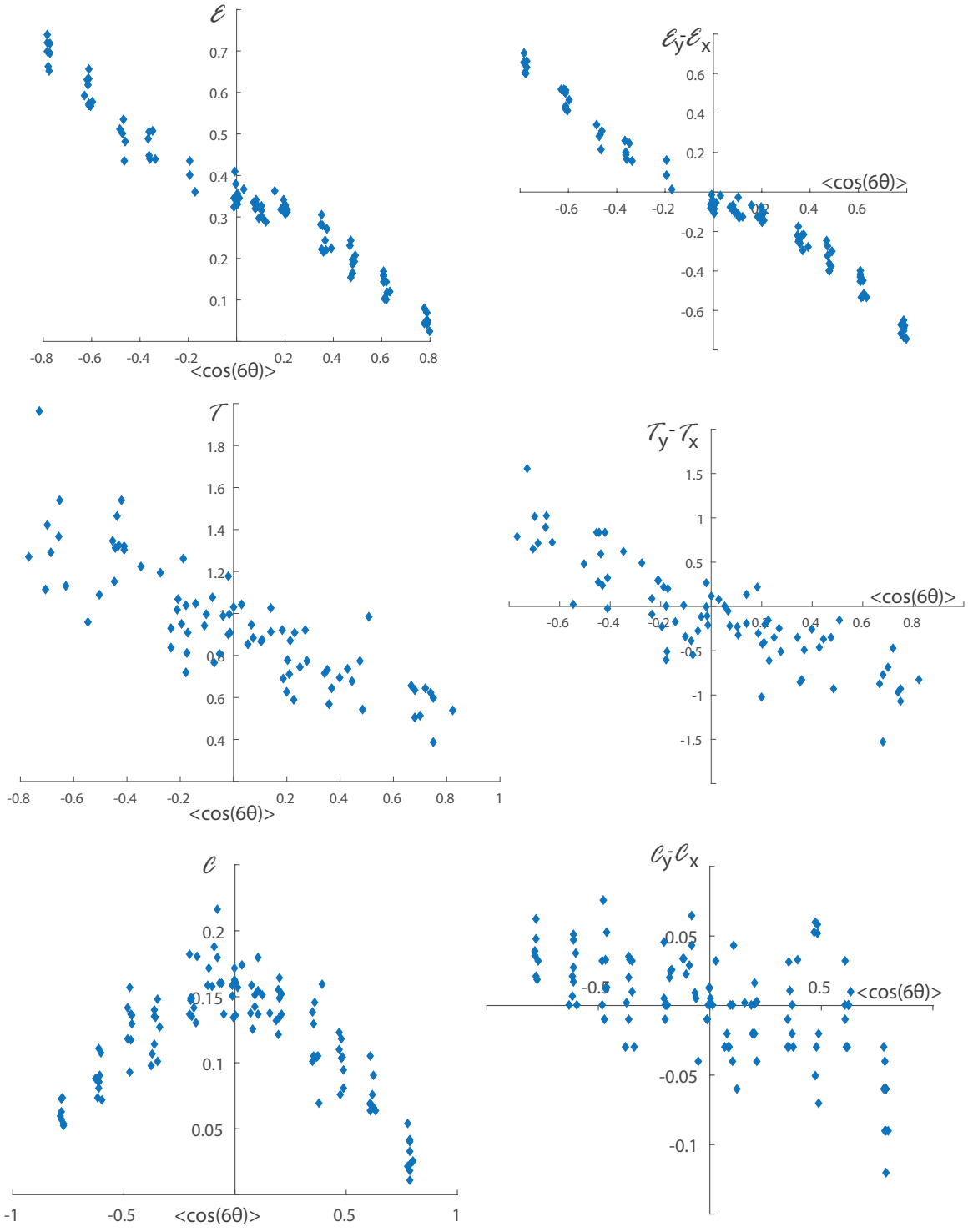

Figure B.2: Cableness measures applied to the forced T1 packings. Note that the convexity measure preforms poorly due to the crystal structure of the packings.

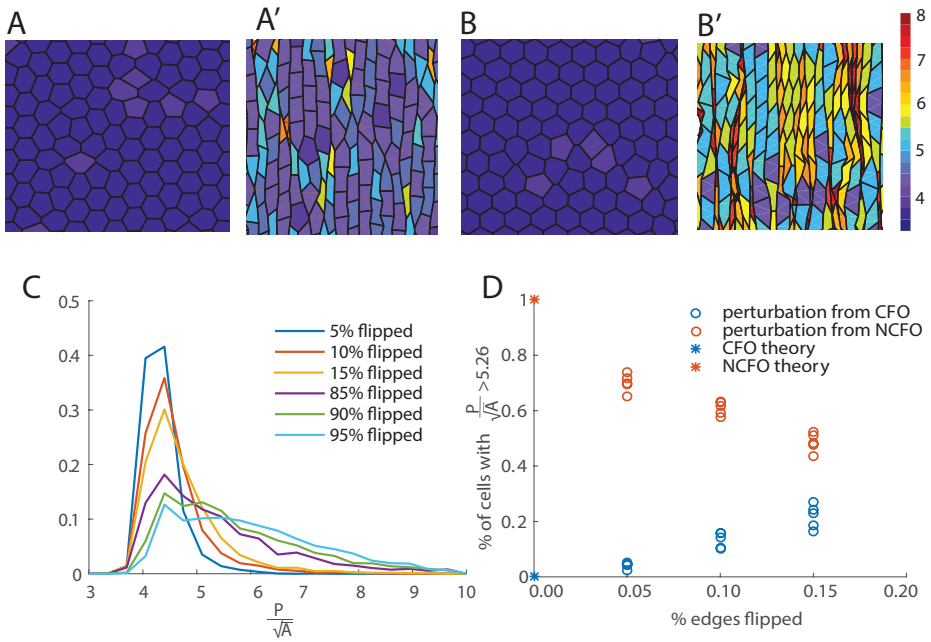

Figure B.3: Elongation based cableness measure. **A**, **A′**: Elongation of the cells in a relaxed and stretched forced T1 packing. Ten percent of edges are flipped. **B, B ′**: Elongation of the cells in a relaxed and stretched forced T1 packing. Ninety percent of edges are flipped. **C:** Histogram of the distribution of cell elongation in stretched forced T1 packings for a range of edges flipped. Less cabley packings have more elongated cells. **D:** Percent of cells with elongation greater than the cutoff of 5.26 in stretched forced T1 packings for a range of edges flipped. We define this measure of cableness as the elongation measure $\mathcal{E}$.

packings [data not shown]. We define $\mathcal{E}$ to be the percentage of cells with elongation greater than 5.26, the elongation of a hexagonal cell with constant perimeter that has half its initial area. Figure B.4A gives the results of applying the elongation based cableness measure to all of our different tissues. It agrees with both the tension and convexity based measures for non-pre-stressed tissue. However, for pre-stressed tissues the elongation measure increases increases, moreover, the magnitude of the difference between the measure applied to a stress free and pre-stressed tissue depends on the cableness of the tissue. This makes it difficult to interpret in pre-stressed tissues, and we therefor discarded it as a measure of cableness. Additionally we expect that it would not correctly identify elongated brick shaped tissues as cabely.

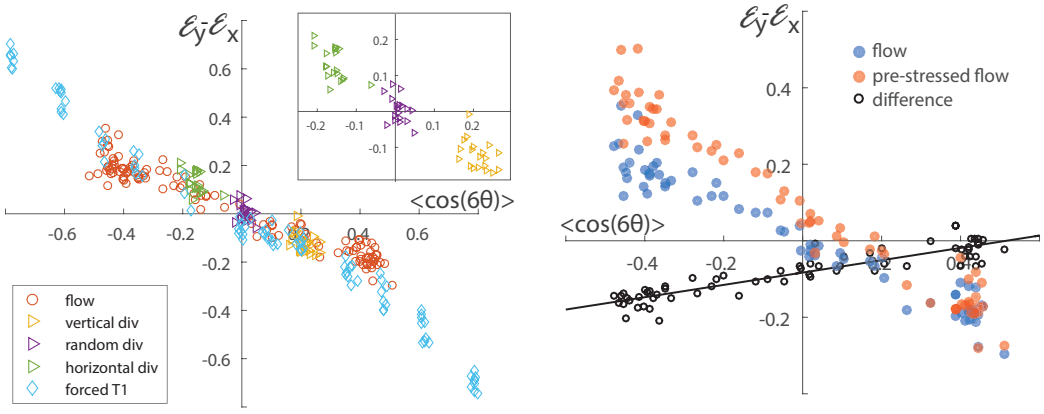

Figure B.4: Elongation base cableness measure **A** Results of applying the difference in elongation measures to a variety of different initial packings. Inset shows only packings created through oriented division. **B** Difference between the results of the elongation measure applied to relaxed or pre-stressed tissues. The elongation measure is effected by the pre-stressing in a non uniform manner. More cabely packings are more effected than less cabley ones.

APPENDIX C

Fiber alignment

The following graphs give the results of the cumulative probability distribution of the normalized mean path length through the tissue for 100 randomly placed fibers. Orange curves are cumulative probability distributions for fibers placed through the tissue, and blue curves are for fibers placed randomly in the same cell. Vertical lines mark the standard deviations. The black line represents the normalized mean path length for the actual fiber positions.

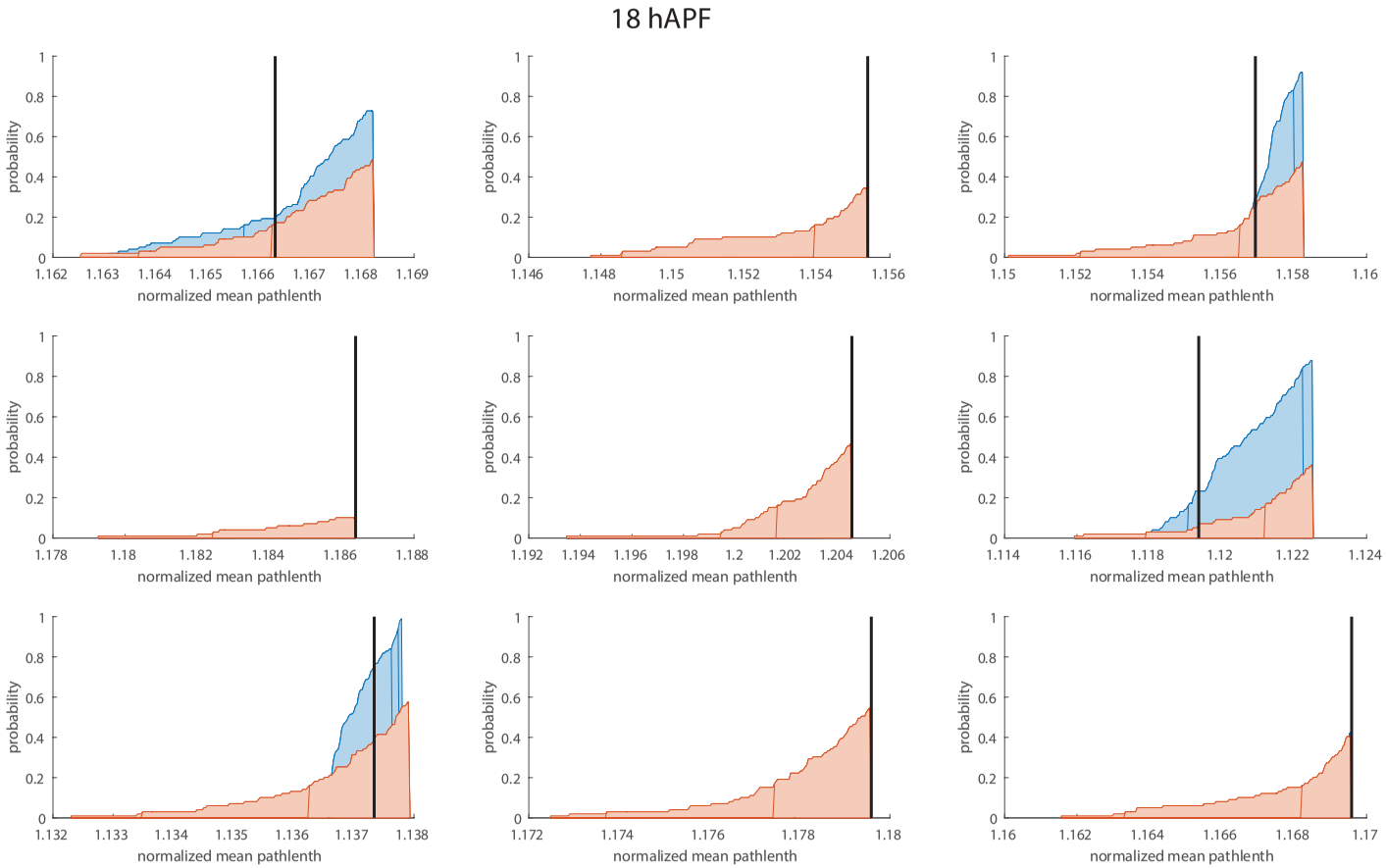

Figure C.1: Cumulative probability distributions for tissues at 18 hours APF.

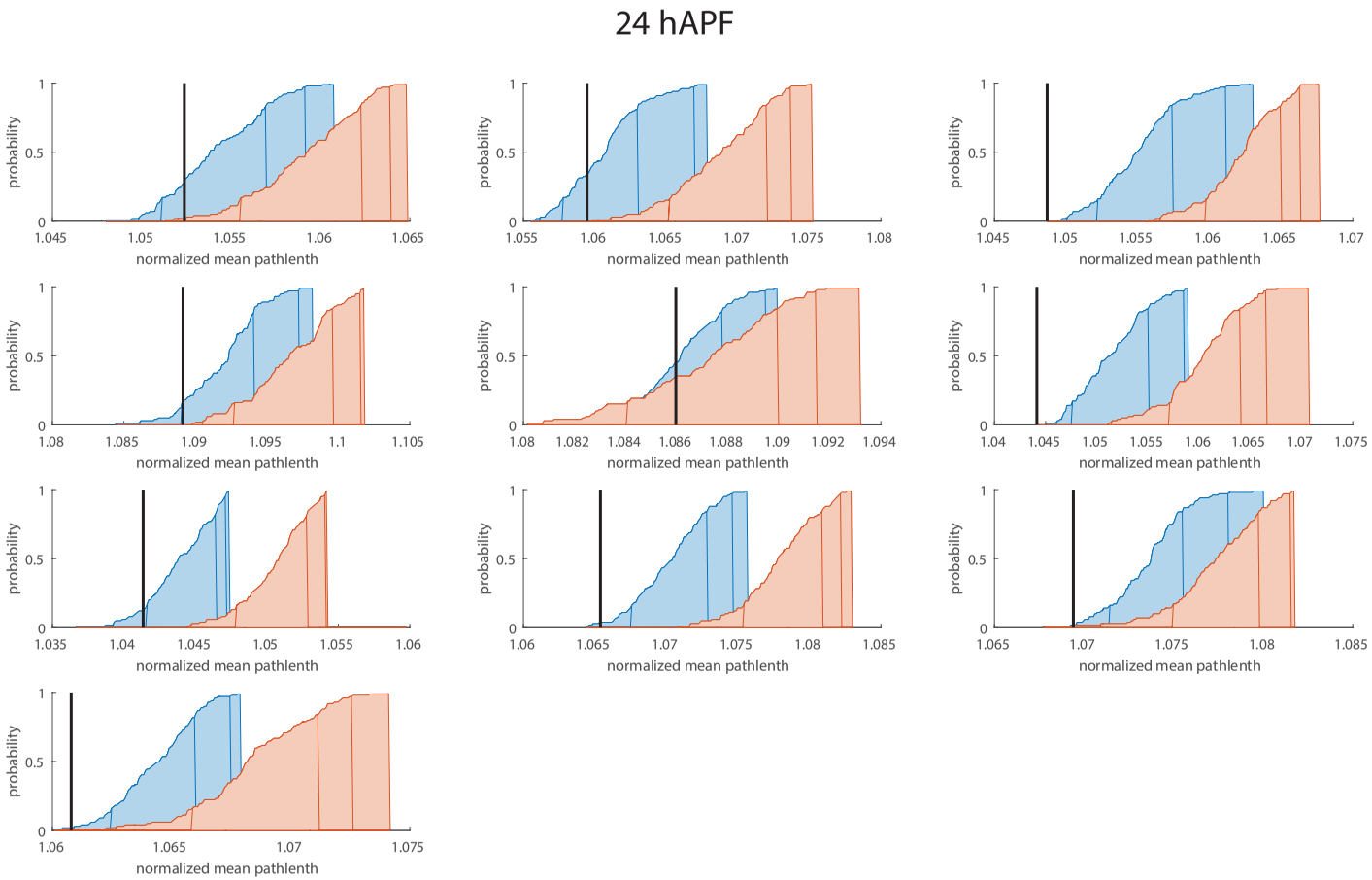

Figure C.2: Cumulative probability distributions for tissues at 24 hours APF.

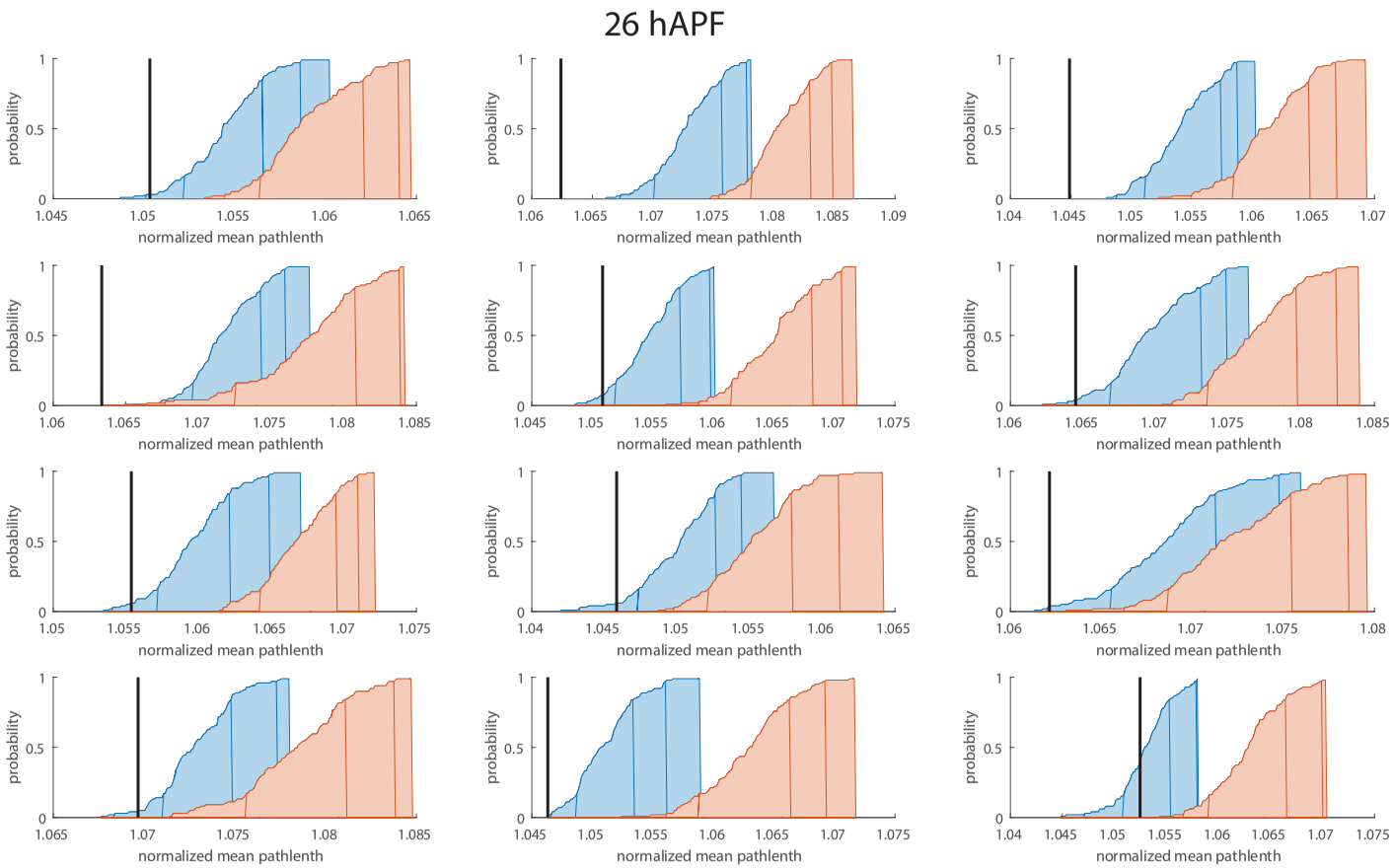

Figure C.3: Cumulative probability distributions for tissues at 26 hours APF.

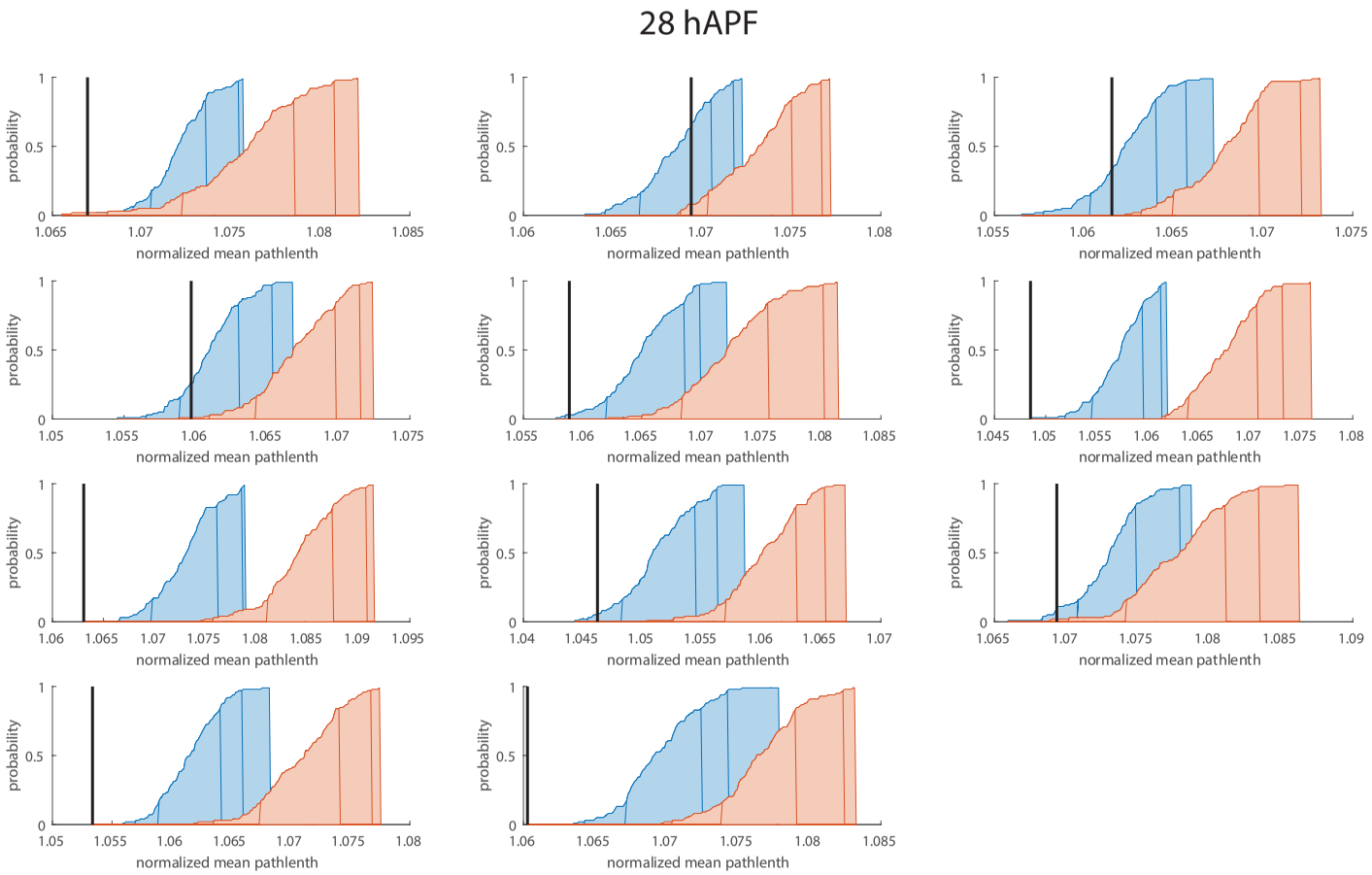

Figure C.4: Cumulative probability distributions for tissues at 28 hours APF.

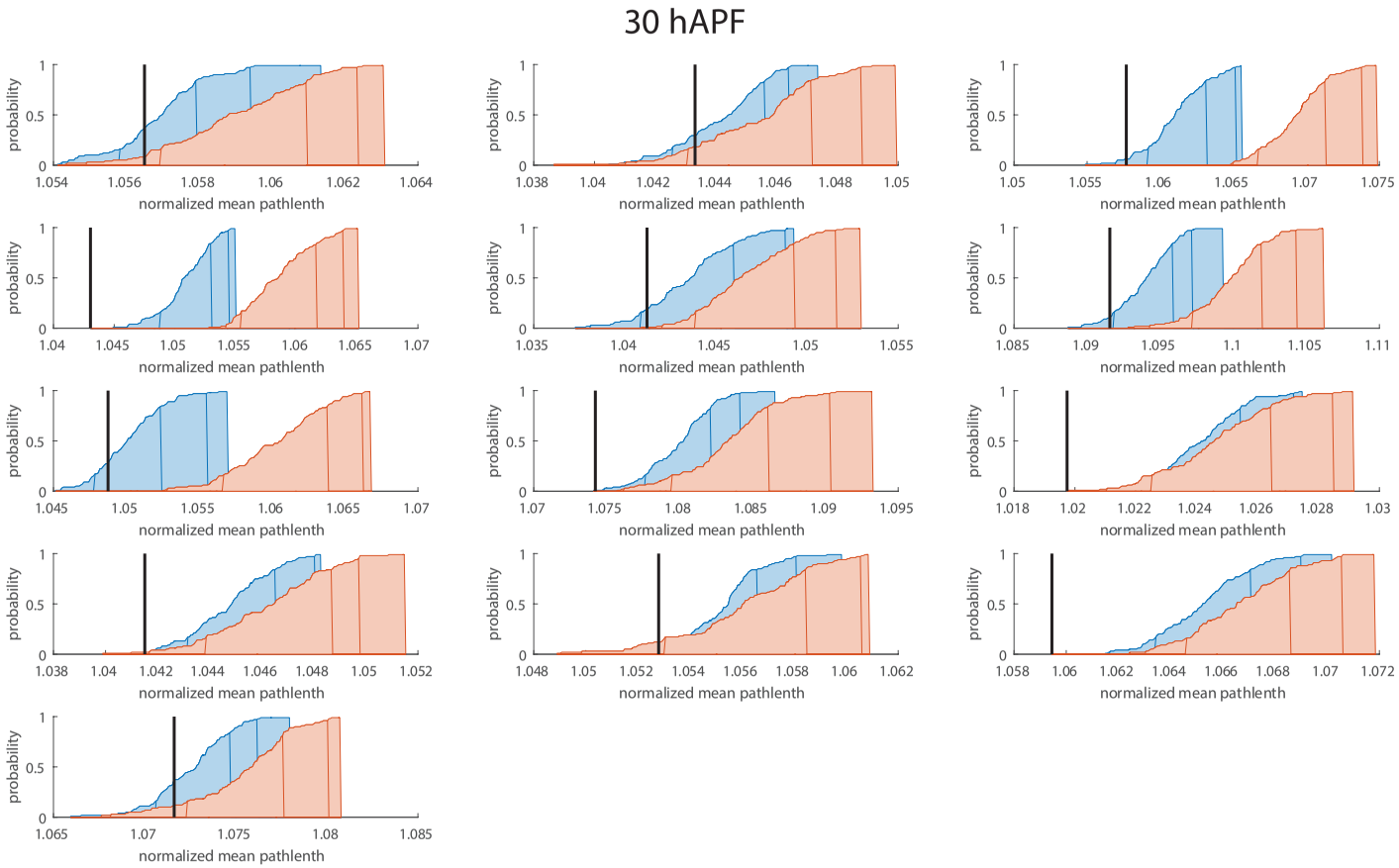

Figure C.5: Cumulative probability distributions for tissues at 30 hours APF.

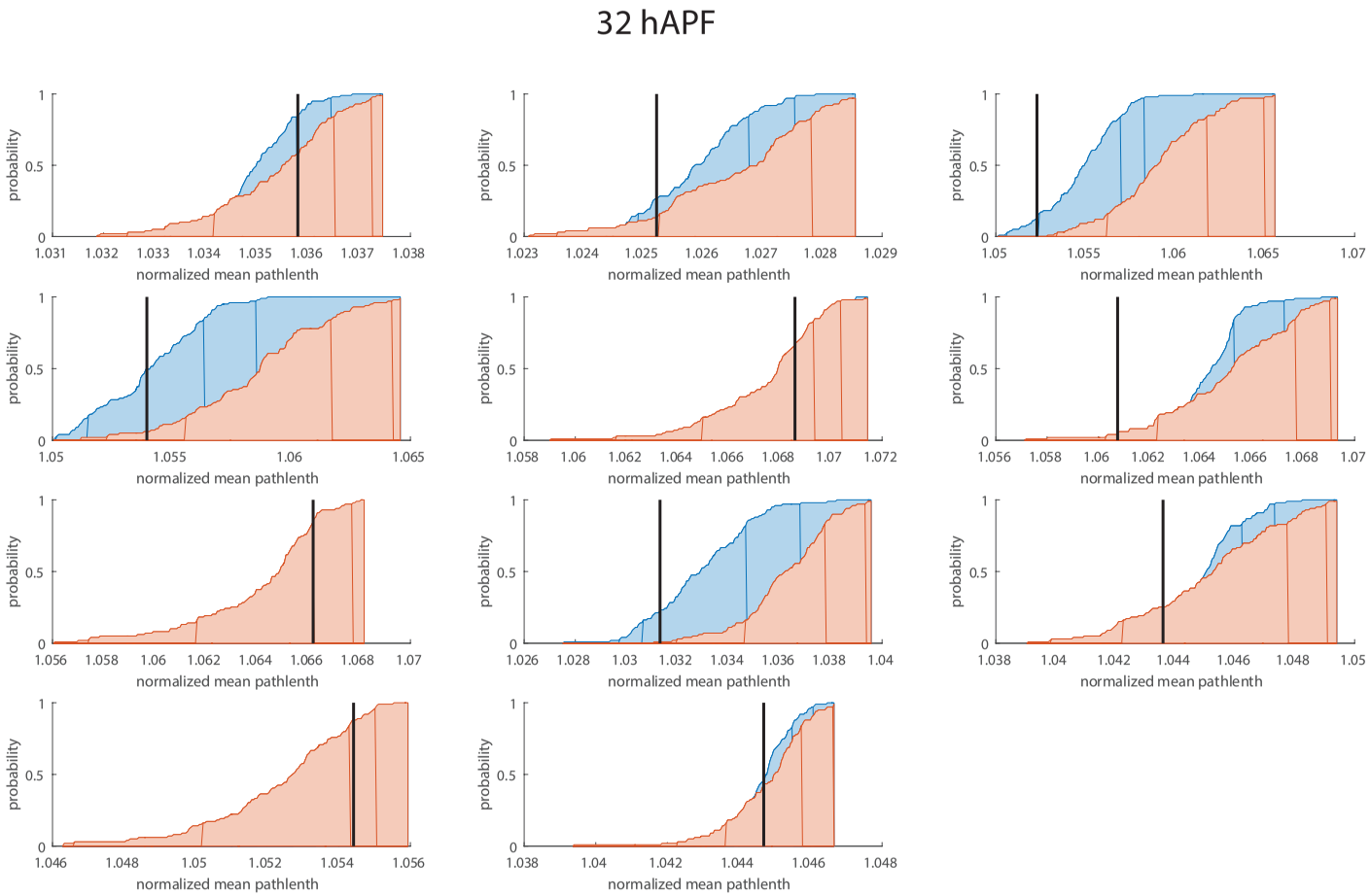

Figure C.6: Cumulative probability distributions for tissues at 32 hours APF.

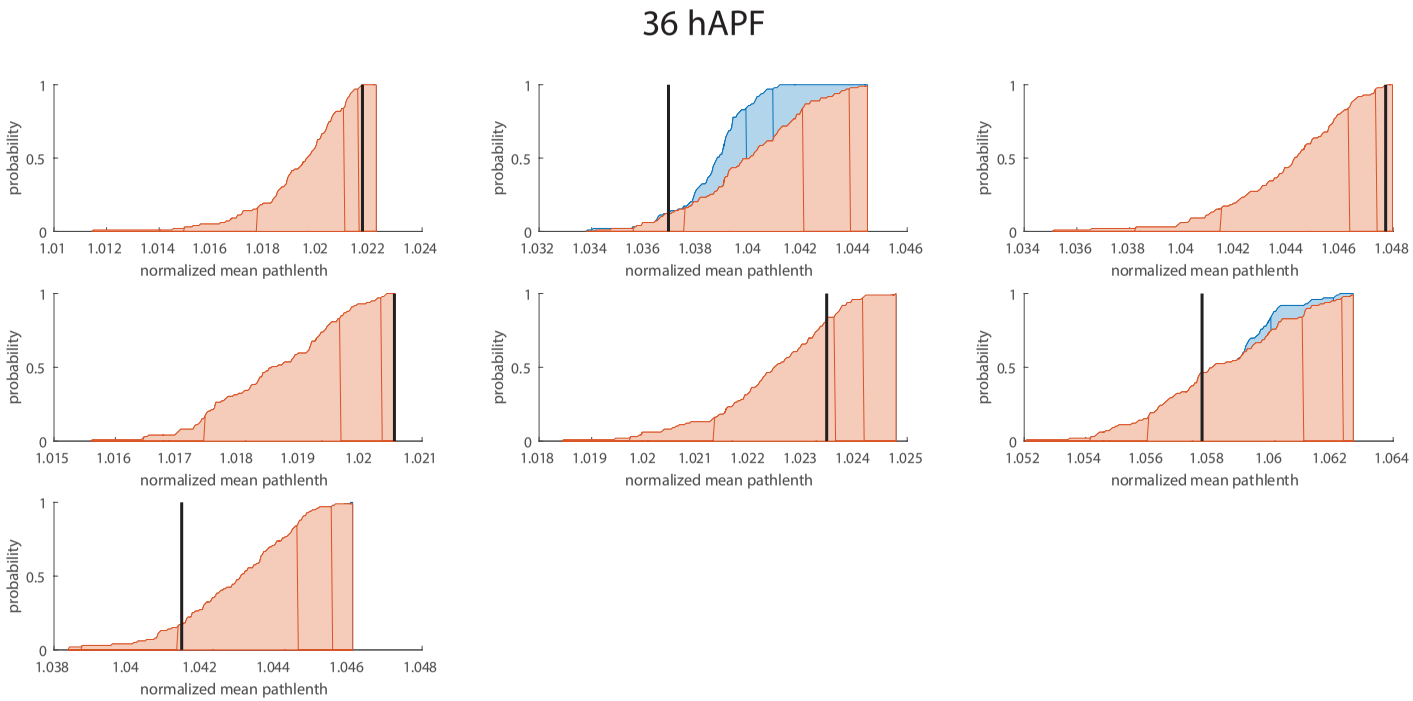

Figure C.7: Cumulative probability distributions for tissues at 36 hours APF.

APPENDIX D

Initializing the DHC Lattice

In order to use algorithm 1 the lattice must be stored as a list of clockwise-ordered vertices around each cell. There are many different solutions which give unique labels to every vertex, and working out the proper boundary conditions and determining the vertices in each tile can be time consuming. We have reproduced our algorithm here for interested parties wishing to produce their own DHC lattices. We used the labeling method shown in figure 7.3 to determine tile and site IDs. Tiles are labeled 0 to $n^2 - 1$ in order by row and then column. The top site of each hexagon has id $\mathrm{ID}_{site} = 2(\mathrm{ID}_{tile})$ and the next clockwise site has $\mathrm{ID}_{site} = 2(\mathrm{ID}_{tile}) + 1$. The following pseudocode produces a two-dimensional array of $n \times n$ hexagons. The ith row of the array gives the clockwise ID's of vertices of tile i where the first entry is the top vertex of the tile.

Algorithm 6: Initialize Lattice

```
1  for z=0 to n^2 − 1 do
2      α = (z − (z (mod n))) / n (mod 2)
3      z' = 2(z + n (mod n))
4      tiles[z][0] = 2z
5      tiles[z][1] = 2z + 1
6      if α = 1 then
7          tiles[z][2] = z'
8          if z (mod n) = 0 then
9              tiles[z][3] = 2(z + 2n − 1 (mod n)^2) + 1
10             tiles[z][4] = 2(z + 2n − 1 (mod n)^2)
11         else
12             tiles[z][3] = z' − 1
13             tiles[z][4] = z' − 2
14     if z (mod n) = n − 1 then
15         tiles[z][2] = 2z + 2
16         tiles[z][2] = z' + 2
17     else
18         tiles[z][2] = z' + 1
19         tiles[z][2] = z'
20     if z (mod n) ≠ 0 then
21         tiles[z][5] = 2z − 1
22     else
23         tiles[z][5] = 2(z + n − 1) + 1
```

APPENDIX E

Determining percolation thresholds

In section 7.5 we presented data showing the relationship between the percolation threshold of some DHC lattices with the honeycomb variant lattices. The site and bond percolation thresholds for the honeycomb variant lattices are known (see Table 7.2). There are many established ways to estimate the percolation threshold of a lattice using simulations [2, 3], however we found that a novel method based on a cluster growing algorithm was more practical for our problem. In this section we will describe the theory behind our method and outline the computational model we used.

The theory behind the method is straightforward. We found a ratio of properties which we will call y (defined below) that is a constant function of s (for large s) at the percolation threshold. We can then use a normal cluster growing algorithm to determine this ratio at different occupation probabilities, and narrow in on the value of p_c.

The following is a short theoretical derivation proving that our ratio y should be constant for s at the percolation threshold. Let n_s be the number of clusters of size s, and P_s be the probability that any site is in a cluster of size s. It is well known [2] that at p_c the number of clusters of size s, is $As^{-\tau}$, where A is some constant and τ is the Fisher scaling exponent. From this relationship it follows that the probability that a site on the lattice belongs to a cluster of size s is given by

$$P_s = sn_s \sim As^{1-\tau}. \tag{E.1}$$

for large s, which implies that the probability a site is in a cluster of at least size s is given by

$$P_{\geq s} = \int_s^\infty P_{s'} ds' \sim \int_s^\infty A(s')^{1-\tau} ds', \tag{E.2}$$

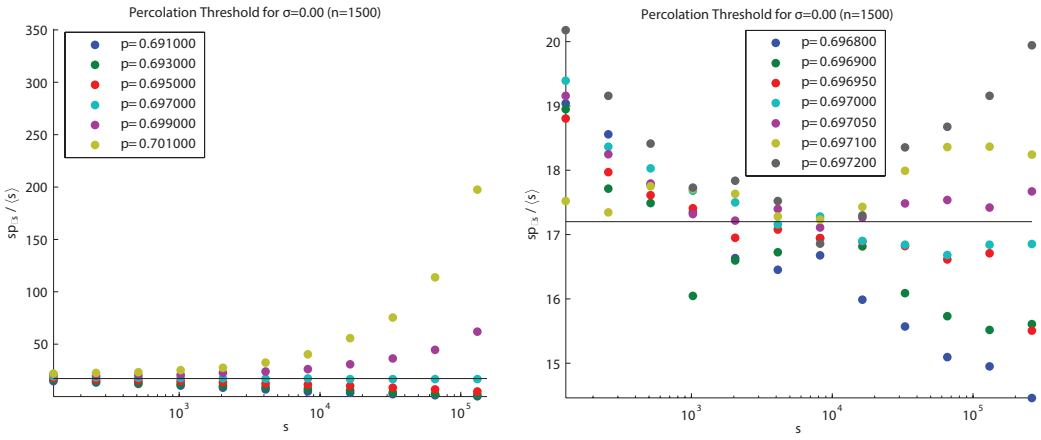

Figure E.1: Example of finding the percolation threshold using our method. We grew 10000 clusters on a honeycomb lattice at various occupation probabilities p. Statistics on the average $P_{\geq s}$ and $\langle s' \leq s \rangle$ were gathered and the ratio $y = sP_{\geq s}/\langle s' \leq s \rangle$ was plotted versus the cluster size s. At the percolation threshold the resulting curve should be flat. On the left we get the approximate value of p_c for site percolation on the honeycomb lattice. On the right we narrow in on the exact value. Using this method we determined that the site percolation threshold was 0.69700 ± 0.00005 which agrees with the known value of 0.697040 [1].

at p_c. The value of τ is between 2 and 2.5 in every dimension so the integral in (E2) converges and evaluates to

$$P_{\geq s} \sim \frac{As^{2-\tau}}{\tau - 2}. \tag{E.3}$$

We want our ratio of lattice properties y to be constant at p_c, so we incorporated the average value of the size of all clusters up to size s:

$$\langle s' \leq s \rangle = \int_1^s s' P_{s'} \sim \int_1^s A(s')^{2-\tau} ds' = \frac{As^{3-\tau} - 1}{3 - \tau}. \tag{E.4}$$

It follows that the ratio y, defined as

$$y = \frac{sP_{\geq s}}{\langle s' \leq s \rangle} \sim \frac{3 - \tau}{\tau - 2} = \frac{86}{5}. \tag{E.5}$$

is constant for large s.

In order to use the ratio y to find the value of p_c we ran a standard epidemic cluster growing algorithm with occupation probability p and recorded the number of sites in each cluster until appropriate statistics had been collected. Clusters were grown until a preset maximum cutoff $s_{\max}$ was hit. Data were binned by $\log_2 s$, that is, bins for $s = 1$, 2-3, 4-7, 8-15, ... and those that hit $s_{\max}$, with bins both for the

number in each bin, and the total s of all the clusters in each bin. Summing these bins respectively above s and below s allowed us to calculate y for $s = 2, 4, 8, \ldots$, $s_{\max}$. We have verified that this gives the correct threshold for other two dimensional systems where the threshold is known exactly.

When the occupation threshold is above p_c the curve for f is above 17.2, and when the occupation threshold is below p_c the curve is below 17.2. Therefore, we can narrow in on the percolation threshold by testing various values of p and graphing them as shown in figure E.1. First we scan through a range of p values using a large step size and short simulations to determine the approximate p_c. We then run extensive simulations on just a few values of p near the threshold to get an accurate value of p_c. This method is easy to program and efficient when one is trying to find just the threshold, and not the behavior of the system for all values of p, in which case other methods [4, 5] are more efficacious.

For this method to work well, the lattice has to be big enough so that the maximum cluster size $s_{\max}$ could be reached before the boundaries are hit. Then there are no finite-size effects due to the boundaries. However, there are still finite-size (lattice) effects for small clusters; these are generally unimportant for $s > 1000$.

Bibliography

[1] J. L. Jacobsen, "High-precision percolation thresholds and Potts-model critical manifolds from graph polynomials," *J. Phys. A: Math. Th.*, vol. 47, 2014.

[2] D. Stauffer and A. Aharony, *Introduction to Percolation Theory, 2nd ed.* CRC Press, 1994.

[3] P. L. Leath, "Cluster size and boundary distrabution near percolation threshold," *Phys. Rev. B*, 1976.

[4] M. E. J. Newman and R. M. Ziff, "Efficient Monte Carlo algorithm and high-precision results for percolation," *Phys. Rev. Lett.*, vol. 85, pp. 4104–4107, Nov 2000.

[5] M. E. J. Newman and R. M. Ziff, "Fast Monte Carlo algorithm for site or bond percolation," *Phys. Rev. E*, vol. 64, p. 016706, Jun 2001.