COMPUTATIONAL MODELING
OF METASTATIC CANCER MIGRATION
THROUGH A REMODELING EXTRACELLULAR MATRIX

By

YEN T. NGUYEN

Bachelor of Science in Chemical Engineering
Oklahoma State University
Stillwater, Oklahoma
2017

Submitted to the Faculty of the
Graduate College of
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF SCIENCE
July, 2018

COMPUTATIONAL MODELING

OF METASTATIC CANCER MIGRATION

THROUGH A REMODELING EXTRACELLULAR MATRIX

Thesis Approved:

Dr. Ashlee N. Ford Versypt

Thesis Advisor

Dr. Jindal K. Shah

Dr. Yu Feng

ACKNOWLEDGEMENTS

My deepest gratitude goes to my advisor, Dr. Ashlee Ford Versypt. It's impossible to express how lucky I feel to be one of her students. I have a deep respect for her wholehearted devotion to students learning and development. Having her introduced me to research during my undergraduate study, the experience led me to explore what excites me and has truly inspired my ambition of pursuing my graduate studies and the dream of becoming an academic one day. Thanks to her I had the opportunity to attend and present at many local and national conferences, to meet new people and grow my professional skills. I am grateful for her dedicated guidance and vital support, academically and emotionally, every step of the way throughout my journey at Oklahoma State University (OSU).

I am also immensely grateful to Dr. Jindal Shah and Dr. Yu Feng for their expert advice and assistance while serving on my thesis committee.

There are many fathomless thanks I want to send to the faculty and staff of the School of Chemical Engineering at OSU whose services have turned my academic journey here a success. The endless support and resources offered by the department proved to be a milestone in the accomplishment of my goals at OSU. I feel honored and proud to represent such a productive and high-performing department. I could not have asked for a better environment to thrive in the last four years of my undergraduate and master study. I also wish to present my special thank to Dr. Rob Whiteley, our department head, for his

words of wisdom that motivated me to embrace the power of assertion and confidence which will stick with me for life.

A gigantic thank is due to all my supportive teammates from Ford Versypt Lab and colleagues at OSU for their constant encouragement, offering inputs and feedbacks that made positive impacts on my research. In particular, I would like to send my sincere gratitude to Minu Pilvankar and Steve Ruggiero for their enthusiasm and willingness to help me in so many ways from my very first day joining the group. Their effort and accomplishments set a good example for me to learn from and strive harder in my work. Also, thank you for all the productive working-dates with coffee and sweet treats!

None of this would have been possible without the tremendous support from my family and closest friends. To my dearest Mother, I can never thank you enough for your endless love and encouragement to make me become the best version of myself in life. I am grateful to my Father, my siblings, Phat, Peter and Lucy, and my best friend, Trang, who have provided me with moral and emotional support. I am also grateful to my kindest and loving in-laws who have supported me along the way.

Lastly, thanks to my selfless and sweet husband, Steven, for providing me with unfailing support and continuous encouragement throughout my years of study, for taking me to countless coffee shops to write this thesis in my best state of mind, for driving me almost halfway across the country to defend my thesis, and, mostly, for always believing in me and respect my career decision!

Name: YEN T. NGUYEN

Date of Degree: July, 2018

Title of Study: COMPUTATIONAL MODELING OF METASTATIC CANCER MIGRATION THROUGH A REMODELING EXTRACELLULAR MATRIX

Major Field: CHEMICAL ENGINEERING

Abstract: The spreading of cancer cells, also known as metastasis, remains a lethal and unstoppable aspect of cancer treatment. Several cancer studies have suggested the remodeling of collagen fibers in the extracellular matrix (ECM) facilitates the migration of cancer cells during metastasis. ECM remodeling refers to the following activities: the ECM degradation caused by the enzyme matrix metalloproteinases (MMPs) and the ECM alignment due to the cross-linking enzyme lysyl oxidase (LOX). Such modifications of the collagen fibers induce changes in physical and biomechanical properties of the ECM that affect cancer cell migration through the ECM. However, the underlying mechanism of how these changes will give way favorably for the directional motility of cancer cells through the pool of collagen fibers in the ECM remains an open question. In this thesis, we employed the art of multiscale modeling of cancer to gain more insight into the complex interplay between metastatic cancer cells and the ECM while it undergoes remodeling. Two *in silico* models following different modeling approaches are proposed in this work. The first model is developed via the continuum modeling approach. The mathematical model is a system of five coupled partial differential equations (PDEs). The second model is built via the open-source software CompuCell3D upon the insight and framework gained from the continuous model. Modeling method applied in CompuCell3D is a composite of discrete and continuum modeling approach in which cells are treated as discrete while other components such as the ECM and chemicals are described through continuum fields. Both models include the effect of LOX, an enzyme that has not been included in any previous cancer invasion models. LOX are considered to enable transport of enzymes and migration of cells through a dynamic, reaction tumor microenvironment that is modulated during cell migration. Our models thus far have extended other existing relevant models with improved features showing the influential role of LOX as well as MMPs on the remodeling of ECM and metastatic cancer migration.

Contents

List of Tables

Table Page

List of Figures

Chapter 1

Introduction

1.1 Cancer Metastasis in the Early Stage

Every year there are approximately 8.2 million cancer-related deaths worldwide. Metastasis is the primary cause of cancer death. Cancer metastasis occurs when the disease reaches its lethal stage via the uncontrolled spreading of cancer cells to invade a nearby connective tissue and other key organs in the human body. The metastasizing primary tumor cells are not the only agents that drive the progression of metastasis. Instead, metastasis is a systematic process that involves the interaction of cancer cells among a community of various biochemical and cellular factors localized in the tumor microenvironment at both the primary and secondary tumor sites [24]. Many recent cancer-related studies have pointed out that the primary tumor microenvironment contains many important factors that determine whether the primary tumor progresses and proceeds to metastasize or remains dormant, staying a benign tumor [54, 6, 41, 87, 42]. In the early stage of metastatic cancer invasion, cancer cells migration first takes place by breaking away from the primary tumor site and breaching the basement membrane of the tumor. This thin barrier underlying the tumor mass is made up of mostly type IV collagen fibers that separate the tumor from the extracellular matrix (ECM) [48]. After perforating the basement membrane, the escaped cancer cells then must invade the ECM and travel through a meshwork of collagen fibers prior to intravasation into a blood vessel [20]. While maneuvering through the ECM, the direction and behavior of the migrating cells are greatly influenced by the physical and biomechanical properties of the ECM [70, 52, 62].

1.2 Rationale

Recent cancer research has suggested the remodeling of collagen fibers in ECM facilitates the migration of cancer cells during metastasis. However, the mechanism of such cancer migration through a remodeling ECM remains not well understood. Our work considers mathematical modeling approach to unravel the complexity of many interacting processes during ECM remodeling. Modeling serves as a tool for researchers to formularize and quantitatively understand physical and chemical phenomena observed in reality via the language of mathematics [73]. Mathematical modeling has been a prominent practice in almost all fields of science and engineering in general and in cancer system biology in particular. Scientists and researchers find benefit from a well-developed mathematical model that could deliver one or more of the following useful functions: test hypotheses, lead to new insights and new approaches, suggest and refine experiments, interpret experiments, trace chains of causation, carry out sensitivity analyses, and investigate coupling and feedback [14]. Numerous mathematical models of cancer invasion have been established in the last twenty years and are constantly being improved upon [33, 34, 60, 65, 66, 3, 4, 86, 2, 18, 19, 32, 39, 35, 72, 5, 68, 83, 63, 64, 1, 46, 85, 25, 28]. However, no model has yet considered the enzyme LOX and its promising influence on metastasis in ECM. In our new mathematical models of metastatic cancer invasion, we not only added a new equation for LOX but also extended the existing models with improved features showing how LOX affects ECM remodeling and metastatic cancer migration.

1.3 Objectives

This work aims to develop *in silico* models targeting a metastatic tumor microenvironment. Two *in silico* models following different modeling approaches are addressed in this thesis to study the complex interplay between metastatic cancer cells and the ECM while it undergoes remodeling. One model is developed via the continuum

modeling approach, while the other follows the hybrid continuum-discrete modeling technique. The first model is a continuous system of five coupled partial differential equations (PDEs) reflecting the spatiotemporal dynamics and interconnection of the following components: population density of cancer cells and concentrations for two types of ECM collagen fibers and the enzymes MMP and LOX. Via the open-source software CompuCell3D, the second model includes both discrete and continuous variables to simulate a multiscale tumor microenvironment in which cells are treated as discrete while other components such as the ECM and chemicals are described through continuum fields. The setting of both models assume that the primary tumor is approaching the metastatic stage and that cancer cells have already penetrated the basement membrane surrounding the tumor. These simulations consider malignant cancer cells that are ready to detach away from the tumor mass and invade the ECM.

Chapter 2

Overview of the Tumor Microenvironment Components

The tumor microenvironment (TME) has emerged as an important tissue in the study of cancer, especially in the prognostic guiding of tumor growth and tumor behaviors [10]. TME is the domain that surrounds a tumor (Figure 2.1) including multiple cellular components (e.g. fibroblast, macrophages, immune cells, inflammatory cells, lymphocytes, and vascular endothelial cells) non-cellular components (ECM and collagens) or biochemical components (e.g. growth factors, oxygen, glucose, and remodeling enzymes) residing within that surrounding. TME and its diverse microenvironmental components, hence, are becoming interesting targets for the study and modeling of cancer attracting both experimental and computational researchers.

2.1 Extracellular Matrix

In a TME, ECM acts as a scaffold supporting the surrounding cells, helping them communicate with each other and with the ECM itself through both physical and biochemical signaling [11]. The direct interaction of ECM with tumor cells via adhesion contact and the highly dynamic characteristics of ECM together promote cancer growth and spread [7]. The ECM is an insoluble non-cellular environment outside the cells present in all tissues of a human body [12]. Generally, ECM consists of water, polysaccharides, and proteins. Various types of fibrous proteins are present in the ECM including collagens, elastins, fibronectins, and laminins; among these, collagen is the most abundant and the main structural protein in the ECM [31]. The collagen fibers are constantly being remodeled [52]. The remodeling of collagen fibers results in

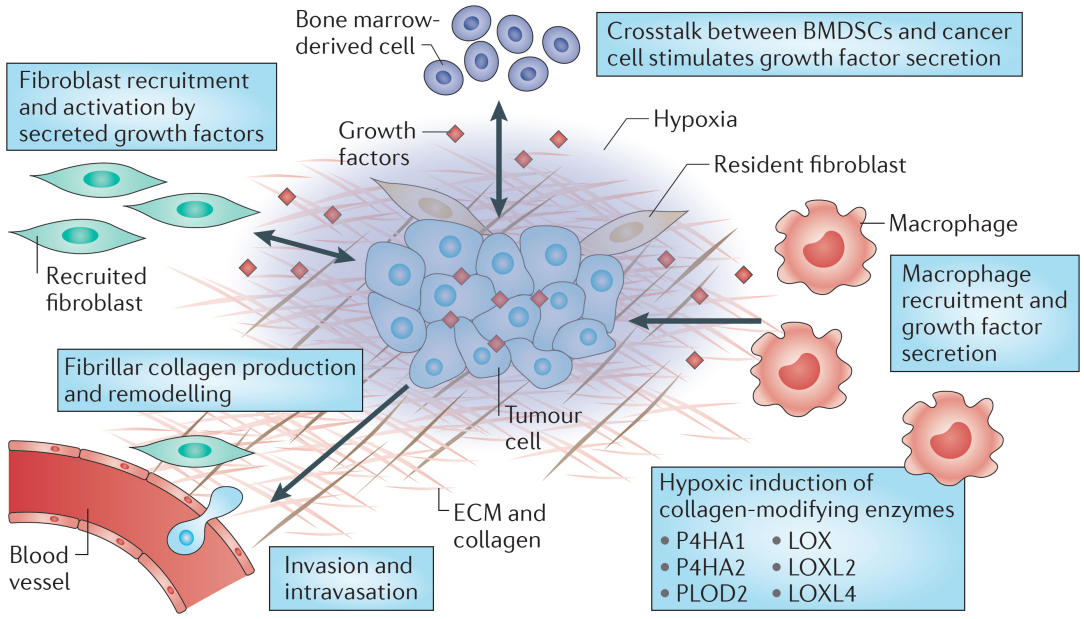

Figure 2.1: Dynamics during tumor evolution in TME involved multiple interacting biophysical and biological processes among various components including blood vessel, tumor cells, stromal cells like fibroblast and macrophage, ECM collagen fibers, chemicals like remodeling enzymes, oxigen and growth factors. Components are labeld and key cancer-related processes are described in the boxes [37]

dynamic changes in the physical, chemical, and biomechanical properties of the ECM [90]. ECM remodeling refers to the following two processes: (a) ECM degradation caused by enzyme matrix metalloproteinases (MMPs) and (b) ECM alignment due to cross-linking enzyme lysyl oxidase (LOX) (Figure 2.2). In a pathological condition, like cancer, such changes in the ECM properties facilitate the movement of cells [71]. Via cell-ECM interaction and signaling pathways, these changes elicit cell responses to secrete chemicals like MMPs and LOX to shape the orientation of ECM from a barrier of randomly oriented fibers into a more propitious fibrous "expressway" for migration [22, 88, 53].

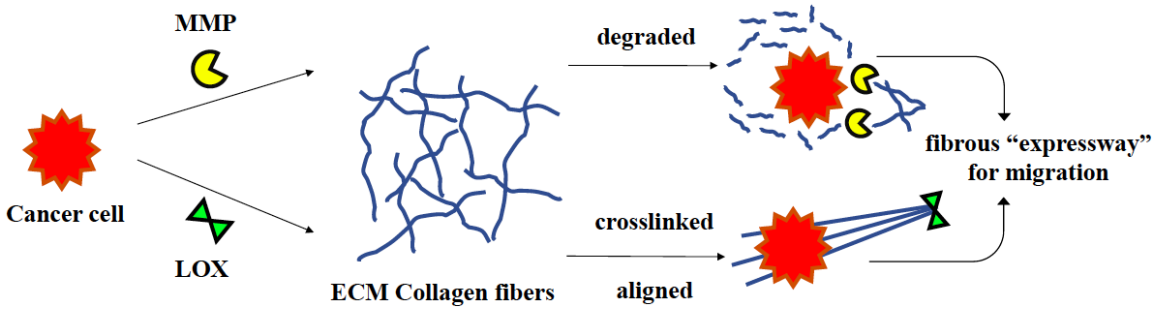

Figure 2.2: Influence of ECM remodeling on cancer migration

The directional motility of cells through the ECM is guided mainly by cells sensing and responding to haptotactic and durotactic cues from the ECM components and their properties and characteristics (Figure 2.3). Haptotaxis is the ability of cells to sense and respond to a gradient of immobilized chemoattractants, while durotaxis is the directional motility of cell up a gradient of mechanical stiffness. In our work, we currently address only the effect of haptotaxis on cell migration in term of the gradient of fiber concentration as a surrogate for the ECM stiffness.

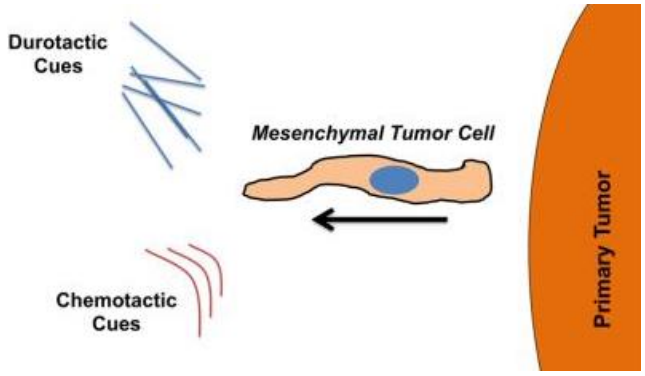

Figure 2.3: Directed migration of mesenchymal cells for various types of directional cues [9].

2.2 Matrix Metalloproteinases

The family of MMPs is the most prominent enzymes in ECM remodeling, responsible for proteolytic degradation of the collagen fibers in the ECM [17]. Here, we focus on the collagen fibers residing in the interstitial ECM, which are mostly collagen type I

and are degraded by MMP-14, one of the MMP family members, secreted by cancer cells [52]. As cancer cells becomes metastatic, they secretes MMPs to degrade the collagen fibers in the ECM. MMPs hence first help the metastatic cell to detach away from the primary tumor by breaching the basement membrane encapsulating the tumor and then go on to generate spaces in the matrix for cells to maneuver through the ECM more easily (Figure 2.4). Dysregulation of the activities of MMPs intervenes in almost all cancer types [16] and in the hallmarks of cancer from migration and initial invasion to angiogenesis and metastasis [36]. Due to its important role as a driving factor for cancer progression, MMPs have been and continue to be appealing and promising targets of cancer therapy despite the failure of several clinical trials for MMP inhibitors conducted within the past three decades [23, 29, 16]. For simplification, we lump all the family of MMPs together in this work, referring to them collectively as MMP hereafter.

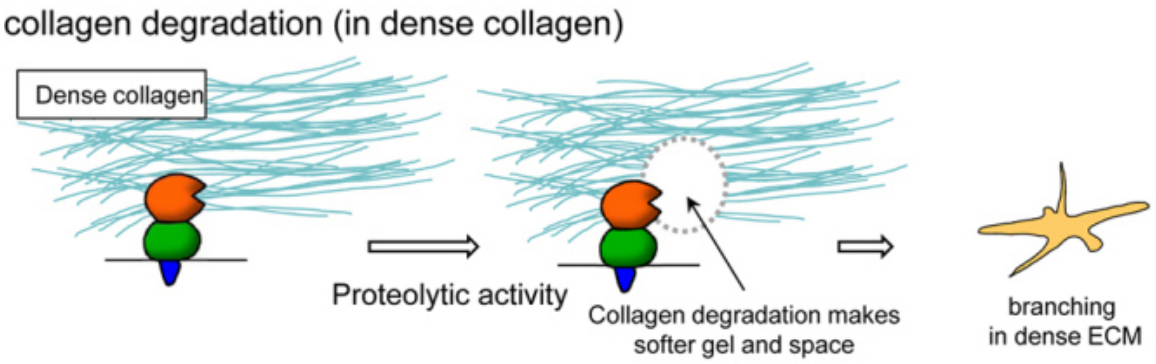

Figure 2.4: Proteolytic activities of MMP degrading and invading the microenvironment of collagen [56].

2.3 Lysyl Oxidase

LOX oxidizes the primary amine substrates to reactive aldehydes (Figure 2.5). Via this amine oxidase activity, collagens are crosslinked and aligned in the ECM [79]. Bundles of crosslinked collagen fibers stiffen the ECM and aid the migration of cancer cells via growth factor signaling pathways, for instance, the transforming growth

factor-β (TGF-β) and the fibroblast growth factor (FGF) [27, 76]. Evidence of thickening and aligned collagen fibers due to LOX has been observed in areas of active tissue invasion and tumor vasculature [22, 70]. While the expression of LOX is found to be upregulated during cancer invasion and metastasis [61, 55], the underlying mechanism of how LOX modulates the cell-ECM interaction during invasion and metastasis to facilitate cancer cells migration remains poorly understood.

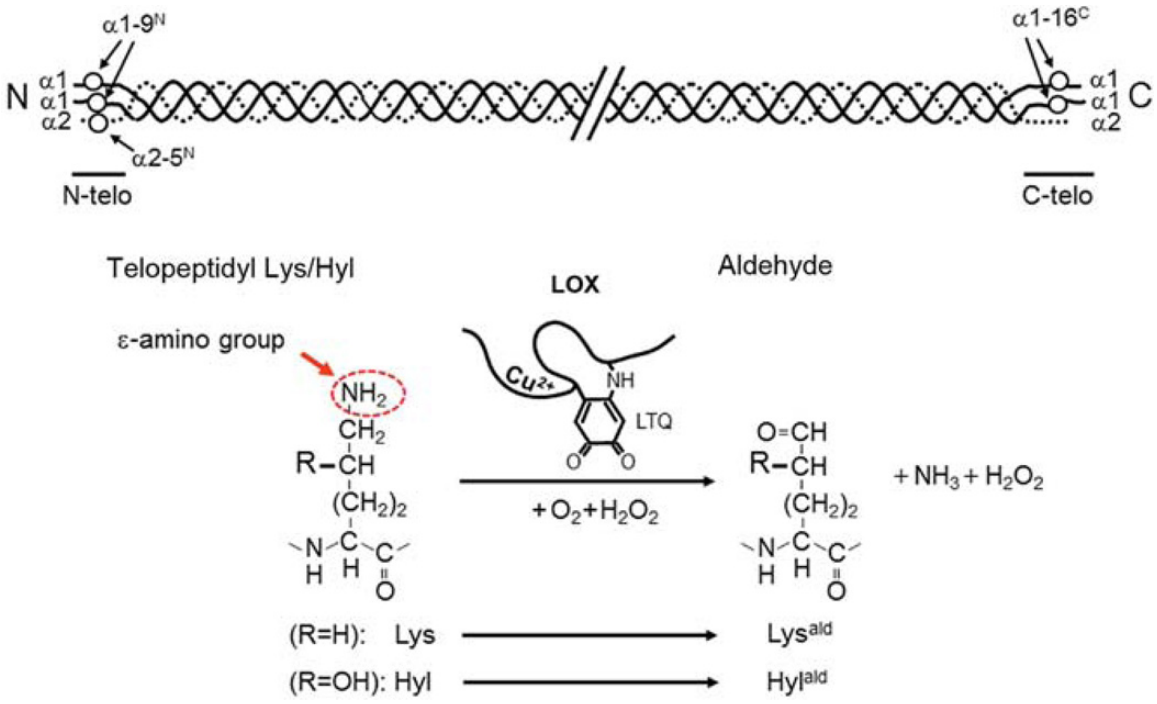

Figure 2.5: Schematic of the oxidative deamination of lysine and hydroxylysine in type I collagen by LOX [89].

Chapter 3

Continuum Modeling of Metastatic Cancer Migration through a Remodeling ECM

Most of the work in this chapter has been published in ***Processes*** *journal in May, 2018* [57].

In the following sections, we derive and formulate a system of five partial differential equations (PDEs) to describe the local invasion and migration of cancer cells through a remodeling ECM. We also briefly introduce the method used to numerically solve our model in MATLAB (R2017b, MathWorks, Natick, MA, USA, 2017) and detail in Tables 3.1 and 3.2 all the parameter values applied in our simulation. We present our results obtained from the numerical simulation and interpret our *in silico* findings from a biological and pathological perspective. We analyze three case studies to highlight the significance of adding the enzyme LOX into the model and the impact from different modes of haptotaxis caused by LOX on the ECM and on the migration of cancer cells. Lastly, we conclude with remarks on the potential usefulness of the proposed model to further clinical understanding of the critical role of ECM remodeling in the early stages of cancer metastasis.

3.1 Formulation of the Model Equations

The setting of the model established in this work is an *in silico* metastatic tumor microenvironment (Figure 3.1). The model aims to unravel the interconnection of the main concepts of metastasis in the ECM: the spreading of cancer cells, the remodeling

of collagen fibers, and the reaction and transport dynamics of the chemicals involved. We start the model time period just after cancer cells have penetrated the basement membrane of the primary tumor (Figure 3.1A). At this stage, the malignant cancer cells detach away from the primary tumor mass and squeeze through the gaps in the degraded basement membrane. These motile cancer cells proceed to invade the ECM and maneuver their way through a barrier of ECM collagen fibers undergoing remodeling (Figures 3.1B–D). Such modification of the ECM facilitates the migration of cancer cells via haptotactic sensing and response from cancer cells toward the degraded and crosslinked areas of ECM collagen fibers. Beyond what our model covers, the escaped cancer cells will eventually reach and intravasate nearby blood vessels or invade other connective tissues to travel to other parts of the body and initiate secondary tumors.

The model established in this section is a continuous system of five coupled partial differential equations (PDEs) describing the dynamics and interaction of cancer cells, collagen fibers, and the enzymes MMP and LOX. Two population of collagen fibers are considered: those that are oriented randomly and those that have been crosslinked.

Cancer Cells

The population balance of cancer cells in the system is governed by three main factors: random diffusion, proliferation, and haptotaxis due to the remodeling ECM collagen fibers:

$$\frac{\partial c}{\partial t} = D_c \frac{\partial^2 c}{\partial x^2} + \gamma c(1 - v_1 c - v_2 f - v_3 f_{cl}) - \frac{\partial}{\partial x}\left[\rho(1 - v_1 c - v_2 f - v_3 f_{cl}) c \frac{\partial f}{\partial x}\right] + g \quad (3.1)$$

where c is the number of cancer cells per volume of the spacial domain x in one dimension at time t. The first term accounts for the diffusive migration of cells with a constant diffusion coefficient D_c. We assume any nutrient transport supplied to the tumor from nearby blood vessels is unaffected by the remodeling and thus is ignored

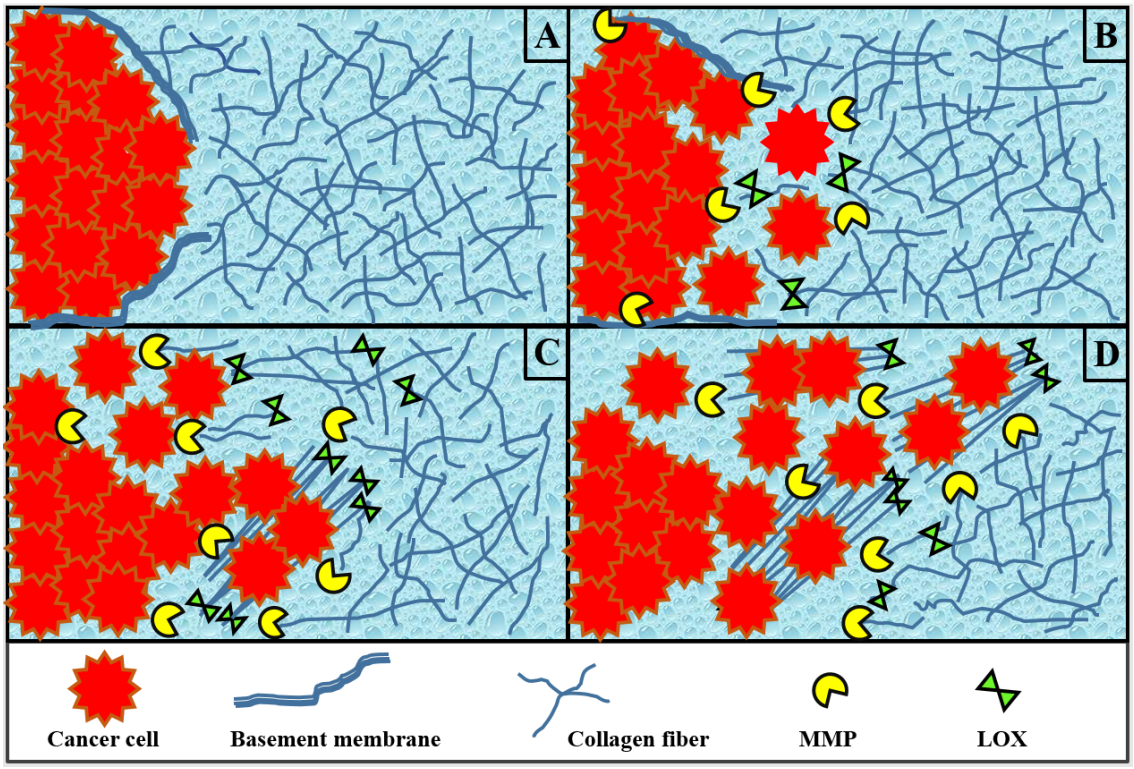

Figure 3.1: Dynamics in a metastatic tumor microenvironment: (**A**) The basement membrane around the edge of the primary tumor (cluster of cancer cells in red) has already been perforated by cancer cells. The surrounding collagen fibers are randomly oriented in the extracellular matrix (ECM). (**B**) Enzymes metalloproteinase (MMP) and lysyl oxidase (LOX) are secreted by cancer cells to degrade and cross-link collagen fibers. MMP generates spaces for cancer cells to begin detaching away from the primary tumor mass to invade the ECM. (**C**) Meanwhile, aligned and crosslinked collagen fibers form a fibrous pathway along which cancer cells prefer to travel. (**D**) Collagen fibers continue to be crosslinked to aid the maneuvers of cancer cells further through the matrix.

in our model. The second term is included to account for cell proliferation using a first-order rate expression for c with specific reatction rate γ as an exponential decay function $\gamma = \exp(-x^2/\epsilon)$, where ϵ is an adjustable parameter. The third and fourth terms in the equation implement the haptotaxis effect that directs motility of cancer cells towards the area of remodeling ECM with ρ and ρ_{cl} denoting the haptotaxis parameters for non-crosslinked and crosslinked ECM collagen fibers, respectively. Based on studies on cancer migration using live imaging, cancer cells are prone to move more rapidly on collagen fibers in the collagen enriched area [22, 27]. Hence, we assume the effect of haptotaxis on motility of cancer cells toward the crosslinked area of ECM, f_{cl}, will be stronger than toward the area of degraded ECM that has not yet been crosslinked. We incorporate the logistic growth factor $(1 - v_1 c - v_2 f - v_3 f_{cl})$ in the last three terms of (3.1) where v_1, v_2, and v_3 are the occupied fractions of one unit volume of physical space by the densities of cancer cells c, regular collagen fibers f, and crosslinked collagen fibers f_{cl}, respectively. The logistic growth factor is implemented to ensure the migration of cancer cells only into space that is not already occupied by the c, f, and f_{cl} present in the system domain [35]. The fourth term g represents the attraction or the haptotactic migration of cancer cells toward the crosslinked ECM collagen fibers:

$$g = -\frac{\partial}{\partial x}\left[\rho_{cl}(1 - v_1 c - v_2 f - v_3 f_{cl}) c \frac{\partial f_{cl}}{\partial x}\right]. \tag{3.2}$$

The term g is turned on or off ($g \equiv 0$) in the simulation to investigate the impact of LOX on driving the ECM stiffness via cross-linking and how that influences the overall migration behavior of cancer cells in the system.

Extracellular Matrix Collagen Fibers

Changes in the morphology and the biochemical and physical properties of a tumor-associated ECM make a substantial impact on regulating cancer progression during

metastasis [12]. Since ECM collagen fibers do not diffuse [74] and can be treated as a static structural support network, none of the terms incorporated in the ECM model relate to motility. Instead, the dynamics of ECM is expressed via the remodeling of its collagen fibers due to LOX and MMP. Under disease conditions like cancer, MMP and LOX are known to be dysregulated and often overexpressed, especially in carcinoma. While MMP digests the collagen fibers and generate spaces within the ECM, LOX crosslinks and linearizes the fibers, hence further organizing the matrix of fibers. Although they have different roles in ECM remodeling, both enzymes contribute greatly to transforming ECM from a barrier of fibers into a more navigable fibrous structure that facilitates the migration of cancer cells through the ECM [52, 53].

In our model, we include two species to distinguish the structure and function of ECM collagen fibers. The randomly oriented collagen fibers that have not been crosslinked are represented by f. The collagen fibers that are crosslinked, linearized, and aligned are denoted by f_{cl}. Assuming a domain of constant volume, the balances for the two collagen fibers species are

$$\frac{\partial f}{\partial t} = -\alpha_f m f + \mu_f(1 - v_1 c - v_2 f - v_3 f_{cl}) - h \tag{3.3}$$

$$\frac{\partial f_{cl}}{\partial t} = -\alpha_f m f_{cl} + h \tag{3.4}$$

where the first term in both models accounts for the degradation of collagen fibers by the concentration of MMP, m, present in the system domain, α_f is the MMP cleavage rate of collagen fibers, and μ_f is the rate constant for production due to the regular collagen synthesis in ECM. The logistic growth fraction $(1-v_1c-v_2f-v_3f_{cl})$ again is incorporated in the source term to check for the availability of unoccupied space for the normal synthesis of collagen fibers to take place. The term h is included in both equations indicating the rate of converting regular collagen fibers into crosslinked fibers. The h term depends on the presence of LOX, l, and collagen fibers according

to

$$h = \beta_f f l \tag{3.5}$$

where β_f is the reaction rate for cross-linking of collagen fibers by LOX. Like the g term in Equation (3.1), the h term in Equations (3.3) and (3.4) is turned on and off in the simulation depending on the existence of LOX in the system.

Enzyme MMP

The evolution of the chemical signal MMP in the system is modeled using reaction-diffusion dynamics

$$\frac{\partial m}{\partial t} = D_m \frac{\partial^2 m}{\partial x^2} - \alpha_m m + \beta_m \, c \tag{3.6}$$

where the first term accounts for MMP diffusion with a constant diffusion coefficient D_m, the second term considers the natural decay of MMP with α_m denoting the MMP decay rate constant, and the third term accounts for the secretion of MMP by cancer cells with a production rate constant β_m. In reality, MMP is secreted by many different types of cells in the malignant tumor microenvironment including cancer-associated fibroblasts (CAFs), inflammatory cells, macrophages, and cancer cells [43]. However, the collagen fibers residing in the interstitial ECM are mostly collagen type I, which is degraded mostly by MMP-1 secreted by CAFs and MMP-14 secreted by cancer cells [67, 52]. In the current model, we have not included the existence of CAFs. The secretion of MMP, generally, is considered from cancer cells.

Enzyme LOX

Similar to that for MMP dynamics, the reaction-diffusion equation for LOX is constructed as

$$\frac{\partial l}{\partial t} = D_l \frac{\partial^2 l}{\partial x^2} - \alpha_l l + \beta_l c \tag{3.7}$$

where D_l is the chemical diffusion coefficient of LOX, α_l is the natural decay rate constant for LOX, and β_l is the rate constant for the secretion rate of LOX by cells

3.2 Nondimensionalization and Parameter Estimation

To numerically solve the system of PDEs, non-dimensionalization is first performed. Nondimensionalization helps establish fewer parameters in turn allowing for a much more efficient way to analyze the system of PDEs. The dimensionless variables are

$$\hat{x} \equiv \frac{x}{L}, \quad \hat{t} \equiv \frac{t}{\tau}, \quad \hat{c} \equiv \frac{c}{c_o}, \quad \hat{f} \equiv \frac{f}{f_o}, \quad \hat{f_{cl}} \equiv \frac{f_{cl}}{f_o}, \quad \hat{m} \equiv \frac{m}{m_o}, \quad \hat{l} \equiv \frac{l}{l_o} \tag{3.8}$$

where parameters L, τ, c_o, f_o, m_o, and l_o are appropriate reference values for scaling x, t, c, f and f_{cl}, m, and l, respectively. Table 3.1 summarizes the reference values and parameters that are available from the literature. Considering the early stages of metastatic cancer cell invasion, the length scale L is taken to be 1 cm but could be in the range of 0.1 to 1 cm. The residence time, τ, is taken to be 32 h, which is a representative average *in vitro* doubling time for the well-established human cancer cell lines of A549 lung carcinoma cells, U87MG glioma cells, and MCF-7 and MDA-MB 231 breast cancer cells [15, 47, 51, 59]. The diffusion coefficient of the cancer cell, D_c, has been previously determined in an experiment of cell movement by [13]. The diffusion coefficient for the MMP, D_m, is taken as the higher end value in the range of diffusion coefficient for membrane proteins from 10^{-8} to 10^{-10} cm^2 s^{-1} reported in [40]. The haptotaxis parameter, ρ, is taken from [3]. The reference chemical diffusion coefficient, D, is from [13]. The reference concentration of ECM collagen fiber, f_o, is in the range of 10^{-11} to 10^{-8} M based on [84]. Again, the lower end value of f_o, which is 10^{-11}, is applied in our model. A value of 0.1×10^{-9} M is taken from [1] as an appropriate reference chemical concentration for both MMP and LOX, m_o and l_o, respectively.

Table 3.1: Parameter values available from literature used in the model of metastatic invasion of cancer through remodeling ECM.

Term	Description	Value	Unit	Sources
L	Reference length	1	cm	[4]
τ	Reference time	32	hours	[15, 47, 51, 59]
c_o	Reference number of cancer cells per volume	6.7×10^7	cells/cm^3	[4]
f_o	Reference value for f, f_{cl}	10^{-11}	M	[84]
m_o, l_o	Reference value for m, l	0.1×10^{-9}	M	[1]
D_c	Diffusion coefficient of cancer cells	10^{-9}	cm^2/s	[13]
D_m	Diffusion coefficient of MMP	10^{-8}	cm^2/s	[40, 4, 21]
ρ	Haptotaxis coefficient toward f	2600	cm^2 M^{-1} s^{-1}	[3]

Introducing the dimensionless quantities defined in (3.8) into Equations (3.1)–(3.7), the model can be written as

$$\begin{aligned}\frac{\partial \hat{c}}{\partial \hat{t}} =& \left(\frac{D_c\tau}{L^2}\right)\frac{\partial^2 \hat{c}}{\partial \hat{x}^2} + (\gamma\tau)\hat{c}\big(1 - (v_1c_o)\hat{c} - (v_2f_o)\hat{f} - (v_3f_o)\hat{f}_{cl}\big) \\ &- \frac{\partial}{\partial \hat{x}}\left[\left(\frac{\rho f_o\tau}{L^2}\right)\big(1 - (v_1c_o)\hat{c} - (v_2f_o)\hat{f} - (v_3f_o)\hat{f}_{cl}\big)\hat{c}\frac{\partial \hat{f}}{\partial \hat{x}}\right] + \hat{g}\end{aligned} \tag{3.9}$$

$$\hat{g} = -\frac{\partial}{\partial \hat{x}}\left[\left(\frac{\rho_{cl} f_o\tau}{L^2}\right)\big(1 - (v_1c_o)\hat{c} - (v_2f_o)\hat{f} - (v_3f_o)\hat{f}_{cl}\big)\hat{c}\frac{\partial \hat{f}_{cl}}{\partial \hat{x}}\right] \tag{3.10}$$

$$\frac{\partial \hat{f}}{\partial \hat{t}} = -(\alpha_f\tau m_o)\hat{m}\hat{f} + \left(\frac{\mu_f\tau}{f_o}\right)\big(1 - (v_1c_o)\hat{c} - (v_2f_o)\hat{f} - (v_3f_o)\hat{f}_{cl}\big) - \hat{h} \tag{3.11}$$

$$\frac{\partial \hat{f}_{cl}}{\partial \hat{t}} = -(\alpha_f\tau m_o)\hat{m}\hat{f}_{cl} + \hat{h} \tag{3.12}$$

$$\hat{h} = (\beta_f\tau l_o)\hat{f}\hat{l} \tag{3.13}$$

$$\frac{\partial \hat{m}}{\partial \hat{t}} = \left(\frac{D_m\tau}{L^2}\right)\frac{\partial^2 \hat{m}}{\partial \hat{x}^2} - (\alpha_m\tau)\hat{m} + \left(\frac{\beta_m\tau c_o}{m_o}\right)\hat{c} \tag{3.14}$$

$$\frac{\partial \hat{l}}{\partial \hat{t}} = \left(\frac{D_l\tau}{L^2}\right)\frac{\partial^2 \hat{l}}{\partial \hat{x}^2} - (\alpha_l\tau)\hat{l} + \left(\frac{\beta_l\tau c_o}{l_o}\right)\hat{c}. \tag{3.15}$$

The dimensionless parameters that emerge within parentheses are defined in Table 3.2. Values for these dimensionless parameters are obtained either through calculation (if sufficient information is provided from literature listed in Table 3.1, these are labeled as "Calculated from") or by being given tentative values that are reasonable

for the model (those with related values obtained from other previous models are listed as "Estimated from" followed by the source).

Table 3.2: Dimensionless expression and values of parameters used in MATLAB simulation of metastatic invasion of cancer through a remodeling ECM.

Term	Description	Value	Sources
$\hat{D_c} = D_c\tau/L^2$	Diffusion coefficient of cell	1.152×10^{-4}	Calculated from [13]
$\hat{\gamma} = \gamma\tau$	Rate expression for tumor proliferation	$\exp(-\hat{x}^2/\hat{\epsilon}), \quad \hat{\epsilon} = 0.001$	Assumed
$\hat{\rho} = \rho f_o\tau/L^2$	Haptotaxis toward f	0.003	Calculated from [84, 13]
$\hat{\rho_{cl}} = \rho_{cl} f_o\tau/L^2$	Haptotaxis toward f_{cl}	0.05	Assumed
$\hat{v_1} = v_1 c_o$	Space fraction per unit $\hat{c}$	1	By definition from [35]
$\hat{v_2} = v_2 f_o$	Space fraction per unit $\hat{f}$	1	By definition from [35]
$\hat{v_3} = v_3 f_o$	Space fraction per unit $\hat{f_{cl}}$	1	By definition from [35]
$\hat{\alpha_f} = \alpha_f\tau m_o$	Rate constant for MMP cleavage of f	10	Estimated from [4]
$\hat{\mu_f} = \mu_f\tau/f_o$	Rate constant for production of f	0.15	Estimated from [1]
$\hat{\beta_f} = \beta_f\tau l_o$	Rate constant for LOX remodeling of f	18	Assumed
$\hat{D_m} = D_m\tau/L^2$	Diffusion of MMP	1.152×10^{-3}	Calculated from [40, 15]
$\hat{\alpha_m} = \alpha_m\tau$	Rate constant for decay of MMP	1×10^{-3}	Estimated from [4, 49]
$\hat{\beta_m} = \beta_m c_o\tau/m_o$	Rate constant for secretion of MMP by cells	0.1	Estimated from [4]
$\hat{D_l} = D_l\tau/L^2$	Diffusion coefficient of LOX	2.304×10^{-3}	Assumed
$\hat{\alpha_l} = \alpha_l\tau$	Rate constant for decay of LOX	1×10^{-3}	Assumed
$\hat{\beta_l} = \beta_l c_o\tau/l_o$	Rate constant for secretion of LOX by cells	0.1	Assumed

3.3 Initial and Boundary Conditions

Some of the initial conditions for the concentrations in our models follow a similar set of initial conditions proposed in several previous cancer invasion mathematical models [2, 19, 35, 1]. The center of the primary tumor mass resides at the left edge of the system domain $\hat{x} = 0$. Initially, it is assumed that a fixed cluster of cancer cells already exists in the system domain from $\hat{x} \in [0, 0.25]$. Additionally, at $\hat{x} = 0.25$ is the edge where the basement membrane of the primary tumor resides before being degraded. The initial distribution of the population density of cancer cells in the system given by

$$\hat{c}(\hat{x}, 0) = \begin{cases} \exp(\frac{-\hat{x}^2}{\sigma}), & \hat{x} \in [0, 0.25] \\ 0, & \hat{x} \in [0.25, 1] \end{cases} \tag{3.16}$$

where σ is a positive constant given a value of 0.01 [4].

Initially, the ECM is a mesh of randomly oriented collagen fibers that are not yet

crosslinked, so

$$\hat{f}_{cl}(\hat{x}, 0) = 0 \tag{3.17}$$

To ensure the physical space of the system domain is not congested, the initial condition for the concentration of ECM collagen fibers must satisfy $\hat{c}(\hat{x}, 0) + \hat{f}(\hat{x}, 0) \leq 1$ [35], so

$$\hat{f}(\hat{x}, 0) = 1 - \hat{c}(\hat{x}, 0). \tag{3.18}$$

We assume a zero concentration of MMP and LOX presented in the system initially:

$$\hat{m}(\hat{x}, 0) = 0 \tag{3.19}$$

$$\hat{l}(\hat{x}, 0) = 0. \tag{3.20}$$

Zero flux boundary conditions for symmetry are imposed on the left edge of the spatial domain ($\hat{x} = 0$) for all components ($\hat{c}, \hat{f}, \hat{f}_{cl}, \hat{m}$, and $\hat{l}$). On the right edge of the spatial domain, it is assumed that there are no cancer cells, crosslinked ECM fibers, MMP, or LOX ($\hat{c}, \hat{f}_{cl}, \hat{m}$, and $\hat{l}$, respectively). On the other hand, there exists a constant amount of non-crosslinked ECM fibers, $\hat{f}$, on the right edge. Hence, the set of boundary conditions imposed on the system is taken as below

$$\left.\frac{\partial \hat{c}}{\partial \hat{x}}\right|_{\hat{x}=0} = \left.\frac{\partial \hat{f}}{\partial \hat{x}}\right|_{\hat{x}=0} = \left.\frac{\partial \hat{f}_{cl}}{\partial \hat{x}}\right|_{\hat{x}=0} = \left.\frac{\partial \hat{m}}{\partial \hat{x}}\right|_{\hat{x}=0} = \left.\frac{\partial \hat{l}}{\partial \hat{x}}\right|_{\hat{x}=0} = 0 \tag{3.21}$$

$$\hat{c}(1, \hat{t}) = \hat{f}_{cl}(1, \hat{t}) = \hat{m}(1, \hat{t}) = \hat{l}(1, \hat{t}) = 0 \tag{3.22}$$

$$\hat{f}(1, \hat{t}) = 1. \tag{3.23}$$

3.4 Numerical Methods and Code Repository

The system of coupled PDEs (3.9)–(3.23) in this paper is numerically solved in the domain $\Omega = [0, 1] \times (0, 20]$ utilizing the `pdepe` function, an internal PDE solver in

MATLAB that discretizes the equations in space to obtain a system of ordinary differential equations in time that is then solved along the discrete grid points. This function can handle solving initial-boundary value problems for systems of parabolic and elliptic PDEs in one spatial variable x and time t [38]. The PDEs that the function `pdepe` can solve must follow the general form

$$a_1\left(x,t,u,\frac{\partial u}{\partial x}\right)\frac{\partial u}{\partial t} = x^{-n}\frac{\partial}{\partial x}\left(x^n a_2\left(x,t,u,\frac{\partial u}{\partial x}\right)\right) + a_3\left(x,t,u,\frac{\partial u}{\partial x}\right) \tag{3.24}$$

Our model follows this form with $n = 0$, indicating rectangular coordinates for `pdepe`.

To enable code reuse, we wrote the model in MATLAB and shared the code including parameter values and documentation in an open-source software repository [58].

3.5 One Dimensional Results in MATLAB

In this section, we present and analyze in silico experimental results in one dimension for three case studies of the model: (I) No LOX nor its effects on ECM and cancer cells (for all time steps: $\hat{l} \equiv 0$, $\hat{f}_{cl} \equiv 0$, $\hat{h} \equiv 0$, and $\hat{g} \equiv 0$); (II) LOX present but without its effect on the haptotactic migration of cancer cells toward the crosslinked ECM fibers (for all time steps: $\hat{g} \equiv 0$); (III) LOX present and including all of its effects on ECM fibers and cancer cells haptotactic migration. For all results, we have dropped the "hat" notation for convenience. Only dimensionless quantities were used in the model equations.

Case I: LOX is Absent

In this case study, LOX is not present in the system, so $l \equiv 0$. This leads to no cross-linking of ECM collagen fibers, so $h \equiv 0$ and $f_{cl} \equiv 0$. Hence, the term g in (3.2), representing the haptotactic migration of cancer cells toward the crosslinked

ECM collagen fibers, is also turned off in the simulation, hence $g \equiv 0$. The system of PDEs is thus left with only three active species: c, f, and m.

The purpose of excluding LOX from the model in Case I is to validate our model before further exploration by comparing our results with those obtained from a highly cited published cancer invasion model [4]. The model in [4] included three PDEs for c, f, and m. Our results (not shown here) in Case I for the same three components combined with the same initial and boundary conditions used in [4] show the same results as the published model.

We propose starting the initial condition from $m = 0$, deviating from [4], to consider the onset of MMP secretion from cancer cells. Figure 3.2 shows the spatial profiles of the tumor microenvironment at different dimensionless simulation times ($t = 0, 1, 10$, and 20). The results obtained for the population density of cancer cells capture a slow detachment of the cells away from the center the primary tumor mass (the left edge of the spatial domain) as time evolves. We also observe that ECM fiber concentration is low when and where MMP concentration is high. Such trends are expected due to the fact that MMP degrades and digests ECM collagen fibers.

Case II: LOX is Present But Not Coupled to Haptotactic Migration of Cancer Cells

In Case II, LOX, l, is introduced to the system. Since LOX is present, the rate of cross-linking of collagen fibers, h, must be activated in the model. However, the haptotactic migration effect on the population density of cancer cells in Equation (3.1) remains inactive in the simulation ($g \equiv 0$).

Our primary aim for the settings in Case II is to confirm that LOX can perform its main function on ECM via the activation of h, which transforms regular collagen fibers to crosslinked ones. Meanwhile, in the absence of the attraction of cancer cells toward the crosslinked fibers ($g \equiv 0$), the migration of cancer cells is only influenced

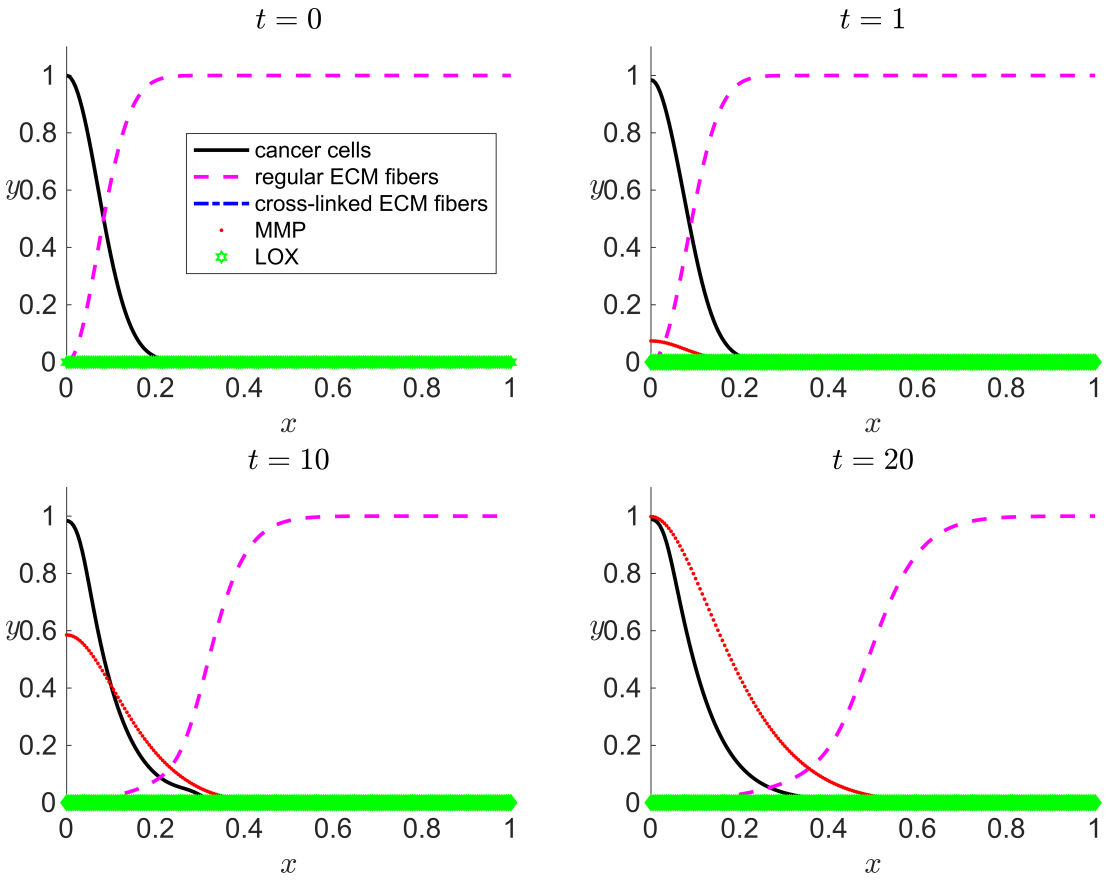

Figure 3.2: One dimensional numerical results for Case I when there is zero concentration of LOX ($l \equiv 0$) and hence, zero crosslinked ECM collagen fibers, f_{cl}, in the system. Results are snapshots of the system dynamics at four simulation times: (**A**) $t = 0$, (**B**) $t = 1$, (**C**) $t = 10$, and (**D**) $t = 20$. For all four plots, the horizontal axis, x, indicates a dimensionless spatial position, and the vertical axis, y, indicates the dimensionless population density or concentration of the species indicated in the legend.

by haptotactic migration toward the pristine collagen fibers, f, which are present in lower concentrations compared with Case I since they can be crosslinked.

In Figure 3.3, the slow detachment of cancer cells away from the center the primary tumor mass and the effect of ECM degradation by MMP are captured in similar trends as in the Case I results and as in [4]. The main difference between the results of Cases I and II is the existence of the crosslinked ECM fibers and f_{cl} dynamics. With h active and g inactive, the concentration of regular ECM fibers decreases more quickly and to a larger penetration distance due to cross linking (compare the regular ECM fiber curve in Figure 3.3D with that in Figure 3.2D). This slightly reduces the cancer cell haptotactic driving force. Additionally, the volume occupancy fractions of other species are affected by f_{cl} in Case II.

Case III: LOX is Present and Is Coupled to Haptotactic Migration of Cancer Cells

In Case III, LOX is present in the system, and both the h and g terms are turned on in the model. In the simulation results (Figure 3.4), besides the recurrence of the expected phenomena from Cases I and II, a peak for cancer cells evolves over time away from the primary tumor ($x > 0.25$) (Figure 3.4C,D). The location of the peak of cancer cell population density corresponds to trailing the wave of crosslinked ECM fiber concentration. This suggests that cancer cells that have invaded the ECM via migration are clustered in the area where there is a high concentration of crosslinked ECM collagen fibers. Such behavior is the consequence of the haptotaxis effect caused by the directional motility of cancer cells toward the crosslinked ECM collagen fibers. Additionally, a smaller, secondary peak appears to the left of the migrating cell front where conditions are favorable for the secretion of new regular ECM fibers. Results of Case III have confirmed the capability of our new model to successfully implement the extended features to capture the cross-linking effect that LOX performs on ECM.

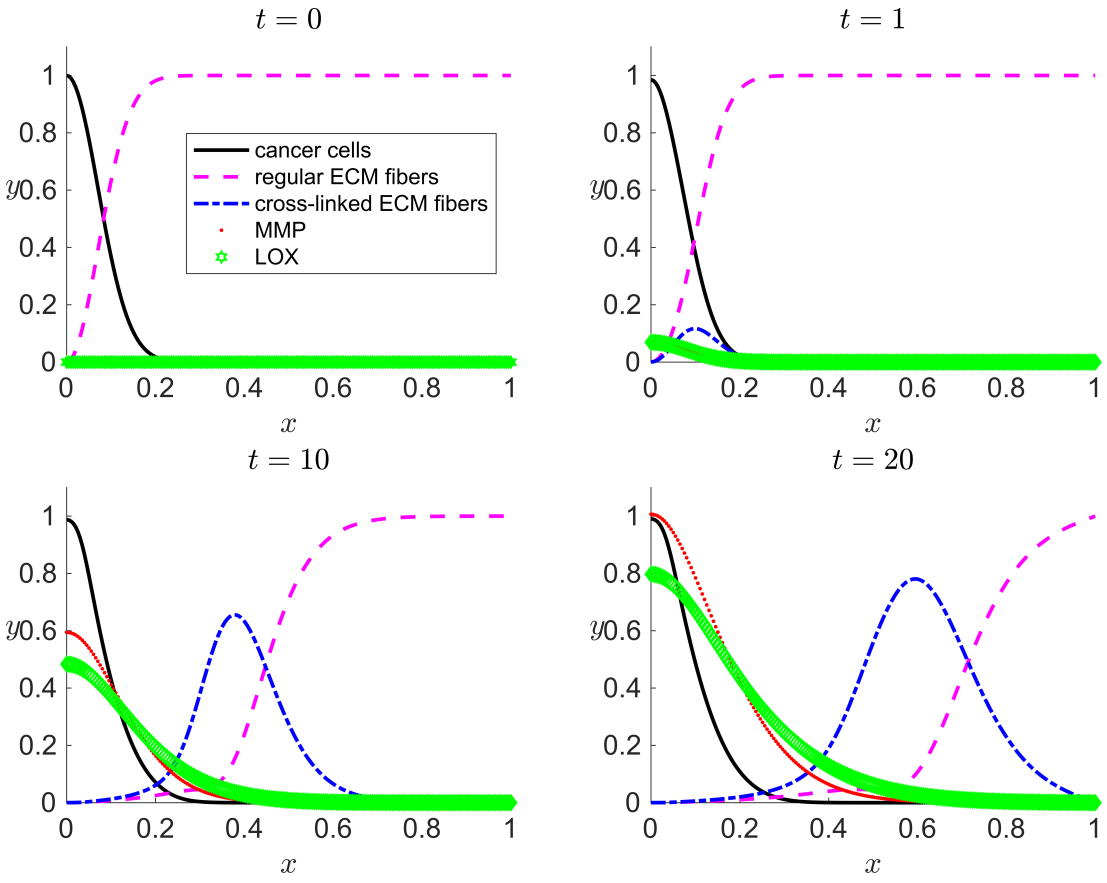

Figure 3.3: One dimensional numerical results for Case II when LOX ($l \not\equiv 0$) and its effect only on the ECM collagen fibers and not on the cancer haptotaxis ($g \equiv 0$) is considered in the modeled system. The results are snapshots of the system dynamics at four simulation times: (**A**) $t = 0$, (**B**) $t = 1$, (**C**) $t = 10$, and (**D**) $t = 20$. For all four plots, the horizontal axis, x, indicates a dimensionless spatial position, and the vertical axis, y, indicates the dimensionless population density or concentration of the species indicated in the legend.

The results demonstrate how crosslinked fibers enhance the overall migration of cancer cells.

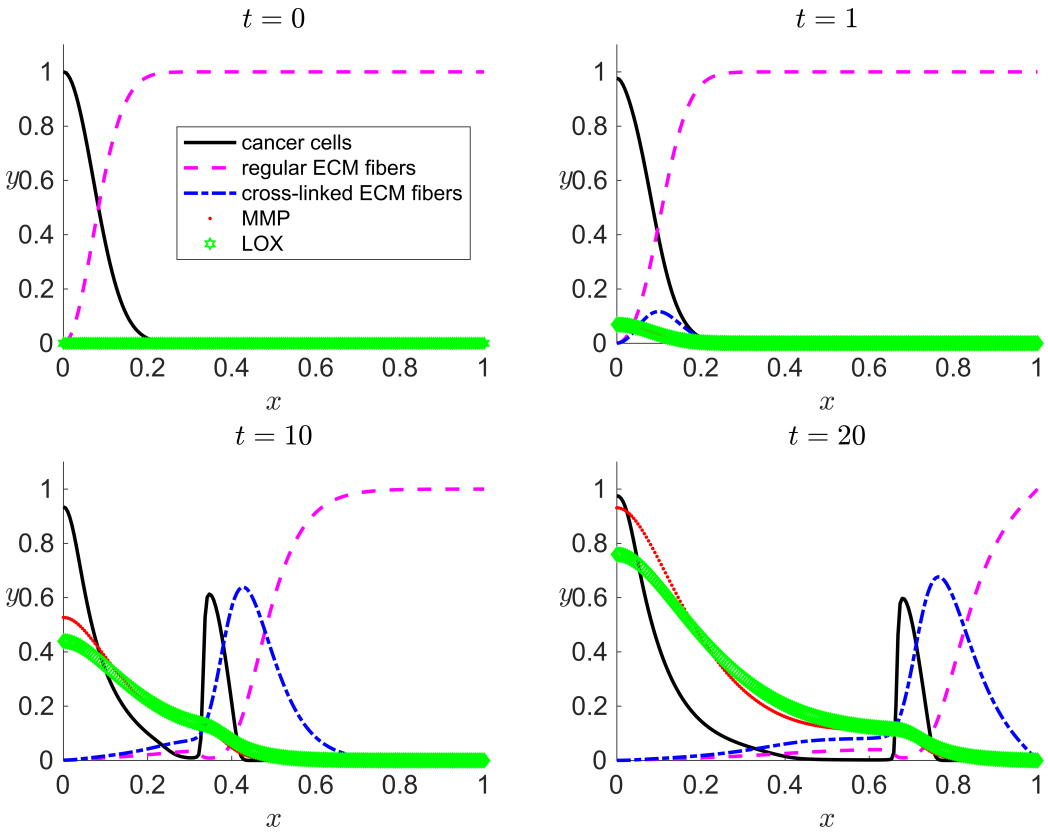

Figure 3.4: One dimensional numerical results for Case III when LOX ($l \not\equiv 0$) and its effects both on the ECM collagen fibers and cancer cells motility are considered in the modeled system. Results are snapshots of the system dynamics at four simulation times: (**A**) $t = 0$, (**B**) $t = 1$, (**C**) $t = 10$, and (**D**) $t = 20$. For all four plots, the horizontal axis, x, indicates a dimensionless spatial position, and the vertical axis, y, indicates the dimensionless population density or concentration of the species indicated in the legend.

3.6 Local Sensitivity Analysis

Local sensitivity assesses the impact of variations in each parameter on model outputs. The Case III conditions discussed in Section 3.5 are used. All of the dimensionless parameters listed in Table 3.2 are considered for the local sensitivity analysis, except for $\hat{v}_1$, $\hat{v}_2$, and $\hat{v}_3$. The nominal values for the parameters are those listed in Table 3.2. The sensitivity analysis is conducted by varying each parameter, P_j, at a time by a

small perturbation ΔP_j while keeping the other parameters fixed. We define the time-dependent local sensitivity index $S(t)$ defined as

$$S(t) = \frac{\max_x |Y_i(P_j + \Delta P_j, P_{k\neq j}, x, t) - Y_i(P_j, P_{k\neq j}, x, t)|}{\Delta P_j} \tag{3.25}$$

where $Y_i(P, x, t)$ is the model prediction of the output variable i at position x and time t evaluated at parameter set P. The parameters and model equations are all dimensionless, so Equation (3.25) is already normalized. Maximum deviations sorted with respect to x from the nominal case $Y_i(P_j, P_{k\neq j}, x, t)$ allow for the straightforward comparison of offsets over time due to parameter variations. ΔP_j is taken to be an increase of 10%.

We consider $S = 1$ to be a baseline threshold for categorizing parameter effects to be sensitive (i.e., a parameter is labeled "sensitive" if $S \geq 1$). In Figure 3.5, we show that, besides $\hat{D}_c$, $\hat{D}_m$, $\hat{D}_l$, $\hat{\alpha}_f$, and $\hat{\beta}_m$, the model outputs up to simulation time $t = 20$ are relatively insensitive to the other parameters. c is the output most sensitive to changes in input parameters, especially $\hat{D}_c$, $\hat{D}_m$, $\hat{D}_l$, $\hat{\alpha}_f$, and $\hat{\beta}_m$ (Figure 3.5A). The two local sensitivity curves of parameters $\hat{\beta}_m$ and $\hat{\alpha}_f$ consistently overlap each other, indicating that any changes in these two parameters will influence the model outputs with the same magnitudes over time (Figure 3.5A–C). Additionally, $\hat{\beta}_m$ and $\hat{\alpha}_f$ appear to impact the model outputs of c, f, and f_{cl} more profoundly than others. The outputs of m and l are the most sensitive to particularly the input parameters relating to chemical production by cancer cells $\hat{\beta}_m$ and $\hat{\beta}_l$, respectively (Figure 3.5D–E). From visual inspection of Figure 3.5F, it is clear that the maximum deviation from the baseline with respect to x occurs for $t = 20$ near $x = 0$, and that $\hat{\beta}_l$ is the most sensitive parameter for this scenario. The sensitivity index accounts for the same visual detection in a more concise metric. The maximum with respect to x is considered because the peaks may shift in different parameter variation scenarios compared to the locations of the peaks in Figure 3.4, making it difficult to choose an

x-position *a priori* at which the sensitivities can be fairly compared.

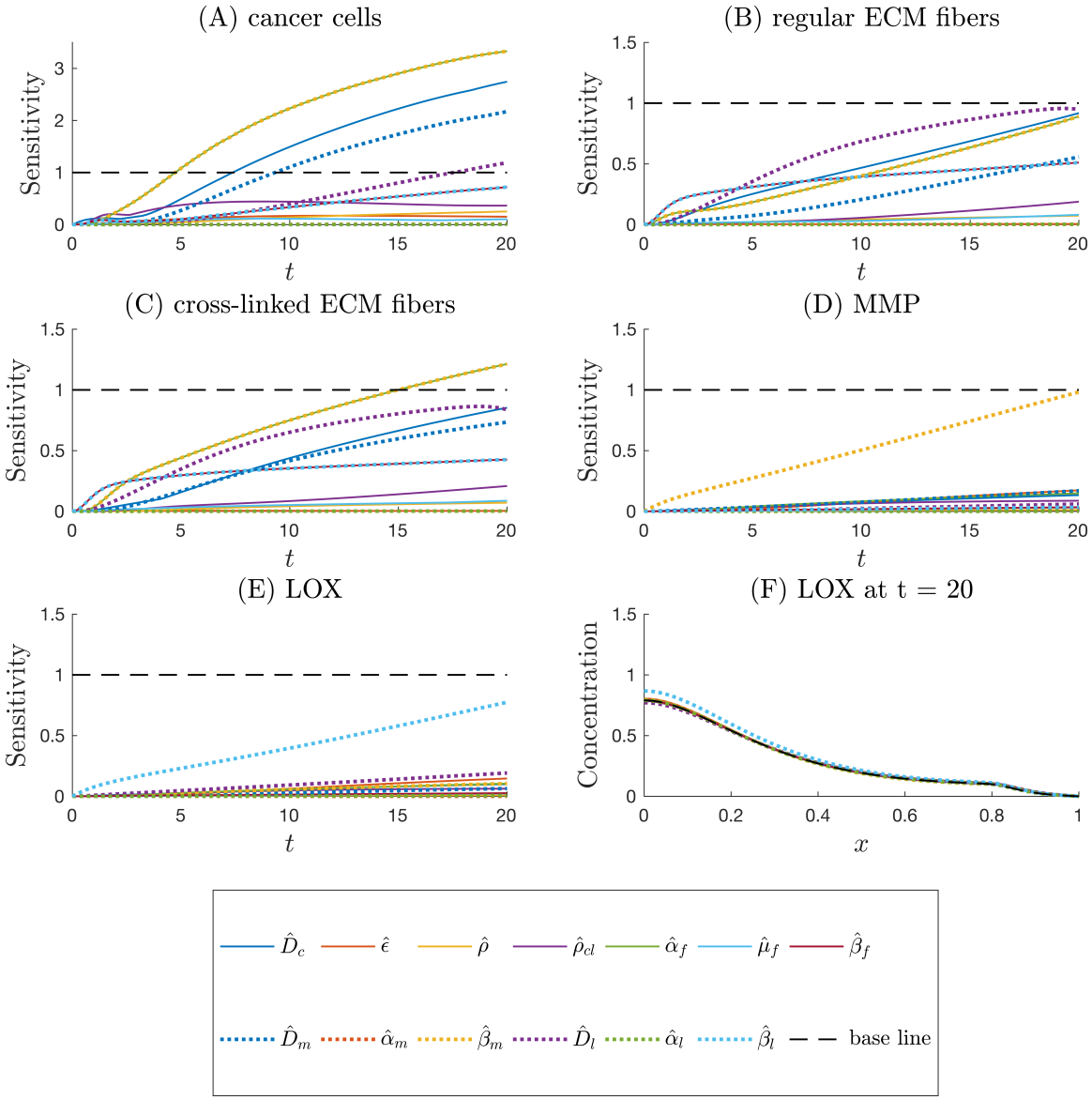

Figure 3.5: Local sensitivity index as a function of time assessing the impacts of 10% one-at-a-time increases in dimensionless parameters listed in Table 3.2 on the following model output variables: (**A**) the population density of cancer cells, (**B**) the concentration of regular ECM fibers, (**C**) the concentration of crosslinked ECM fibers, (**D**) the concentration of MMP, and (**E**) the concentration of LOX. (**F**) Model output profiles for concentration of LOX as a function of position x at at simulation time $t = 20$ corresponding to 10% changes in each parameter input. In (**A**–**E**), the baseline marks the threshold value of $S = 1$. In (**F**) the baseline marks the nominal LOX profile.

Chapter 4

Hybrid Continuum-Discrete Modeling via CompuCell3D of a Metastatic Tumor Microenvironment

In this chapter, we first introduce the algorithmic framework implemented behind the operation of a CompuCell3D simulation and the main features of the software package. CompuCell3D is the main tool we used to build a hybrid model for cancer migration in a metastatic tumor microenvironment. The simulation developed in this work consists of two types of interacting environments: (i) the discrete cell field representing cancer cells; (ii) the chemical fields representing the ECM collagen and the remodeling enzymes. Components in the chemical fields are modeled by continuous variables. Cancer cells are modeled as discrete entities residing and confined to a fixed 2D square lattice structure. The collective behavior and evolution of the cellular agent-based structures are modeled following the framework of the cellular Potts model, also called the Glazier-Granner-Hogeweg model in CompuCell3D. The focus of this work is on the quantitative study of how different ECM configurations in term of collagen fiber concentrations influence the spreading trajectories of cells, individually and collectively, as well as cell migration efficiency.

4.1 Introduction to CompuCell3D

CompuCell3D is a fully open-source multi-cell, multi-scale modeling environment. One of the advantages using this software is that it does not require a user to build sophisticated models from scratch, which also mean users are not required to have advanced knowledge in programming. A CompuCell3D project often consists

of CC3DML scripts (XML-based script) where cells' properties and behaviors are specified, Python scripts to determine cells' functions, and .piff files generated from CellDraw for the model's initial configuration. The biological aspects of the CompuCell3D simulation are implemented based primarily on the Glazier-Graner-Hogeweg (GGH) model. [80].

The Glazier-Graner-Hogeweg Model

Unlike other approaches to build physical models of tissues that ignore important properties and characteristics at a single cell level, GGH model is a discrete cell-based model. Due to its versatility and extensibility, the GGH model is able to not only define a single biological model but also incorporate multiple interacting processes and phenomenon at the cell level occurring at different length- and time-scales. The biological elements in a GGH model, called generalized cells, reside on a cell lattice which could be either two dimensional or three dimensional. Each generalized cell in a GGH model can be considered as a single biological cell, a group of cells of the same type, non-cellular components such as the ECM fibers, the diffusing chemical field or the medium fluid in the surrounding. A model could comprise several different cell types. Each cell type could be made up by one or more generalized cells. Each generalized cell is assigned a unique index and is occupied by a group of one or many pixels carrying the same index number as illustrated in Figure 4.1 [80]. .

The framework of a GGH model (Figure 4.2) utilizes the effective energy or Hamiltonian to define the thermodynamics of behaviors and interactions among the generalized cells in the simulation. Moreover, the GGH model follows the Metropolis algorithm to implement the stochastic changes in the cell evolution. The dynamics of any chemical signals involved is described via a set of reaction-diffusion partial differential equations (PDEs) being solved using one of the built-in PDE solver tools in CompuCell3D [80].

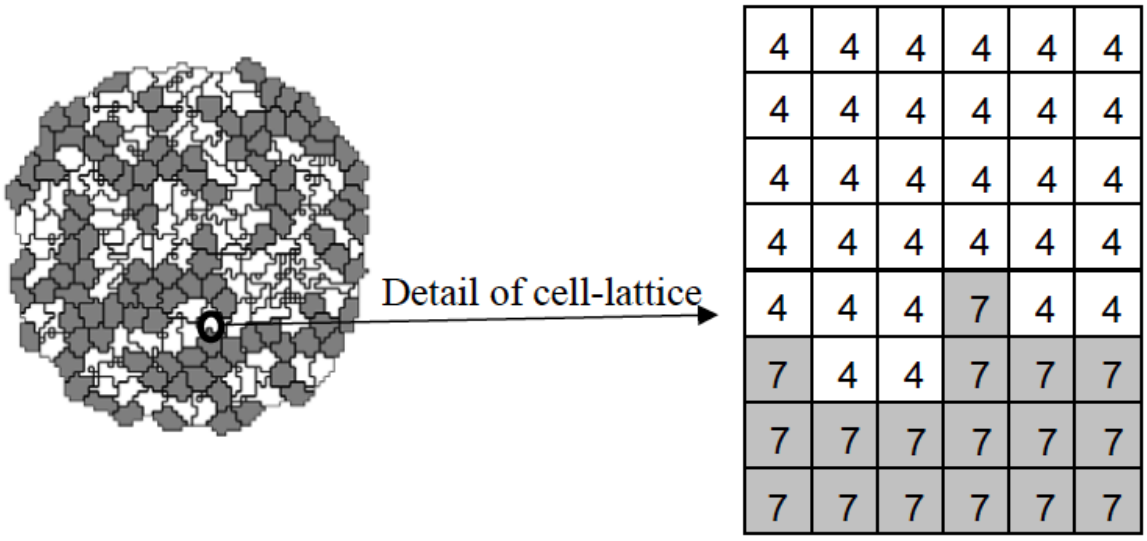

Figure 4.1: Example of a two-dimensional GGH cell-lattice configuration. Different color denotes different cell type, $J(\tau(\sigma_{\vec{i}})$. Each cell type encompasses one or more generalized cells represented by discrete entities, here in white or dark color. Each generalized cell carries a unique index, $\sigma_{\vec{i}}$, here 4 or 7, and is a composite of one or several pixels, i [80].

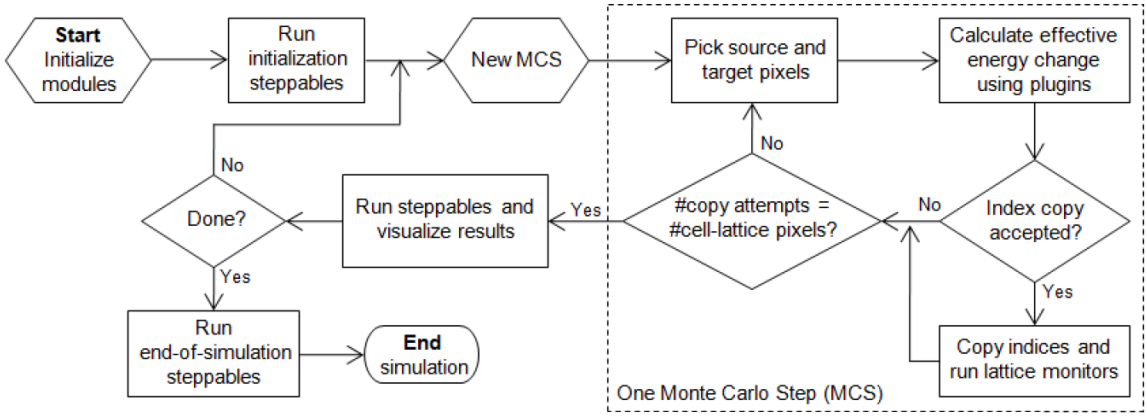

Figure 4.2: Flow chart of the GGH framework implemented in CompuCell3D [80].

The Hamiltonian in Term of Effective Energy

The effective energy is determined using the energy function Hamiltonian. This term is considered the heart of the GGH model. [80] noted that the term "energy" in the effective energy does not denote the actual physical energy of the cell but rather a convenient convention to implement a relatively desirable series of cell's properties and behaviors.

The equation for the Hamiltonian used in CompuCell3D is

$$H = \sum_{\vec{i},\vec{j}_{neighbors}} J\Big(\tau(\sigma_{\vec{i}}), \tau(\sigma_{\vec{j}})\Big)\Big(1 - \delta(\sigma_{\vec{i}}, \sigma_{\vec{j}})\Big) + \sum_{\sigma} \lambda_{vol}(\sigma)\Big(v(\sigma) - V_t(\sigma)\Big)^2 \quad (4.1)$$

Description of all variables and the two summation terms accounted in (4.1) are listed in Table 4.1.

Table 4.1: Hamiltonian variables and descriptions

Variable	Description
H	Effective energy
$\vec{i}$ and $\vec{j}$	Neighboring lattice sites
J	Contact energy for a pair of cells
$\sigma_{\vec{i}}$	Cell at site $\vec{i}$
δ	Kronecker delta
λ_{vol}	Inverse compressibility, behaves like a Young's modulus
$\tau(\sigma)$	Cell type of cell σ
$v(\sigma)$	Volume of cell σ
$V_t(\sigma)$	Target volume
Summation 1	Contact energy for adhesive interactions
Summation 2	Effective energy from volume constraint

Metropolis Algorithm

The stochastic, modified Metropolis algorithm is implemented in the GGH framework to perform the cell dynamics in the simulation. The algorithm calculates and assigns a probability of movement to a given cell site (or pixel). Specifically, showing in Figure

4.3 is one of the index-copy attempt examples that is applied every Monte Carlo step (MCS) to all the pixels present in the simulation lattice, changing from type $\vec{i}$ to type $\vec{j}$. In the piecewise function shown in (4.2)), the probability of the index-copy attemptl occurs with a probability of 100% if the copied pixel goes from the site of higher to lower H, which results in a negative change in system effective energy ΔH. Otherwise, the probability reduces exponentially depending on the cell-membrane fluctuation T_m (a fixed temperature-like parameter) specified in the XML file (see Appendix B.1) [80]. T_m represents the probability of acceptance by the system for an unfavorable thermodynamic move. From the expression defined in (4.2), higher T_m will result in higher rate of acceptance of any move while lower T_m will be more applicable for a deterministic model. which might cause the move to be trapped at the local minima [50]. The stochastic, modified Metropolis algorithm is given as

$$P(\sigma_{\vec{i}} \rightarrow \sigma_{\vec{j}}) = \begin{cases} 1 & \Delta H \leq 0 \\ \exp\left(\frac{-\delta H}{Tm}\right) & \Delta H > 0 \end{cases} \tag{4.2}$$

where P is the probability of cell movement, ΔH is the effective energy change of the system for every attempt of index-copy activity, and T_m is the amplitude parameter of cell-membrane fluctuations.

Python Steppable

Steppables are CompuCell3D modules written in Python and are called at a fixed interval of MCS (specified by users) during the a simulation. The built-in steppables are mainly used to tailor cell parameters for adapting and responding to simulation events, to solve PDEs, to customize simulation initial conditions, or to log simulation data. Beyond the built-in steppables, CompuCell3D allows users to write their own steppable(s) to account for distinct functions or phenomena that occur to certain cell types or chemicals. This feature is what makes CompuCell3D a versatile tool

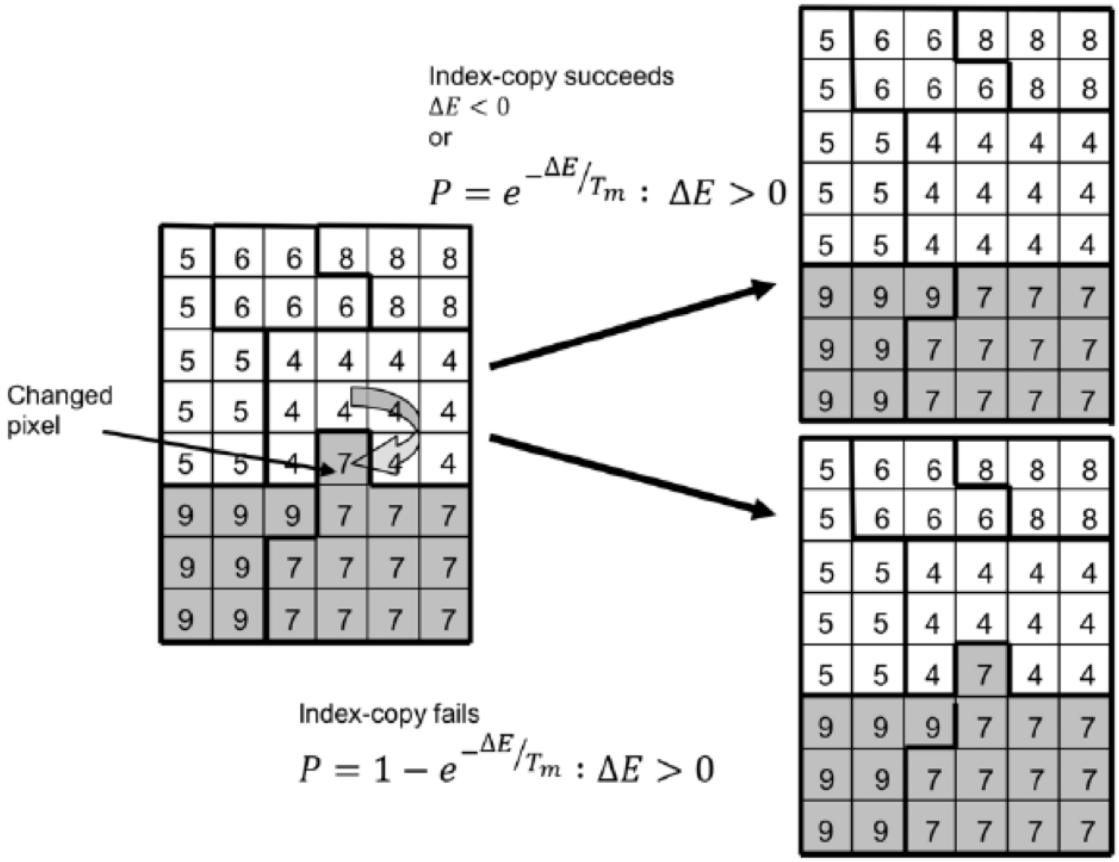

Figure 4.3: GGH representation of an index-copy attempt for two cell site on a 2D square lattice in CompuCell3D. The pixel in white color (source) attempts to replace the pixel in grey (target). The probability of accepting the index copy is given by (4.2) [80].

and attractive to users with or without strong experience in programming who might want to build a simple biological model or a more advanced and complicated one that requires flexibility from user-defined coding [80].

The Main CompuCell3D Environments for Users

Besides the computational kernel of CompuCell3D, the software package provides three main user-friendly environments including: (i) Twedit++CC3D, a code generator and editor platform; (ii) CellDraw, where users can customize the shape and compose the cell lattice as desired; (iii) CC3D player, a graphical user interface that serves the purpose of running, replaying, and analyzing simulations.

Table 4.2 records the system of units suggested by CompuCell3D for properties specified in the XML file.

Table 4.2: Fundamental properties and units in Compucell3D

Property	**Unit**
Mass	10^{-15} kg
Length	10^{-6}m
Time	sec
Volume	10^{-18}m^3
Lambda volume	$10^{9}\text{kg}\cdot\text{m}^{-4}\text{sec}^{-2}$
Surface area	10^{-12}m^2
Energy	$10^{-27}\text{kg}\cdot\text{m}^2\text{sec}^{-2}$
Contact energy	10^{-15} kg-sec^{-2}
Chemotaxis strength	$10^{-27}\text{kg}\cdot\text{m}^2\text{sec}^{-2}$
Diffusion coefficient	$10^{-12}\text{m}^2\text{sec}^{-1}$
Rate constant for decay	sec^{-1}
Rate constant for secretion	sec^{-1}
Rate constant for uptake	sec

4.2 Simulation Setup

Cell Size and Lattice Dimension

The lattice size of a CC3D model is assigned based on the spatial resolution of the system being modeled. We start by setting a length scale of 2 μm for each pixel.

Each generalized cell in our CC3D model is treated as one biological tumor cell. The diameter of each cell is reported to be approximately 20 μm [77]. Hence, we set the width of each generalized cell to 10 pixels corresponding to 20 μm. The target cell size in the simulation is anticipated to be $400\mu\text{m}^2$ according to [50]. The simulated tumor mass is a blob of multiple generalized tumor cells congregated in the center of the simulation lattice in which the initial tumor centroid is positioned at (150,150) as shown in the red border in Figure 4.4. The blob of tumor cells in the simulation as a whole called a tumor mass is specified with an initial radius of 50 pixel. The simulated tumor mass in our case is made up of approximately 69 generalized tumor cells. This design makes our simulated tumor seems small compared to a realistic tumor since the number of cells encompassed within one tumor could range from thousands to millions of cells [77]. Clinically, tumors vary in shape and size. However, tumor size matters when it comes to planning surgeries or treatment, whereas the characteristics and migrating behavior of individual tumor cells, especially once they start metastasizing, do not seem to be much impacted by the size of the tumor itself [78, 69]. Additionally, our model considers a 2D cross-section of a tumor that has hundreds of cells in 3D.

Most malignant tumor cells are located at a distance of at most 200 μm away from the blood vessel [82]. The tumor cells reside near blood vessels to get access to nutrients, or else the tumor cells either remain quiescent (dormant) or result in necrosis (death) due to the lack of the blood supply. Hence, we choose a lattice size of 300×300 pixels^2 to ensure that each cell is able to travel a distance of at least 100 pixels which is equivalent to 200 μm before reaching the lattice boundaries.

Cell Motility and Time Scale

Regarding the cell migration speed, one approach considered by [81] and [50] in order to relate simulation step from MCS to real time is to compare and match

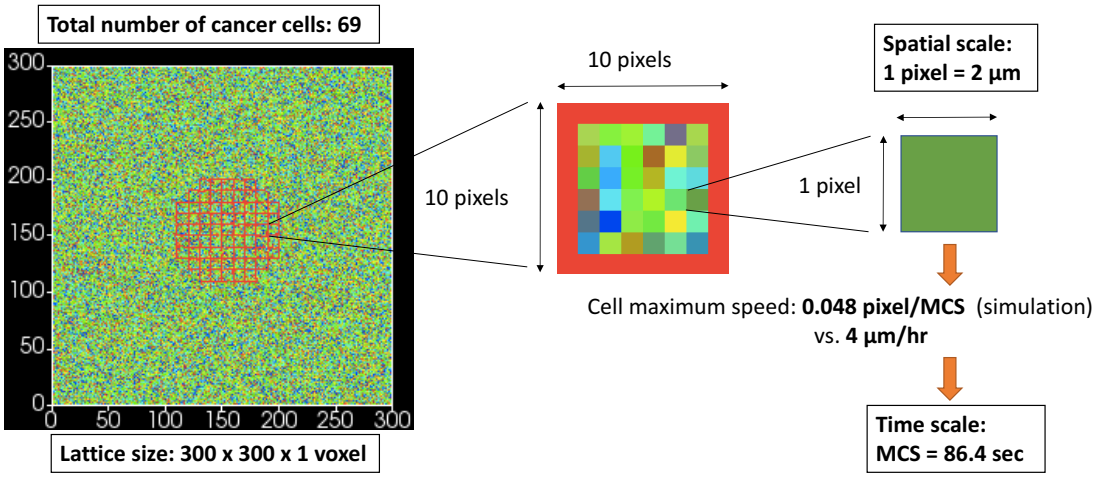

Figure 4.4: Spatial and temporal scales setting for the initial configuration of the simulated tumor (in red borders) in CompuCell3D.

the migration speed of cells obtained from experiments with the speed measured from the *in silico* generalized cells. Based on experiments done by [91], the migration speed for a tumor cell could vary from 2 to 12 μm/hour. To ensure the level of statistical significance, at least 10 simulations were performed repeatedly using a tentative set of parameter values and without the haptotaxis effect (which is known as the chemotaxis plugin in CC3D manuals) turned on. We recorded a maximum migration speed of 0.048 ± 0.00618 pixel/MCS for a simulated cancer cell. Matching this recorded migration speed of 0.048 pixel/MCS to 4 μm/hr reported in [91], each MCS corresponds to a time scale of 86.4 sec. From this new determined time scale, a new set of simulated parameter values are calculated and are reported in Table 4.3. These calculated parameters are mostly related to diffusion and reaction properties of chemical components accounted for in the simulation including the uncrosslinked and crosslinked fibers and enzymes MMP and LOX. Re-running the simulation for at least 10 more times with the new adjusted set of parameter values, a maximum cell speed of 0.047 ± 0.00270 pixel/MCS was recorded confirming that the migration

speed converged.

Chemical Fields

The chemicals involved in the hybrid model are similar to the continuous model, which includes the uncrosslinked and crosslinked collagen fiber in the ECM and enzymes MMP, and LOX. Their reaction-diffusion models remain the same as (3.3)–(3.7) introduced in Chapter 3. The PDEs are solved using the ReactionDiffusionSolver –one of the PDE solvers provided within CompuCell3D that is able to solve a system of N coupled PDEs of the form

$$\frac{\partial c_1}{\partial t} = D_1 \nabla^2 c_1 + f_1(c_1, c_2, ..., c_N) \tag{4.3}$$

$$\frac{\partial c_2}{\partial t} = D_2 \nabla^2 c_2 + f_2(c_1, c_2, ..., c_N) \tag{4.4}$$

$$\cdots$$

$$\frac{\partial c_N}{\partial t} = D_N \nabla^2 c_N + f_N(c_1, c_2, ..., c_N) \tag{4.5}$$

where c_j with $j = 1, 2, ..., N$ represents the concentration of the chemical j, D_j is the diffusion coefficient of chemical j, and f_j is any functions with coupled variables involving any set of the chemical species in the system.

In case of large diffusion coefficients and an unstable method being used within the ReactionDiffusionSolver, instability checks are performed for the equations of MMP and LOX. Both have the calculated value of $D\frac{\Delta t}{\Delta x^2}$ (where D is the diffusion coefficient of each chemical, t is the time step, and x is the spatial step) exceed the instability threshold of 0.25 for the 2D model. Hence, adjusting the step size is necessary for the PDE solvers of each chemical term. This task can be done by adding to each PDE solver in the XML file N number of <ExtraTimesPerMCS> (see Appendix B.1), where N satisfies $D\frac{\Delta t/N}{\Delta x^2} < 0.25$ [80]. The larger value calculated for N, which

is 175 for the case of the LOX diffusion field, is used to rescale the step size of the ReactionDiffusionSolver for the other chemical fields as well. We did not have to perform instability checks for the equations for uncrosslinked and crosslinked fibers because their diffusion constants are assumed to be zero since fibers do not diffuse.

Haptotaxis Strength toward Fields of Uncrosslinked and Crosslinked Fibers

It has been indicated that haptotaxis plays a vital role in the ability of cancer cells to metastasize [45]. Haptotaxis is a directional behavior of motile cells up a gradient of the chemoattractant that is bound on a surface, which are the concentrations of uncrosslinked and crosslinked fibers in our case. We implement haptotaxis in our simulation via the chemotaxis plugin in CompuCell3D. This plugin evaluates the change in the system effective energy ΔH_{chem} for pixel-copy attempts associated with chemotactic motility. ΔH_{chem} is defined in CompuCell3D

$$\Delta H_{chem} = -\lambda\Big(c(\vec{x}) - c(\vec{x}_{source})\Big) \tag{4.6}$$

where $\lambda > 0$ is the chemotaxis strength, or haptotaxis strength in our case, $c(\vec{x})$ and $c(\vec{x}_{source})$ indicate the chemical concentrations at the destination and source pixels, respectively, during the pixel-copy attempts [80]. Accounting for the effective energy due to chemotactic motility into the Hamiltonian of the GGH model in (4.1), the total system energy calculated for each pixel-copy attempt now becomes

$$H = \sum_{\vec{i},\vec{j}_{neighbors}} J\Big(\tau(\sigma_{\vec{i}}), \tau(\sigma_{\vec{j}})\Big)\Big(1 - \delta(\sigma_{\vec{i}}, \sigma_{\vec{j}})\Big) + \sum_{\sigma} \lambda_{vol}(\sigma)\Big(v(\sigma) - V_t(\sigma)\Big)^2 - \sum_{\vec{i}} \lambda c_{\vec{i}} \tag{4.7}$$

where $c_{\vec{i}}$ is the chemical concentration at pixel $\vec{i}$

Simulations were tested at different haptotaxis strength (λ) varied from 0, 50, 100, 200, 500, 1000, 2000, and 5000 to determine the baseline of haptotaxis strength where the direction of cell motility under influence of haptotaxis dominates the random

motion effect. Again for the purpose of statistical significance, 10 repeated simulations were run up to 10000 MCS for each value of haptotaxis strength used. For λ values of 50, 100, and 200, the visual results (not shown here) from simulations are not significantly different from the results without haptotaxis in which $\lambda = 0$. Specifically, these observations include that the tumor mass remains intact and none of the cells show an intention of moving toward the haptotactic zones of fibers. This indicates that a λ value of equal to or less than 200 is not high enough for haptotaxis to influence the cell motility in our case. On the other hand, for λ values 500 –5000, at around 5000 MCS, an outward dissemination of cells away from the center of the tumor mass is clearly noticed (Figure 4.5). In conclusion, a haptotaxis strength $\lambda = 500$ is chosen to be the baseline for the cell haptotactic motion to prevail over the free motion of cells in the simulation.

In Figures 4.6 and 4.7, the simulations were conducted to confirm that haptotaxis is performed stronger toward crosslinked fibers than toward the uncrosslinked ones. The haptotaxis tests were performed under the two different cases of initial distributions of uncrosslinked and crosslinked fiber concentrations: randomly distributed in Figure 4.6 and uniformly in Figure 4.7. The initial average dimensionless concentration of both uncrosslinked and crosslinked fibers in both types of distributions were set to 0.5. Most of the parameter values used in the simulations followed the same conditions specified in Table 4.3, except that the rate constants for production of fibers and LOX crosslinking of fibers were set to 0. The reason for turning off these two terms in this test is to assure that half of the simulation domain was purely crosslinked fibers on the left and the other half of the simulation domain is purely uncrosslinked fibers on the right.

Qualitatively, we see that over time more cells are migrating to the left side of the domain, which is the crosslinked fibers side. This phenomenon is much more profound in Figure 4.7 where the initial concentration distribution of fibers is uniform.

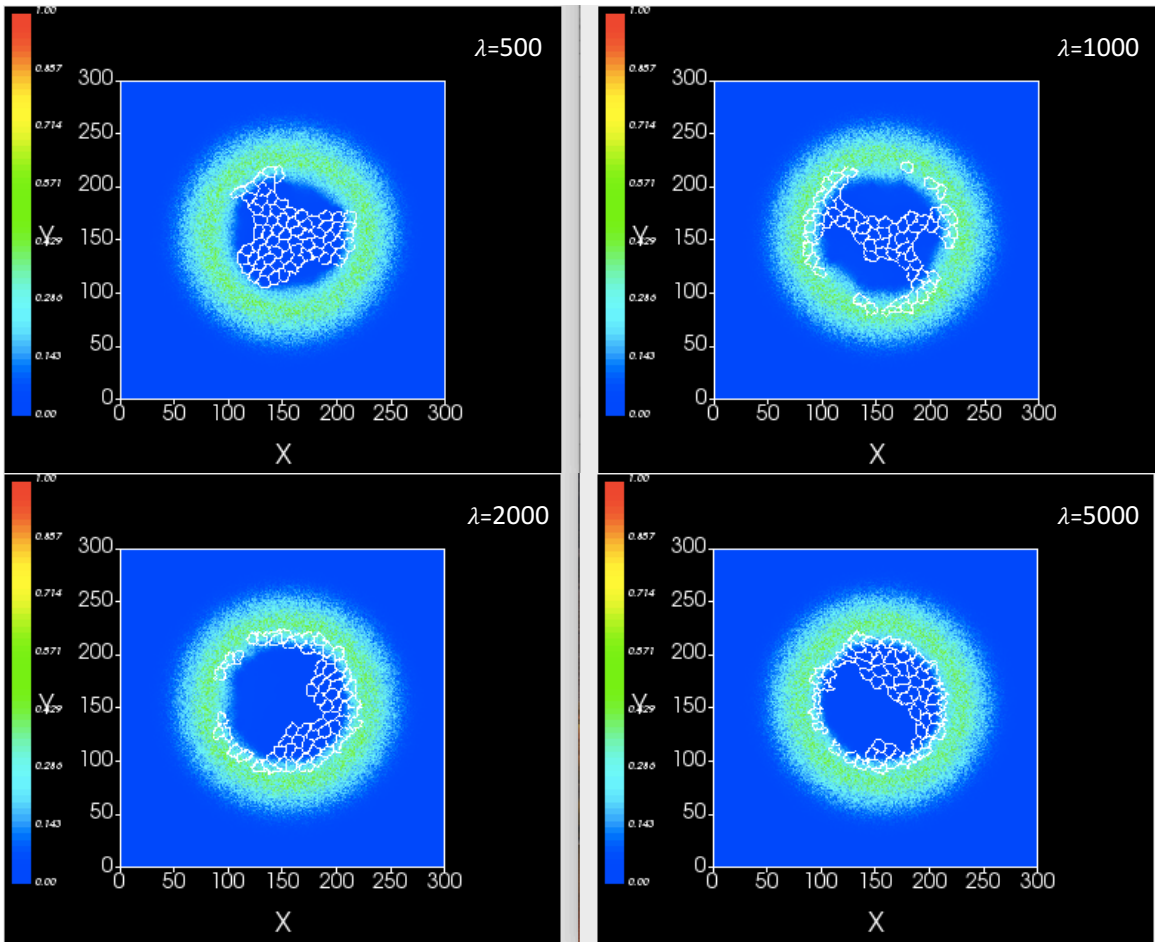

Figure 4.5: Representative snapshots of the crosslinked fiber field at MCS = 5000 from four different simulations for varying values of haptotaxis strength λ: 500, 1000, 2000, and 5000.

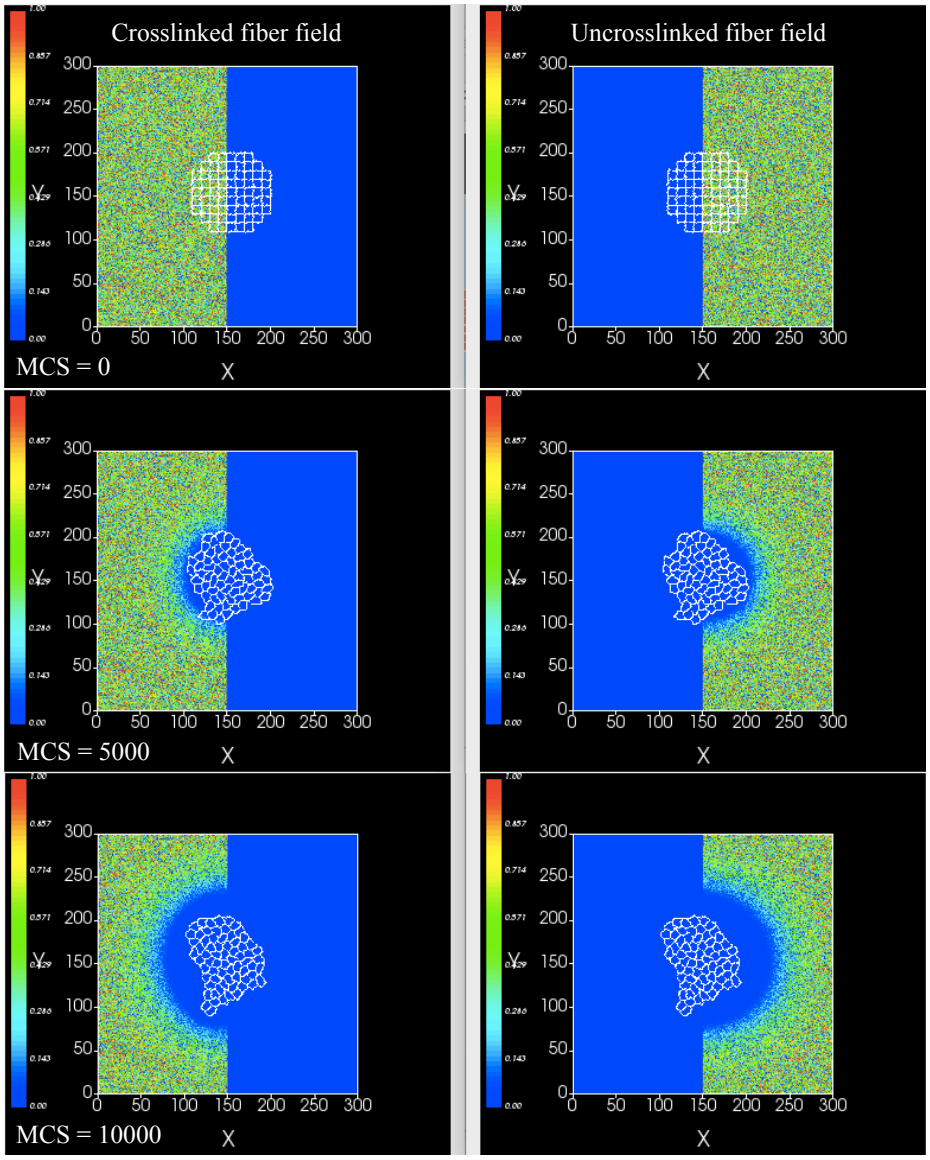

Figure 4.6: Simulation snapshots of the crosslinked fiber field (left) and uncrosslinked fiber field (right) at 0, 5000, 8000 MCS (top to bottom) testing the effect of haptotaxis for an initial randomly distributed crosslinked and uncrosslinked fibers restricted to the left and right halves of the domain, respectively.

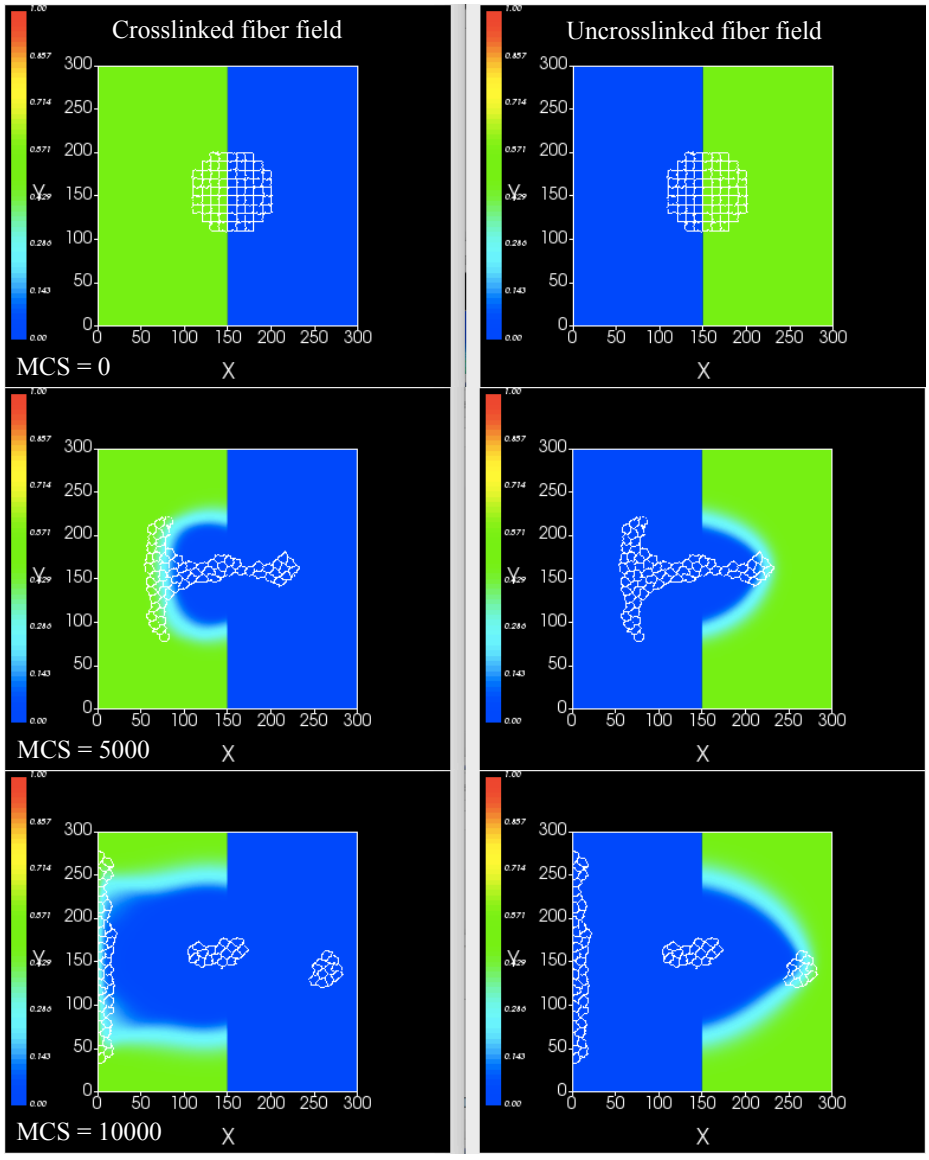

Figure 4.7: Simulation snapshots of the crosslinked fiber field (left) and uncrosslinked fiber field (right) at 0, 5000, 8000 MCS (top to bottom) testing the effect of haptotaxis for an initial uniformly distributed crosslinked and uncrosslinked fibers restricted to the left and right halves of the domain, respectively.

4.3 Partial Nondimensionalization and Parameter Estimation

For the hybrid model introduced in this chapter, non-dimensionalization is performed for every chemical species in the reaction-diffusion PDEs for fiber, crosslinked fiber, MMP, and LOX in (3.3)–(3.7), respectively, following similar methods as in Section 3.2 of Chapter 3. We leave the time (t) and spatial (x) terms in units of MCS and pixels, respectively. Hence, the subset of dimensionless variables used from (3.8) is

$$\tilde{f} \equiv \frac{f}{f_o}, \quad \tilde{f_{cl}} \equiv \frac{f_{cl}}{f_o}, \quad \tilde{m} \equiv \frac{m}{m_o}, \quad \tilde{l} \equiv \frac{l}{l_o} \tag{4.8}$$

Introducing the dimensionless quantities defined in (4.8) into (3.3)–(3.7), the partial dimensionless reaction diffusion PDEs for all chemical fields utilized for CompuCell3D model can be re-written. For the uncrosslinked fiber,

$$\frac{\partial \tilde{f}}{\partial t} = -(\alpha_f m_o)\tilde{m}\tilde{f} + \left(\frac{\mu_f}{f_o}\right)\left(1 - (v_1 c_o)\tilde{c} - (v_2 f_o)\tilde{f} - (v_3 f_o)\tilde{f_{cl}}\right) - \tilde{h} \tag{4.9}$$

where $\tilde{c}$ represents the state of physical contact with the discrete cancer cell in the lattice. $\tilde{c} = 1$ if pixel x belongs to the cancer cell type which is cell type 1 in our case (see Appendix B.1) . Otherwise, $\tilde{c} = 0$ if pixel x is the Medium.

The remainder of the chemical field reaction-diffusion PDEs follow as

$$\frac{\partial \tilde{f_{cl}}}{\partial t} = -(\alpha_f m_o)\tilde{m}\tilde{f_{cl}} + \tilde{h} \tag{4.10}$$

$$\tilde{h} = (\beta_f l_o)\tilde{f}\tilde{l} \tag{4.11}$$

$$\frac{\partial \tilde{m}}{\partial t} = D_m \frac{\partial^2 \tilde{m}}{\partial x^2} - \alpha_m \tilde{m} + \left(\frac{\beta_m c_o}{m_o}\right)\tilde{c} \tag{4.12}$$

$$\frac{\partial \tilde{l}}{\partial t} = D_l \frac{\partial^2 \tilde{l}}{\partial x^2} - \alpha_l \tilde{l} + \left(\frac{\beta_l c_o}{l_o}\right)\tilde{c} \tag{4.13}$$

The partial dimensionless parameters that emerge within parentheses are defined

in Table 4.3. Values for these partial dimensionless parameters are obtained through calculation from information reported in [57], which are also Table 3.1 and 3.2 of this document.

Table 4.3: Parameters values used in CompuCell3D model of cancer migration under the haptotaxis effect from a remodeling ECM

Property	**Value**	**Unit/ID**	Comment
Lattice dimension	$300 \times 300 \times 1$	pixel^3	
Length per pixel	2	μm	
Simulation time	10000	MCS	
Boltzmann energy (kT)	10^{-27}	$\text{kg} \cdot \text{m}^2\text{s}^{-2}$	Assumed
Cell membrane fluctuation (T_m)	50 kT		[81]
Pixel copy neighbor order	2		[80]
Adhesion contact neighbor order	2		[80]
Number density of cells	1.25×10^9	$\text{cells} \cdot \text{cm}^3$	Calculated from [50]
Radius of the tumor mass	50	pixel	
Size per generalized cell	$10 \times 10 \times 1$	pixel^3	
Cell migration speed	4	$\mu\text{m} \cdot \text{hr}^{-1}$	[91]
	0.048	$\text{pixel} \cdot \text{MCS}^{-1}$	Determined
Real time per MCS	86.4	sec	Calculated
Target cell size	400	μm^2	[50]
	100	pixel^2	Calculated
Volume constraint of a cell	1 kT/L^6		[50]
Contact energy (J)			
medium-medium	0 kT/L^2		[80]
medium-cell	32 kT/L^2		[8, 50]
cell-cell	40 kT/L^2		[8, 50]
Chemotaxis strength toward regular fiber	500 kT		Baseline of haptotaxis strength
Chemotaxis strength toward crosslinked fiber	1000 kT		Assumed
Diffusion coefficient of all fibers	0	$\text{pixel}^2\text{MCS}^{-1}$	Fibers do not diffuse
Rate constant for natural decay of all fibers	0	MCS^{-1}	Assumed
Rate constant for MMP cleavage of fiber $\tilde{\alpha}_f = \alpha_f m_o$	7.5×10^{-3}	MCS^{-1}	Calculated from [4, 57]
Rate constant for LOX remodeling of fiber $\tilde{\beta}_f = \beta_f l_o$	1.35×10^{-2}	MCS^{-1}	Calculated from [57]
Rate constant for production of fiber $\tilde{\mu}_f = \mu_f / f_o$	1.13×10^{-4}	MCS^{-1}	Calculated from [1, 57]
Space fraction per unit volume of fiber concentration $\tilde{v}_2 = v_2 f_o$	1		Calculated from [35, 57]
Diffusion coefficient of MMP D_m	21.6	$\text{pixel}^2\text{MCS}^{-1}$	Unit conversion from [40, 57]
Rate constant for decay of MMP $\tilde{\alpha}_m = \alpha_m$	7.5×10^{-7}	MCS^{-1}	Calculated from [50, 57]
Rate constant for secretion of MMP by cell $\tilde{\beta}_m = \beta_m c_o / m_o$	7.5×10^{-5}	MCS^{-1}	Calculated from [49, 57]
Diffusion coefficient of LOX D_l	43.2	$\text{pixel}^2\text{MCS}^{-1}$	Assumed
Rate constant for decay of LOX $\tilde{\alpha}_l = \alpha_l$	7.5×10^{-7}	MCS^{-1}	Assumed
Rate constant for secretion of LOX by cell $\tilde{\beta}_l = \beta_l c_o / l_o$	7.5×10^{-5}	MCS^{-1}	Assumed

4.4 Initial and Boundary Conditions

Initially, the ECM is a mesh of randomly oriented collagen fibers that are not yet crosslinked. Hence

$$\tilde{f}_{cl}(x, y, 0) = 0 \tag{4.14}$$

We also assume a zero concentration of MMP and LOX present in the lattice

initially

$$\tilde{m}(x, y, 0) = 0 \tag{4.15}$$

$$\tilde{l}(x, y, 0) = 0 \tag{4.16}$$

We apply Dirichlet-type boundary conditions along all the x and y boundaries. Constant values of zero concentrations of crosslinked fibers, MMP, and LOX are assumed at all the four walls of simulation boundaries. Meanwhile, suppose beyond the simulation boundaries, there are blood vessels that surround the tumor microenvironment. The outer layer of blood vessels consists of mostly collagen fibers. Hence, we assume there is a constant nonzero amount of uncrosslinked ECM fibers, $\tilde{f}$, residing at all the simulation boundaries. Hence, the set of boundary conditions imposed on each chemical field in the simulation is taken as

$$\tilde{f}_{cl}(x = 0, x = 300, y = 0, y = 300, t) = 0 \tag{4.17}$$

$$\tilde{m}(x = 0, x = 300, y = 0, y = 300, t) = 0 \tag{4.18}$$

$$\tilde{l}(x = 0, x = 300, y = 0, y = 300, t) = 0 \tag{4.19}$$

$$\tilde{f}(x = 0, x = 300, y = 0, y = 300, t) = 1 \tag{4.20}$$

4.5 Methods of Quantifying Cell Migration

Following similar methods used in [50] to quantify the migration of the cancer cells in the simulations, two different methods of distance measurements are considered in our work: the Euclidean distance and the gyradius, or also known as the radius of gyration. Each simulation being analyzed is run for a total of 10000 MCS $\approx$ 10 days.

Euclidean distance is applied to determine the average displacement for a population of 69 simulated cancer cells over a time course of 10000 MCS. The displacement of a cell from its initial position at MCS = 0 also suggests how persistently on its mi-

gration path the migrating cell maintains a certain direction. The population average displacement ($\bar{D}$) is given as

$$\bar{D} = \frac{1}{N}\sum_{j=1}^{N}\sqrt{(x_{j,10000} - x_{j,0})^2 + (y_{j,10000} - y_{j,0})^2} \tag{4.21}$$

where N is number of cancer cells present in the simulation domain, which is 69 in our case. The two terms ($x_{j,time}$,$y_{j,time}$) indicate the centroid position (also called the center of mass in CompuCell3D) of the j^{th} cell at a given MCS here, the initial time is chosen as MCS = 0 and the selected final time is MCS = 10000.

In cell migration, the gyradius quantity (R_g) characterizes the degree to which the cell population has spread from the initial tumor centroid. R_g is evaluated at the end of the simulation at 10000 MCS. The gyradius for a tumor mass of 69 tumor cells with respect to the center of initial population at a given time is defined as

$$R_g = \sqrt{\frac{1}{N}\sum_{j=1}^{N}(x_{j,10000} - x_{mean})^2 + (y_{j,10000} - y_{mean})^2} \tag{4.22}$$

where (x_{mean},y_{mean}) represents the initial tumor centroid position which is (150,150) in our case.

4.6 Results

Random Distribution of Fiber Concentration Diminishes the Invasion Rate

Multiple physical properties of the ECM including stiffness, crosslinking density, pore size, and alignment of fibers have been indicated as important factors governing the motility of cells through the matrix [30]. In the simulations testing the haptotactic movement of cancer cells up gradients toward the uncrosslinked and crosslinked fibers (Figures 4.6 and 4.7), the invasion extent of the disseminated cells is greater in the uniform ECM than in the random ECM. Hence, in this section, we conducted sim-

ulations for varying between two different distribution modes of fiber concentration, random and uniform, in the 2D lattice to test if our model can recapitulate the effect of ECM fiber distribution on the spreading capability of cells. Both simulations with the random and uniform ECM had initial fiber dimensionless concentrations with the average value of 0.5. To implement this in the model, we set a value 0.5 everywhere for the initial condition of fiber concentration in uniform ECM and a random value in the range $[0, 1]$ for concentration in each fiber pixel for the initial condition of random ECM. Properties of cancer cells and the tumor mass as well as the other parameters related to the continuous variables were kept fixed at values in Table 4.3. For each condition, simulations were run for 10000 MCS and repeated 10 times for statistics. Data relating to positions of every cancer cell were collected at the initial time when MCS = 0 and at the final time when MCS = 10000 ($\approx$ 10 days).

Comparing the system evolution in Figures 4.8 and 4.9, in both of the ECM conditions, cancer cells follow a similar invasion pattern of an arc shape as they disperse in multiple directions. Since the contact energy between cells specified in Table 4.3 is set quite high ($J_{cell-cell} = 40$) indicating a weak cell adhesion, it is anticipated that the cell invasion mode is going to result in a small cluster of cells or individual cells. Rate of invasion appears to be higher in the case of uniform ECM (Figure 4.9) in comparison to random ECM (Figure 4.8). After 10000 MCS, the majority of the disseminated cells in Figrue 4.9 have already reached the lattice boundaries while those in Figure 4.8 are still a short distance away from reaching the boundaries. Notice that higher invasion rate does not imply higher cell speed. Every cell has different migration trajectories. One that moves actively might not necessarily invade effectively as it is possible for a cell to move back to its previous position. To quantitatively verify our observations, the invasion distance and the scattering level of cells are quantified by the average displacement ($\bar{D}$) and the gyradius (R_g), respectively. Results shown in Figure 4.10 confirms that a greater migration (higher $\bar{D}$ and R_g)

is demonstrated in the condition where the concentration of ECM fibers is uniformly distributed. Also, the numerical result from Figure 4.10 for cell displacement $\bar{D}$ in random ECM is found to agree to a certain extent with the experimental data for cell invasion depth over 11 days of the SW620 colon cancer cell line reported from [75] shown in Figure 4.11. Hence, we recommend the model of random ECM with the initial condition of fiber concentration distributed randomly in the range $[0, 1]$ with other properties reported in Table 4.3 as the nominal case for further study beyond the scope of the present thesis.

Increasing Fiber Concentration Enhances Migration Efficiency

Given that the way fiber concentration is distributed in the domain does have an influence on cell movement, we propose that fiber concentration is another ECM attribute regulating cell motility, beside other attributes of stiffness, crosslink density, pore size, and alignment of fibers [30] as mentioned earlier. Varying fiber concentration results in changing in pore size, which has been shown to be related to cell migration [26]. Hence, in the next test, for each ECM distribution mode, random or uniform, the initial condition of fiber concentration average value was set to 0.25, 0.5, or 0.75. In the case of random ECM, the selected average values of fiber concentrations 0.25, 0.5, and 0.75 are generated by keeping width of the distribution the same while varying the random intervals of fiber concentration in the ranges $[0, 0.5]$, $[0.25, 0.75]$, and $[0.5, 1]$, respectively. Similar procedures as in previous tests were followed. 10 repeated simulations were run to 10000 MCS, and cell's positions were collected, and analyzed to quantify the migration with $\bar{D}$ and R_g for each condition. As a result, though the effect of varying fiber concentration might not be straightforwardly demonstrated in Figures 4.12 and 4.13, the quantitative results in Figures 4.14 and 4.15 clearly indicate that in both randomly and uniformly distributed ECM initial fiber condition, an increasing in fiber concentration also leads to a higher mi-

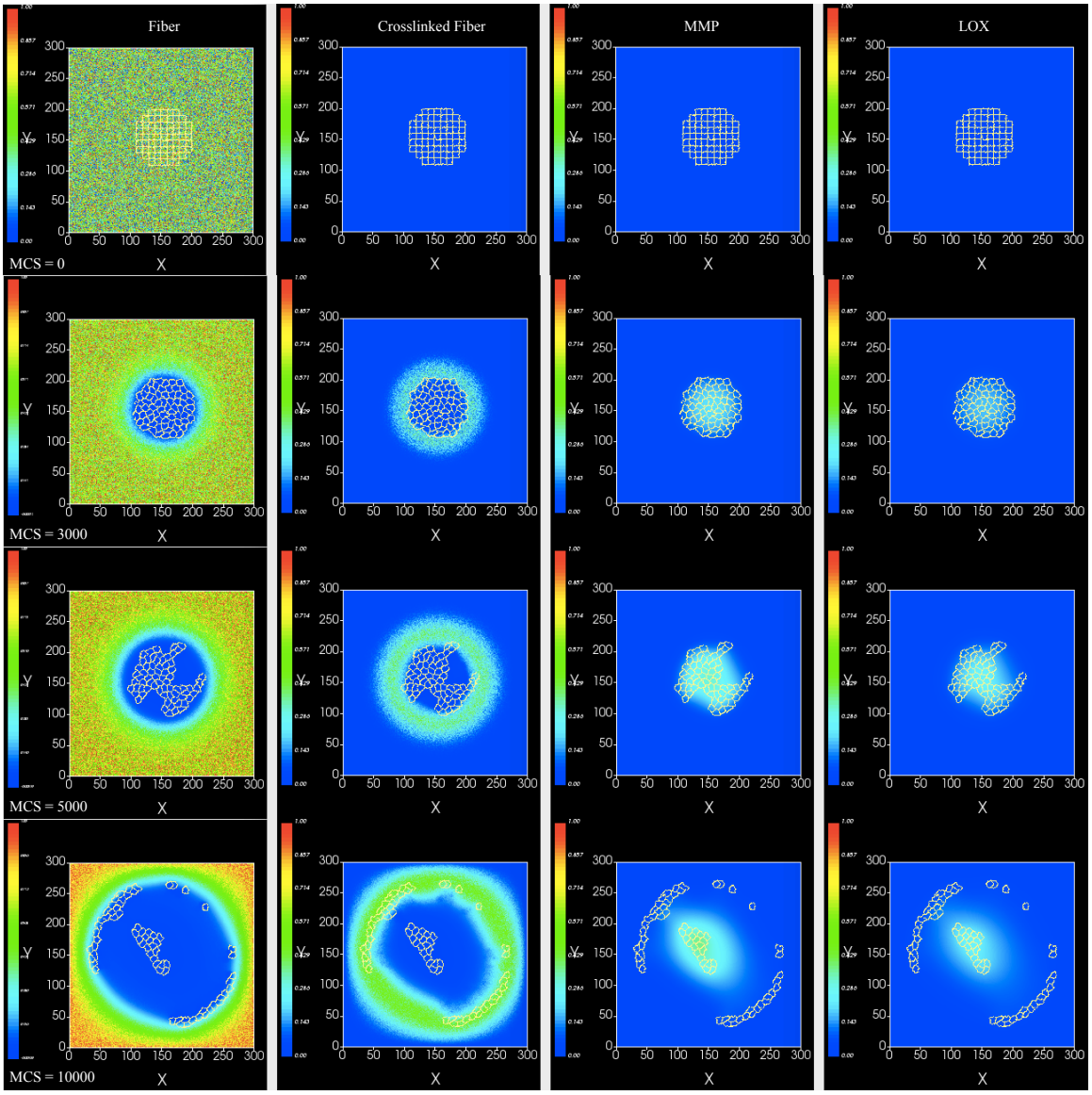

Figure 4.8: Simulation snapshots at four selected simulation times in MCS: 0, 3000, 5000, and 10000 (rows from top to bottom) showing the reaction-diffusion dynamics of the chemical fields: uncrosslinked fiber, crosslinked fiber, MMP, and LOX (columns from left to right) and the trajectories of cancer migration over time through a randomly distributed fiber.

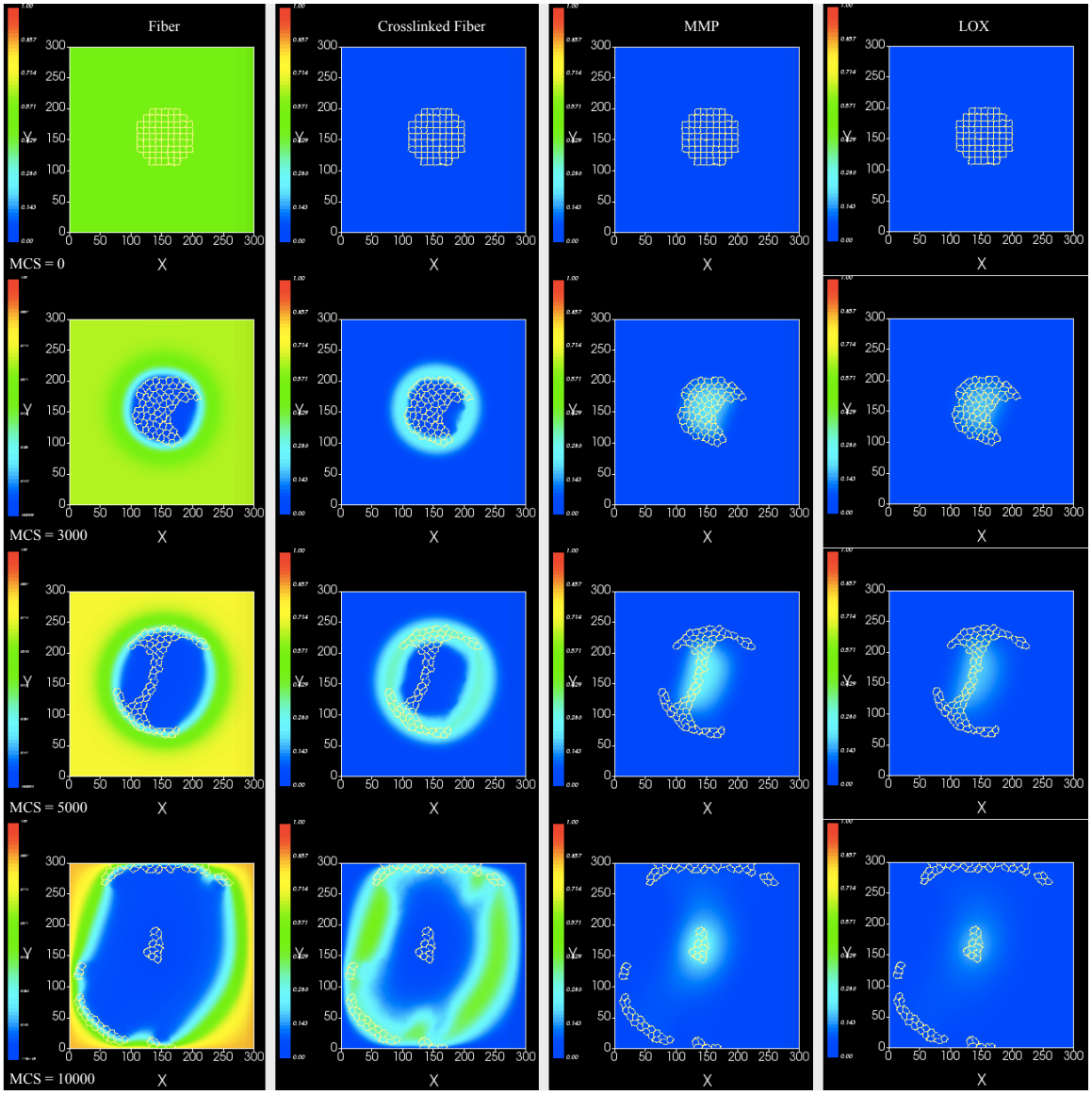

Figure 4.9: Simulation snapshots at four selected simulation times in MCS: 0, 3000, 5000, and 10000 (rows from top to bottom) showing the reaction-diffusion dynamics of the chemical fields: uncrosslinked fiber, crosslinked fiber, MMP, and LOX (columns from left to right) and the trajectories of cancer migration over time through a uniformly distributed fiber.

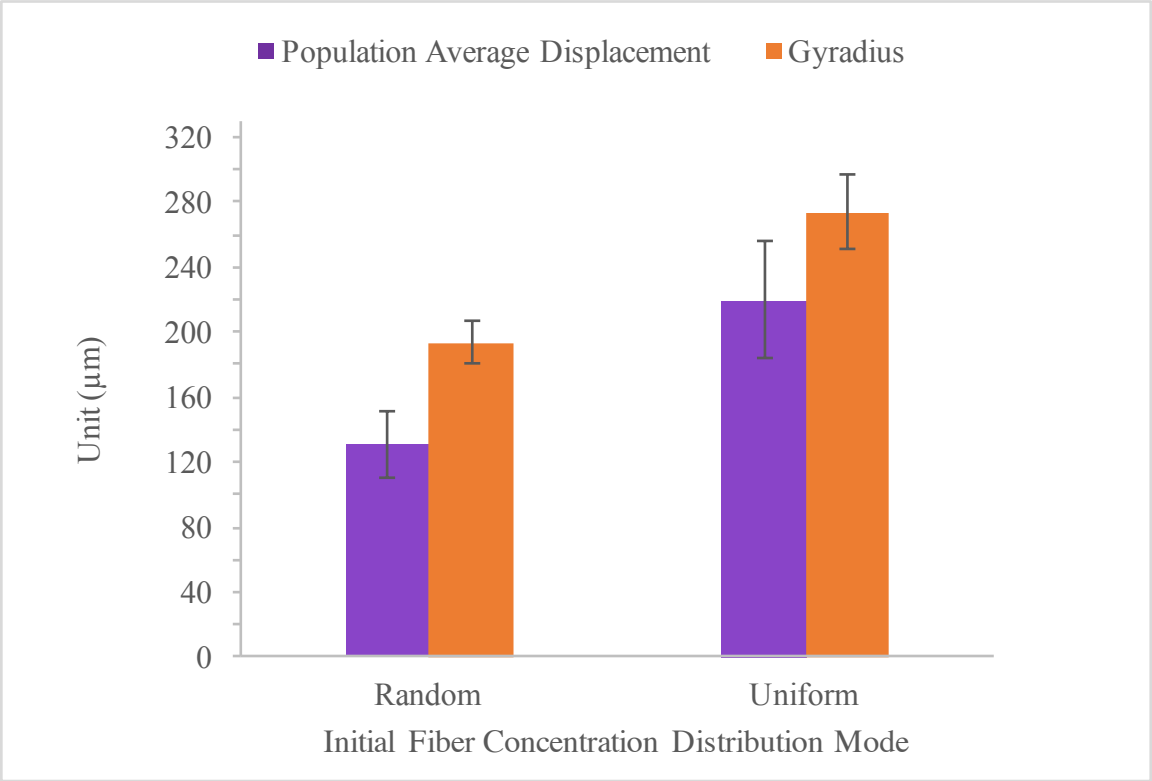

Figure 4.10: Compared to the uniform distribution, the random distribution of fiber concentration in the domain hinders the spreading capability of cancer cells resulting in a lower cancer migration efficiency. Cell migration metrics under the influence of two different modes of fiber distribution, random and uniform, for the same average dimensionless fiber concentration of 0.5. The migration is quantified in terms of the population average net displacement ($\bar{D}$) and the gyradius (R_g) for 69 simulated cancer cells after 10000 MCS $\approx$ 10 days. Error bars represent $\pm$ standard deviation of 10 repeated runs.

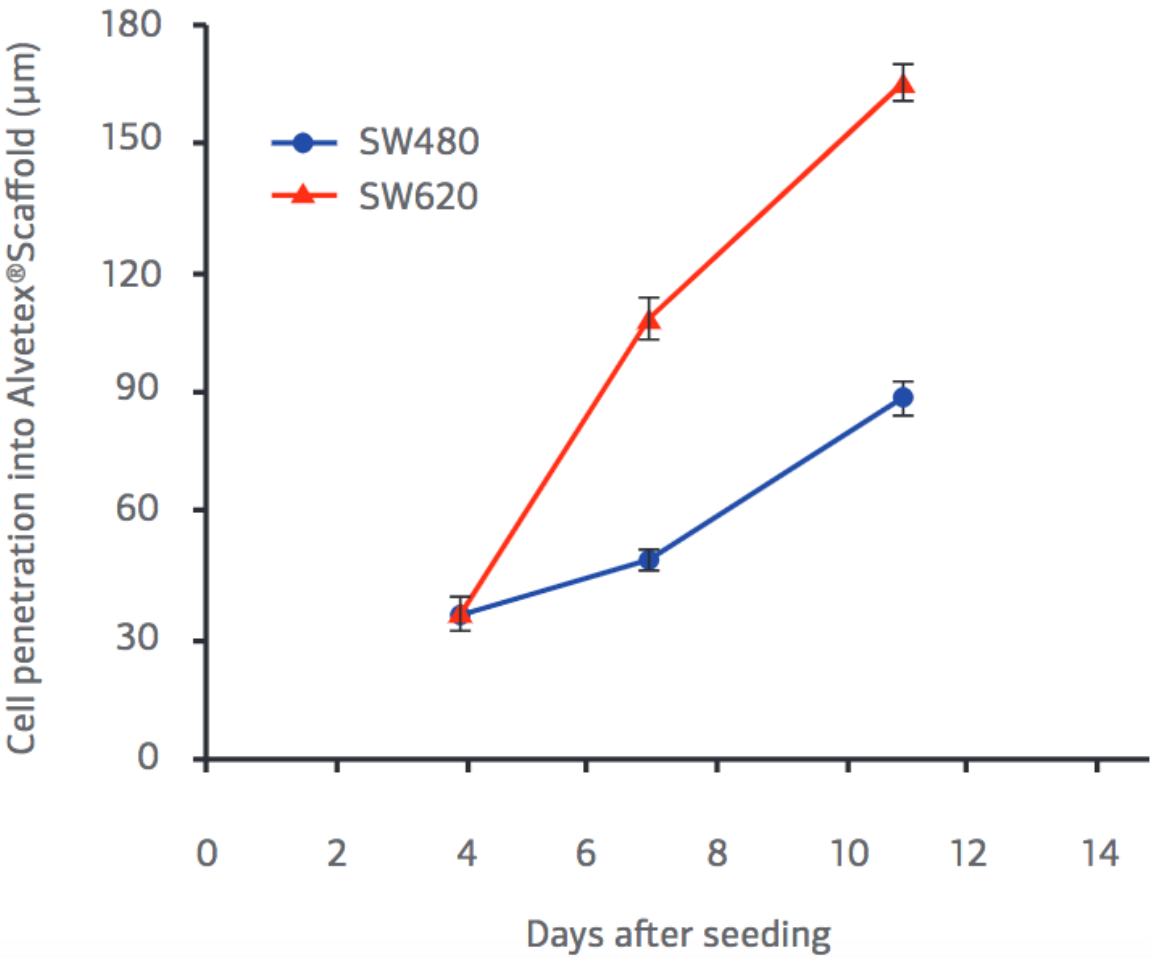

Figure 4.11: Data represent mean cell penetration depth over 11 days for a pair of isogenic colon carcinoma cell lines SW480 and SW620 in a 3D culture called Alvetex Scaffold [75]. Error bars represent ± standard error of mean.

gration efficiency (largers values for $\bar{D}$ and R_g). As in the previous analysis of Figure 4.8–4.10, the cancer cells have a greater migration efficiency through the uniformly distributed domains (Figure 4.15) than through the randomly distributed domains (Figure 4.14).

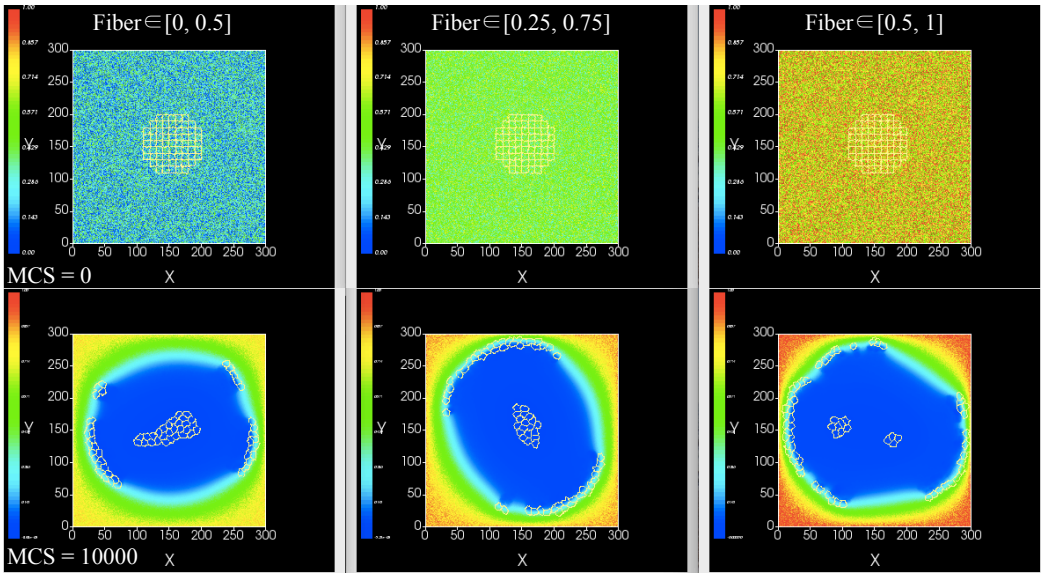

Figure 4.12: CompuCell3D simulation snapshots at two selected simulation times: 0 and 10000 MCS (rows from top to bottom) depicting effects on cancer spread for varying initial fiber concentrations in a random ECM. The initial average fiber dimensionless concentration is set to 0.25, 0.5, or 0.75 by varying the random distribution range while keeping widths the same by randomly sampling within the ranges [0, 0.5], [0.25,0.75], or [0.5, 1], respectively (columns from left to right).

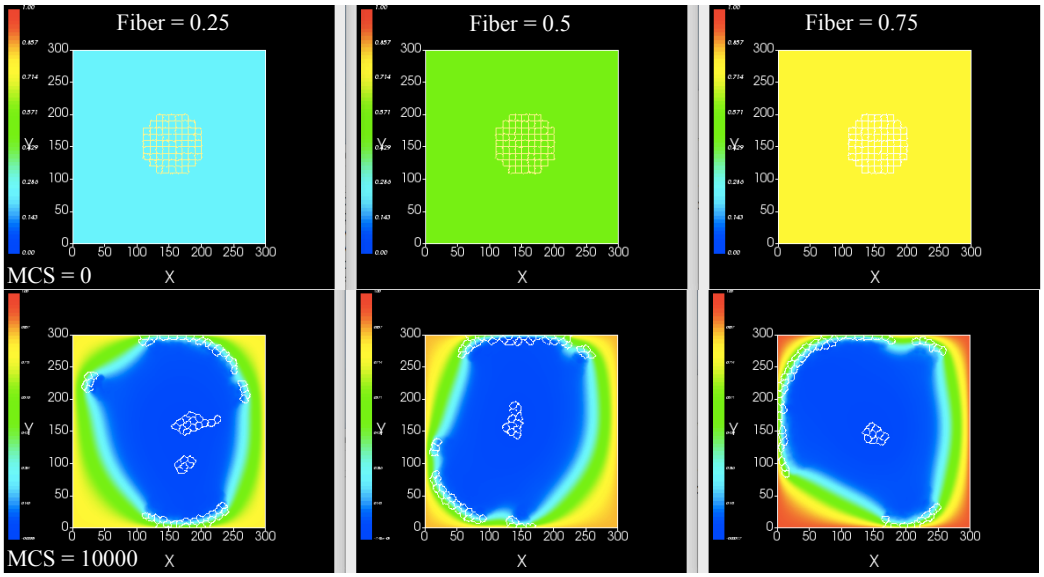

Figure 4.13: CompuCell3D simulation snapshots at two selected simulation times: 0 and 10000 MCS (rows from top to bottom) depicting effects on cancer spread for varying initial fiber concentrations in a uniform ECM. The initial average fiber dimensionless concentration is set to 0.25, 0.5, to 0.75 (columns from left to right).

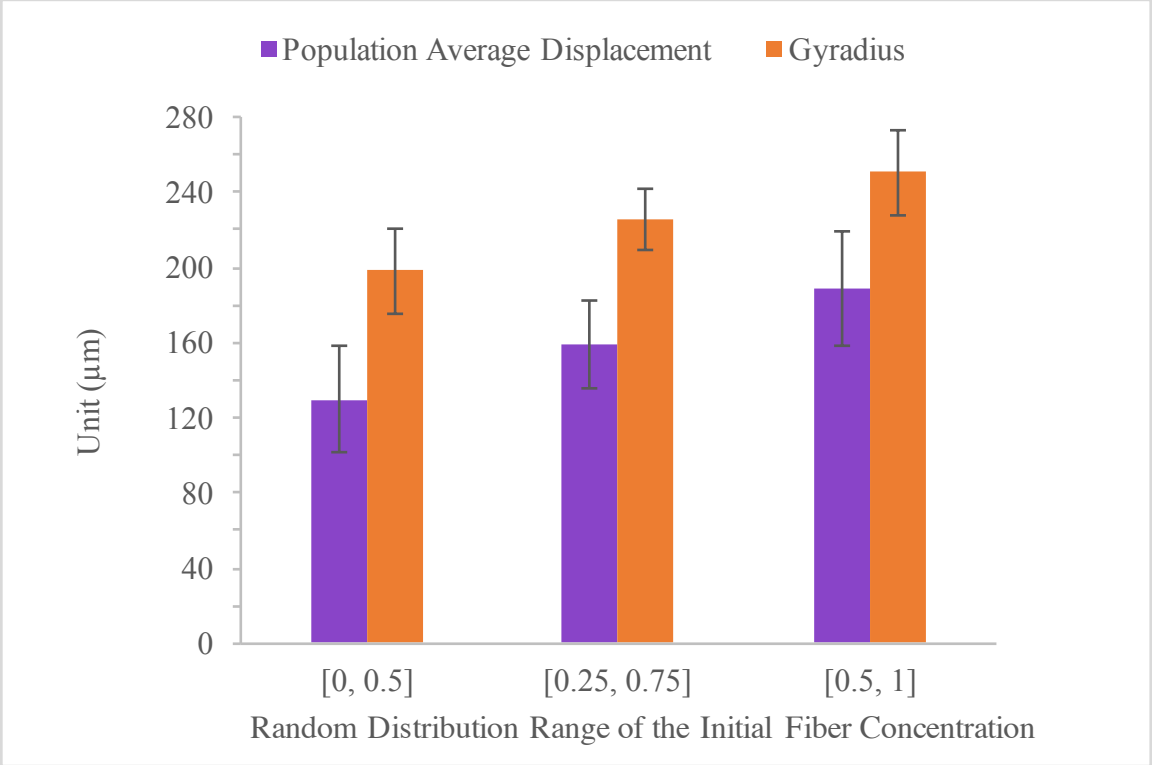

Figure 4.14: Cell migration metrics for varying the initial average fiber concentration in a random ECM. The initial average fiber dimensionless concentration is set to 0.25, 0.5, to 0.75 by varying the random distribution range while keeping widths the same by randomly sampling with the ranges [0, 0.5], [0.25,0.75], or [0.5, 1], respectively. The migration is quantified in terms of the population average displacement ($\bar{D}$) and the gyradius (R_g) for 69 simulated cancer cells after 10000 MCS $\approx$ 10 days. Error bars represent $\pm$ standard deviation of 10 repeated runs. For random distribution of fiber concentration, a low porosity ECM (high concentration of fibers) induces the migration capability of cancer cells resulting in a higher cancer migration efficiency.

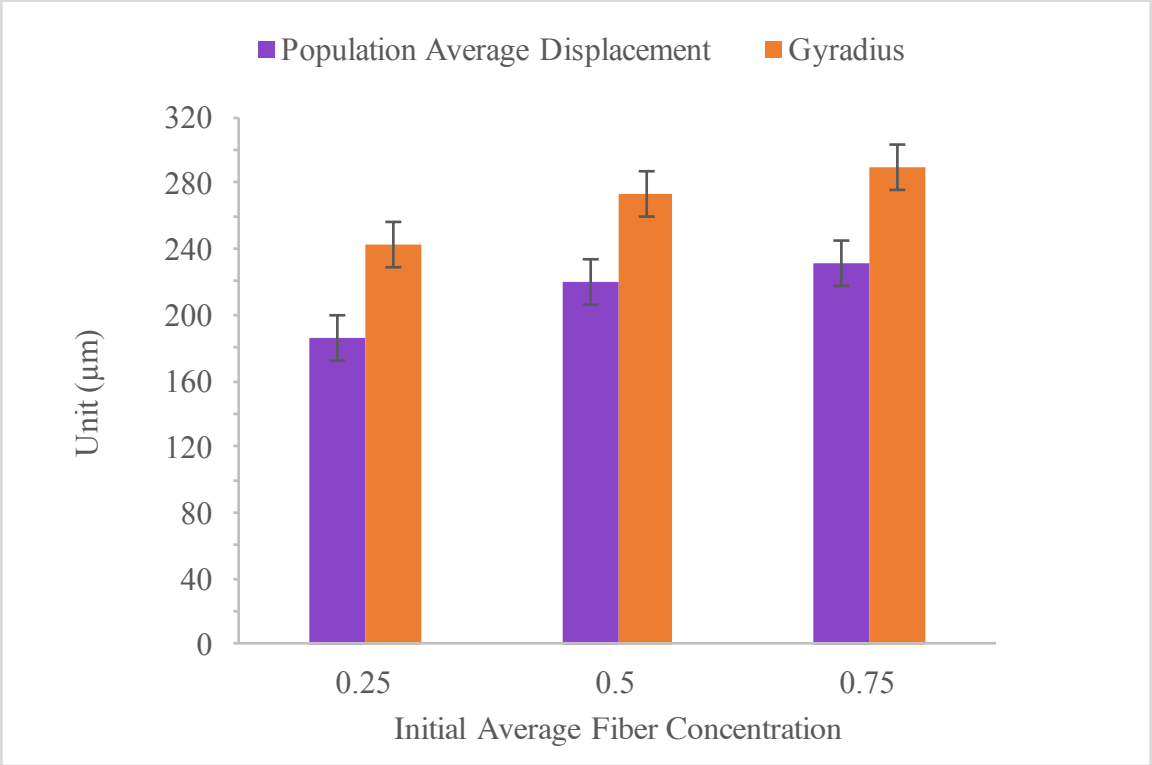

Figure 4.15: Cell migration metrics for varying the initial average fiber concentration in a uniform ECM. The initial average fiber dimensionless concentration is set to 0.25, 0.5, to 0.75. The migration is quantified in terms of the population average displacement ($\bar{D}$) and the gyradius (R_g) for 69 simulated cancer cells after 10000 MCS $\approx$ 10 days. Error bars represent $\pm$ standard deviation of 10 repeated runs. For uniform distribution of fiber concentration, a low porosity ECM (high concentration of fibers) induces the migration capability of cancer cells resulting in a higher cancer migration efficiency.

Chapter 5

Conclusions

In the first work presented in Chapter 3, a continuous model for reactions, diffusion, migration, and proliferation in the ECM undergoing dynamic remodeling has been proposed and analyzed. Case I, which only covers cancer cells, ECM density, and MMP concentration, has been verified and validated via comparison with a previous model [4]. The extended features of considering the enzyme LOX and its effect on ECM and cancer migration are successfully implemented in our new model demonstrated in Cases II and III. Simulation results of Case III confirmed the capability of the model to capture the cross-linking effect that LOX performs on ECM and how cross-linked fibers enhance the overall migration of cancer cells. Based on the current model, an additional PDE could be included to potentially aid in optimizing drug transportation into the tumor through the remodeling ECM. This could result in a better understanding of the various processes that take place within the specific microenvironment and in the determination of tissue and/or chemical factors that may inhibit an administered drug from infiltrating the tumor [44].

In the second work presented in Chapter 4, a ybrid model for cancer migration has been adopted by combining a continuum description of the fibers and remodeling enzymes encompassed within a metastatic tumor microenvironment and an agent-based Glazier-Granner-Hogeweg model of stochastic behavior and movement for discrete individual cells on a 2D square lattice. Results obtained from the model suggest that ECM fiber concentration is potentially a regulator of cell motility. For validation, the quantitative measurement of cell displacement in random ECM from our model

is found to be comparable to the experimental value of a similar quantity collected from a 3D culture of colon carcinoma cell line [75]. Further, accounting for additional biological activities could allow for the model to represent a more complex system.

Our models, in particular, include the role of the enzyme LOX in mathematical modeling of cancer migration and provide a fundamental understanding for the influence of ECM remodeling on the migration efficiency. Future extensions to models of this kind could potentially guide patient-specific therapies by accounting for drug actions on inhibiting the effects of LOX or altering or slowing the remodeling rate of the ECM to slow down or prevent metastasis.

References

[1] V. Andasari, A. Gerisch, G. Lolas, A. P. South, and M. A. J. Chaplain. Mathematical modeling of cancer cell invasion of tissue: Biological insight from mathematical analysis and computational simulation. *J Math Biol*, 63(1):141–171, 2011.

[2] A. R. A Anderson. A hybrid mathematical model of solid tumour invasion: the importance of cell adhesion. *Math Med Biol*, 22(2):163–186, 2005.

[3] A. R. A. Anderson and M. A. J. Chaplain. Continuous and discrete mathematical models of tumor-induced angiogenesis. *Bull Math Biol*, 60(5):857–99, 1998.

[4] A. R. A. Anderson, M. A. J. Chaplain, E. L. Newman, R. J. C. Steele, and A. M. Thompson. Mathematical modelling of tumour invasion and metastasis. *J Theor Med*, 2(2):129–154, 2000.

[5] A.R.A. Anderson, K.A. Rejniak, P. Gerlee, and V. Quaranta. Microenvironment driven invasion: a multiscale multimodel investigation. *J Math Biol*, 58:579–624, 2009.

[6] R.G. Bagley. *The Tumor Microenvironment*. Springer, New York, NY, 2010.

[7] F. R. Balkwill, M. Capasso, and T. Hagemann. The tumor microenvironment at a glance. *J Cell Sci*, 125(23):5591–5596, 2012.

[8] A. L. Bauer, T. L. Jackson, and Y. Jiang. A cell-based model exhibiting branching and anastomosis during tumor-induced angiogenesis. *Biophys J*, 92(9):3105–31021, 2007.

[9] J. E. Bear and J. M. Haugh. Directed migration of mesenchymal cells: where signaling and the cytoskeleton meet. *Curr Opin Cell Biol*, 30:74–82, 2014.

[10] M. J. Bissell and W. C. Hines. Why don't we get more cancer? A proposed role of the microenvironment in restraining cancer progression. *Nat Med*, 17(3):320–329, 2011.

[11] M. J. Bissell and M. A. LaBarge. Context, tissue plasticity, and cancer. *Cancer Cell*, 7(1):17–23, 2005.

[12] C. Bonnans, J. Chou, and Z. Werb. Remodelling the extracellular matrix in development and disease. *Nat Rev Mol Cell Biol*, 15(12):786–801, 2014.

[13] D. Bray. *Cell Movements: From Molecules to Motility*. Garland Publishing, New York, NY, 2001.

[14] G. W. Brodland. How computational models can help unlock biological systems. *Semin Cell Dev Biol*, 47–48(5):62–73, 2015.

[15] M. Brower, D. N. Carney, H. K. Oie, A. F. Gazdar, and J. D. Minna. Growth of cell lines and clinical specimens of human non-small cell lung cancer in serum-free defined medium. *Cancer Res*, 46(2):798–806, 1986.

[16] J. Cathcart, A. Pulkoski-Gross, and J. Cao. Targeting matrix metalloproteinases in cancer: Bringing new life to old ideas. *Genes Dis*, 2(1):26–34, 2015.

[17] T. E. Cawston and D. A. Young. Proteinases involved in matrix turnover during cartilage and bone breakdown. *Cell Tissue Res*, 339(1):221–235, 2010.

[18] M. A. J. Chaplain and G. Lolas. Mathematical modeling of cancer cell invasion of tissue: the role of the urokinase plasminogen activation system. *Math Models Methods Appl Sci*, 15(11):1685–1734, 2005.

[19] M. A. J. Chaplain and G. Lolas. Mathematical modeling of cancer invasion of tissue: dynamic heterogeneity. *Networks Heterog Media*, 1(3):399–439, 2006.

[20] A. G. Clark and D. M. Vignjevic. Modes of cancer cell invasion and the role of the microenvironment. *Curr Opin Cell Biol*, 36:13–22, 2015.

[21] I. E. Collier, W. Legant, B. Marmer, O. Lubman, S. Saffarian, T. Wakatsuki, E. Elson, and G. I. Goldberg. Diffusion of MMPs on the surface of collagen fibrils: the mobile cell surface - collagen substratum interface. *PLoS ONE*, 6(9):1–14, 2011.

[22] J. Condeelis and J. E. Segall. Intravital imaging of cell movement in tumours. *Nat Rev Cancer*, 3(12):921–930, 2003.

[23] L. M. Coussens, B. Fingleton, and L. M. Matrisian. Matrix metalloproteinase inhibitors and cancer: trials and tribulations. *Science*, 295(5564):2387–92, 2002.

[24] T. R. Cox, A. Gartland, and J. T. Erler. Lysyl oxidase, a targetable secreted molecule involved in cancer metastasis. *Cancer Res*, 76(2):188–192, 2016.

[25] N. E. Deakin and M. A. J. Chaplain. Mathematical modeling of cancer invasion: The role of membrane-bound matrix metalloproteinases. *Front Oncol*, 3:70, 2013.

[26] A. D. Doyle, R. J. Petrie, M. L. Kutys, and K. M. Yamada. Dimensions in Cell Migration. *Curr Opin Cell Biol*, 25(5):642–649, 2013.

[27] M. Egeblad, M. G. Rasch, and V. M. Weaver. Dynamic interplay between the collagen scaffold and tumor evolution. *Curr Opin Cell Biol*, 22(5):697–706, 2010.

[28] S. Evje. An integrative multiphase model for cancer cell migration under influence of physical cues from the microenvironment. *Chem Eng Sci*, 165:240–259, 2017.

[29] B. Fingleton. Chapter 36: MMP inhibitor clinical trials - the past, present, and future. In D. Edwards, G. Hoyer-Hansen, F. Blasi, and B.F. Sloane, editors, *The Cancer Degradome: Proteases and Cancer Biology*, pages 759–785. Springer, New York, NY, 2008.

[30] S. I. Fraley, P. Wu, L. He, Y. Feng, R. Krisnamurthy, G. D. Longmore, and D. Wirtz. Three-dimensional matrix fiber alignment modulates cell migration and MT1-MMP utility by spatially and temporally directing protrusions. *Sci Rep*, 5:14580, 2015.

[31] C. Frantz, K. M. Stewart, and V. M. Weaver. The extracellular matrix at a glance. *J Cell Sci*, 123:4195–4200, 2010.

[32] H. B. Frieboes, X. Zheng, C. Sun, B. Tromberg, R. Gatenby, and V. Cristini. An integrated computational/experimental model of tumor invasion. *Cancer Res*, 66(3):1597–1604, 2006.

[33] R. A. Gatenby. Models of tumor-host interaction as competing populations: implications for tumor biology and treatment. *J Theor Biol*, 176(4):447–55, 1995.

[34] R. A. Gatenby and E. T. Gawlinski. A reaction-diffusion model of cancer invasion. *Cancer Res*, 56(24):5745–53, 1996.

[35] A. Gerisch and M. A. J. Chaplain. Mathematical modelling of cancer cell invasion of tissue: local and non-local models and the effect of adhesion. *J Theo Biol*, 250(4):684–704, 2008.

[36] C. Gialeli, A. D. Theocharis, and N. K. Karamanos. Roles of matrix metalloproteinases in cancer progression and their pharmacological targeting. *Cell Tissue Res*, 339(1):221–235, 2010.

[37] D. M. Gilkes, G. L. Semenza, and D. Wirtz. Hypoxia and the extracellular matrix: drivers of tumour metastasis. *Nat Rev Cancer*, 14(6):430–439, 2014.

[38] D. J. Higham. *MATLAB Guide*. Society for Industrial and Applied Mathematics, Philadelphia, 1992.

[39] T. Hillen. M5 mesoscopic and macroscopic models for mesenchymal motion. *J Math Biol*, 53(4):585–616, 2006.

[40] K. Jacobson, A. Ishihara, and R. Inman. Lateral diffusion of proteins in membranes. *Annu Rev Physiol*, 49(1):163–175, 1987.

[41] C. Kahlert and R. Kalluri. Exosomes in tumor microenvironment influence cancer progression and metastasis. *J Mol Med*, 91(4):431–437, 2013.

[42] S. Keeratichamroen, K. Lirdprapamongkol, and J. Svasti. Mechanism of ECM-induced dormancy and chemoresistance in A549 human lung carcinoma cells. *Oncol Rep*, 39(4):1765–1774, 2018.

[43] K. Kessenbrock, V. Plaks, and Z. Werb. Matrix metalloproteinases: Regulators of the tumor microenvironment. *Cell*, 141(1):52–67, 2010.

[44] M. J. Kim, R. J. Gillies, and K. A. Rejniak. Current advances in mathematical modeling of anti-cancer drug penetration into tumor tissues. *Front Oncol*, 3:278, 2013.

[45] J. Klominek, K. Robért, and K. Sundqvist. Chemotaxis and haptotaxis of human malignant mesothelioma cells: effects of fibronectin, laminin, type IV collagen, and an autocrine motility factor-like substance. *Cancer Res*, 53(18):4376, 1993.

[46] M. Kolev and B. Zubik-Kowal. Numerical solutions for a model of tissue invasion and migration of tumour cells. *Comput Math Methods Med*, 2011:1–16, 2011.

[47] T. Kremmer, I. Palyi, D. Daubner, M. Boldizsar, B. Vincze, E. Paulik, J. Sugar, E. Pokorny, and E. Tury. Comparative studies on the polyamine metabolism and DFMO treatment of MCF-7 and MDA-MB-231 breast cancer cell lines and xenografts. *Anticancer Res*, 11(5):1807–13, 1991.

[48] K. Kuhn. Basement membrane (type IV) collagen. *Matrix Biol*, 14(6):439–445, 1995.

[49] S. Kumar, A. Das, A Barai, and S. Sen. MMP secretion rate and inter-invadopodia spacing collectively govern cancer invasiveness. *Biophys J*, 114(3):650–662, 2018.

[50] S. Kumar, A. Kapoor, S. Desai, M. M. Inamdar, and S. Sen. Proteolytic and non-protelytic regulation of collective cell invasion: Tuning by ECM density and organization. *Sci Rep*, 6(1), 2016.

[51] R. Limame, A. Wouters, B. Pauwels, E. Fransen, M. Peeters, F. Lardon, O. de Wever, and P. Pauwels. Comparative analysis of dynamic cell viability, migration and invasion assessments by novel real-time technology and classic endpoint assays. *PLoS ONE*, 7(10):e46536, 2012.

[52] P. Lu, K. Takai, V. M. Weaver, and Z. Werb. Extracellular matrix degradation and remodeling in development and disease. *Cold Spring Harb Perspect Biol*, 3(12):a005058, 2011.

[53] P. Lu, V. M. Weaver, and Z. Werb. The extracellular matrix: A dynamic niche in cancer progression. *J Cell Biol*, 196(4):395–406, 2012.

[54] M. Mendoza and C. Khanna. Revisiting the seed and soil in cancer metastasis. *Int J Biochem Cell Biol*, 41(7):1452–1462, 2009.

[55] B. W. Miller, J. P. Morton, M. Pinese, G. Saturno, N. B. Jamieson, E. McGhee, P. Timpson, J. Leach, L. McGarry, E. Shanks, P. Bailey, D. Chang, K. Oien,

S. Karim, A. Au, C. Steele, C. R. Carter, C. McKay, K. Anderson, T. R. J. Evans, R. Marais, C. Springer, A. Biankin, J. T. Erler, and O. J. Sansom. Targeting the LOX/hypoxia axis reverses many of the features that make pancreatic cancer deadly: inhibition of LOX abrogates metastasis and enhances drug efficacy. *EMBO Mol Med*, 7(8):1063–76, 2015.

[56] H. Mori, A. T. Lo, J. L. Inman, J. Alcaraz, C. M. Ghajar, J. D. Mott, C. M. Nelson, C. S. Chen, H. Zhang, J. L. Bascom, M. Seiki, and M. J. Bissell. Transmembrane/cytoplasmic, rather than catalytic, domains of Mmp14 signal to MAPK activation and mammary branching morphogenesis via binding to integrin β1. *Development*, 140(2):343–352, 2013.

[57] Y. T. Nguyen Edalgo and A. N. Ford Versypt. Mathematical modeling of metastatic cancer migration through a remodeling extracellular matrix. *Processes*, 6(5):58, 2018.

[58] Y. T. Nguyen Edalgo and A. N. Ford Versypt. *MetastaticCancerECMRemodeling*, 2018. DOI: 10.5281/zenodo.1226605. http://github.com/ashleefv/MetastaticCancerECMRemodeling (accessed 18.04.22).

[59] M.-E. Oraiopoulou, E. Tzamali, G. Tzedakis, A. Vakis, J. Papamatheakis, and V. Sakkalis. *In vitro/in silico* study on the role of doubling time heterogeneity among primary glioblastoma cell lines. *BioMed Res Int*, 2017(22):Article 8569328, 2017.

[60] M. E. Orme and M. A. Chaplain. A mathematical model of the first steps of tumour-related angiogenesis: capillary sprout formation and secondary branching. *IMA J Math Appl Med Biol*, 13(2):73–98, 1996.

[61] T. Osawa, N. Ohga, K. Akiyama, Y. Hida, K. Kitayama, T. Kawamoto, K. Ya-

mamoto, N. Maishi, M. Kondoh, Y. Onodera, M. Fujie, N. Shinohara, K. Nonomura, M. Shindoh, and K. Hida. Lysyl oxidase secreted by tumour endothelial cells promotes angiogenesis and metastasis. *Br J Cancer*, 109(8):2237—-47, 2013.

[62] M. J. Oudin, O. Jonas, T. Kosciuk, L. C. Broye, B. C. Guido, J. Wyckoff, D. Riquelme, J. M. Lamar, S. B. Asokan, C. Whittaker, D. Ma, R. Langer, M. J. Cima, K. B. Wisinski, R. O. Hynes, D. A. Lauffenburger, P. J. Keely, J. E. Bear, and F. B. Gertler. Tumor cell-driven extracellular matrix remodeling drives haptotaxis during metastatic progression. *Cancer Discovery*, 6(5):516–31, 2016.

[63] K. J. Painter. Modelling cell migration strategies in the extracellular matrix. *J Math Biol*, 58(4–5):511–543, 2009.

[64] K. J. Painter, N. A. Armstrong, and J. A. Sherratt. The impact of adhesion on cellular invasion processes in cancer and development. *J Theor Biol*, 264:1057–1067, 2010.

[65] A. J. Perumapanani, J. A. Sherratt, J. Norbury, and H. M. Byrne. Biological inferences from a mathematical model for malignant invasion. *Invasion Metastasis*, 16(4–5):209–21, 1996.

[66] A. J. Perumapanani, D. L. Simmons, A. J. H. Gearing, K. M. Miller, G. Ward, J. Norbury, M. Schneemann, and J. A. Sherratt. Extracellular matrix-mediated chemotaxis can impede cell migration. *Proc Biol Sci*, 265(1413):2347, 1998.

[67] P. Pittayapruek, J. Meephansan, O. Prapapan, M. Komine, and M. Ohtsuki. Role of matrix metalloproteinases in photoaging and photocarcinogenesis. *Int J Mol Sci*, 17(6), 2016.

[68] N. J. Poplawski, U. Agero, J. S. Gens, M. Swat, J. A. Glazier, and A. R. A.

Anderson. Front instabilities and invasiveness of simulated avascular tumors. *Bull Math Biol*, 71(5):1189–1227, 2009.

[69] L. Provencher, C. Diorio, J. Hogue, C. Doyle, and S. Jacob. Does breast cancer tumor size really matter that much? *Breast*, 21(5):682–685, 2012.

[70] P. P. Provenzano, K. W. Eliceiri, J. M. Campbell, D. R. Inman, J. G. White, and P. J. Keely. Collagen reorganization at the tumor-stromal interface facilitates local invasion. *BMC Med*, 4(1):38, 2006.

[71] V. Quaranta. Cell migration through extracellular matrix: membrane-type metalloproteinases make the way. *J Cell Biol*, 149(6):1167–1170, 2000.

[72] I. Ramis-Conde, D. Drasdo, A. R. A. Anderson, and M. A. J. Chaplain. Modeling the influence of the E-cadherin-β-catenin pathway in cancer cell invasion: a multiscale approach. *Biophys J*, 95(1):155–165, 2008.

[73] A. Rasmuson, B. Andersson, L. Olsson, and R. Andersson. *Mathematical Modeling in Chemical Engineering*. Cambridge University Press, Cambridge, 2014.

[74] K. A. Rejniak. *Systems Biology of Tumor Microenvironment: Quantitative Modeling and Simulations*. Springer International Publishing, Switzerland, 2016.

[75] ReproCELL. Modelling Cancer Cell Migration and Invasion Using Novel Technology for Three Dimensional Cell Culture. `https://www.reprocell.com/pub/media/wysiwyg/alvetex_pdfs/AN-SC-03_AppNote_Scaffold_03.pdf`. [Online; accessed 25-June-2018].

[76] M. Ruchi, I. L. Peter, and C. Edna. Biomechanical and biochemical remodeling of stromal extracellular matrix in cancer. *Trends Biotechnol*, 33(4):230–236, 2015.

[77] A Countercurrents Series and S. A. Narod. Disappearing breast cancers. *Curr Oncol*, 19(2):59–60, 2012.

[78] C. Shah, E. B. Johnson, E. Everett, H. Tamimi, B. Greer, E. Swisher, and B. Goff. Does size matter? Tumor size and morphology as predictors of nodal status and recurrence in endometrial cancer. *Gynecol Oncol*, 99(3):564–570, 2005.

[79] L. I. Smith-Mungo and H. M. Kagan. Lysyl oxidase: properties, regulation and multiple functions in biology. *Matrix Biol*, 16(7):387–398, 1998.

[80] M. H. Swat, G. L. Thomas, J. M. Belmonte, A. Shirinifard, D. Hmeljak, and J. A. Glazier. Multi-Scale modeling of tissues using CompuCell3D. *Methods Mol Biol*, 110:325–366, 2012.

[81] M. H. Swat, G. L. Thomas, A. Shirinifard, S. G. Clendenon, and J. A. Glazier. Emergent stratification in solid tumors selects for reduced cohesion of tumor cells: a multi-cell, virtual-tissue model of tumor evolution using CompuCell3D. *PLoS ONE*, 10(6):1–40, 2015.

[82] Z. Szymanska, M. Cytowski, E. Mitchell, C. K. Macnamara, and M. A. J. Chaplain. Computational modelling of cancer development and growth: modelling at multiple scales and multiscale modelling. *Bull Math Biol*, 80(5):1366—-1403, 2018.

[83] Z. Szymanska, C. M. Rodrigo, M. Lachowicz, and M. A. J. Chaplain. Mathematical modelling of cancer invasion of tissue: the role and effect of nonlocal interactions. *Math Models Methods Appl Sci*, 19(2):257–281, 2009.

[84] V. P. Terranova, R. Diflorio, R. M. Lyall, S. Hic, R. Friesel, and T. Maciag. Human endothelial cells are chemotactic to endothelial cell growth factor and heparin. *J Cell Biol*, 101(6):2330–2334, 1985.

[85] A. Toma, A. Mang, T. A. Schuetz, S. Becker, and T. M. Buzug. A novel method for simulating the extracellular matrix in models of tumour growth. *Comput Math Methods Med*, 2012, 2012.

[86] S. Turner and J. A. Sherratt. Intercellular adhesion and cancer invasion: a discrete simulation using the extended Potts model. *J Theor Biol*, 16(1):85–100, 2002.

[87] M. Wang, J. Zhao, L. Zhang, F. Wei, Y. Lian, Y. Wu, Z. Gong, S. Zhang, J. Zhou, K. Cao, X. Li, W. Xiong, G. Li, Z. Zeng, and C. Guo. Role of tumor microenvironment in tumorigenesis. *J Cancer*, 8(5):761–773, 2017.

[88] J. B. Wyckoff, Y. Wang, E. Y. Lin, J. F. Li, S. Goswami, E. R. Stanley, J. E. Segall, J. W. Pollard, and J. Condeelis. Direct visualization of macrophage-assisted tumor cell intravasation in mammary tumors. *Cancer Res*, 67(6):2649–2656, 2007.

[89] M. Yamauchi and M. Sricholpech. Lysine post-translational modifications of collagen. *Essays Biochem*, 52:113–133, 2012.

[90] H. Yu, J. K. Mouw, and V. M. Weaver. Forcing form and function: biomechanical regulation of tumor evolution. *Trends Cell Biol*, 21(1):47–56, 2011.

[91] M. H. Zaman, L. M. Trapani, A. L. Sieminski, D. MacKellar, H. Gong, R. D. Kamm, A. Wells, D. A. Lauffenburger, and P. Matsudaira. Migration of tumor cells in 3D matrices is governed by matrix stiffness along with cell-matrix adhesion and proteolysis. *Proc Natl Acad Sci USA*, 103(29):10889–10894, 2006.

Appendix A

MATLAB Codes

A.1 PDEs Model Numerical Solution File

Contents

- solve_pdepe_CancerECM
- Author: Ye Nguyen
- Description of Input and output parameters or variables:
- Input
- Units of Input and output parameters or variables:
- Parameters value (either calculated or tentative)
- PDEs solver pdepe set up and called
- Parameters considered for localsensitivity_CancerECM
- Retrive values for each variables from output soln
- Plotting
- Plot 1D numerical solution for the system at t = 0
- Legend set up:
- Plot 1D numerical solution for the system at t = 1
- Plot 1D numerical solution for the system at t = 10
- Plot 1D numerical solution for the system at t = 20
- Export_fig
- Define system of PDEs
- Define PDEs system into the framework of pdepe
- Logistic growth/ physical space filling TERM
- Cases:
- Case 1: Withoutout LOX effect
- Case 2: With LOX, NO haptotaxis effect toward cross-link ECM
- Case 3: With LOX AND haptotaxis effect toward cross-link ECM
- Model set up in pdepe form:
- Define the initial conditions at t = t0
- Anderson (2000) conditions for case 1
- Define the boundary condions at x = a = 0 and x = b = 1
- Zero-flux in the left edge
- Anderson (2000) conditions: Zero-flux in both edges

solve_pdepe_CancerECM

```
% % Ouput are tumor cells density; density of non-crosslink and cross-link
% % ECM; concentration of enzyme MMPs and LOXs
% % in a metastatic tumor microenvironment.
% % The model describes the remodeling of ECM due to MMP and LOX and the
% % migration of tumor cells through a remodeling ECM

%  The current model is inspired
%  from the following primariy previous models of
%  Anderson (2000), Gerisch (2008), and Andasari (2011)
%  This new model is further improved with extended features
%  related to LOX and its effect on the whole system.

function sol = solve_pdepe_CancerECM(varargin)

% default values first

ploton = 0;
```

Author: Ye Nguyen

Description of Input and output parameters or variables:

```
% x             : Ind. spatial variable
% L             : reference length

    % t             : Ind. simulation time variable
    % tau           : residence time

    % c             : Dep. tumor cells density variable
    % co            : ref. value for c

    % D             : reference chemical diffusion coefficient
    % Dc            : diffusion coefficient of tumor cells
    % Dc_hat        : dimensionless coefficient of Dc

    % rho           : haptotaxis toward regular ECM
    % rho_hat       : dimensionless rho

    % rho_cl_hat    : dimensionless haptotaxis toward cross-link ECM

    % gamma         : dimensionless proliferation of the primary tumor

    % f             : dep. non-cross-link ECM density variable
    % f_cl          : cross-link ECM density
    % fo            : reference parameter for f and f_cl
```

```
% alpha_f_hat   : dimensionless MMPs uptake rate of ECM for degradation
% beta_f_hat    : dimensionless LOXs uptake rate of ECM for crosslinkings
% mu_f_hat      : dimensionless remodeling rate of ECM

% m             : MMPs concentration
% mo            : ref value for m

% Dm            : diffusion coefficient of enzyme MMPs
% Dm_hat        : dimensionless Dm

% alpha_m       : decay coefficient of MMPs
% alpha_m_hat   : dimensionless alpha_m

% beta_m        : secretion rate of MMPs
% beta_m_hat    : dimensionless production rates of MMPs

% l             : LOXs concentration

% Dl_hat        : dimensionless diffusion coefficient of enzyme LOX

% alpha_l_hat   : dimensionless decay coefficient of LOXs

% beta_l_hat    : dimensionless production rates of LOXs

% epsilon       : A positive constant used in I.C.s
```

Units of Input and output

```
% L         : cm
% x         : cm

% t         : s
% t0        : s

% c         : cells/cm3

% D         : cm2/s
% Dc        : cm2/s

% rho       : cm^2/(s*M)

% gamma     : s^(-1)

% f         : M
```

```
% f_cl      : M
% fo        : M

% m, mo     : M

% Dm        : cm2/s

% l         : M

% Dl        : cm2/s

% alpha_m   : s^(-1)
% beta_m    : s^(-1)
```

Parameter value (calculated or estimated)

```
co         = 6.7*10^7     ; % cell/cm3  Anderson (2000)
fo           = 10^(-11)     ; % 10^-8 to 10^-11  from Anderson, 2000
mo           = 0.1*10^(-9) ; % Andasari(2011)

L          = 1            ; % 0.1 to 1 cm
D          = 10^(-6)      ; % Bray 1992
tau      = 32*3600   ; % Anderson(2000)- 8 to 24 hours, take 8 hrs

Dc           = 10^(-9)     ; %Bray 1992
Dc_hat       = Dc*tau/L^2       ; %10^(-3) to 10^(-5) Chaplain (2006)

rho             = 2600         ; % Anderson
rho_hat      = rho*fo*tau/(L^2);
rho_cl_hat = 0.05     ; % Estimated dominates over non-cross-link ECM

epsilon     = 0.001       ; % positive constant used in gamma

alpha_f_hat = 10           ; % Anderson (2000)
mu_f_hat    = 0.15       ; % Andasari (2000)
beta_f_hat  = 18           ; % Estimated

Dm           = 10^(-8) ; % 10^-8 to 10^-10, Anderson(2000), Kumar(2015)
Dm_hat       = Dm*tau/L^2        ;

alpha_m      = 0.002       ; % sec^(-1) Kumar (2018)
alpha_m_hat = 0.001    ; % estimated based on magnitude ratio between...
 % secretion and degration rate in Kumar (2018),...
 % i.e. 0.1/0.002~100
beta_m       = 0.1        ; % sec^(-1) Kumar (2018) varied rates...
```

```
      % from 0.005, 0.1, and 0.5 s^-1
    beta_m_hat  = 0.1     ; % estimated (as alpha) by Anderson (2000)

Dl_hat      = 2*Dm_hat       ;

   alpha_l_hat = alpha_m_hat ; % assumed equals alpha_m_hat
   beta_l_hat  = 0.1         ;
```

PDEs solver "pdepe" set up

```
n = 0; %rectangular coordinate for pdepe

xmesh = linspace(0,1,100);
tspan = linspace(0,20,100);

% Read in chang in values if varargin is supplied to
% solve_pdepe_CancerECM(parameters,tspan,ploton)

if nargin > 0

    parameters = varargin{1};
```

Parameters considered for localsensitivity_CancerECM.m file

unpack parameters

```
    Dc_hat      = parameters(1);
    epsilon     = parameters(2);
    rho_hat     = parameters(3);
    rho_cl_hat  = parameters(4);
    alpha_f_hat = parameters(5);
    mu_f_hat    = parameters(6);
    beta_f_hat  = parameters(7);
    Dm_hat      = parameters(8);
    alpha_m_hat = parameters(9);
    beta_m_hat  = parameters(10);
    Dl_hat      = parameters(11);
    alpha_l_hat = parameters(12);
    beta_l_hat  = parameters(13);

    tspan = varargin{2};
    ploton = varargin{3};
end
```

PDEs solver "pdepe" gets called

```
PDEs solver "pdepe
\end{par} \vspace{1em}
options = odeset('RelTol',1e-8,'AbsTol',1e-10);
sol = pdepe(n,@mbpde,@mbic,@mbbc,xmesh,tspan,options);
```

Retrive values for each variables from output solution

```
c           = sol(:,:,1);
f           = sol(:,:,2);
f_cl        = sol(:,:,3);
m           = sol(:,:,4);
l           = sol(:,:,5);
spaceCheck  = 1-c-f-f_cl;

%desired output for global sensitivity may be
% cMaxwrtXatFinalT = max(sol(tspan(end),:,1)); % max with respect to x, not
% max change in output wrt x
% fMaxwrtXatFinalT = ...
% ...
% output = [cMaxwrtXatFinalT; fMaxwrtXatFinalT; ...]; % might need to be
% columns instead of rows
```

Plotting

```
if ploton == 1
```

Plot 1D numerical solution for the system at t = 0

```
        %subplot_tight(2.5,2,1, [0.1 0.1])
        subplot(2, 2, 1)
        hold on
        p1 = plot(xmesh,sol(1,:,1),'k',...
             xmesh,sol(1,:,2),'m--',...
             xmesh,sol(1,:,3),'b-.',...
             xmesh,sol(1,:,4),'r.',...
             xmesh,sol(1,:,5),'gh')

%         plot(xmesh, 1-sol(1,:,1)-sol(1,:,2)-sol(1,:,3),'c')

        p1(1).LineWidth = 2;
        p1(2).LineWidth = 2;
        p1(3).LineWidth = 2;
        p1(4).LineWidth = 2;
        ylim([0 1.1])
```

```
xlabel('$x$','Interpreter','latex','FontSize',15)
ylabel('$y$','Interpreter','latex','FontSize',15,'Rotation',0)
title('$t = 0$','Interpreter','latex','FontSize',15)
get(gca);set(gca,'FontSize',15,'FontName','Arial');
```

Legend set up:

```
ledg = legend('cancer cells',...
        'regular ECM fibers',...
        'cross-linked ECM fibers',...
        'MMP',...
        'LOX',...
        'space check',...
        'position',[50 50 450 0]) %[left right across height]

set(ledg,'location','east');
ledg.FontSize = 12;
ledg.FontName ='Arial';
```

Plot 1D numerical solution for the system at t = 1

```
%subplot_tight(2.5,2,2, [0.1 0.1])
subplot(2, 2, 2)
hold on
p2 = plot(xmesh,sol(5,:,1),'k',...
     xmesh,sol(5,:,2),'m--',...
     xmesh,sol(5,:,3),'b-.',...
     xmesh,sol(5,:,4),'r.',...
     xmesh,sol(5,:,5),'gd')

%         plot(xmesh, 1-sol(5,:,1)-sol(5,:,2)-sol(5,:,3),'c')

p2(1).LineWidth = 2;
p2(2).LineWidth = 2;
p2(3).LineWidth = 2;
p2(4).LineWidth = 2;
ylim([0 1.1])
xlabel('$x$','Interpreter','latex','FontSize',15)
ylabel('$y$','Interpreter','latex','FontSize',15,'Rotation',0)
title('$t = 1$','Interpreter','latex','FontSize',15)
get(gca);set(gca,'FontSize',15,'FontName','Arial');
```

Plot 1D numerical solution for the system at t = 10

```
%subplot_tight(2.5,2,3, [0.1 0.1])
subplot(2, 2, 3)
```

```
        hold on
        p3 = plot(xmesh,sol(50,:,1),'k',...
             xmesh,sol(50,:,2),'m--',...
             xmesh,sol(50,:,3),'b-.',...
             xmesh,sol(50,:,4),'r.',...
             xmesh,sol(50,:,5),'gd')

%         plot(xmesh, 1-sol(50,:,1)-sol(50,:,2)-sol(50,:,3),'c')

        p3(1).LineWidth = 2;
        p3(2).LineWidth = 2;
        p3(3).LineWidth = 2;
        p3(4).LineWidth = 2;
        ylim([0 1.1])
        xlabel('$x$','Interpreter','latex','FontSize',15)
        ylabel('$y$','Interpreter','latex','FontSize',15,'Rotation',0)
        title('$t = 10$','Interpreter','latex','FontSize',15)
        get(gca);set(gca,'FontSize',15,'FontName','Arial');
```

Plot 1D numerical solution for the system at t = 20

```
        %subplot_tight(2.5,2,4, [0.1 0.1])
        subplot(2, 2, 4)
        hold on
        p4 = plot(xmesh,sol(100,:,1),'k',...
             xmesh,sol(100,:,2),'m--',...
             xmesh,sol(100,:,3),'b-.',...
             xmesh,sol(100,:,4),'r.',...
             xmesh,sol(100,:,5),'gd')

%         plot(xmesh, 1-sol(100,:,1)-sol(100,:,2)-sol(100,:,3),'c')

        p4(1).LineWidth = 2;
        p4(2).LineWidth = 2;
        p4(3).LineWidth = 2;
        p4(4).LineWidth = 2;
        ylim([0 1.1])
        xlabel('$x$','Interpreter','latex','FontSize',15)
        ylabel('$y$','Interpreter','latex','FontSize',15,'Rotation',0)
        title('$t = 15$','Interpreter','latex','FontSize',15)
        get(gca);set(gca,'FontSize',15,'FontName','Arial');
```

Export_fig - NOTE; Rename cases

```
        set(gcf, 'color','w','Units','inches','Position', [0 0 10 7]);
        saveas(gcf, 'Plots/Case.png')
```

```
        export_fig ('Plots/Case','-m10','-painters','-png')
end
```

Define system of PDEs

```
    function [a1, a2, a3] = mbpde(x_hat,t_hat,u,DuDx)
```

Unpack variable y into descriptive variables

```
        c   = u(1);
        f   = u(2);
        f_cl= u(3);
        m   = u(4);
        l   = u(5);

        dcdx    = DuDx(1);
        dfdx    = DuDx(2);
        df_cldx = DuDx(3);
        dmdx    = DuDx(4);
        dldx    = DuDx(5);
```

Define PDEs system into the framework of "pdepe"

Logistic growth/ physical space filling TERM

```
        gamma      = exp(-x_hat^2/epsilon);
        spaceCheck = (1-c-f-f_cl);
        growth     = gamma*c*spaceCheck;
```

Cases:

Case 1: Withoutout LOX effect

```
           g = 0;
           h = 0;
           D_l = 0;
           beta_l_hat = 0;
```

Case 2: With LOX, NO haptotaxis effect toward cross-link ECM

```
           g = 0;
           h = beta_f_hat*f*l;
```

Case 3: With LOX AND haptotaxis effect toward cross-link ECM

```
            g = -rho_cl_hat*spaceCheck*c*df_cldx;
            h = beta_f_hat*f*l;
```

Model set up in "pdepe" form:

```
            a1 = [1; 1; 1; 1; 1];

            a2 = [Dc_hat*dcdx-rho_hat*spaceCheck*c*dfdx+g;...
                  0;...
                  0;...
                  Dm_hat*dmdx;...
                  Dl_hat*dldx];

            a3 = [growth;...
                  -alpha_f_hat*m*f+mu_f_hat*spaceCheck-h;...
                  -alpha_f_hat*m*f_cl+h;...
                  beta_m_hat*c-alpha_m_hat*m;
                  beta_l_hat*c-alpha_l_hat*l];

    end
```

Define the initial conditions at t = t0

```
    function u0 = mbic(x)

        sigma = 0.01; % a positive constant, Anderson (2000)

        u0 = [exp(-x^2/sigma);...
              1-exp(-x^2/sigma);...
              0;...
              0;...
              0];
```

Anderson (2000) conditions for case 1

```
       u0 = [exp(-x^2/sigma);...
             1-0.5*exp(-x^2/sigma);...
             0;...
             0.5*exp(-x^2/sigma);...
             0];

    end
```

Define the boundary condions at x = a = 0 and x = b = 1

```
    function [pa, qa, pb, qb] = mbbc(xa,ua,xb,ub,t)
```

Zero-flux in the left edge

```
        pa = [0; 0; 0; 0; 0];
        qa = [1; 1; 1; 1; 1];
```

```
        pb = [ub(1); ub(2)-1; ub(3); ub(4); ub(5)];
        qb = [0; 0; 0; 0; 0];
```

Anderson(2000) conditions: Zero-flux in both edges

```
       pa = [0; 0; 0; 0; 0];
       qa = [1; 1; 1; 1; 1];
       pb = [0; 0; 0; 0; 0];
       qb = [1; 1; 1; 1; 1];

    end

end
```

A.2 Local Sensitivity Analysis File

Contents

- Project: CancerECM mathematical models
- Parameters considered and their base case values:
- Simulation set up
- Create updated parameters
- Create Ycalc matrix for model output at time values
- Plotting

Project: CancerECM mathematical 1D model

```
clear all
```

Parameters considered and their base case values:

```
    Dc_hat      = 0.001  ;
    epsilon     = 0.001  ;
    rho_hat     = 0.003 ;
    rho_cl_hat  = 0.05   ;
    alpha_f_hat = 10     ;
    mu_f_hat    = 0.15   ;
    beta_f_hat  = 18     ;
    Dm_hat      = 0.001  ;
    alpha_m_hat = 0.001  ;
    beta_m_hat  = 0.1    ;
    Dl_hat      = 0.002  ;
    alpha_l_hat = 0.001  ;
    beta_l_hat  = 0.1    ;

vector_of_default_parameters = [Dc_hat, epsilon, rho_hat, rho_cl_hat,...
                                alpha_f_hat, mu_f_hat, beta_f_hat,...
```

```
                                Dm_hat, alpha_m_hat, beta_m_hat,...
                                Dl_hat, alpha_l_hat, beta_l_hat];
```

Simulation set up

```
xmesh = linspace(0,1,500);
tspan = linspace(0,20,100);

num_cases = 2;

number_of_scenarios = num_cases*length(vector_of_default_parameters);
% each of the 13 default params has two cases, up & down by
% sensitive_range*100%

sensitivity_range = 0.1;

for i = 1:number_of_scenarios+1
    parameters(i,:) = vector_of_default_parameters;
end
```

Create updated parameters

```
for j = 1:length(vector_of_default_parameters)
    parameters(j*num_cases,j) =...
    vector_of_default_parameters(j)*(1+sensitivity_range);
    parameters(j*num_cases+1,j) =...
    vector_of_default_parameters(j)*(1-sensitivity_range);
end

Ycalc_baseline = solve_pdepe_CancerECM(parameters(1,:),tspan,0);
[a,b,~] = size(Ycalc_baseline);
% baseline has no change
normalized_change_Y1 =...
 zeros(a,b,2*length(vector_of_default_parameters)+1);
normalized_change_Y2 =...
 zeros(a,b,2*length(vector_of_default_parameters)+1);
normalized_change_Y3 =...
 zeros(a,b,2*length(vector_of_default_parameters)+1);
normalized_change_Y4 =...
 zeros(a,b,2*length(vector_of_default_parameters)+1);
normalized_change_Y5 =...
 zeros(a,b,2*length(vector_of_default_parameters)+1);

for j = 2: number_of_scenarios+1
```

Create Ycalc matrix for model output at time values

```
    Ycalc = solve_pdepe_CancerECM(parameters(j,:),tspan,0);
    % for this example Ycalc has five output vs. distance: Y1 vs. x, Y2,
    % Y3, Y4 and Y5 corresponding to c, f, f_cl, m and l in CancerECM model
    Y1(:,:,j) = Ycalc(:,:,1);
    Y2(:,:,j) = Ycalc(:,:,2);
    Y3(:,:,j) = Ycalc(:,:,3);
    Y4(:,:,j) = Ycalc(:,:,4);
    Y5(:,:,j) = Ycalc(:,:,5);
    change_Y1(:,:,j) = abs(Ycalc(:,:,1)-Ycalc_baseline(:,:,1));
    change_Y2(:,:,j) = abs(Ycalc(:,:,2)-Ycalc_baseline(:,:,2));
    change_Y3(:,:,j) = abs(Ycalc(:,:,3)-Ycalc_baseline(:,:,3));
    change_Y4(:,:,j) = abs(Ycalc(:,:,4)-Ycalc_baseline(:,:,4));
    change_Y5(:,:,j) = abs(Ycalc(:,:,5)-Ycalc_baseline(:,:,5));

end

for j = 1:length(vector_of_default_parameters)
    normalized_change_Y1(:,:,2*j)   =...
     change_Y1(:,:,2*j)./Ycalc_baseline(:,:,1)/sensitivity_range;
    normalized_change_Y1(:,:,2*j+1) =...
     change_Y1(:,:,2*j+1)./Ycalc_baseline(:,:,1)/sensitivity_range;
    normalized_change_Y2(:,:,2*j)   =...
     change_Y2(:,:,2*j)./Ycalc_baseline(:,:,2)/sensitivity_range;
    normalized_change_Y2(:,:,2*j+1) =...
     change_Y2(:,:,2*j+1)./Ycalc_baseline(:,:,2)/sensitivity_range;
    normalized_change_Y3(:,:,2*j)   =...
     change_Y3(:,:,2*j)./Ycalc_baseline(:,:,3)/sensitivity_range;
    normalized_change_Y3(:,:,2*j+1) =...
     change_Y3(:,:,2*j+1)./Ycalc_baseline(:,:,3)/sensitivity_range;
    normalized_change_Y4(:,:,2*j)   =...
    change_Y4(:,:,2*j)./Ycalc_baseline(:,:,4)/sensitivity_range;
    normalized_change_Y4(:,:,2*j+1) =...
     change_Y4(:,:,2*j+1)./Ycalc_baseline(:,:,4)/sensitivity_range;
    normalized_change_Y5(:,:,2*j)   =...
     change_Y5(:,:,2*j)./Ycalc_baseline(:,:,5)/sensitivity_range;
    normalized_change_Y5(:,:,2*j+1) =...
     change_Y5(:,:,2*j+1)./Ycalc_baseline(:,:,5)/sensitivity_range;
end
```

Plotting

Y1-cancer cells

```
    subplot(4, 2, 1)
```

```
for i = 1:length(tspan)
increasingchange=change_Y1(i,:,2:2:number_of_scenarios+1);
decreasingchange=change_Y1(i,:,3:2:number_of_scenarios+1);

[maxinc,index_maxinc]=max(increasingchange,[],2);
index_maxinc = reshape(index_maxinc,13,1);
increasingVStime(:,i)=reshape(maxinc,13,1)./sensitivity_range;

[maxdec,index_maxdec] = max(decreasingchange,[],2);
index_maxdec = reshape(index_maxdec,13,1);
decreasingVStime(:,i)=reshape(maxdec,13,1)./sensitivity_range;
end

hold on
for i = 1:7
    plot(tspan,increasingVStime(i,:),'linewidth',1)
end
for i = 8:13
    plot(tspan,increasingVStime(i,:),':','linewidth',2)
end
plot(tspan, ones(size(tspan)),'k--','linewidth',1)
ylim([0 3.5])

xlabel('$t$','Interpreter','latex','FontSize',15)
ylabel('Sensitivity','Interpreter','latex','FontSize',15)
title('(A) cancer cells','Interpreter','latex','FontSize',15)
```

Y2-regular ECM

```
subplot(4, 2, 2)

for i = 1:length(tspan)
increasingchange=change_Y2(i,:,2:2:number_of_scenarios+1);
decreasingchange=change_Y2(i,:,3:2:number_of_scenarios+1);

[maxinc,index_maxinc]=max(increasingchange,[],2);
index_maxinc = reshape(index_maxinc,13,1);
increasingVStime(:,i)=reshape(maxinc,13,1)./sensitivity_range;

[maxdec,index_maxdec] = max(decreasingchange,[],2);
index_maxdec = reshape(index_maxdec,13,1);
decreasingVStime(:,i)=reshape(maxdec,13,1)./sensitivity_range;
end

hold on
```

```
for i = 1:7
    plot(tspan,increasingVStime(i,:),'linewidth',1)
end
for i = 8:13
    plot(tspan,increasingVStime(i,:),':','linewidth',2)
end
plot(tspan, ones(size(tspan)),'k--','linewidth',1)
ylim([0 1.5])


xlabel('$t$','Interpreter','latex','FontSize',15)
ylabel('Sensitivity','Interpreter','latex','FontSize',15)
title('(B) regular ECM fibers','Interpreter','latex','FontSize',15)
```

Y3-cross-linked ECM

```
subplot(4, 2, 3)

for i = 1:length(tspan)
increasingchange=change_Y3(i,:,2:2:number_of_scenarios+1);
decreasingchange=change_Y3(i,:,3:2:number_of_scenarios+1);

[maxinc,index_maxinc]=max(increasingchange,[],2);
index_maxinc = reshape(index_maxinc,13,1);
increasingVStime(:,i)=reshape(maxinc,13,1)./sensitivity_range;

[maxdec,index_maxdec] = max(decreasingchange,[],2);
index_maxdec = reshape(index_maxdec,13,1);
decreasingVStime(:,i)=reshape(maxdec,13,1)./sensitivity_range;
end

hold on
for i = 1:7
    plot(tspan,increasingVStime(i,:),'linewidth',1)
end
for i = 8:13
    plot(tspan,increasingVStime(i,:),':','linewidth',2)
end
plot(tspan, ones(size(tspan)),'k--','linewidth',1)
ylim([0 1.5])

xlabel('$t$','Interpreter','latex','FontSize',15)
ylabel('Sensitivity','Interpreter','latex','FontSize',15)
title('(C) cross-linked ECM fibers','Interpreter','latex','FontSize',15)
```

```
%     set(gca,'Position',[0.1 .1 0.5 0.1]) %[left bottom width height]
```

Y4-MMP

```
    subplot(4, 2, 4)

    for i = 1:length(tspan)
    increasingchange=change_Y4(i,:,2:2:number_of_scenarios+1);
    decreasingchange=change_Y4(i,:,3:2:number_of_scenarios+1);

    [maxinc,index_maxinc]=max(increasingchange,[],2);
    index_maxinc = reshape(index_maxinc,13,1);
    increasingVStime(:,i)=reshape(maxinc,13,1)./sensitivity_range;

    [maxdec,index_maxdec] = max(decreasingchange,[],2);
    index_maxdec = reshape(index_maxdec,13,1);
    decreasingVStime(:,i)=reshape(maxdec,13,1)./sensitivity_range;
    end

    hold on
    for i = 1:7
        plot(tspan,increasingVStime(i,:),'linewidth',1)
    end
    for i = 8:13
        plot(tspan,increasingVStime(i,:),':','linewidth',2)
    end
    plot(tspan, ones(size(tspan)),'k--','linewidth',1)
    ylim([0 1.5])

    xlabel('$t$','Interpreter','latex','FontSize',15)
    ylabel('Sensitivity','Interpreter','latex','FontSize',15)
    title('(D) MMP','Interpreter','latex','FontSize',15)
```

Y5-LOX

```
    subplot(4, 2, 5)

    for i = 1:length(tspan)
    increasingchange=change_Y5(i,:,2:2:number_of_scenarios+1);
    decreasingchange=change_Y5(i,:,3:2:number_of_scenarios+1);

    [maxinc,index_maxinc]=max(increasingchange,[],2);
    index_maxinc = reshape(index_maxinc,13,1);
    increasingVStime(:,i)=reshape(maxinc,13,1)./sensitivity_range;
```

```
[maxdec,index_maxdec] = max(decreasingchange,[],2);
index_maxdec = reshape(index_maxdec,13,1);
decreasingVStime(:,i)=reshape(maxdec,13,1)./sensitivity_range;
end

hold on
for i = 1:7
    plot(tspan,increasingVStime(i,:),'linewidth',1)
end
for i = 8:13
    plot(tspan,increasingVStime(i,:),':','linewidth',2)
end
plot(tspan, ones(size(tspan)),'k--','linewidth',1)
ylim([0 1.5])

xlabel('$t$','Interpreter','latex','FontSize',15)
ylabel('Sensitivity','Interpreter','latex','FontSize',15)
title('(E) LOX','Interpreter','latex','FontSize',15)
```

YCalc of only Y5-LOX

```
p1 = subplot(4, 2, 6)
time = 100;
hold on
for j = 2:2:14
    plot(xmesh,Y5(time,:,j),'linewidth',1)
end
for j = 16:2:number_of_scenarios+1
    plot(xmesh,Y5(time,:,j),':','linewidth',2)
end
plot(xmesh, Ycalc_baseline(time,:,5),'k--','linewidth',1)
ylim([0 1.5])

xlabel('$x$','Interpreter','latex','FontSize',15)
ylabel('Concentration','Interpreter','latex','FontSize',15)
title('(F) LOX at t = 20','Interpreter','latex','FontSize',15)
```

Set up legend

```
hL    = subplot(4, 2, 7.5);
poshL = get(hL,'position');
leg1 = legend([p1],...
        {'$\hat{D}_c$','$\hat{\epsilon}$','$\hat{\rho}$',...
       '$\hat{\rho}_{cl}$','$\hat{\alpha}_f$','$\hat{\mu}_f$',...
       '$\hat{\beta}_f$', '$\hat{D}_m$','$\hat{\alpha}_m$',...
```

```
           '$\hat{\beta}_m$', '$\hat{D}_l$','$\hat{\alpha}_l$',...
           '$\hat{\beta}_l$','base line'},...
          'Interpreter','latex',...
          'Orientation','horizontal','NumColumns',7); % customize this

    set(leg1,'Interpreter','latex'); %'location','northeastoutside');
    leg1.FontSize = 12
    set(leg1,'position',poshL);      % Adjusting legend's position
axis(hL,'off');
```

Export_fig - NOTE

```
    set(gcf, 'color','w','Units','inches','Position', [0 0 10 11]);
    saveas(gcf, 'LocalSensitivity/FigureCombinedMATLAB.png')
    export_fig('LocalSensitivity/LocalSensitivity','-m10','-painters','-png')
```

Appendix B

CompuCell3D Codes

B.1 XML File

```
<CompuCell3D Revision="20171121" Version="3.7.7">

   <Potts>

      <!-- Basic properties of CPM (GGH) algorithm -->
      <Dimensions x="300" y="300" z="1"/>
      <Steps>30001</Steps>
      <Temperature>50.0</Temperature>
      <NeighborOrder>2</NeighborOrder>
      <LatticeType>Square</LatticeType>

   </Potts>

   <Plugin Name="CellType">

      <!-- Listing all cell types in the simulation -->
      <CellType TypeId="0" TypeName="Medium"/>
      <CellType TypeId="1" TypeName="cell"/>
   </Plugin>

   <Plugin Name="Volume">
      <VolumeEnergyParameters CellType="cell"
      LambdaVolume="1.0" TargetVolume="100"/>
   </Plugin>

   <Plugin Name="CenterOfMass">
   </Plugin>

   <Plugin Name="PixelTracker">
    </Plugin>


   <Plugin Name="Contact">
      <Energy Type1="Medium" Type2="Medium">0.0</Energy>
      <Energy Type1="Medium" Type2="cell">32.0</Energy>
```

```
    <Energy Type1="cell" Type2="cell">40.0</Energy>
    <NeighborOrder>2</NeighborOrder>
</Plugin>

<Plugin Name="Chemotaxis">
    <ChemicalField Name="fiber"
    Source="ReactionDiffusionSolverFE">
        <ChemotaxisByType Lambda="500.0" Type="cell"/>
    </ChemicalField>

    <ChemicalField Name="fiber_cl"
    Source="ReactionDiffusionSolverFE">
        <ChemotaxisByType Lambda="1000.0" Type="cell"/>
    </ChemicalField>
</Plugin>

<Steppable Type="ReactionDiffusionSolverFE">
    <DiffusionField Name="fiber">
        <DiffusionData>
            <FieldName>fiber</FieldName>
            <DiffusionConstant>0</DiffusionConstant>
            <ExtraTimesPerMCS>175</ExtraTimesPerMCS>
            <AdditionalTerm>-7.50E-03*MMP*fiber
            -1.35E-02*LOX*fiber
            +1.13E-04*(1- fiber-fiber_cl
            -(CellType==1 ? 1: 0))</AdditionalTerm>
        </DiffusionData>

        <BoundaryConditions>
            <Plane Axis="X">
                <ConstantValue PlanePosition="Min" Value="1.0"/>
                <ConstantValue PlanePosition="Max" Value="1.0"/>
            </Plane>
            <Plane Axis="Y">
                <ConstantValue PlanePosition="Min" Value="1.0"/>
                <ConstantValue PlanePosition="Max" Value="1.0"/>
            </Plane>
        </BoundaryConditions>

    </DiffusionField>

    <DiffusionField Name="fiber_cl">
        <DiffusionData>
            <FieldName>fiber_cl</FieldName>
            <DiffusionConstant>0</DiffusionConstant>
```

```
        <DecayConstant>0</DecayConstant>
        <ExtraTimesPerMCS>175</ExtraTimesPerMCS>
        <AdditionalTerm>-7.50E-03*MMP*fiber_cl
        +1.35E-02*LOX*fiber</AdditionalTerm>
    </DiffusionData>

    <BoundaryConditions>
        <Plane Axis="X">
            <ConstantValue PlanePosition="Min" Value="0.0"/>
            <ConstantValue PlanePosition="Max" Value="0.0"/>
        </Plane>
        <Plane Axis="Y">
            <ConstantValue PlanePosition="Min" Value="0.0"/>
            <ConstantValue PlanePosition="Max" Value="0.0"/>
        </Plane>
    </BoundaryConditions>
</DiffusionField>

<DiffusionField Name="MMP">
    <DiffusionData>
        <FieldName>MMP</FieldName>
        <DiffusionConstant>21.6</DiffusionConstant>
        <ExtraTimesPerMCS>175</ExtraTimesPerMCS>
        <AdditionalTerm>-7.50E-07*MMP
        +(CellType==1 ? 7.50E-05: 0)</AdditionalTerm>
    </DiffusionData>

    <BoundaryConditions>
        <Plane Axis="X">
            <ConstantValue PlanePosition="Min" Value="0.0"/>
            <ConstantValue PlanePosition="Max" Value="0.0"/>
        </Plane>
        <Plane Axis="Y">
            <ConstantValue PlanePosition="Min" Value="0.0"/>
            <ConstantValue PlanePosition="Max" Value="0.0"/>
        </Plane>
    </BoundaryConditions>
</DiffusionField>

<DiffusionField Name="LOX">
    <DiffusionData>
        <FieldName>LOX</FieldName>
        <DiffusionConstant>43.2</DiffusionConstant>
        <ExtraTimesPerMCS>175</ExtraTimesPerMCS>
        <AdditionalTerm>-7.50E-07*LOX
```

```
            +(CellType==1 ? 7.50E-05: 0)</AdditionalTerm>
        </DiffusionData>

        <BoundaryConditions>
            <Plane Axis="X">
                <ConstantValue PlanePosition="Min" Value="0.0"/>
                <ConstantValue PlanePosition="Max" Value="0.0"/>
            </Plane>
            <Plane Axis="Y">
                <ConstantValue PlanePosition="Min" Value="0.0"/>
                <ConstantValue PlanePosition="Max" Value="0.0"/>
            </Plane>
        </BoundaryConditions>
    </DiffusionField>

</Steppable>

<Steppable Type="BlobInitializer">
    <Region>
        <Gap>0</Gap>
        <Width>10</Width>
        <Radius>50</Radius>
        <Center x="150" y="150" z="0"/>
        <Types>cell</Types>
    </Region>
</Steppable>

</CompuCell3D>
```

B.2 Main Python File

```
import sys
from os import environ
from os import getcwd
import string

sys.path.append(environ["PYTHON_MODULE_PATH"])

import CompuCellSetup

sim,simthread = CompuCellSetup.getCoreSimulationObjects()

# add extra attributes here

CompuCellSetup.initializeSimulationObjects(sim,simthread)
```

```
# Definitions of additional Python-managed fields go here

#Add Python steppables here
steppableRegistry=CompuCellSetup.getSteppableRegistry()

from HaptotaxisTest2DSteppables import HaptotaxisTest2DSteppable
steppableInstance=HaptotaxisTest2DSteppable(sim,_frequency=1)
steppableRegistry.registerSteppable(steppableInstance)

#*****FIBER CONCENTRATINON CASES STUDY******

from HaptotaxisTest2DSteppables \
        import FiberConcentrationCaseARandom50
instanceOfFiberConcentrationCaseARandom50 \
        =FiberConcentrationCaseARandom50(_simulator=sim,_frequency=1)
steppableRegistry.registerSteppable \
        (instanceOfFiberConcentrationCaseARandom50)

#from HaptotaxisTest2DSteppables \
        #import FiberConcentrationCaseAUniform50
#instanceOfFiberConcentrationCaseAUniform50 \
        #=FiberConcentrationCaseAUniform50(_simulator=sim,_frequency=1)
#steppableRegistry.registerSteppable \
        #(instanceOfFiberConcentrationCaseAUniform50)

#from HaptotaxisTest2DSteppables \
         #import FiberConcentrationCaseBRandom25
#instanceOfFiberConcentrationCaseBRandom25 \
        #=FiberConcentrationCaseBRandom25(_simulator=sim,_frequency=1)
#steppableRegistry.registerSteppable \
        #(instanceOfFiberConcentrationCaseBRandom25)

#from HaptotaxisTest2DSteppables \
        #import FiberConcentrationCaseBRandom50
#instanceOfFiberConcentrationCaseBRandom50 \
        #=FiberConcentrationCaseBRandom50(_simulator=sim,_frequency=1)
#steppableRegistry.registerSteppable \
        #(instanceOfFiberConcentrationCaseBRandom50)

#from HaptotaxisTest2DSteppables \
        #import FiberConcentrationCaseBRandom75
#instanceOfFiberConcentrationCaseBRandom75 \
        #=FiberConcentrationCaseBRandom75(_simulator=sim,_frequency=1)
#steppableRegistry.registerSteppable \
        #(instanceOfFiberConcentrationCaseBRandom75)
```

```
#from HaptotaxisTest2DSteppables \
        #import FiberConcentrationCaseCUniform25
#instanceOfFiberConcentrationCaseCUniform25 \
        #=FiberConcentrationCaseCUniform25(_simulator=sim,_frequency=1)
#steppableRegistry.registerSteppable \
        #(instanceOfFiberConcentrationCaseCUniform25)

#from HaptotaxisTest2DSteppables \
        #import FiberConcentrationCaseCUniform75
#instanceOfFiberConcentrationCaseCUniform75 \
        #=FiberConcentrationCaseCUniform75(_simulator=sim,_frequency=1)
#steppableRegistry.registerSteppable \
        #(instanceOfFiberConcentrationCaseCUniform75)

#*****FIBER CONCENTRATINON CASES STUDY******

from HaptotaxisTest2DSteppables import LogData
instanceOfLogData=LogData(_simulator=sim,_frequency=100)
steppableRegistry.registerSteppable(instanceOfLogData)

#from HaptotaxisTest2DSteppables import ChemotaxisTest
#instanceOfChemotaxisTest=ChemotaxisTest(_simulator=sim,_frequency=1)
#steppableRegistry.registerSteppable(instanceOfChemotaxisTest)

CompuCellSetup.mainLoop(sim,simthread,steppableRegistry)
```

B.3 Steppables Python File

```
from PySteppables import *
import CompuCell
import sys
import random

from PlayerPython import *
import CompuCellSetup
from math import *
import numpy as np
from random import uniform

class HaptotaxisTest2DSteppable(SteppableBasePy):

    def __init__(self,_simulator,_frequency=1):
        SteppableBasePy.__init__(self,_simulator,_frequency)
    def start(self):
```

```
        # any code in the start function runs before MCS=0
        pass
    def step(self,mcs):
        pass

class FiberConcentrationCaseARandom50(SteppableBasePy):
    def __init__(self,_simulator,_frequency=1):
        SteppableBasePy.__init__(self,_simulator,_frequency)

    def start(self):
        field_fiber    = self.getConcentrationField("fiber")
        field_fiber_cl = self.getConcentrationField('fiber_cl')

        for x,y,z in self.everyPixel():
            field_fiber[x,y,z]  = random.uniform(0, 1)
            field_fiber_cl[x,y,z]  = 0

    def step(self,mcs):
        fiber_concentration    = 0
        fiber_cl_concentration = 0

        field_fiber    = self.getConcentrationField("fiber")
        field_fiber_cl = self.getConcentrationField('fiber_cl')

        for x,y,z in self.everyPixel():
            fiber_concentration    += field_fiber[x,y,z]
            fiber_cl_concentration += field_fiber_cl[x,y,z]

        fileName='CaseA_Random_50.csv'
        try:
            fileHandle,fullFileName=self.open \
                FileInSimulationOutputDirectory(fileName,"a")
        except IOError:
            print "Could not open file ", fileName," for writing. "
            return

        print >>fileHandle,mcs,",", \
                fiber_concentration,",", \
                fiber_cl_concentration
        fileHandle.close()

    def finish(self):
        return

class FiberConcentrationCaseAUniform50(SteppableBasePy):
```

```
def __init__(self,_simulator,_frequency=1):
    SteppableBasePy.__init__(self,_simulator,_frequency)

def start(self):
    field_fiber    = self.getConcentrationField("fiber")
    field_fiber_cl = self.getConcentrationField('fiber_cl')

    #****Average 0.5 everywhere:
    for x,y,z in self.everyPixel():
        field_fiber[x,y,z]  = 0.5
            field_fiber_cl[x,y,z]  = 0

    #****Half  0 Half 1
    #for x,y,z in self.everyPixel():
    #    if x >=150:
    #        field_fiber[x,y,z]  = 1
    #        field_fiber_cl[x,y,z]  = 0
    #    else:
    #        field_fiber[x,y,z]  = 0
    #        field_fiber_cl[x,y,z]  = 0

def step(self,mcs):
    fiber_concentration    = 0
    fiber_cl_concentration = 0

    field_fiber    = self.getConcentrationField("fiber")
    field_fiber_cl = self.getConcentrationField('fiber_cl')

    for x,y,z in self.everyPixel():
        fiber_concentration    += field_fiber[x,y,z]
        fiber_cl_concentration += field_fiber_cl[x,y,z]

    fileName='CaseA_Uniform_50.csv'
    try:
        fileHandle,fullFileName=self.open \
            FileInSimulationOutputDirectory(fileName,"a")
    except IOError:
        print "Could not open file ", fileName," for writing. "
        return

    print >>fileHandle,mcs,",", \
            fiber_concentration,",", \
            fiber_cl_concentration
    fileHandle.close()
```

```
    def finish(self):
        return

class FiberConcentrationCaseBRandom25(SteppableBasePy):
    def __init__(self,_simulator,_frequency=1):
        SteppableBasePy.__init__(self,_simulator,_frequency)

    def start(self):
        field_fiber      = self.getConcentrationField("fiber")
        field_fiber_cl = self.getConcentrationField('fiber_cl')

        for x,y,z in self.everyPixel():
            field_fiber[x,y,z]  = random.uniform(0, 0.5)
            field_fiber_cl[x,y,z]  = 0

    def step(self,mcs):
        fiber_concentration     = 0
        fiber_cl_concentration = 0

        field_fiber      = self.getConcentrationField("fiber")
        field_fiber_cl = self.getConcentrationField('fiber_cl')

        for x,y,z in self.everyPixel():
            fiber_concentration     += field_fiber[x,y,z]
            fiber_cl_concentration += field_fiber_cl[x,y,z]

        fileName='CaseB_Random_25.csv'
        try:
            fileHandle,fullFileName=self.open \
                FileInSimulationOutputDirectory(fileName,"a")
        except IOError:
            print "Could not open file ", fileName," for writing. "
            return

        print >>fileHandle,mcs,",", \
                fiber_concentration,",", \
                fiber_cl_concentration
        fileHandle.close()

    def finish(self):
        return

class FiberConcentrationCaseBRandom50(SteppableBasePy):
    def __init__(self,_simulator,_frequency=1):
        SteppableBasePy.__init__(self,_simulator,_frequency)
```

```
    def start(self):
        field_fiber      = self.getConcentrationField("fiber")
        field_fiber_cl = self.getConcentrationField('fiber_cl')

        for x,y,z in self.everyPixel():
            field_fiber[x,y,z]  = random.uniform(0.25, 0.75)
            field_fiber_cl[x,y,z]  = 0

    def step(self,mcs):
        fiber_concentration     = 0
        fiber_cl_concentration = 0

        field_fiber      = self.getConcentrationField("fiber")
        field_fiber_cl = self.getConcentrationField('fiber_cl')

        for x,y,z in self.everyPixel():
            fiber_concentration     += field_fiber[x,y,z]
            fiber_cl_concentration += field_fiber_cl[x,y,z]

        fileName='CaseB_Random_50.csv'
        try:
            fileHandle,fullFileName=self.open \
                FileInSimulationOutputDirectory(fileName,"a")
        except IOError:
            print "Could not open file ", fileName," for writing. "
            return

        print >>fileHandle,mcs,",", \
                fiber_concentration,",", \
                fiber_cl_concentration
        fileHandle.close()

    def finish(self):
        return

class FiberConcentrationCaseBRandom75(SteppableBasePy):
    def __init__(self,_simulator,_frequency=1):
        SteppableBasePy.__init__(self,_simulator,_frequency)

    def start(self):
        field_fiber      = self.getConcentrationField("fiber")
        field_fiber_cl = self.getConcentrationField('fiber_cl')

        for x,y,z in self.everyPixel():
```

```
            field_fiber[x,y,z]   = random.uniform(0.5,1)
            field_fiber_cl[x,y,z]  = 0

    def step(self,mcs):
        fiber_concentration      = 0
        fiber_cl_concentration = 0

        field_fiber      = self.getConcentrationField("fiber")
        field_fiber_cl = self.getConcentrationField('fiber_cl')

        for x,y,z in self.everyPixel():
            fiber_concentration     += field_fiber[x,y,z]
            fiber_cl_concentration += field_fiber_cl[x,y,z]

        fileName='CaseB_Random_75.csv'
        try:
            fileHandle,fullFileName=self.open \
                FileInSimulationOutputDirectory(fileName,"a")
        except IOError:
            print "Could not open file ", fileName," for writing. "
            return

        print >>fileHandle,mcs,",", \
                fiber_concentration,",", \
                fiber_cl_concentration
        fileHandle.close()

    def finish(self):
        return

class FiberConcentrationCaseCUniform25(SteppableBasePy):
    def __init__(self,_simulator,_frequency=1):
        SteppableBasePy.__init__(self,_simulator,_frequency)

    def start(self):
        field_fiber      = self.getConcentrationField("fiber")
        field_fiber_cl = self.getConcentrationField('fiber_cl')

        #****Average 0.25 everywhere:
        for x,y,z in self.everyPixel():
            field_fiber[x,y,z]  = 0.25
            field_fiber_cl[x,y,z]  = 0

        #****Half  0 Half 1
        #for x,y,z in self.everyPixel():
```

```
            #    if x >=150:
            #        field_fiber[x,y,z]   = 0.5
            #        field_fiber_cl[x,y,z]   = 0
            #    else:
            #        field_fiber[x,y,z]   = 0
            #        field_fiber_cl[x,y,z]   = 0

    def step(self,mcs):
        fiber_concentration     = 0
        fiber_cl_concentration = 0

        field_fiber      = self.getConcentrationField("fiber")
        field_fiber_cl = self.getConcentrationField('fiber_cl')

        for x,y,z in self.everyPixel():
            fiber_concentration     += field_fiber[x,y,z]
            fiber_cl_concentration += field_fiber_cl[x,y,z]

        fileName='CaseC_Uniform_25.csv'
        try:
            fileHandle,fullFileName=self.open \
                FileInSimulationOutputDirectory(fileName,"a")
        except IOError:
            print "Could not open file ", fileName," for writing. "
            return

        print >>fileHandle,mcs,",", \
                fiber_concentration,",", \
                fiber_cl_concentration
        fileHandle.close()

    def finish(self):
        return

class FiberConcentrationCaseCUniform75(SteppableBasePy):
    def __init__(self,_simulator,_frequency=1):
        SteppableBasePy.__init__(self,_simulator,_frequency)

    def start(self):
        field_fiber      = self.getConcentrationField("fiber")
        field_fiber_cl = self.getConcentrationField('fiber_cl')

        #****Average 0.25 everywhere:
        for x,y,z in self.everyPixel():
            field_fiber[x,y,z]   = 0.75
```

```
            field_fiber_cl[x,y,z]  = 0

        #****Half  0 Half 1
        #for x,y,z in self.everyPixel():
         #    if x >=150:
         #        field_fiber[x,y,z]  = 1
         #        field_fiber_cl[x,y,z]  = 0
         #    else:
         #        field_fiber[x,y,z]  = 0.5
         #        field_fiber_cl[x,y,z]  = 0

    def step(self,mcs):
        fiber_concentration     = 0
        fiber_cl_concentration = 0

        field_fiber      = self.getConcentrationField("fiber")
        field_fiber_cl = self.getConcentrationField('fiber_cl')

        for x,y,z in self.everyPixel():
            fiber_concentration     += field_fiber[x,y,z]
            fiber_cl_concentration += field_fiber_cl[x,y,z]

        fileName='CaseC_Uniform_75.csv'
        try:
            fileHandle,fullFileName=self.open \
                FileInSimulationOutputDirectory(fileName,"a")
        except IOError:
            print "Could not open file ", fileName," for writing. "
            return

        print >>fileHandle,mcs,",", \
                fiber_concentration,",", \
                fiber_cl_concentration
        fileHandle.close()

    def finish(self):
        return

class LogData(SteppableBasePy):
    def __init__(self,_simulator,_frequency=10):
        SteppableBasePy.__init__(self,_simulator,_frequency)

    def start(self):
        IDCount = 1
```

```
        for cell in self.cellListByType(1):
            cell_attribute=self.getDictionaryAttribute(cell)

            # Way to count the amount of
                #generalized cells for a given cell type
            cell_attribute["id"] = IDCount
            IDCount += 1

    def step(self,mcs):

        #*****For every cell_ID of the same cell type,
                #log cell position in term of xCOM and yCOM
        for cell in self.cellListByType(1):
            cell_attribute=self.getDictionaryAttribute(cell)

            #Log data into multiple separate cvs file by cell_ID
            #fileName='CellPosition_COM_' \
                #+str(cell_attribute["id"])+'.csv'

            #Log all data into one cvs file
            fileName='CellPosition_COM.csv'

            try:
                fileHandle,fullFileName=self.open \
                        FileInSimulationOutputDirectory(fileName,"a")
            except IOError:
                print "Could not open file ", fileName," for writing. "
                return

            cell_attribute=self.getDictionaryAttribute(cell)
            print >>fileHandle, cell.id,",", \
                mcs,",", \
                cell.xCOM,",", \
                cell.yCOM
            fileHandle.close()

        #******Log MMP and LOX Concentration
        MMP = 0
        fieldMMP=self.getConcentrationField("MMP")
        for x in xrange(self.dim.x):
            for y in xrange(self.dim.y):
                for z in xrange(self.dim.z):
                    MMP += fieldMMP[x,y,z];

        fileName='MMP_LOX.csv'
```

```
        try:
            fileHandle,fullFileName=self.open \
                FileInSimulationOutputDirectory(fileName,"a")
        except IOError:
            print "Could not open file", fileName," for writing."
            return

        print >>fileHandle,mcs,",", MMP
        fileHandle.close()

    def finish(self):
        return

class ChemotaxisTest(SteppableBasePy):
    def __init__(self,_simulator,_frequency=1):
        SteppableBasePy.__init__(self,_simulator,_frequency)

    def start(self):
        field_fiber      = self.getConcentrationField("fiber")
        field_fiber_cl = self.getConcentrationField('fiber_cl')

        #for x,y,z in self.everyPixel():
        #    field_fiber[x,y,z]  = 0.5
        #    field_fiber_cl[x,y,z]  = 0

        # The half uncrosslink and half crosslink test
        for x,y,z in self.everyPixel():
            if x >=150:
                #field_fiber[x,y,z]  = random.uniform(0,1)
                field_fiber[x,y,z]  = 0.5
                field_fiber_cl[x,y,z]  = 0
            else:
                field_fiber[x,y,z]  = 0
                #field_fiber_cl[x,y,z]  = random.uniform(0,1)
                field_fiber_cl[x,y,z]  = 0.5
```

VITA

YEN T. NGUYEN

Candidate for the Degree of

Master of Science

Thesis: COMPUTATIONAL MODELING OF METASTATIC CANCER MIGRATION THROUGH A REMODELING EXTRACELLULAR MATRIX

Major Field: Chemical Engineering

Biographical:

Education:

Completed the requirement for the Master of Science in Chemical Engineering at Oklahoma State University, Stillwater, Oklahoma in July 2018.

Completed the requirement for the Bachelor of Science in Chemical Engineering at Oklahoma State University, Stillwater, Oklahoma in May 2017.

Experience & Honors:

OSU CEAT Academic Coach in 2015-2016
OSU ChE Grader in 2016
Undergraduate Research Assistant in 2016-2017
Graduate Teaching Assistant in 2017-2018
Women's Faculty Council Students Research Award Winner in 2018
William Cunningham Award for National AIChE Student Design Competition–Team Category in 2017

Professional Membership:

American Institute of Chemical Engineers (AIChE)
Biomedical Engineering Society (BMES)
Society of Women Engineers (SWE)
Omega Chi Epsilon –The National Honor Society for Chemical Engineering
Pi Mu Epsilon –U.S. Honorary National Mathematics Society
Phi Kappa Phi –Honor Society