Incorporating Mechanics Into the Cellular Potts Model

Citation
Chen, Ruoxi (Michelle). 2020. Incorporating Mechanics Into the Cellular Potts Model. Bachelor's thesis, Harvard College.

Incorporating Mechanics into the Cellular Potts Model

A THESIS PRESENTED
BY
RUOXI (MICHELLE) CHEN

IN PARTIAL FULFILLMENT OF THE HONORS REQUIREMENTS
FOR THE DEGREE OF
BACHELOR OF ARTS
IN THE SUBJECT OF
APPLIED MATHEMATICS

HARVARD UNIVERSITY
CAMBRIDGE, MASSACHUSETTS
APRIL 2020

Thesis advisor: Chris Rycroft

Ruoxi (Michelle) Chen

Incorporating Mechanics into the Cellular Potts Model

Abstract

Lattice-based computational models such as the cellular Potts model (CPM) display a versatility capable of representing systems on both the tissue and cellular level, yet fail to incorporate mechanical interactions crucial for understanding the mechanisms of cell growth and movement. In contrast, the reference map technique (RMT) based on a fixed regular lattice uses a special reference map field that tracks deformations and mechanical stresses to simulate solid mechanics. In this project, I couple the cellular Potts model with the RMT to understand the mechanical behavior of multicellular entities in a dynamically changing environment. This hybrid model allows the CPM to take into account the complex mechanics of the deformation and strain of the material around the cells, while expanding the mechanical model to encompass multicellular effects. To do so, I use the Hamiltonian energy function in the CPM to influence cellular growth and movement. To account for mechanical forces, I model the system as a global continuum field in which cells placed on top of the material should apply a force to the underlying material while also being affected by the material. From a computational perspective, this technique can serve as a generalizable framework for future studies of mechanical interactions between cells.

Contents

Listing of figures

Acknowledgments

This thesis was a project that developed out of a single email from a young and hopeful freshman who had a vague inkling of what she was interested in studying. I would like to express my immense gratitude to my mentor Chris Rycroft for readily accepting this nebulous thought and helping me discover an entire project that could actually play a big role in biological modeling and scientific research as a whole. I am very grateful for his endless patience with my woeful ignorance in the use of C++, git, and shell scripting, and for bearing with me for four years both as a mentor and as an advisor. I would like to shout out the members of the Rycroft lab for being the cutest human beings and being so welcoming to a baby undergrad. Also, even though I ended up not using the CompuCell3D framework for the thesis, I would still like to thank Dr. Maciej Swat, the creator of the CompuCell3D software, for over four years of emails and being very receptive to and encouraging of my initial attempts to code in mechanics into the software. Without CompuCell3D, I would likely not have become interested in this topic almost six years ago. On a more personal level, I want to thank my blinkmates, many of whom have been by my side since the first semester and even before college, for the unending entertainment and companionship over the years. There are also so many other close friends, whether in the Quad or in the faraway River, that have brightened my college experience. Our time together may have been cut short, but I will forever remember these experiences and the friendships I have built over the four years. Finally, I would like to shout out my family and the other special people in my life. I hope that we will stay happy together for a lifetime.

1

Introduction

For years, the growth and development of stem cells, especially in malignant ways, has been a key focus of research, but in many aspects still remains poorly understood. Of particular importance has been the methods in which stem cells divide and differentiate, with a focus on how normal stem cells can transform into malignant cancers. Various studies have attempted to illuminate this process, with a landmark paper in 2000, in which Hanahan and Weinberg introduce the Hallmarks of Cancer: six physiological changes, or rules, that allow cells and tissues to break through their natural cellular defenses and become malignant growth[8]. However, with further progress in cancer and stem cell research, it has become apparent that tumor growth involves much more than the malignant cancer cells themselves—instead, tumors are complex tissues composed of many different cell types that all interact with each other. In this depiction, the benign, normal cells outside of the proliferating cancer cells are actively involved in tumorigenesis, and contribute to the creation of a "tumor microenvironment" that can greatly impact tumor development and metastasis[7].

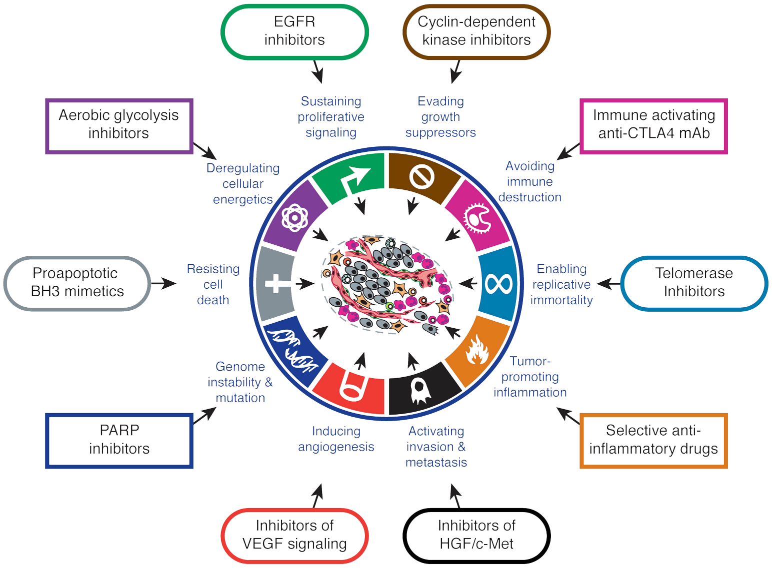


Figure 1.1: The updated ten Hallmarks of Cancer[7].

Indeed, the physical effects between cells and the surrounding environment have proven to be crucial to determining cellular behavior. A key challenge in stem cell research is determining the direction of differentiation in the cell lineage, and Engler finds that the elasticity of the extracellular matrix (ECM) microenvironment can exert very strong influences on this lineage specification[6]. Naive mesenchymal stem cells (MSCs) are shown to have extreme sensitivity to mechanical elasticity, which affects lineage specification and phenotypes. In particular, soft matrices similar to the brain cause MSCs to differentiate into neurons, stiffer matrices similar to muscle cause differentiation into myoblasts, and rigid matrices similar to bone cause differentiation into osteoblasts. While addition of soluble induction factors to influence the lineage can cause reprogramming initially, over time the cells are shown to commit to the lineage specified by the level of matrix stiffness.

By influencing stem cell lineage specification, matrix stiffness and microenvironment conditions can also predict the presence of a tumor or the development of a higher risk of malignant transformation. Paszek et al. find that even small increases in matrix rigidity can greatly influence compatibility with normal tissue morphogenesis and thus contribute to aberrant tissue growth, destabilizing tissue architecture and pushing mammary acini toward the malignant phenotype[12]. But both the immediate microenvironment as well as the extended microenvironment can play crucial roles in tumor growth and metastasis by impacting cell-matrix interactions or host tissue structure[3]. Results have shown that harsher tumor microenvironments that include hypoxia and matrix heterogeneity influence tumors to only grow invasively with fingering margins with a few aggressive clones. In contrast, mild microenvironment conditions, such as normoxia and matrix homogeneity, would be less selective, allowing aggressive clones to grow along with less aggressive tumor phenotypes with smooth and noninvasive margins[4]. These results suggest that the increased aggressiveness of various tumors may be a consequence of developing in the context of harsher environments, illustrating the importance of the ECM on tumor development, and the fact that a variety of physical and environmental factors can greatly influence the development and behavior of cells in biological systems.

1.2 Cellular Behavior Models

To simulate all aspects of cellular behavior, researchers have adopted many different methods at varying levels of detail. On the cellular level, one study introduces a two-dimensional model that focuses on a single cell by representing the eukaryotic cell as a deformable body using elastic springs to define cell shape as the cell plasma membrane and a viscous incompressible fluid to define cell mass, with a single point to represent the cell nucleus[15]. Five phenotypically different cell populations are incorporated into this model, each undergoing four cellular processes, including growth, division, apoptosis, and polarization. Cellular homeostasis is implemented using sources and sinks of fluid on opposites of the cellular membrane, and the fluid transport is adjusted accordingly to represent cell growth and apoptosis. Of particular interest is the incorporation of extracellular interactions through cell membrane receptors, which allows for cell adhesion and cell proliferation depending on the nearby environment[15].

On a different scale, the Proliferation-Invasion-Hypoxia-Necrosis-Angiogenesis (PIHNA) model involves much larger quantities of cells and tissues, incorporating normoxic glioma tissue, hypoxic glioma tissues, vascular tissue, necrotic cells, as well as angiogenic factors to characterize glioma through various coupled reaction-diffusion equations[9]. In this model, many different cell populations and tissues are all competing for space, each with different phenotypes and actions. Rather than predicting cellular behaviors as in many other studies, the PIHNA model mainly focuses on quantifying and explaining cell behavior and evolution on an imaging scale. To do so, since the PIHNA model only captures cell tissues and angiogenesis, Hawkins-Daarud et al. incorporated edema into the original model, allowing for more accurate descriptions of many MR imaging cases of glioblastoma in which vasogenic edema would emerge due to breakdown in the blood-brain barrier[9].

Modeling cancer cell invasion commonly incorporates multiple pathways, including the intracellular pathway, extracellular pathway, as well as interactions connecting the intracellular dynamics with the extracellular stroma[14]. Thus, different representations can be combined to introduce a hybrid discrete-continuum model. Ramis-Conde et al. introduce one such model in which the continuum model describes the interaction of chemicals with the ECM, where cells act as sources and sinks to incorporate the dynamics of the intracellular and extracellular environments. In turn, the discrete model represents the cells as discrete particles, which interact with each other and the surrounding environment through a potential function[14]. Invasion is triggered by contact between peripheral cancer cells and the ECM, adjacent stroma configurations can stimulate cells to migrate via chemotactic or haptotactic gradients, and mitosis results from cells absorbing growth factors in the ECM[14].

In contrast, the Subcellular Element Model can be used both to represent the dynamics of many three-dimensional deformable cells in multicellular systems or a single cell in more detail. In the first case, the model serves as an efficient off-lattice model for simulating thousands of three-dimensional cells in a complex three-dimensional environment by representing the cytoskeleton as a network of dynamically cross-linked filaments. To model a highly detailed single cell, many subcellular elements form the cell and are systematically placed into ordered close-packing arrangements that adjust into metastable states of equilibrium, which can be influenced by environmental fluctuations[17].

Each model implements a different way to visualize cellular behavior and represents individual cells with different forms, as well as varying levels of detail. With the large variety of computational methods, different modeling techniques are chosen based on the conditions of the cells and the environment to be modeled.

1.3 Cellular Potts Model

One widely-used computational model is the cellular Potts model, or the Glazier-Graner-Hogeweg Model. The cellular Potts model is a lattice-based computational modeling method to describe the behavior of cells and their interactions as a result of the initial conditions and the evolution of the cellular and extracellular properties over time. A Hamiltonian function describes the various interactions in the system, such as terms that preserve the cell volumes and describe the surface energy of their boundaries. A stochastic update procedure is employed, where random perturbations to the cell shapes are considered, and they are rejected or accepted depending the change in the value of the Hamiltonian. A wide variety of additional cellular behaviors can be implemented by incorporating different energy terms into the overall Hamiltonian energy function.

The versatility of this computational model has led it to be used in a variety of systems, including angiogenesis[13], cell sorting[28], and tumor growth[2,5]. One example of a three dimensional simulation of tumor growth modeled using the cellular Potts model can be seen

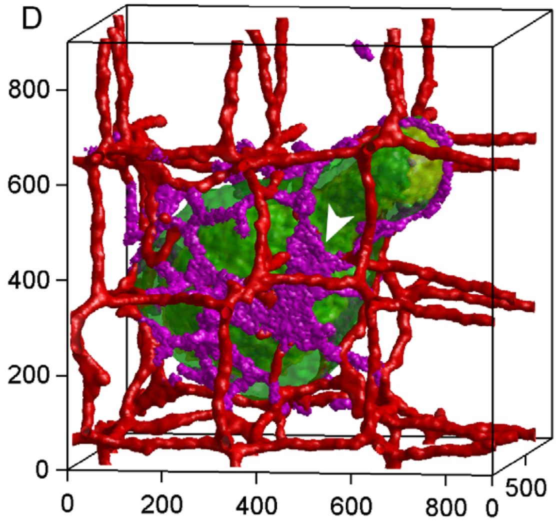


Figure 1.2: A three-dimensional simulation of tumor growth and angiogenesis using the cellular Potts model. In the figure, the green cells are normal tumor cells, the purple cells are active neovascular cells, and the red vessels are the preexisting vasculature.[20]

in Figure 1.2. The cellular Potts model is capable of representing systems on both the tissue level, as in the study of its continuum limits[1], as well as at the cellular level, as in the movement of individual cells[18].

The cellular Potts model is an example of an agent-based model. Individual cells are defined by a program that dictates their internal state, their laws of motion, and their interactions with other. Agent-based models are a viable candidate for investigating multicellular interactions. However, inherent to this model is the assumption that cells were organized in a densely packed regular structure and that the environment was not being manipulated in any way other than through the release of various morphogens. Thus, while the lattice-based viewpoint is simple and effective, it is highly idealized and is not well-suited to incorporating certain physical effects. In particular, it completely neglects the mechanics of how cells pull and push on each other and their environment, which is known to be of crucial importance in many situations.

Various studies have attempted to extend the cellular Potts model to incorporate cell-ECM interactions. Rubenstein and Kaufman utilize the cellular Potts model to study the complex interactions between glioma cells and their extracellular matrix, appreciating the ease of use, transparency, and access to subcellular details offered by the model[16]. In their model, tumor cells were introduced to a lattice representing the ECM, which was composed of two components, one of which was a fibrous scaffold inhomogeneous on the cellular scale, and the other which was distributed homogeneously throughout the lattice. This two-component ECM was consistent with tissue approximations of fibrous proteins used to study cell migratory behavior, and was similar to the structure of collagen gels used in glioma studies. Here, each of the glioma cells, as well as the ECM elements, were modeled using the cellular Potts model, along with rules for mitosis when qualified cells touch at least one collagen or matrix site[16]. The two-dimensional model was interpreted as essentially a slice through the centers of three-dimensional spheroids, to reproduce behaviors associated with spheroids grown in three-dimensional extracellular matrices. While the model was able to qualitatively reproduce spheroid growth and invasion patterns in collagen I matrices, the model had limited agreement with experimental changes when allowing cells to alter their environment[16].

In a different study, Kabla uses the cellular Potts model to explore mechanisms leading to coordinated cellular motion in a number of biological processes, such as cancer metastasis or embryo development[10]. The tissue is represented as a densely packed two-dimensional layer of epithelial cells on an underlying substrate, where cell-cell cohesion is homogeneous across the population. To model the interaction between the cell and the mechanical environment, there is feedback from earlier displacements to the polarization itself, in which when a cell attempts to move along its direction of polarization, the polarization direction evolves toward the cell's net displacement over time[10]. Motility is created through each cell, which generates a motile force along its polarization direction. These forces act between the cell and the substrate through an additional cell migration term in the overall Hamiltonian energy in which the direction of the motile force is determined by feedback from earlier movement, oriented along the mean velocity of the cell. The model does reproduce the coordinated cellular movement associated with wound healing, but ultimately incorporates cell movement solely based on the motile forces produced by individual cells on an unchanging substrate, which can be complicated by nonlinear changes and cellular interactions from the underlying substrate.

1.4 Objective

Many models that do take into account physical effects and mechanics are based on a moving mesh, which makes these models very difficult to combine with the versatility of the cellular Potts model, which is based on a lattice. To circumvent these issues, we implement the reference map technique (RMT)[11,25]. This recently developed numerical method is based on a fixed regular lattice and uses a reference map field to track the deformation of the material and the generated mechanical stresses. Thus, despite acting upon a lattice grid, the model still grants the flexibility to formulate nonlinear dynamics, allowing the model to take into consideration both the changing physical properties of the simulation space due to deformation while still being able to track individual cells. The model has been used to investigate mechanical interactions between tumor cells mediated through the extracellular environment[19]. The method is well-suited to modeling interactions that depend on the nonlinear properties of biological materials[22] that become apparent at large deformations.

In this paper, we seek to incorporate mechanical effects in the cellular Potts model. Given its lattice-based foundation, the RMT appears to be the ideal approach to implementing consideration for nonlinear physical effects into the CPM. We believe that this technique has broad applicability to study mechanical interactions between cells and the influences of such interactions on cellular behavior and development.

2

Methods

2.1 The Cellular Potts Model

The cellular Potts model (CPM) serves as a lattice-based computational modeling method to describe the behavior of cells and their interactions as a result of the initial conditions and the evolution of the cellular and extracellular properties over time.

In the CPM, the simulation involves parameters and properties delineated in table 2.1.

Table 2.1: List of variable parameters in the cellular Potts model.

Parameter	Description
$\sigma(\vec{\imath})$	Index of lattice pixel $\vec{\imath}$
$\tau\left(\sigma(\vec{\imath})\right)$	Cell type of cell with index $\sigma(\vec{\imath})$
$\mathcal{J}_{\tau(\sigma(\vec{\imath}))}$	Adhesion energy
λ_V	Volume constraint constant
v_σ	Volume of cell $\vec{\imath}$
V_T	Target Volume

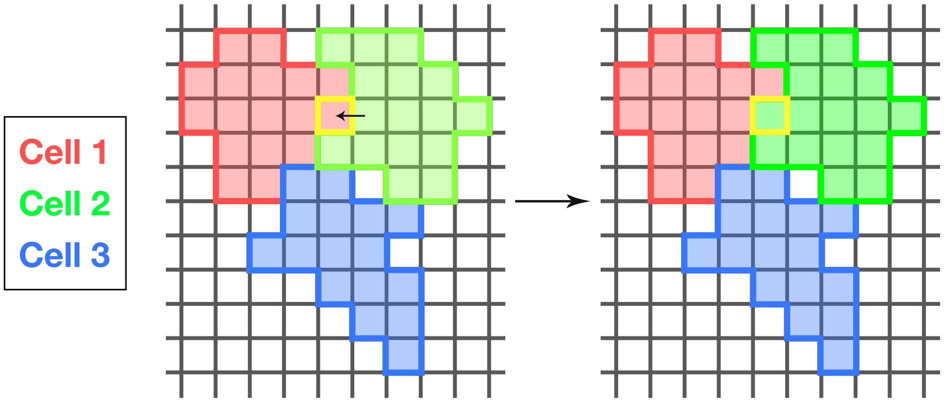


Figure 2.1: Schematic of cells in the cellular Potts model and an index copy attempt. Pixel delineated in yellow is the randomly chosen target pixel, and an arrow is pointing from the randomly chosen neighboring source pixel to the target pixel. In this case, the index copy attempt succeeds, and the pixel copy is made.

Within a regular lattice, each cell is defined as the collection of lattice pixels with the same identification index, $\sigma(\vec{\imath})$, where $\vec{\imath}$ is a lattice pixel. Using a Monte Carlo algorithm dependent on an effective energy, at each step a target pixel $\vec{\imath}$ is selected randomly, and a source pixel $\vec{\imath}'$ is chosen randomly from the cells neighboring the target pixel, as seen in figure 2.1. An attempt is made to change the index of the target pixel $\sigma(\vec{\imath})$ to that of the source pixel $\sigma' = \sigma(\vec{\imath}')$. If $\sigma(\vec{\imath}) = \sigma(\vec{\imath}')$, the two pixels are within the same cell and no pixel copy is made. If the pixels are from different cells, then $\sigma(\vec{\imath}) \neq \sigma(\vec{\jmath}')$. The change in the overall effective energy of the system $\Delta\mathcal{H}$ is calculated, and the pixel copy is accepted with a probability $P(\Delta\mathcal{H})$ given by the Boltzmann acceptance function,

$$P(\Delta\mathcal{H}) = \begin{cases} 1, & \Delta\mathcal{H} \leq 0, \\ e^{-\frac{\Delta\mathcal{H}}{T_m}}, & \Delta\mathcal{H} > 0, \end{cases}$$

where T_m is a parameter representative of the temperature in the system due to its effect of increasing the rate of successful pixel reassignments[23]. Thus, this algorithm describes evolution that tends to minimize the overall effective energy, which is thought to be representative of actual cell behavior. A Monte Carlo step (MCS) in the simulation consists of one index-copy attempt for each lattice pixel.

The effective energy $\mathcal{H}$, or Hamiltonian energy, is defined as a sum of energies that incorporate different cellular behaviors, including components for adhesion energy and volume constraints:

$$\mathcal{H}_{\text{CPM}} = \mathcal{H}_{\text{adhesion}} + \mathcal{H}_{\text{volume}}$$

The adhesion energy component between the cells is defined as:

$$\mathcal{H}_{\text{adhesion}} = \sum_{\vec{\imath},\vec{\jmath}\,\text{neighbors}} \mathcal{J}_{\tau(\sigma(\vec{\imath})),\tau(\sigma(\vec{\jmath}))} \left(1 - \delta_{\sigma(\vec{\imath}),\sigma(\vec{\jmath})}\right)$$

where $\mathcal{J}$ is defined as the boundary energy between two cells $\sigma(\vec{\imath})$, $\sigma(\vec{\jmath})$ of cell type $\tau\left(\sigma(\vec{\imath})\right)$, $\tau\left(\sigma(\vec{\jmath})\right)$ at the interface between two neighboring pixels[26]. The adhesion energy sum iterates over all neighboring cell pairs $\vec{\imath}, \vec{\jmath}$. We have $\mathcal{J}_{\tau(\sigma(\vec{\imath})),\tau(\sigma(\vec{\jmath}))} = \mathcal{J}_{\tau(\sigma(\vec{\jmath})),\tau(\sigma(\vec{\imath}))}$, and δ is defined as

$$\delta_{x,y} = \begin{cases} 1 & x = y \\ 0 & x \neq y \end{cases}$$

The volume constraint energy component is defined as

$$\mathcal{H}_{\text{volume}} = \sum_{\sigma} \lambda_V (v_\sigma - V_T)^2,$$

where λ_V represents the stiffness of the cell, v_σ indicates the cell volume, and V_T indicates the target volume of the cell. Deviations of the cell volume from the target volume of the cell are penalized using this energy.

2.1.1 Mitosis

Mitosis is implemented in the cellular Potts model in a relatively simple fashion. The conditions for a cell undergoing mitosis are straightforward: as long as the number of cells present is lower than the specified cell number cap and the cell volume is larger than the mitotic cell volume specified, the cell will undergo mitosis.

During mitosis, the cell is divided into two daughter cells of equal volume. To do so, the center of mass of the cell is calculated as

$$(c_i, c_j) = \left(\frac{\sum\limits_{i \in \sigma} m_i}{v_\sigma}, \frac{\sum\limits_{j \in \sigma} m_j}{v_\sigma} \right)$$

where σ is the index of the cell, v_σ is the volume of the cell, c_i, c_j are the x and y coordinates of the cell center of mass, respectively, and m_i, m_j are the x and y coordinates of the pixels contained in the cell σ.

Given the center of mass, we divide the cell along an axis of division determined by the eigenvector of the cell's covariance matrix corresponding to the largest eigenvalue. We do so by computing the variance $\sigma(x,x)$ in the x-direction and the variance $\sigma(y,y)$ in the y-direction, as well as the covariance

$$\sigma(x,y) = E[(x - E(x))(y - E(y))],$$

where here $E(z)$ is the z-coordinate of the center of mass of the cell and z is either x or y. We summarize the variance of the cell in the form of the covariance matrix,

$$\Sigma = \begin{bmatrix} \sigma(x,x) & \sigma(x,y) \\ \sigma(y,x) & \sigma(y,y) \end{bmatrix}.$$

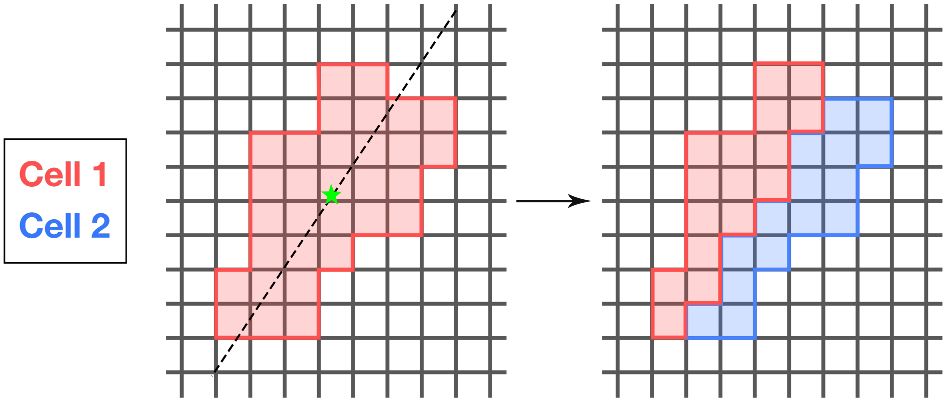


Figure 2.2: Schematic of a cell undergoing mitosis in the cellular Potts model. The original cell is divided along the axis corresponding to the largest variance in the cell, splitting into two daughter cells of approximately equal volume.

Now, the covariance matrix defines both the spread and the orientation of the cell. We find the vector that points in the direction of the largest spread of the cell, given by the largest eigenvector of the covariance matrix, and divide the cell using that axis. Upon computing the slope of the eigenvector and setting the mitotic axis as the line with that slope passing through the center of mass of the cell, all points in the cell above the axis are classified in one new daughter cell and all points in the cell below the axis are classified in another new daughter cell, as seen in figure 2.2. The volumes and centers of mass of the two daughter cells are updated accordingly. The new cells continue to grow and divide as long as the conditions for mitosis are satisfied.

2.2 The Reference Map Technique

2.2.1 Finite Strain Theory

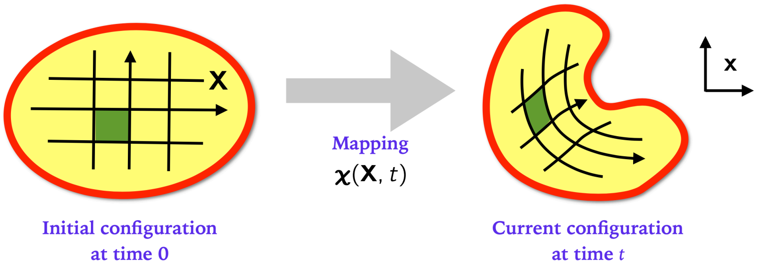


Figure 2.3: Basis of finite strain theory. A grid is imposed over the simulation space, and the deformation of a small section is described by a linear transformation.

Through modelling the complex simulation space as a material similar to collagen in order to take into consideration changing physical properties and mechanics, the reference map technique uses the mathematical formalism of finite strain theory. Considering deformation in only two dimensions, the formalism divides a region into a set of small sections. By eval-

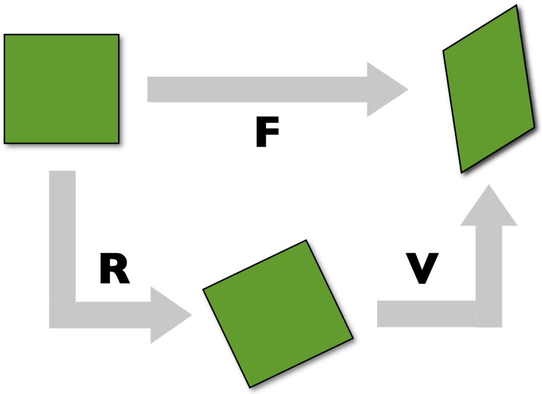


Figure 2.4: The linear transformation, known as the deformation gradient **F**, can be decomposed into orthogonal rotation matrix **R** and a symmetric stretch matrix **V**.

uating the Jacobian of the relative positions of the deformed material, the model represents the transformation of an initial square into a deformed quadrilateral with the deformation gradient tensor **F** in the form of a 2×2 matrix.

The tensor **F** is then decomposed using the polar decomposition theorem into an orthogonal rotation matrix **R** and a symmetric stretch matrix **V**:

$$\mathbf{F} = \mathbf{V}\mathbf{R}.$$

It follows that the Cauchy Stress is

$$\sigma = \frac{1}{\lambda_1 \lambda_2} \left(\lambda_1 \frac{\partial W}{\partial \lambda_1} \mathbf{n}_1 \otimes \mathbf{n}_1 + \lambda_2 \frac{\partial W}{\partial \lambda_2} \mathbf{n}_2 \otimes \mathbf{n}_2 \right), \tag{2.1}$$

where λ_1, λ_2 are defined as the eigenvalues of $\mathbf{V}$ and $\mathbf{n}_1, \mathbf{n}_2$ are defined as the eigenvectors of $\mathbf{V}$. $W(\lambda_1, \lambda_2)$ is defined as the strain-energy density function, defines the elastic energy stored in the material in terms of λ_1 and λ_2 and will be explained further in section 2.3.3[27].

We define a velocity field $\mathbf{u}(\mathbf{x}, t)$ and a position field $\mathbf{X}(\mathbf{x}, t)$ for the material based on position $\mathbf{x}$ in the deformed configuration. Based on Newton's second law, the velocity field satisfies

$$\rho \frac{d\mathbf{u}}{dt} = -\phi \mathbf{u} + \nabla \cdot \sigma \tag{2.2}$$

where ρ is the density and $-\phi\mathbf{u}$ is a global drag of the material on the the underlying substrate. We work in the overdamped limit where the drag is assumed to be much larger than the acceleration, so that Eq. (2.2) can be simplified to

$$\mathbf{u} = \phi \nabla \cdot \sigma. \tag{2.3}$$

The reference map is updated using the transport equation

$$\frac{\partial \mathbf{X}}{\partial x} = -\left(\mathbf{u} \cdot \nabla\right) \mathbf{X}$$

where we define $\frac{d}{dt} = \frac{\partial}{\partial t} + (\mathbf{u} \cdot \nabla)$ as the advective derivative.

2.2.2 Numerical Simulation Technique

The reference map technique allows for numerically simulating the evolution of a biological material on a regular fixed grid with an Eulerian time step, simplifying calculations for each cell[25].

A schematic of the spatial discretization of the reference map technique is shown in figure 2.5. The simulation domain covers the coordinate ranges $x \in [a_x, b_x]$ and $y \in [a_y, b_y]$, which

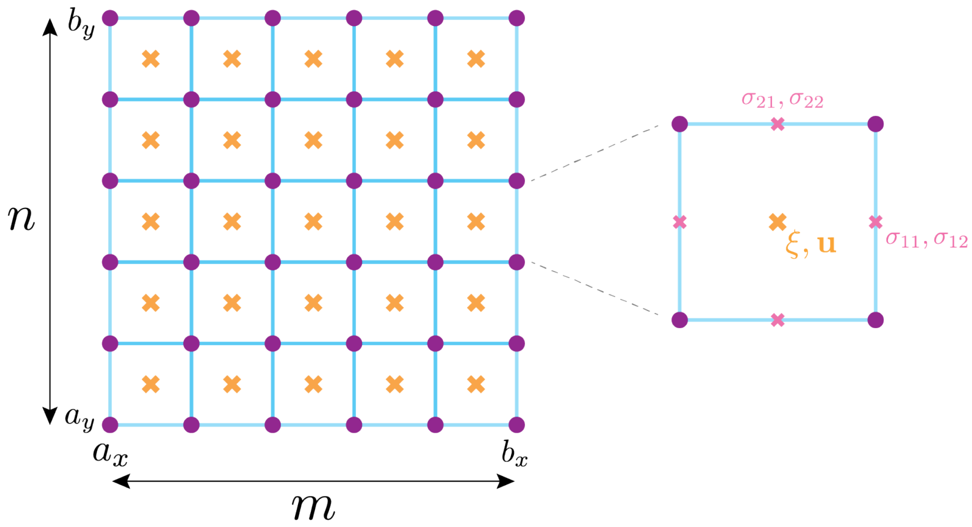


Figure 2.5: The simulation space is divided into a grid of square cells each of width h_x and height h_y. At the center of each cell, the reference map ξ and the velocity field **u** is stored, and on the edges, the stress tensor components are stored.

is discretized into an $m \times n$ grid of lattice cells. The cells have side lengths spacings

$$h_x = \frac{b_x - a_x}{m}, h_y = \frac{b_y - a_y}{n}$$

in the horizontal and vertical directions, respectively. For simplicity, we use equal spacings along the two axes, so each lattice cell has side length $h = h_x = h_y$.

We can represent each cell as a square lattice point. We store the velocity field $\mathbf{u}(\mathbf{x}, t)$ and the reference map field $\xi(\mathbf{x}, t)$ at the center of cell. The reference map ξ serves to track the deformation of the material. On the edges of each cell, we store the stress tensor components,

$$\sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{21} & \sigma_{22} \end{bmatrix}$$

.

We divide the components of the simulation into different steps labeled **A** through **E**.

Step A: Field initialization. We initialize the reference map to the undeformed configuration with

$$\xi(\mathbf{x}, t = 0) = \mathbf{x}$$

and the velocity field $\mathbf{u}$ to rest.

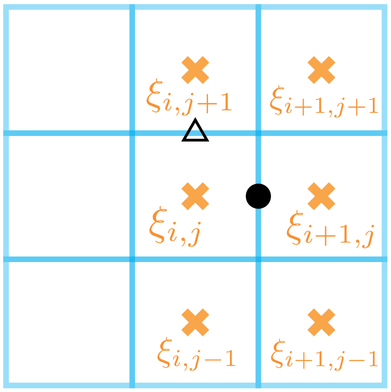


Figure 2.6: The stress is computed at the edges of the cell and stored at the black dot and triangle. The stress function for the cell at i, j depends on the neighbors of the cell.

Step B: Stress computation. We first define the function to compute stress σ. At the black dot in the figure 2.6, the stress for cell i, j is stored. We have

$$\begin{aligned} \left[\frac{\partial \xi}{\partial x}\right] &= \frac{\xi_{i+1,j} - \xi_{i,j}}{h}, \\ \left[\frac{\partial \xi}{\partial y}\right] &= \frac{\xi_{i+1,j+1} + \xi_{i,j+1} - \xi_{i+1,j-1} - \xi_{i,j-1}}{4h}, \end{aligned}$$

where $\xi_{i,j}$ is the value of the reference map ξ at the cell at i, j and $\xi = (\xi_x, \xi_y)$. Now our deformation gradient is

$$\mathbf{F} = \begin{pmatrix} \frac{\partial \xi_x}{\partial x} & \frac{\partial \xi_x}{\partial y} \\ \frac{\partial \xi_y}{\partial x} & \frac{\partial \xi_y}{\partial y} \end{pmatrix}^{-1} .$$

From here, the stress can be computed using the constitutive law in Eq. (2.1). Upon calculating the stress, we only store components σ_{11} and σ_{12} at the black dot. We repeat the same procedure to calculate the stress at the triangle using stencils with x and y transposed, and we only store components σ_{21} and σ_{22} there.

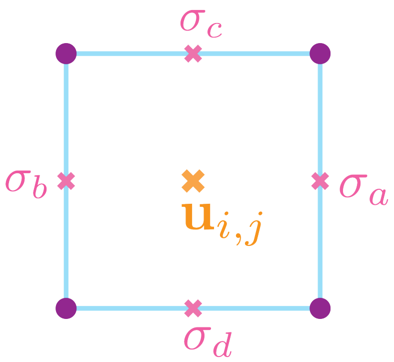


Figure 2.7: The velocity is computed and stored at the center of the cell, based on the stress components stored at the edges of the cells.

Step C: Velocity computation. The velocity follows Eq. (2.3). Given the previously computed stresses on edges as shown in figure 2.6, the velocity components $\mathbf{u} = (u_1, u_2)$ at a point are given by

$$(u_k)_{i,j} = \phi \left(\frac{(\sigma_{1k})_a - (\sigma_{1k})_b}{h} + \frac{(\sigma_{2k})_c - (\sigma_{2k})_d}{h} \right),$$

where the stresses are arranged as shown in figure 2.7.

Step D: Reference map update. To update the reference map with the given velocity, we employ a second-order essentially non-oscillatory (ENO) method[21]. We have

$$\frac{f_{i,j}^{k+1} - f_{i,j}^{k}}{\Delta t} = -u_1 \left(\frac{\partial f}{\partial x}\right)_{i,j} - u_2 \left(\frac{\partial f}{\partial y}\right)_{i,j}$$

where k is the index of the collagen simulation time step.

When $u_1 > 0$, the ENO derivative in the x direction is computed as

$$\left[\frac{\partial f}{\partial x}\right]_{i,j} = \frac{1}{2h} \begin{cases} 3f_{i,j} - 4f_{i-1,j} + f_{i-2,j} & \text{if } |[f_{xx}]_{i,j}| > |[f_{xx}]_{i-1,j}|, \\ f_{i+1,j} - f_{i-1,j} & \text{otherwise,} \end{cases}$$

where we compute the second derivative f_{xx} using the stencil of

$$\left[\frac{\partial^2 f}{\partial x^2}\right]_{i,j} = \frac{f_{i+1,j} - 2f_{i,j} + f_{i-1,j}}{h^2}.$$

If the $u_1 < 0$ then the mirror image of the stencil is taken. The ENO derivative in the y direction is handled similarly[27].

Step E: Cell contraction. Now we compute the function to implement contraction, in which the cell pulls with a radially inwards force on the collagen. This is done quite simply. For a cell at lattice site i, j, the contraction stress constant c_{str} is added to the stress components on all four edges: to σ_{11} on both sides and σ_{22} on the top and bottom edges.

With all of these components computed, the simulation follows the following steps:

1. At the start of the simulation, we initialize the simulation with step **A**.

2. After initializing the simulation, we set up the stress and velocity fields with steps **B** and **C**. This ensures that the simulation fields are all set up and consistent at time $t = 0$.

3. During the course of one collagen simulation time step, we implement contraction from the cell on the collagen through step **E**.
4. After adding contraction, we compute the velocity from the updated stress components as in step **C**.
5. After calculating the velocities, we can move the collagen as in step **C**.
6. After moving the collagen, we recalculate the stress tensors based on the updated reference map as in step **B**.
7. The steps above are repeated until the end of the simulation.

2.3 Coupling the Models

2.3.1 Reconciling Units

The first step to coupling two different models, the cellular Potts model and the reference map technique, lay in reconciling two different sets of measurements and units. The collagen simulation is described using physical units. For the purposes of this study, we work with simulation units that are non-dimensionalized according to reference a length scale L_{ref}, time scale T_{ref}, and mass scale M_{ref}. Any quantity in the simulations can be converted back to a physical quantity by multiplying by the appropriate scales; for example a simulation velocity can be multiplied to $L_{\text{ref}}/T_{\text{ref}}$ in order to obtain the corresponding physical value. Since our purpose in this thesis is to demonstrate the numerical method and the simulation as opposed to applying it to a specific problem, we do not provide a detailed calibration of the physical scales.

In our simulations, the collagen simulation takes place in a domain $x \in [-3, 3], y \in [-3, 3]$ with a 150×150 grid. This exactly aligns with a 150×150 grid in the Potts model that is exactly overlaid.

Connecting the two models temporally is more complex. The reference map simulation of the collagen uses a timestep Δt, while the cellular Potts model advances based on Monte Carlo steps, as detailed in Section 2.1. A single Monte Carlo step in the cellular Potts model only implements a single index copy attempt, and thus with larger lattice grids, more index copy attempts and thus more Monte Carlo steps would be needed to enact the same level of change in a single physical time unit of the reference map technique. Every reference map technique time step, the collagen simulation is updated and forces are adjusted and implemented. Thus, the number of Monte Carlo steps associated with each reference map technique physical time unit is dependent upon the size of both grids.

Let Δt be a physical time step of the reference map technique, which is adjusted based on the size of the lattice grid. Then we want to find d_{MCS}, the number of Monte Carlo steps that is associated with one Δt, and calibrate using d_{MCS} to find the best number of Monte Carlo steps to implement between stages of adjusting and coupling to the collagen–that is, how many Monte Carlo steps to associate with one Δt collagen time step.

Here, we have τ, the physical time that passes per Monte Carlo step per grid point. τ is a parameter that could have an actual time scale accorded to it, which would be adjustable corresponding to the simulation. Since τ is the time per Monte Carlo step for all grid points, per Monte Carlo step and index copy attempt, we have a different value τ_2, which is $\frac{\tau}{mn}$, where mn is the total number of grid points in the lattice grid. This gives us the physical time that passes per Monte Carlo step. Now, Δt is the physical time that passes per collagen simulation time step, which is calculated relative to the overall grid size. This gives us an overall d_{MCS}:

$$d_{\text{MCS}} = \frac{\Delta t}{\tau_2},$$

where d_{MCS} gives the number of Monte Carlo steps associated with one collagen simulation time step.

Based on the grid and parameters, we have two possible cases for the value of d_{MCS}:

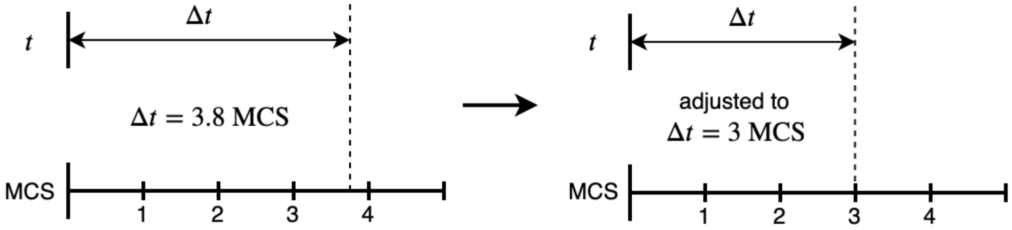


Figure 2.8: Calibration for Δt in the case when $d_{\text{MCS}} > 1$. In this example, we assume the calculated $\Delta t = 3.8$ Monte Carlo steps.

The first case is when d_{MCS} is larger than 1. This is more likely as many Monte Carlo steps would be needed in order to iterate over an entire frame, thus leading to more Monte Carlo steps needed than collagen simulation physical time steps. In this case, multiple Monte Carlo steps would need to be implemented for each collagen simulation time step. We then calibrate our time steps with the following parameters:

parameter	variable	value
number of physical time steps to take per MCS	b_{cs}	1
number of MCS to take before applying collagen	f_{cs}	$\lfloor d_{\text{MCS}} \rfloor$
length of a physical time step, in physical time units	Δt_{cs}	$\Delta t \cdot f_{cs}$

We take the floor of d_{MCS} to determine the number of Monte Carlo steps to take per physical time step Δt of the collagen simulation to avoid increasing Δt, which could break the stability of the coupling of our simulations. Reducing the number of MCS per Δt would lead to increased numbers of steps implemented of the collagen simulation, but will not affect the stability of our model.

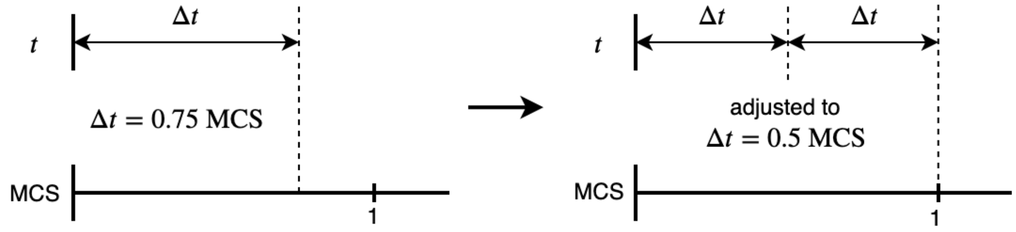


Figure 2.9: Calibration for Δt in the case when $d_{\text{MCS}} < 1$. In this example, we assume the calculated $\Delta t = 0.75$ Monte Carlo steps.

The second case is when the d_{MCS} is smaller than 1. In this case, multiple collagen simulation time steps would need to be implemented for each Monte Carlo step. We then calibrate our time steps using the following parameters:

parameter	variable	value
number of physical time steps to take per MCS	b_{cs}	$\lceil \frac{1}{d_{\text{MCS}}} \rceil$
number of MCS to take before applying collagen	f_{cs}	1
length of a physical time step, in physical time units	Δt_{cs}	$\frac{\Delta t}{b_{cs}}$

Again, we avoid increasing Δt in calculating the number of physical time steps to take per MCS. To do so, we essentially round up the number of Δt that would fit in a Monte Carlo step.

How the parameters in the tables are implemented is as follows. Every f_{cs} Monte Carlo steps, one physical time step is reached, and the collagen simulation is called and adjusted. The collagen simulation steps forward b_{cs} steps, each of which is calculated in the collagen simulation based on Δt_{cs} physical units of time. After these b_{cs} collagen steps are finished, we proceed with the Monte Carlo steps in the cellular Potts model, which is calculated with the updated collagen simulation.

2.3.2 INCORPORATING MECHANICS

After calibrating the time steps of the two models, we seek to implement the mechanics that serve as the basis of our study. In order to incorporate mechanics into the cellular Potts model, we visualize the coupling to involve a lattice grid of material interacting with the cells. The cells would be represented using the cellular Potts model, and the nature of the reference map technique would allow for representation of mechanics at the lattice points. For the two models to couple, the material would need to interact with the cells, which would be from forces exerted from the material on the cells. The basis for the interaction between the cellular Potts model and the reference map technique is the velocity field in the reference map, which would exert a force on the cells to influence their movement.

We can directly allow the cells to move in response to the underlying velocity field from the substrate. In this method, we have the underlying velocity field $\mathbf{u}(x, t)$ from the collagen simulation, as mentioned in Section 2.2.1, and the center of mass $\mathbf{r}_\sigma$ of the cell σ. The center of mass of the cell moves according to the underlying velocity:

$$\frac{d}{dt}\mathbf{r}_\sigma = \mathbf{u}(r_\sigma, t)$$

Now to introduce motility into the model, the overall Hamiltonian energy of the system becomes

$$\mathcal{H}_{\text{CPM}} = \mathcal{H}_{\text{adhesion}} + \mathcal{H}_{\text{volume}} + \mathcal{H}_{\text{velocity}}$$

where the velocity field constraint energy component is defined as:

$$\mathcal{H}_{\text{velocity}} = \sum_{\sigma} \lambda ||r_\sigma - m_\sigma||^2$$

where m_σ is the average of the grid positions of cell σ. Thus, by minimizing this Hamiltonian energy, each cell follows along with the center of mass of the cell, which moves according to the underlying velocity field of the substrate.

The other method by which the cells interact with the collagen is through contraction stress, in which the cell pulls on the collagen. This process is discussed in section 2.2.2.

2.3.3 CONSTITUTIVE LAW

With modeling our substrate on collagen, we want to be able to produce complex physical properties of collagen in our reference map. We do so by implementing a series of constitutive equations that would explain the properties of collagen, starting by initializing our velocity and position fields as indicated in section 2.2.1. The stress σ, defined in Eq. (2.1), depends on $W(\lambda_1, \lambda_2)$, a strain-energy density function.

To derive this relation, we consider the collagen forces in one dimension. The fibers in collagen cause the force to be much stronger in tension than compression, so we can describe the strain-stress relationship as the function:

$$f(x) = d(x-1) + \begin{cases} a(e^{b(x-1)} - 1) & \text{if } x < c, \\ rx + s & \text{if } x \geq c. \end{cases} \tag{2.4}$$

Enforcing continuity and differentiability at $x = c$ requires

$$r = abe^{b(c-1)}, \qquad s = ae^{b(c-1)}(1 - bc) - a. \tag{2.5}$$

with three free parameters: a, b, and c. a controls the linear force response, b controls the nonlinear force response, and c controls the extent when the collagen effectively becomes a linear material[27]. We define our one-dimensional energy potential as

$$g(x) = \int_1^x f(y)dy \tag{2.6}$$

and then derive the strain-energy density function in two dimensions as

$$W(\lambda_1, \lambda_2) = g(\lambda_1 + \alpha\lambda_2 - \alpha) + g(\lambda_2 + \alpha\lambda_1 - \alpha). \tag{2.7}$$

This W function can then be used to calculate the stress for the collagen, as in Eq. (2.1)[27].

In our simulation, we define $a = 0.03, b = 20, c = 1.3, d = 9$ in the equations above. We define the drag parameter in Eq. (2.3) as $\phi = 0.2$.

2.3.4 VELOCITY FIELD COUPLING

As discussed in the introduction in section 1.3, in one study, Kabla et al. explored incorporating cellular motion by implementing a motile force on each cell. The force is generated along the polarization direction n_σ, with amplitude μ_σ. To introduce this motility into the model, the overall Hamiltonian energy of the system becomes

$$\mathcal{H}_{\text{CPM}} = \mathcal{H}_{\text{adhesion}} + \mathcal{H}_{\text{volume}} - \mathcal{H}_{\text{motility}}$$

where the motility constraint energy component is defined as:

$$\mathcal{H}_{\text{motility}} = \sum_{\sigma} \mu_{\sigma} n_{\sigma} \cdot r_{\sigma}$$

where r_σ represents the position of the center of mass of cell σ. The polarization direction of the motile force n is determined by feedback from earlier displacements of the cell σ[10].

While considering methods to incorporate mechanics and movement into the cellular Potts model, we also considered adding motile forces in the same vein as in Kabla's paper. However, upon conducting tests with an array of parameters, the motile force did not seem to be constructive to our work. In particular, it was unclear what the relationship was between the parameter μ_σ and the actual velocity that would result from the cellular Potts model. The velocity seemed to differ for different parameter values, different grid sizes, as well as different polarization directions (velocity directions). Thus, the inconsistency in the force implementation and the resulting velocities in the model itself led us to decide not to incorporate independent motile forces and instead focus on coupling the cells with the velocity field of the underlying collagen.

2.3.5 The Simulation

Upon combining the two separate simulations into one model, there are a lot of moving parts associated with the model. Here we present a step-by-step overview of how the simulation is set up and runs while incorporating both forms of mechanics and adjustments from both techniques:

1. Set up the parameters to be used in the simulation. This includes mitosis volume, the cap on cellular division, target volume of the cells, and temperature, which are all used in the cellular Potts model. This also includes multipliers and rates used in the collagen simulations.

2. Create our simulation with parameters such as the physical grid sizes for the collagen simulation (in our case, -3 to 3 for both axes) and lattice dimensions for the cellular Potts model grid (150×150).

3. Add cells to the cellular Potts model. This could take the form of single cells or cell clusters of different shapes.

4. Initialize parameters for the various terms involved in the Hamiltonian energy of the cellular Potts model. This includes adhesion energy magnitudes, volume constraint terms, and magnitudes for the motility terms.

5. The overall simulation is initialized. This includes calculating all the cell volumes in the model and ensuring that the target positions for the cells, which would be used for the mechanics terms, is set to the center of mass of the cells. The collagen simulation is also initialized using parameters defined previously, and the parameters needed for time step calibration between the two simulations are calculated (see Section 2.3.1).

6. If there is no coupling between the collagen simulation and the cellular Potts model–there is no velocity field term or contraction stress as defined in section 2.3.2–only the

cellular Potts model is run.

(a) One Monte Carlo step iteration is run. In each iteration, mn individual Monte Carlo steps are run, where mn is the number of grid points in the lattice grid. This means that mn pixel copy attempts are run in this one iteration.

(b) One index copy attempt is implemented. To do so, a grid point within a cell, called the target pixel, is randomly chosen on the lattice grid.

(c) A neighbor of the selected grid point, called the source pixel, is randomly chosen.

(d) If the source pixel is part of the same cell as the target pixel, nothing happens in this Monte Carlo step. The simulation moves on to the next Monte Carlo step and repeats the previous two steps.

(e) If the source pixel is part of a different cell from the target pixel, the change in Hamiltonian energy is calculated. This includes the adhesion energies and the volume constraint terms.

(f) The index copy attempt is accepted based on the Boltzmann acceptance function detailed in Section 2.1. However, before the index copy is implemented, connectivity is checked. If the index copy disconnects the cell of the target pixel, the index copy attempt is rejected and the simulation moves on to the next Monte Carlo step.

(g) Upon the acceptance of the index copy attempt, the grid is updated and the volume and centers of mass of the two changed cells are updated. Mitosis is considered, and if the number of cells present is lower than the cell cap defined previously and the volume of the enlarged cell is larger than the mitotic volume defined previously, the cell undergoes mitosis.

    (h) These steps are repeated until the Monte Carlo step iteration is run. Further Monte Carlo step iterations are run until the end of the simulation based on the number of iterations defined by the simulation.

7. If there is coupling between the collagen simulation and the cellular Potts model, both models will be run.

    (a) One Monte Carlo step iteration is run in the cellular Potts model as elaborated in the previous step 6. The difference is that in this case, the Hamiltonian energy calculation in step 6e includes the motility term which draws upon the collagen simulation.

    (b) Individual Monte Carlo steps are run until the number of MCS reaches a multiple of f_{cs}, as defined in Section 2.3.1. At this point, the collagen simulation is updated. The target positions of the cells are adjusted based on the velocities from the collagen simulation and the collagen simulation steps forward by Δt_{cs}.

    (c) The previous step is repeated b_{cs} times. With the updated collagen simulation, Monte Carlo steps continue to be run until another multiple of f_{cs} is reached. Then the previous step is repeated.

    (d) The previous two steps are repeated until mn individual Monte Carlo steps are completed and the Monte Carlo step iteration is finished. Then the previous three steps are repeated and further Monte Carlo step iterations are run until the end of the simulation based on the number of iterations defined by the simulation.

3

Results

WITH THE COMPLEXITY OF OUR SIMULATION, tests to ensure that the coupled model is working are broken up into several parts. In the first part, we will focus on a very basic simulation that purely demonstrates that mitosis is indeed functional and working in our simulation. In the second part, we will run a simple simulation for an array of different adhesion energy parameters to demonstrate how certain characteristics of the cellular Potts model are affected by different adhesion energy parameters. Further parts will include simulations that do contain interactions between the cellular Potts model and the reference map technique to show that the coupling between the cellular Potts model and the collagen simulation is proceeding as planned.

3.1 Mitosis

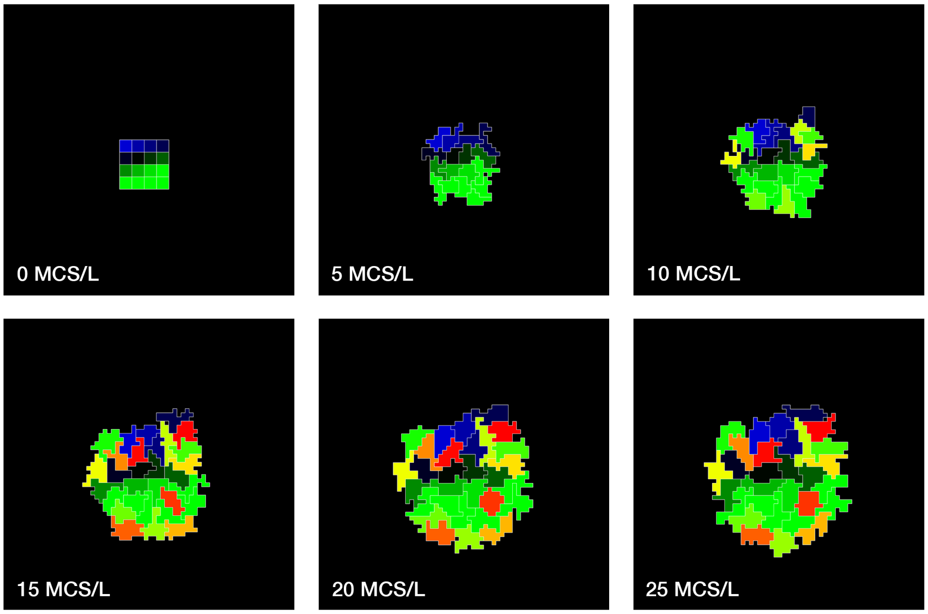


Figure 3.1: Simulation of 16 initial cells undergoing mitosis over 25 Monte Carlo step iterations. Here MCS/L means Monte Carlo step per lattice point, which indicates that 25 Monte Carlo steps were run for every lattice point in the grid.

We start with a very basic simulation to test for mitosis. We initialize our simulation as a group of 16 cells in a 4 by 4 square each with a radius of 0.1 physical grid units. The cells are centered on a lattice grid of size 6 by 6 physical grid units and 150 by 150 lattice points. The mitosis volume is set to 25 grid points, which is the volume a cell must reach in order to be considered for mitosis. The cell division cap is set to 30, thus allowing for 14 mitosis events from the initial 16 cells. The target volume for each cell is set to 30, indicating that every cell should be able to reach the mitotic volume. We set the cell-substrate adhesion constant to 5 and the cell-cell adhesion constant to 10, and the volume constraint constant to 15. We run the simulation to 50 Monte Carlo steps per grid point–this means that since one Monte Carlo step is one pixel copy attempt, we run one Monte Carlo step over every grid point, 50 times. Six frames from the simulation are shown in Figure 3.1.

From the figure, the 16 cells are initialized at a smaller size, and gradually grow over each of the Monte Carlo step iterations to the target volume specified of 30. As each cell grows toward 30, it reaches the mitosis volume of 25 grid points, at which point the cells are considered for mitosis. As the grid points are randomly chosen, the cells are all growing relatively evenly. Thus, as the cells approach mitotic volume, essentially all 16 of the cells reach the conditions for mitosis at the same time, thus leading to 14 of the cells dividing and reaching the cap of 30 cells at around the same time. Thus, we can see at 10 MCS/L that there are the 16 original cells at a larger volume around mitotic volume, and at 15 MCS/L, the cluster has reached the cap of 30 cells, with 14 new daughter cells from mitosis from the 16 cells. As expected, after reaching the cap of 30 cells, the cells in the cluster no longer divide, and simply grow and equilibrate at around the target volume of 30 grid points.

3.2 Adhesion Energy

In these simulations, we start with a similar setup as in the previous section. We initialize our simulation as a group of 16 cells in a 4 by 4 square each with a radius of 0.1 physical grid units. The cells are centered on a lattice grid of size 4 by 4 physical grid units and 100 by 100

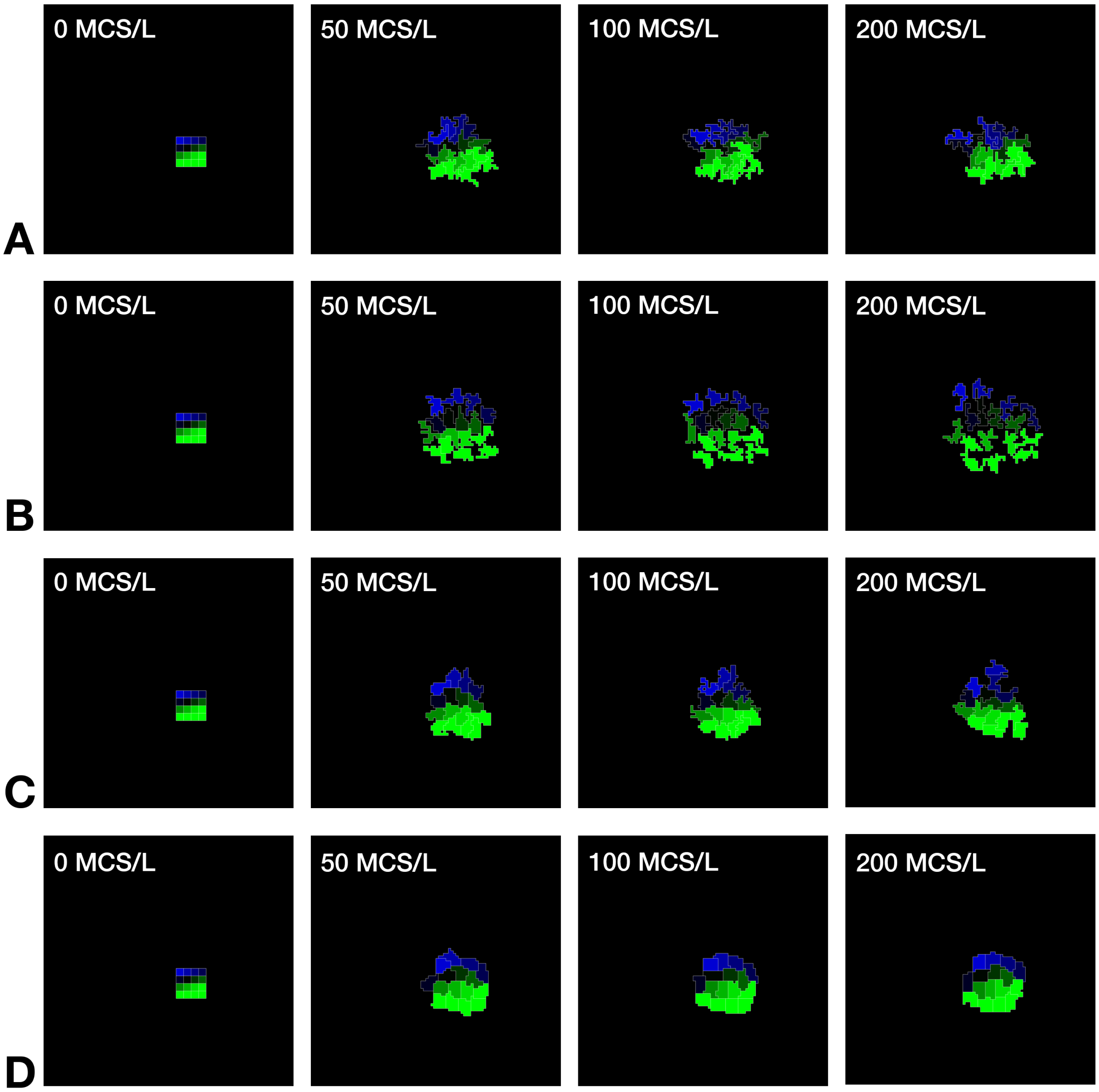


Figure 3.2: A simulation of 16 initial cells was run for a set of different cell-substrate adhesion and cell-cell adhesion constants. Snapshots from the simulations for four different cases are shown to demonstrate the different growth patterns for different adhesion energy constant combinations. The adhesion energy constant parameter combinations shown are as follows:

A: cell-substrate 1, cell-cell 1
B: cell-substrate 1, cell-cell 20
C: cell-substrate 5, cell-cell 10
D: cell-substrate 20, cell-cell 20

lattice points. The mitosis volume is set to 25 grid points, which is the volume a cell must reach in order to be considered for mitosis, but the cell division cap is set to 0, thus disabling mitosis. We do so in order to look purely at the interactions between the cells. The target volume for each cell is set to 30, and the volume constraint constant is set to 15. We iterate the cell-substrate adhesion constant and the cell-cell adhesion constant for a set of values, and run each simulation to 200 Monte Carlo step iterations. For all of these simulations, the motility terms are not included and the collagen simulation is not activated. Thus, we are purely looking at the cellular Potts model and how different adhesion energy constants can affect growth of the cell cluster. We can see a few representative time series of the cluster growth over time in figure 3.2.

We choose to test our simulations for parameters $\{1, 5, 10, 15, 20\}$ for both the adhesion constants. The temperature of our simulation is set to 10, which as defined in section 2.1 plays a role in determining the rate of successful index copy attempts through the Boltzmann acceptance function. Based on our temperature, it would be logical to test constants that are within the order of magnitude of the temperature, and testing for 1 to 20, which are close to but multiples of the temperature, should be able to adequately exhibit the range of behaviors for different energy combinations.

3.2.1 Cluster Characteristics

For each case, we calculate three different parameters from the resulting cluster at the end of the 200 Monte Carlo step iterations, for which a colormap comparing the parameters for the different adhesion energy constants are shown in figure 3.3. One of the parameters is the total edge length, calculated in terms of pixels or grid points from the cellular Potts model. This is

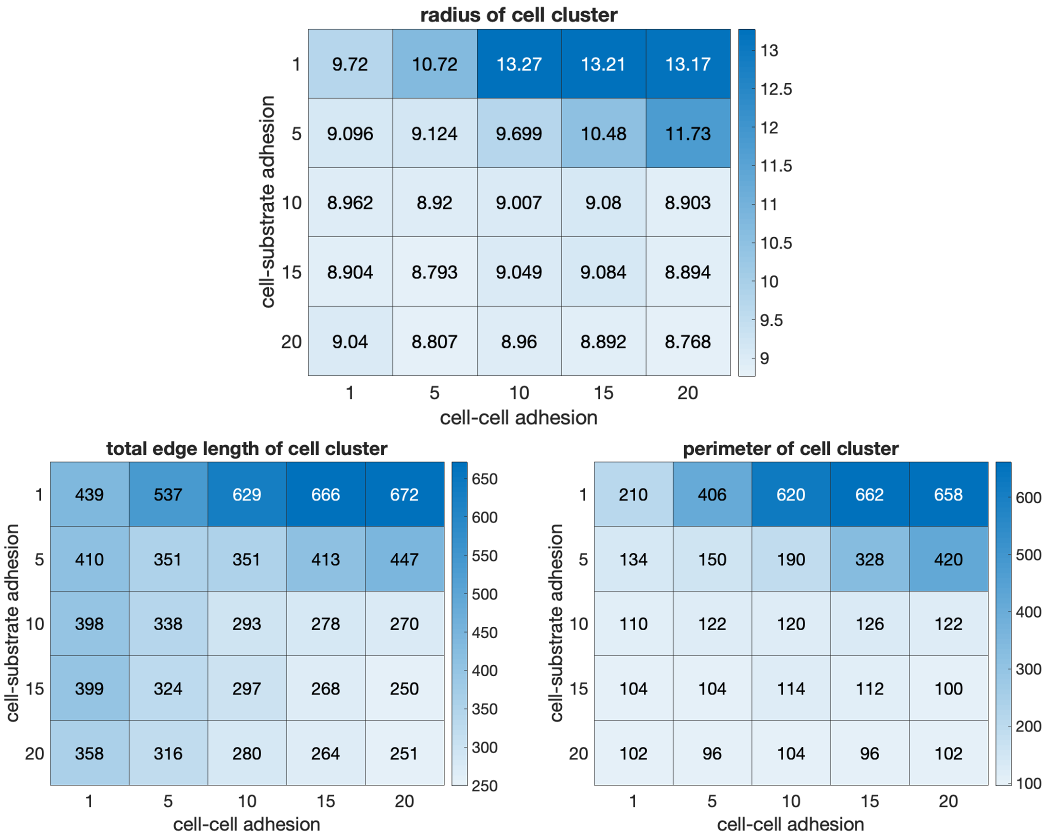


Figure 3.3: A simulation of 16 initial cells was run for a set of different cell-substrate adhesion and cell-cell adhesion constants. The radius of the cell cluster, the total edge length of the cell cluster, and the total perimeter of the cell cluster were calculated for each case.

computed by adding up the perimeters of each cell in the cluster, and provides a comparison to the overall perimeter of the cluster. The overall perimeter of the cluster is calculated by computing the edge lengths of cells in the cluster that have contact with the substrate and not with other cells. With the comparison, larger total edge lengths relative to the cluster perimeter are representative of cells being more fragmented and spread out in the cluster rather than remaining together in a cohesive group.

The radius of the cell cluster is calculated as a relative parameter. The center of mass

cm_i, cm_j was calculated by summing the coordinates of each pixel in the cluster and dividing by the total volume of the cluster:

$$(cm_i, cm_j) = \left(\frac{\sum\limits_{\sigma} \sum\limits_{i \in \sigma} i}{\sum\limits_{\sigma} v_\sigma}, \frac{\sum\limits_{\sigma} \sum\limits_{j \in \sigma} j}{\sum\limits_{\sigma} v_\sigma} \right)$$

We then calculate the effective radius by computing the variance $\sigma(x, x)$ in the x-direction and the variance $\sigma(y, y)$ in the y-direction, as well as the covariance:

$$\sigma(x, y) = \sum_{\sigma} \sum_{i,j \in \sigma} (i - cm_i)(j - cm_j)$$

where i, j are the coordinates of each pixel in the cell σ. We summarize the variance of the cluster in the form of the covariance matrix:

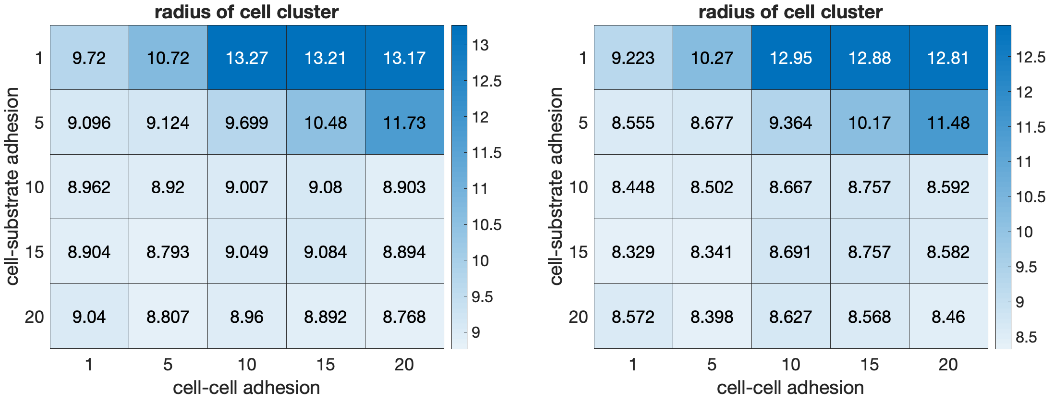
$$\Sigma = \begin{bmatrix} \sigma(x, x) & \sigma(x, y) \\ \sigma(y, x) & \sigma(y, y) \end{bmatrix}$$

Now, the covariance matrix defines both the spread and the orientation of the cell cluster. We then define the effective radius of our cluster as the square root of the trace of the covariance matrix, which is $\sqrt{\sigma(x, x) + \sigma(y, y)}$.

From the charts, we can see that there are very evident trends that are consistent over all three parameters. The cluster radius is significantly larger for lower cell-substrate adhesion energy constants and higher cell-cell adhesion energy constants. This makes sense, since the lower cell-substrate adhesion energy constraint encourages the cells to adhere with the substrate more and the higher cell-cell adhesion energy constraint discourages cell-cell adhesion, thus causing the cells to split apart from each other, thus increasing the effective radius and size of the cluster. This also causes the total edge length and the perimeter of the cluster to become more equivalent as the cells spread out and no longer share edges, thus increasing the perimeter of the cluster.

Otherwise, besides the cases with low cell-substrate adhesion energy constants and high cell-cell adhesion energy constants, the simulations appear to be quite similar. The radius of the clusters are all around 9, and the perimeter stays approximately constant for all of the cases. We can see that edge length shows more significant trends in that for higher cell-substrate adhesion constants, lower cell-cell adhesion constants shows higher total edge lengths. This is due to cells being more "fragmented" with more complex shapes rather than just remaining as simple square-ish blocks, since the low cell-cell adhesion encourages more contact between cells. Thus, with the more complex shapes of the cells, the perimeter of each cell would increase and thus the total edge length would increase correspondingly, even though the overall perimeter of the cluster remains consistent.

3.2.2 RADIUS CONSIDERATIONS



cell-substrate adhesion \ cell-cell adhesion	1	5	10	15	20
1	9.72	10.72	13.27	13.21	13.17
5	9.096	9.124	9.699	10.48	11.73
10	8.962	8.92	9.007	9.08	8.903
15	8.904	8.793	9.049	9.084	8.894
20	9.04	8.807	8.96	8.892	8.768

cell-substrate adhesion \ cell-cell adhesion	1	5	10	15	20
1	9.223	10.27	12.95	12.88	12.81
5	8.555	8.677	9.364	10.17	11.48
10	8.448	8.502	8.667	8.757	8.592
15	8.329	8.341	8.691	8.757	8.582
20	8.572	8.398	8.627	8.568	8.46

Figure 3.4: The radius of the cell cluster, calculated in two different ways. The first is done by looking separately at the pixels within the cluster. The second is done by looking at the centers of mass for the cells within the cluster.

We can calculate the effective radius of the cell cluster in two ways. The first is by individual grid point within the cluster, as delineated in the previous section. The second is by cell rather than individual grid point. Given the center of mass of the cell cluster calculated as before, we then calculate the effective radius by computing the variance $\sigma(x, x)$ in the x-direction and the variance $\sigma(y, y)$ in the y-direction, as well as the covariance:

$$\sigma(x, y) = \sum_{\sigma} (c_i - cm_i)(c_j - cm_j)$$

where c_i, c_j are the coordinates of the center of mass of the cell σ as calculated in section 2.1.1. We create our new covariance matrix:

$$\Sigma = \begin{bmatrix} \sigma(x, x) & \sigma(x, y) \\ \sigma(y, x) & \sigma(y, y) \end{bmatrix}$$

and calculate the new effective radius of our cluster as the square root of the trace of the covariance matrix, which is $\sqrt{\sigma(x, x) + \sigma(y, y)}$.

From the charts in figure 3.4, the radius calculations for the two methods are quite consistent, with the radii from the method using individual grid points being slightly larger than the radii using the center of masses of the cells. This makes sense because the method using cell centers of mass is essentially finding the radius of the cluster based on the centers of the cells, which would not include the radii of the cells themselves, while the radii calculated using the grid points incorporate the entire cluster.

However, in previous iterations of the simulation, the radii calculated by the two methods were not as consistent. For a few cases, the radii calculated using cell COMs would be significantly smaller than the raw radii calculated by grid points, occasionally by a multiple of 2 or more. Upon further investigation, this large discrepancy was caused by cell fragmentation, in which a cell would gradually split into two separate parts that would occasionally separate to opposite sides of the cluster, which would cause the calculated center of mass of the cell to be very close to the center of mass of the overall cluster, thus leading to very small effective cluster radii when looking only at the centers of the cells instead of individual grid points. The severity of the cell fragmentation problem led us to consider methods in order to prevent this, as in real life cells splitting up without undergoing mitosis would be discouraged.

We penalize cell fragmentation effectively in the form of a Hamiltonian term $\mathcal{H}_{\text{connect}}$. During each index copy attempt, we have the target pixel $\vec{i}$ that wants to be changed, and the source pixel $\vec{i'}$ that would be copied to the target pixel. We check the neighbors of the target pixel $\vec{i}$ to see if changing the cell at $\vec{i}$ would cause the original cell that $\vec{i}$ was in to fragment into two separate parts. Upon checking for connectivity of the original $\vec{i}$ cell, if the index copy attempt would cause the cell to split, the connectivity Hamiltonian term $\mathcal{H}_{\text{connect}}$ would essentially be ∞, and if the index copy attempt does not cause a split in the cell, the term $\mathcal{H}_{\text{connect}}$ would be 0. Thus, if there is a split, the index copy attempt would not succeed, and if there is no split, the index copy would proceed given all other conditions are met.

This addition to the index copy procedure has been very effective in removing unintended cell fragmentation from the simulations. In turn, the radii calculated from the cell clusters using both methods have remained very consistent.

3.3 Coupling with One Cluster

In these simulations, we start with a similar setup as in the previous section. We initialize our simulation as a group of 16 cells in a 4 by 4 square each with a radius of 0.1 physical grid units.

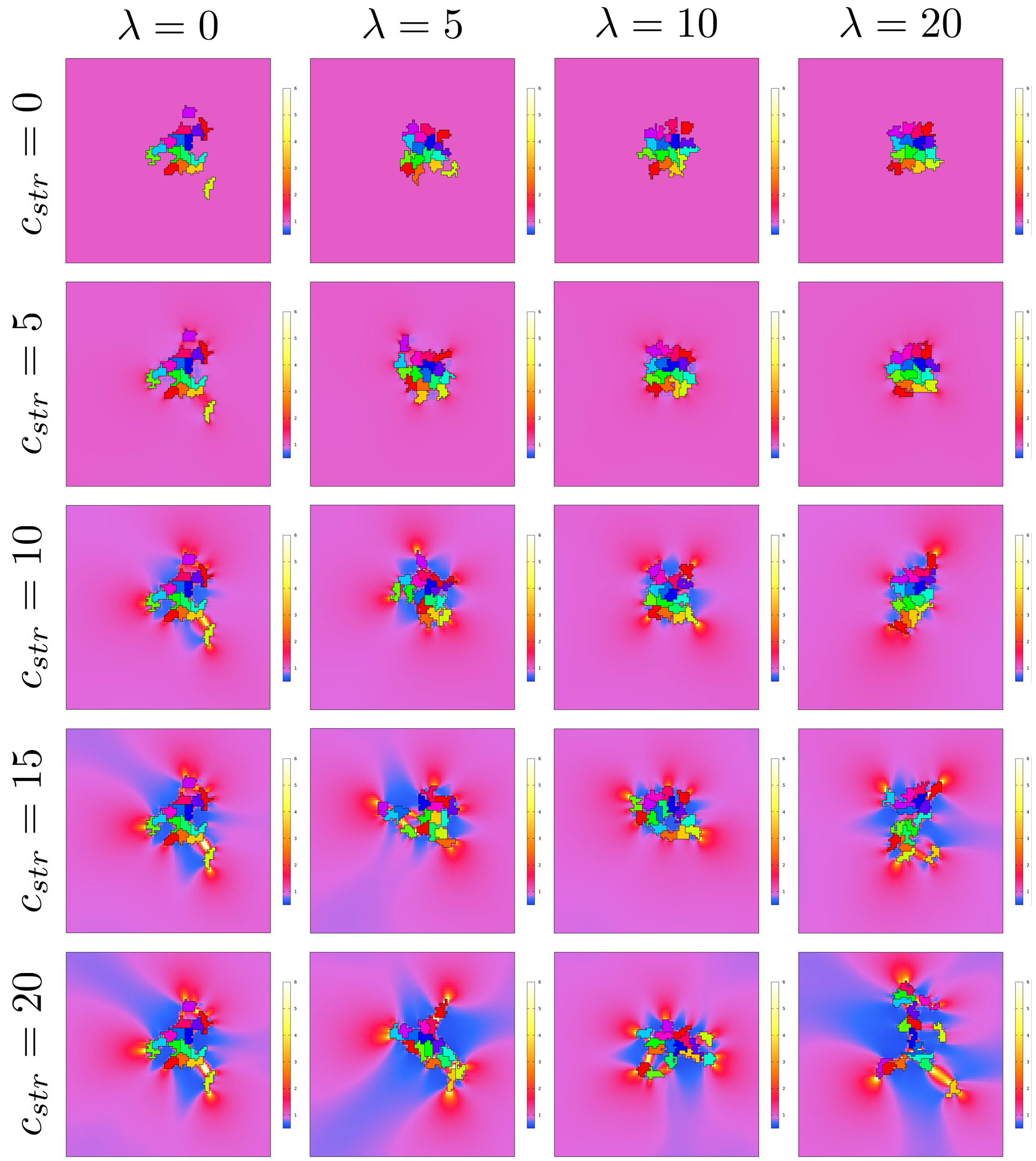


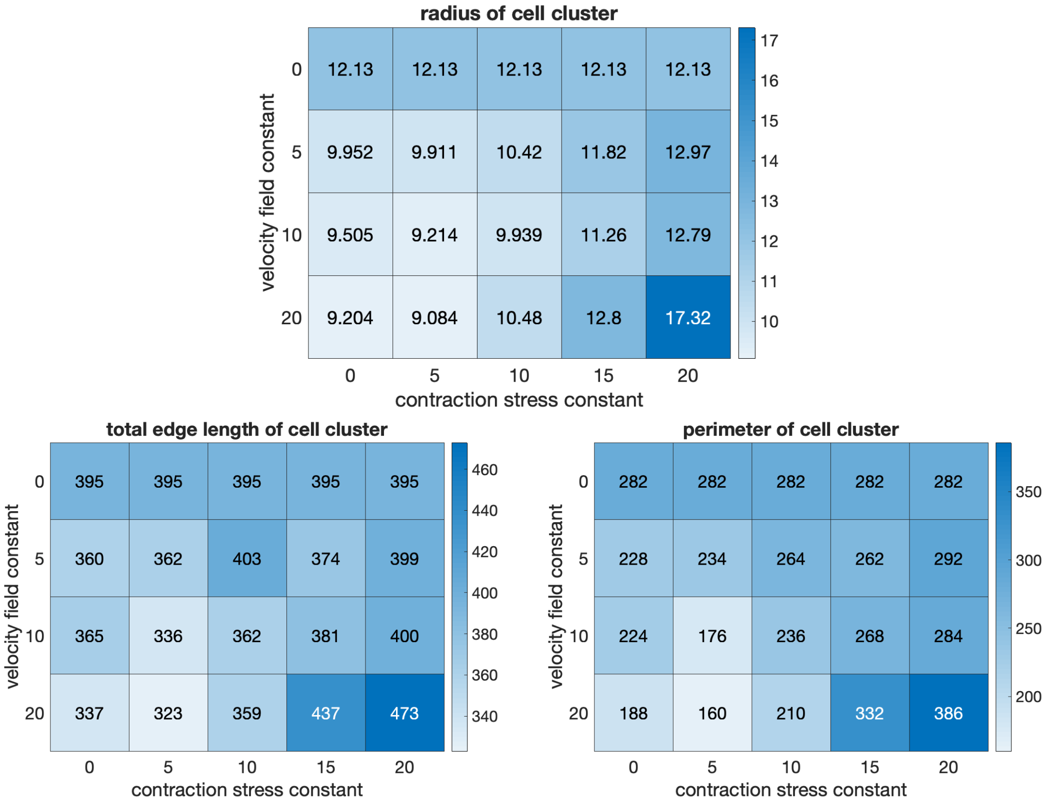
Figure 3.5: A simulation of 16 initial cells was run for a set of different motility term constants: the λ velocity field constant and the c_{str} contraction stress constant. The final simulation state for each of the different parameter cases is shown, including both the cells and the underlying collagen field, colored according to the colormap.

The cells are centered on a lattice grid of size 6 by 6 physical grid units and 150 by 150 lattice points. The mitosis volume is set to 25 grid points, which is the volume a cell must reach in order to be considered for mitosis, but the cell division cap is set to 0, thus disabling mitosis. The target volume for each cell is set to 30, and the volume constraint constant is set to 15. All of these parameters are the same as in the simulations run in section 3.2.

We set the cell-substrate adhesion constant to 5 and the cell-cell adhesion constant to 10 to keep the adhesion energies consistent between the simulations in this set. We iterate the λ and c_{str} constants–for the velocity field and contraction stress, respectively–over a set of values and run each simulation to 200 Monte Carlo step iterations, but with each iteration taking 5 Monte Carlo steps per grid point, thus equivalently doing $5mn \cdot 200$ individual Monte Carlo steps. We can see the final state of each of the simulations after 200 Monte Carlo step iterations in figure 3.5. The lattice grid and number of effective iterations has been increased. In previous tests, we found that the coupling required more individual Monte Carlo steps in order to surface significantly, so increasing the size of the lattice grid and the number of Monte Carlo steps to run per iteration would allow the mechanics effects to exhibit.

We choose to test our simulations for the λ velocity field constant over $\{0, 5, 10, 20\}$ and the c_{str} cell contraction stress constant over $\{0, 5, 10, 15, 20\}$. Again, the temperature of our simulation is set to 10, and we test constants that are within the order of magnitude of the temperature.

For each case, we calculate the same three different parameters from the resulting cluster at the end of the 200 Monte Carlo step iterations: radius, perimeter, and total edge length. The results are shown in figure 3.6. We also show a series of representative snapshots from one simulation for velocity field constant $\lambda = 10$ and cell contraction stress constant $c_{\text{str}} = 15$ in



radius of cell cluster

	0	5	10	15	20
0	12.13	12.13	12.13	12.13	12.13
5	9.952	9.911	10.42	11.82	12.97
10	9.505	9.214	9.939	11.26	12.79
20	9.204	9.084	10.48	12.8	17.32

total edge length of cell cluster

	0	5	10	15	20
0	395	395	395	395	395
5	360	362	403	374	399
10	365	336	362	381	400
20	337	323	359	437	473

perimeter of cell cluster

	0	5	10	15	20
0	282	282	282	282	282
5	228	234	264	262	292
10	224	176	236	268	284
20	188	160	210	332	386

Figure 3.6: A simulation of 16 initial cells was run for a set of different with different coupling velocity field and contraction stress parameters. The radius of the cell cluster, the total edge length of the cell cluster, and the total perimeter of the cell cluster were calculated for each case.

figure 3.7.

From figure 3.5 and the calculations in the charts in figure 3.6, we note that the cell clusters appear to be relatively similar in size across the array of parameters. They also seem to remain mainly cohesive and close together, not showing significant signs of movement until higher values of c_{str}. Despite this, we can note that any movement we see seems to follow the gradients shown in the underlying collagen, suggesting that the cells are interacting with the substrate and that the substrate is influencing the motion of the cells. The effect is particularly evident for high values of c_{str}, especially for $c_{\text{str}} = 20$, where we can see that the shape of the cell cluster

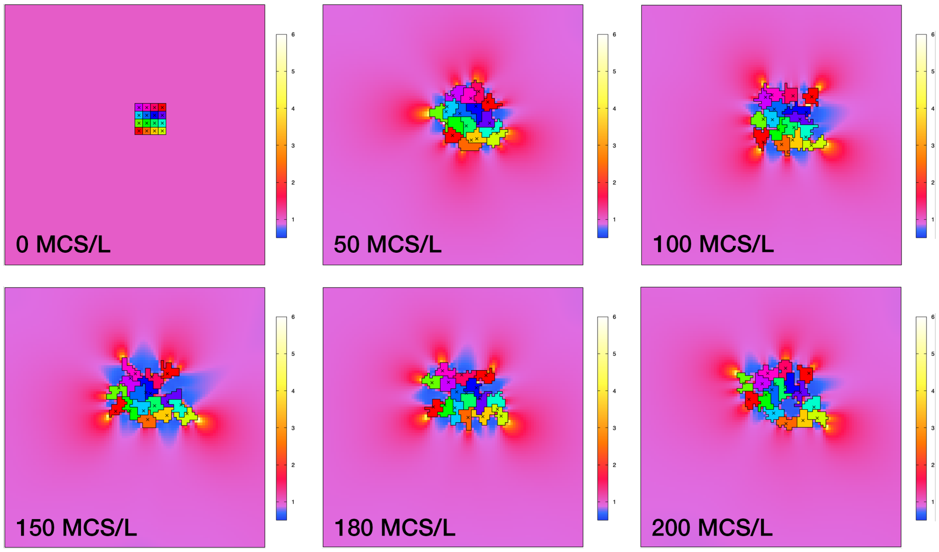


Figure 3.7: A simulation of 16 initial cells was run with velocity field constant $\lambda = 10$ and cell contraction stress constant $c_{\text{str}} = 15$. The simulation states for six different time points in the simulation are shown, including both the cells and the underlying collagen field as well as the target point for each of the cells as a cross.

is essentially molded along the stretches of the collagen. Of particular note is the simulation with $\lambda = 20$ and $c_{\text{str}} = 20$, which has a very diffuse cell cluster structure but all of the cells are adhering to the fluctuations of the collagen underneath. This demonstrates that the coupling between the cells and the underlying substrate is working well.

3.4 INTERACTIONS BETWEEN TWO CLUSTERS

In these simulations, we check to see how two separate cell clusters would interact, influenced by the underlying substrate. We initialize our simulation as two separate clusters of 16 cells, each in a 4 by 4 square with a cell radius of 0.1 physical grid units. As before, the clusters are located on a lattice grid of size 6 by 6 physical grid units and 150 by 150 lattice points. The target volume for each cell is set to 30, and the volume constraint constant is set to 15.

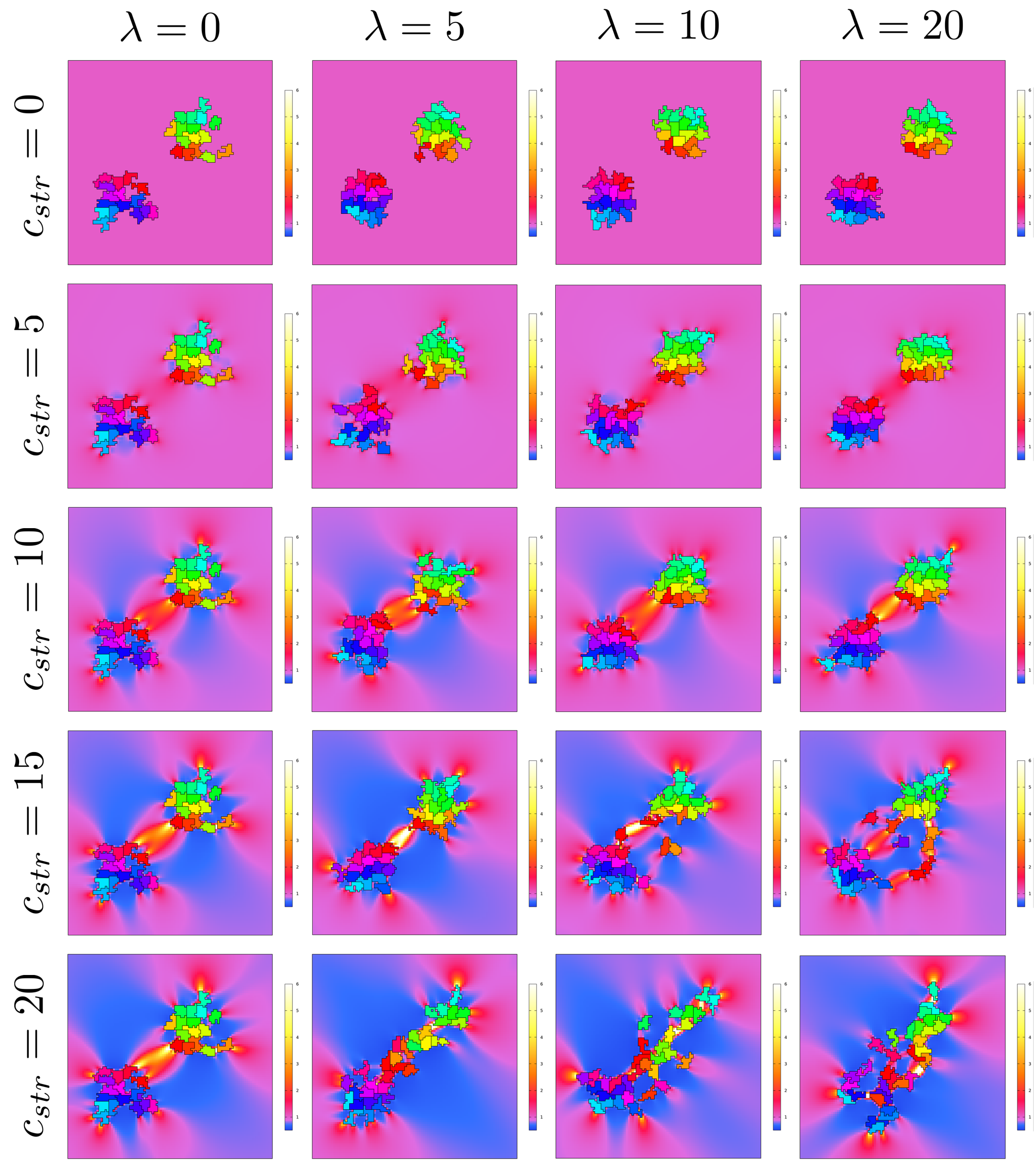


Figure 3.8: A simulation of 2 initial cell clusters each with 16 cells was run for a set of different motility term constants: the λ velocity field constant and the c_{str} contraction stress constant. The final simulation state for each of the different parameter cases is shown, including both the cells and the underlying collagen field, colored according to the colormap.

Just as in section 3.3, we set the cell-substrate adhesion constant to 5 and the cell-cell adhesion constant to 10 to keep the adhesion energies consistent between the simulations in this set. We again iterate over the λ and c_{str} constants and run each simulation to 200 Monte Carlo step iterations, but with each iteration taking 5 Monte Carlo steps per grid point. We choose to test our simulations for the λ velocity field constant over $\{0, 5, 10, 20\}$ and the c_{str} cell contraction stress constant over $\{0, 5, 10, 15, 20\}$. These values are all the same as in the previous simulation with one cell cluster. We can see the final state of each of the simulations after 200 Monte Carlo step iterations in figure 3.8.

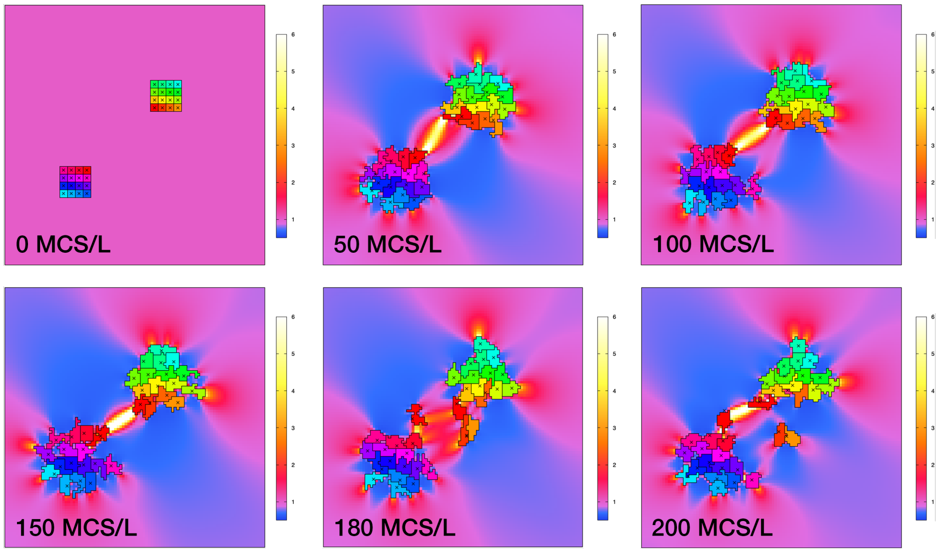


Figure 3.9: A simulation of 2 initial cell clusters each with 16 cells was run with velocity field constant $\lambda = 10$ and cell contraction stress constant $c_{\text{str}} = 15$. The simulation states for six different time points in the simulation are shown, including both the cells and the underlying collagen field as well as the target point for each of the cells as a cross.

From figure 3.8, we can see much greater variation in cell movement for the different sets of parameters. As expected, when there is low cell contraction stress for feedback from the cells upon the collagen, the cells remain relatively motionless based on the adhesion energies provided. The same is true for low velocity field constants, which decreases the influence of the collagen on moving the cells. We note that with high cell contraction stress but no velocity field constant, the cells all finalize in the same configuration despite very different underlying collagen stress values, as seen in the colored background.

When both λ and c_{str} are large enough, we are able to see interesting interactions between the cell clusters. Bands form between the cell clusters, and with enough force, the cells in the two clusters move along the bands and interact with the other cluster. The effect is particularly strong when $c_{\text{str}} = 20$, with the two clusters essentially merging into one continuous cluster along the collagen band that forms.

Six snapshots from a simulation with velocity field constant $\lambda = 10$ and cell contraction stress constant $c_{\text{str}} = 15$ are shown in figure 3.9. Compared to the snapshots in figure 3.7, which was run for the same parameters, we can see that there is much more movement.

3.4.1 Effect of Adhesion Energy on Coupling

While both the velocity field constant and the cell contraction stress constant are vital to determining the cell interactions with the substrate, adhesion energy logically should also play a role in cell movement. In these simulations, we check to see how two separate cell clusters would interact based on different cell-substrate adhesion energy constants. As in the previous simulation, we initialize our simulation as two separate clusters of 16 cells, each in a 4 by 4 square with a cell radius of 0.1 physical grid units. As before, the clusters are located on a lattice grid of size 6 by 6 physical grid units and 150 by 150 lattice points. The target volume for each cell is set to 30, and the volume constraint constant is set to 15.

We set the cell-cell adhesion energy constant to 10 and the contraction stress constant c_{str}

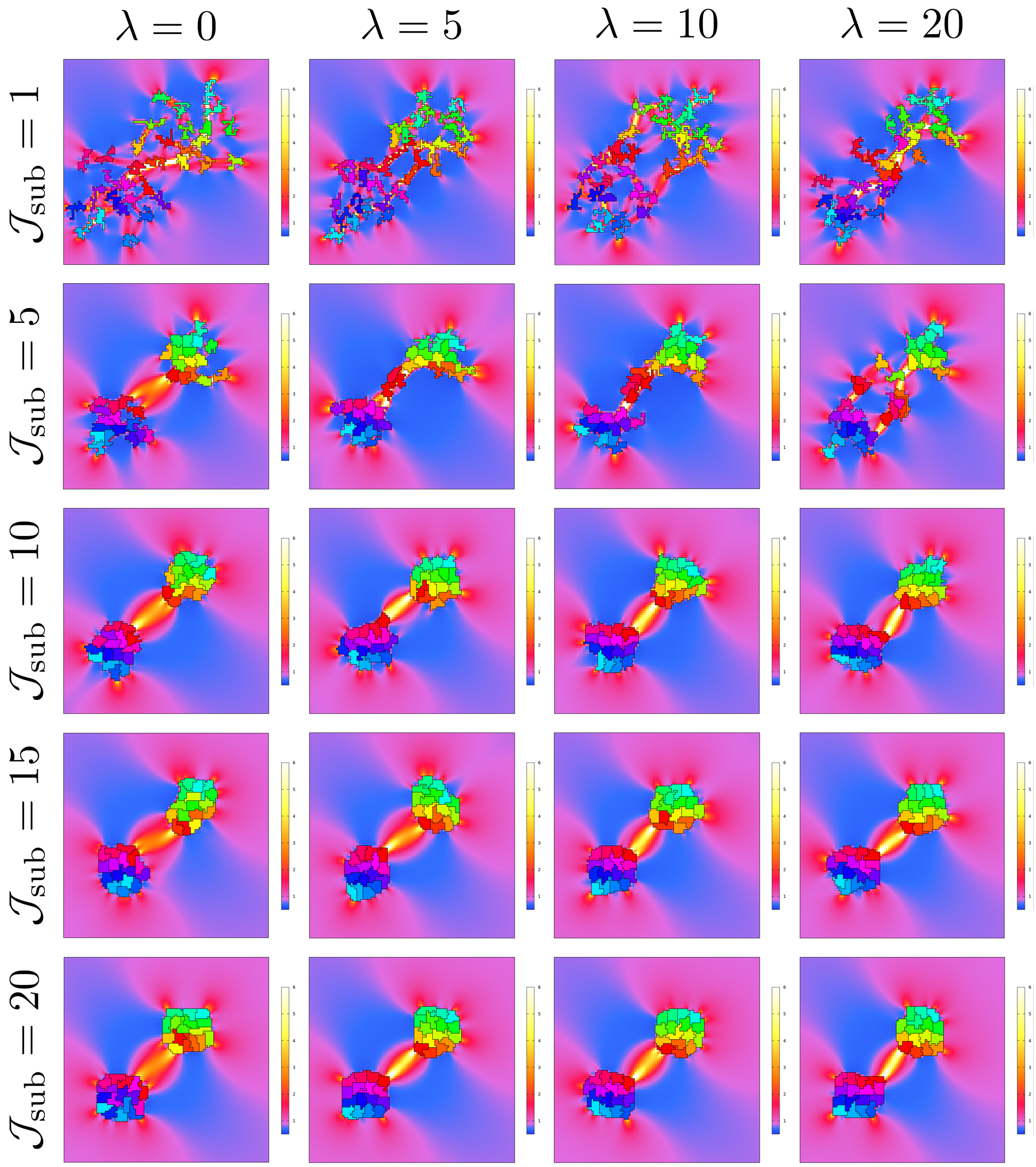


Figure 3.10: A simulation of 2 initial cell clusters each with 16 cells was run for a set of different motility term constants: the λ velocity field constant and the $\mathcal{J}_{\text{sub}}$ cell-substrate adhesion energy constant. The final simulation state for each of the different parameter cases is shown, including both the cells and the underlying collagen field, colored according to the colormap.

to 15. We then iterate over the velocity field constant λ and cell-substrate adhesion energy constant $\mathcal{J}_{\text{sub}}$ and again run each simulation to 200 Monte Carlo step iterations, but with each iteration taking 5 Monte Carlo steps per grid point. We choose to test our simulations for λ over $\{0, 5, 10, 20\}$ and $\mathcal{J}_{\text{sub}}$ over $\{1, 5, 10, 15, 20\}$. We can see the final state of each of the simulations after 200 Monte Carlo step iterations in figure 3.10, and six snapshots from a simulation with velocity field constant $\lambda = 10$ and cell-substrate adhesion energy constant $\mathcal{J}_{\text{sub}} = 5$ are shown in figure 3.11.

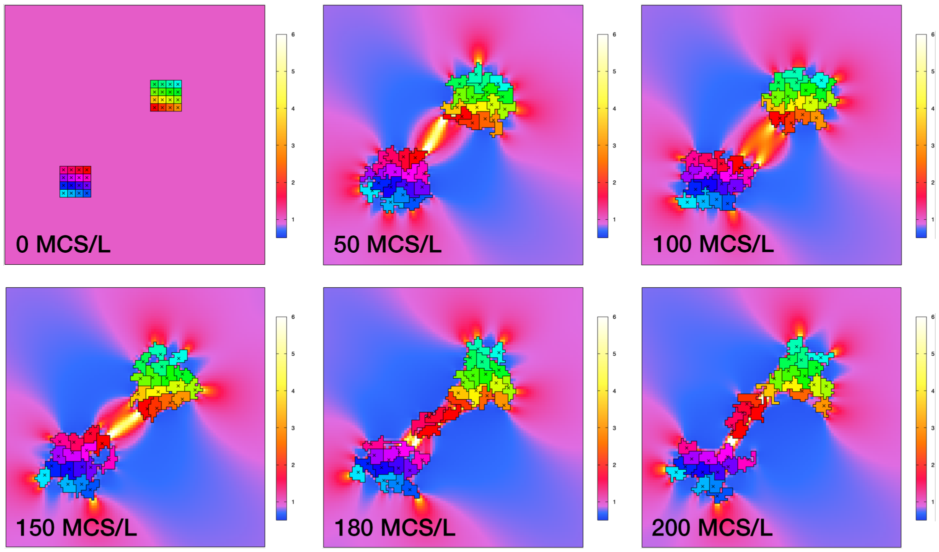


Figure 3.11: A simulation of 2 initial cell clusters each with 16 cells was run with velocity field constant $\lambda = 10$ and cell-substrate adhesion energy $\mathcal{J}_{\text{sub}} = 5$. The simulation states for six different time points in the simulation are shown, including both the cells and the underlying collagen field as well as the target point for each of the cells as a cross.

From figure 3.10, we can see that as expected, adhesion energy, particularly cell-substrate adhesion energy, plays a significant role in determining the extent of cellular movement. High cell-substrate adhesion energy discourages cells from separating and interacting more closely with the substrate, thus discouraging cell disorganization that would have been caused by the velocity field and collagen. This remains true regardless of the strength of the influence of the velocity field on the cells: as we can see, as λ increases from 0 to 20, for high values of $\mathcal{J}_{\text{sub}}$, the clusters still remain very cohesive and largely motionless.

The opposite extreme is true when the cell-substrate adhesion energy is low. At almost no adhesion energy, all of the cells are very dispersed and the cells mold almost exactly to the underlying collagen. Overall, mechanics seems to be best represented by a cell-substrate adhesion energy of 5, which allows for clusters that remain mostly together while also following the underlying collagen closely. In this case, we can notice that the extent of cellular disorganization increases with higher values of λ, which is what we would expect.

4

Conclusions

In this thesis, we seek to develop a method to model cells while incorporating complex mechanics in a manner that does not require excessive amounts of resources and that remains versatile for many different applications. We do so by coupling the cellular Potts model with the reference map technique, allowing the collagen from the reference map technique to interact with the cells in the cellular Potts model to implement physical mechanics. With our model, we are able to replicate representative simulations of cells growing on a complex collagen material and have successfully demonstrated how to couple mechanics to the cellular Potts model.

4.1 POTENTIAL FOR FLEXIBLE MODELING OF MECHANICS

The computational significance of this work lies in the original use of a simple lattice-based agent-based model in combination with a nonlinear continuum modeling technique. Various studies have investigated methods to model nonlinear materials like in the reference map technique, or to use agent-based methods to study various intercellular phenomena, whether with the cellular Potts model or otherwise. However, the methods that are able to model the complex mechanics of nonlinear biomaterials typically do not scale well for larger systems and larger numbers of cells, and those that are able to approximate nonlinear mechanics through simpler methods, such as looking at entire clusters as whole, are typically unable to capture complexities of individual cells[24].

On the other end, intercellular phenomena have been studied extensively using variations of agent-based modeling. Even just looking at the cellular Potts model specifically, the model has proved to be extremely versatile in modeling various biological phenomena and systems, on both the cellular and the tissue level. The versatility of this model did not preclude the model from the innate limitations of a purely lattice-based model. The limitations of a rigid and simple framework are especially apparent in attempts to characterize interactions with a complex and dynamic environment. Various studies have attempted to extend the cellular Potts model to incorporate cellular interactions with a complex substrate, often through chemical signalling or external forces, but while the results often do exhibit some characteristics of cellular interactions in complex environments, the similarities are complicated by nonlinear changes and more intensive interactions between the cells and the substrate.

In this thesis, we develop a generalizable framework for coupling complex physical mechanics with multicellular simulations by utilizing the reference map technique in conjunction with a multicellular simulation in the form of the cellular Potts model. We show that the coupling of the two models allows the cells to act upon the underlying gel and for the gel to influence the cells' movement. The fact that both models are based on a regular grid structure maintains the versatility of the cellular Potts model, allowing it to be used to model other equations and functionality, such as chemical signalling or diffusion simultaneously with the complex mechanics from the underlying substrate.

Of particular note is that a previous study sought to model mechanical interactions specifically between cancerous mammary acini and utilized the reference map technique to successfully demonstrate the complexity of geometries and configurations in the collagen gel. However, despite the successful replications of experimental data, the model suffered from crystallization issues where cell clusters (mammary acini in this case) would form clear hexagonal structures between adjacent cells, which were unrealistic and also heavily affected the model[27]. Our model coupling the cellular Potts model with the reference map technique was able to completely eliminate all of these packing effects, thus avoiding unintended spontaneous structure formation that could affect the interactions between the cells or between the cells and the substrate.

4.2 INSIGHTS ON FACTORS THAT INFLUENCE CELLULAR INTERACTIONS

The versatility of the model, especially due to being based upon the cellular Potts model, allows us to study how the movement of the cells change depending on a combination of different factors.

Crucially, we find that when comparing the simulations with only one cell cluster with those with two cell clusters, the cells in the two cluster systems go through more destabilization and disorganization. As we can see from figure 3.8, in the simulations with two clusters, the cells would move between clusters and generally be more likely to separate from their original cluster. This is representative of what occurs biologically. Shi et. al experimentally showed that when two or more contractile acini are sufficiently close together, they can interact via collagen bands that form between them, and disorganization of interacting acini is more extensive than that of noninteracting acini[19]. Just as we can see in figure 3.9, the two cell clusters are sufficiently close together to see a collagen band forming in the underlying substrate, and the substrate induces movement in the cells to interact between the clusters. In comparison, in figure 3.7, only one cell cluster is implemented in the simulation, and thus while it does seem to move slightly according to the collagen, no large bands are formed and thus cell disorganization is much smaller. Thus, we are able to replicate the significant differences in cell movement and disorganization that are caused by increasing number of cell clusters and distance between them.

Several parameters tested in our simulations included the velocity field constant, the strength of the cell contraction stress, as well as the adhesion energy constants. We found all three factors to have significant influences on the movement of the cells. When the velocity field constant λ is zero, the cells will not move according to the underlying collagen, regardless of the strength of the cell contraction. When the cell contraction is low, the cells do not pull upon the collagen and there is little movement. When cell-substrate adhesion energy is high and cells are penalized for interacting closely with the substrate, regardless of the strength of the velocity field constant or the cell contraction, the clusters will stay very cohesive and essentially motionless. However, with appropriate parameters, the simulation exhibits behavior representative of what we would expect, as well as from experimental data[19].

4.3 Future Directions

Our model has allowed us to understand how changing various parameters and characteristics of the simulation or how cells are connected with each other influences how the simulation destabilizes. Different features of cell clusters can greatly influence the further development of the simulation.

In our study, we did not implement precise parameter matching, instead working toward a proof of principle that this framework can successfully implement our desired type of modeling. We make use of non-dimensionalized quantities for the parameters and results we are presenting. However, we do have access to rheometer data on collagen gel mechanics and how the strain and stress fields of the collagen gel evolves over time. This quantitative data on the underlying collagen substrate would allow us to precisely calibrate the material parameters and assign specific units and numbers to the physical units of the collagen simulation, which would then translate to the cellular Potts model. Our framework can be applied to a variety of systems through the use of appropriate length, time, and mass scales.

We consider the problem in this thesis only in two dimensions, while many experimental models such as the study by Shi et al. involve somewhat three-dimensional cells [19]. The cellular Potts model is commonly used to model three-dimensional systems and thus can be readily extended to a 3D cell culture, and the general framework for combining with the reference map technique remains the same in principle in 3D. Further studies could involve how 3D cells would behave with cell interactions from three dimensions, and further consideration would need to be taken as to how to implement collagen forces in the form of sinks in 3D.

Finally, one of the greatest benefits of the cellular Potts model is the ease with which it can be implemented and used to model a variety of different systems. Of note is the use of CompuCell3D, a three-dimensional software problem solving environment that is conducive for experimentation and testing of biological models using the cellular Potts model. In the preliminary stages of this study, we attempted to implement mechanics through the CompuCell3D framework, but the software proved too limited for the extensive coupling that we needed to produce. Now that we have demonstrated the feasibility of adding physical mechanics through coupling the cellular Potts model and the reference map technique, these basic principles can be incorporated into any cellular Potts model framework, such as CompuCell3D. Doing so would allow for better access and usage of this framework in further simulations that could involve more realistic mechanics. We hope that our proposed framework and model is a useful tool in understanding and modeling mechanical interactions that occur between cells and complex underlying substrates, a concept that is crucial in many biological systems.

References

[1] Alber, M., Chen, N., Lushnikov, P. M., & Newman, S. A. (2007). Continuous macroscopic limit of a discrete stochastic model for interaction of living cells. *Phys. Rev. Lett.*, 99, 168102.

[2] Andasari, V., Roper, R. T., Swat, M. H., & Chaplain, M. A. J. (2012). Integrating intracellular dynamics using compucell3d and bionetsolver: Applications to multiscale modelling of cancer cell growth and invasion. *PLOS ONE*, 7(3), 1–17.

[3] Anderson, A., Rejniak, K., Gerlee, P., & Quaranta, V. (2009). Microenvironment driven invasion: a multiscale multimodel investigation. *Journal of Mathematical Biology*, 58(4), 579–624.

[4] Anderson, A. R., Weaver, A. M., Cummings, P. T., & Quaranta, V. (2006). Tumor morphology and phenotypic evolution driven by selective pressure from the microenvironment. *Cell*, 127(5), 905–915.

[5] Dormann, S. & Deutsch, A. (2002). Modeling of self-organized avascular tumor growth with cellular automata. 2, 35.

[6] Engler, A. J., Sen, S., Sweeney, H. L., & Discher, D. E. (2006). Matrix elasticity directs stem cell lineage specification. *Cell*, 126(4), 677–689.

[7] Hanahan, D. & Weinberg, R. (2011). Hallmarks of cancer: The next generation. *Cell*, 144(5), 646–674.

[8] Hanahan, D. & Weinberg, R. A. (2000). The hallmarks of cancer. *Cell*, 100(1), 57–70.

[9] Hawkins-Daarud, A., Rockne, R. C., Anderson, A. R. A., & Swanson, K. R. (2013). Modeling tumor-associated edema in gliomas during anti-angiogenic therapy and its impact on imageable tumor. *Frontiers in Oncology*, 3.

[10] Kabla, A. J. (2012). Collective cell migration: leadership, invasion and segregation. *Journal of The Royal Society Interface*, 9(77), 3268–3278.

[11] Kamrin, K., Rycroft, C. H., & Nave, J.-C. (2012). Reference map technique for finite-strain elasticity and fluid–solid interaction. *Journal of the Mechanics and Physics of Solids*, 60(11), 1952 – 1969.

[12] Paszek, M. J., Zahir, N., Johnson, K. R., Lakins, J. N., Rozenberg, G. I., Gefen, A., Reinhart-King, C. A., Margulies, S. S., Dembo, M., Boettiger, D., & et al. (2005). Tensional homeostasis and the malignant phenotype. *Cancer Cell*, 8(3), 241–254.

[13] Peirce, S. M., Van Gieson, E. J., & Skalak, T. C. (2004). Multicellular simulation predicts microvascular patterning and in silico tissue assembly. *The FASEB Journal*.

[14] Ramis-Conde, I., Chaplain, M. A., & Anderson, A. R. (2008). Mathematical modelling of cancer cell invasion of tissue. *Mathematical and Computer Modelling*, 47(5-6), 533–545.

[15] Rejniak, K. A. & Anderson, A. R. A. (2008). A computational study of the development of epithelial acini: I. sufficient conditions for the formation of a hollow structure. *Bulletin of Mathematical Biology*, 70(3), 677–712.

[16] Rubenstein, B. M. & Kaufman, L. J. (2008). The role of extracellular matrix in glioma invasion: A cellular potts model approach. *Biophysical Journal*, 95(12), 5661–5680.

[17] Sandersius, S. A. & Newman, T. J. (2008). Modeling cell rheology with the subcellular element model. *Physical Biology*, 5(1), 015002.

[18] Scianna, M. & Preziosi, L. (2012). Multiscale developments of the cellular potts model. *Multiscale Modeling & Simulation*, 10(2), 342–382.

[19] Shi, Q., Ghosh, R. P., Engelke, H., Rycroft, C. H., Cassereau, L., Sethian, J. A., Weaver, V. M., & Liphardt, J. T. (2013). Rapid disorganization of mechanically interacting systems of mammary acini. *Proceedings of the National Academy of Sciences*, 111(2), 658–663.

[20] Shirinifard, A., Gens, J. S., Zaitlen, B. L., Popławski, N. J., Swat, M., & Glazier, J. A. (2009). 3d multi-cell simulation of tumor growth and angiogenesis. *PLOS ONE*, 4(10), 1–11.

[21] Shu, C.-W. & Osher, S. (1988). Efficient implementation of essentially non-oscillatory shock-capturing schemes. *Journal of Computational Physics*, 77(2), 439 – 471.

[22] Storm, C., Pastore, J. J., MacKintosh, F. C., Lubensky, T. C., & Janmey, P. A. (2005). Nonlinear elasticity in biological gels. *Nature*, 435, 191.

[23] Swat, M. H., Thomas, G. L., Belmonte, J. M., Shirinifard, A., Hmeljak, D., & Glazier, J. A. (2012). Multi-scale modeling of tissues using compucell3d. 110(Supplement C), 325 – 366.

[24] Vader, D., Kabla, A., Weitz, D., & Mahadevan, L. (2009). Strain-induced alignment in collagen gels. *PLoS ONE*, 4(6), e5902.

[25] Valkov, B., Rycroft, C. H., & Kamrin, K. (2015). Eulerian method for multiphase interactions of soft solid bodies in fluids. *Journal of Applied Mechanics*, 82(4), 041011–041011–14.

[26] Voss-Böhme, A. (2012). Multi-scale modeling in morphogenesis: A critical analysis of the cellular potts model. *PLOS ONE*, 7(9), 1–14.

[27] Wang, J. B. (2015). Modelling Mechanical Interactions Between Cancerous Mammary Acini. Master's thesis, Harvard University.

[28] Zhang, Y., Thomas, G. L., Swat, M., Shirinifard, A., & Glazier, J. A. (2011). Computer simulations of cell sorting due to differential adhesion. *PLOS ONE*, 6(10), 1–11.