8-2021

Roughening interfaces in spatial population dynamics

Daniel E. Castillo
University of Tennessee, Knoxville, dcastil1@vols.utk.edu

To the Graduate Council:

I am submitting herewith a dissertation written by Daniel E. Castillo entitled "Roughening interfaces in spatial population dynamics." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Physics.

Maxim O. Lavrentovich, Major Professor

We have read this dissertation and recommend its acceptance:

Kenneth Read, Lucas Platter, Alex Bentley

Accepted for the Council:

Dixie L. Thompson

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

Roughening interfaces in spatial population dynamics

A Dissertation Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Daniel Castillo

August 2021

Acknowledgments

I'd like to thank Dr. Maxim Lavrentovich for his guidance and support while writing this thesis. It was not easy jumping into a whole new area of research in physics half-way through my PhD career, and I owe him graititude for his patience and the valuable instruction I received from him throughout that process. Additionally, I'd like to acknowlege the contribution of my committee members to formulating this thesis; by asking tough questions they forced me think more deeply about the real-life applications and motivations of this work.

I'd also like to acknowledge Dr. George Siopsis, for everything I learned from him while working in his research group for the first few years of my graduate career. I learned so much from him about quantum mechanics, quntum information, and quantum computation, and I am grateful for the opportunity to study these fascinating topics.

I would also like to thank my best friend Keith Warren. Not only has he provided support both emotional and financial for the seven long years of my PhD, but also in the years before while I worked on my undergraduate degrees. I'd also like to thank my dogs, Dixie and Data, for being wonderful companions and for putting up with me not taking them on as many walks as they'd like while I worked on this thesis. I'd also like to acknowledge the enduring support of my entire family, especially my parents.

Finally, I owe a tremendous amount of gratitude to the friends and companions who have played an important role in not only my academic and professional development, but in my personal development as well. Those who have had a direct impact are too many to list, but I'd especially like to thank Dr. Bobby Carroll, Dr. Sean Burcher, Dr. Joseph Heideman, and Dr. Phil Dee for being sources of motivation througout my PhD, for countless nights of stimulating conversation, and for their continued friendship.

Abstract

The spatial structure and geometry of biological systems can have a strong effect on that system's evolutionary dynamics. In particular, spatially structured populations may invade one another, giving rise to invasion fronts that may exhibit qualitatively different evolutionary dynamics in different dimensions or geometric configurations. For examples of invasion fronts arising in nature, one might think of a thin layer of bacteria cells growing on a Petri dish, an animal species expanding into new territory, or a cancerous tumor growing into and competing with the surrounding healthy tissue. Perhaps the most well-studied class of invasion fronts in population genetics is the Fisher wave, which was developed to explain how an advantageous gene sweeps throughout a population.

In this thesis, I will focus on the study of invasion fronts which develop an enhanced roughness due to internal dynamics of the invading population; namely, I make use of simple lattice and analytic models to explore how the interface between an unstable, mutating population and a healthy bystander population develops an enhanced roughness as the mutating population approaches a population collapse via mutational meltdown. I track the roughening of an interface by defining a width function on the interface which evolves in time according to a power-law. Departure from the expected diffusive power-law behavior will serve as an indicator for enhanced roughness of the interface. I will do this analysis for flat interfaces in 1 and 2 spatial dimensions. I will additionally show some qualitative results for the case of 3 spatial dimensions.

Table of Contents

List of Figures

Chapter 1

Introduction

The spatial structure and dimensionality of a biological system can have a strong affect on the evolutionary dynamics of that system. In particular, macroscopic phenomena such as travelling wave fronts of a cellular population invading into a new territory can have qualitatively different dynamics as the dimension of the front changes. Understanding the dynamics of travelling wave fronts is interesting in general as such fronts can arise in a myriad of scientific disciplines and phenomena, such as the growth of a bacteria colony (Fig. 1.2(a)), the growth of silicon thin films on a substrate, the propagation of flame or chemical reaction, the invasion of new territory by an animal population in an ecological habitat, the spread of a virus throughout a susceptible population, or the growth of a tumor as it fights for space and resources with the surrounding tissue (Fig. 1.1). In this thesis, I wish to pay particular attention to the evolutionary dynamics of invading population fronts, with a special focus on the roughening properties of such fronts. Indeed, as the fronts propagate in space and time, they typically "roughen" and develop undulations in their shape. We shall see in this thesis that such undulations may be important signatures of the dynamics occurring within the wave.

For the remainer of this chapter, I will introduce the necessary details in order to talk about roughening invasion fronts. First, I will give a brief review of well-mixed population genetics, and I will then introduce the idea of spatially structured populations. Next, I will discuss population invasion fronts and roughening phenomena. In Ch.2 I will discuss the dynamics of roughening invasion fronts in 1 and 2 spatial dimensions. The particular focus

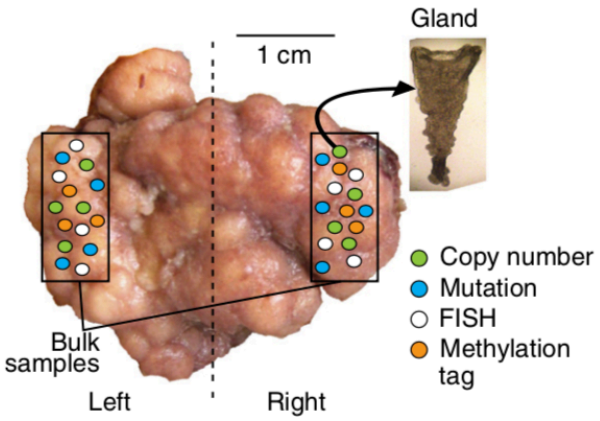

Figure 1.1: Spatially-resolved tumor showing an irregular, "rough" shape, taken from Ref. [1]. The colored dots in the figure are meant to indicate mutational heterogeneities present in the tumor from the early stages of tumor growth. The presence of such heterogeneities support the so-called Big Bang model for tumor growth [1].

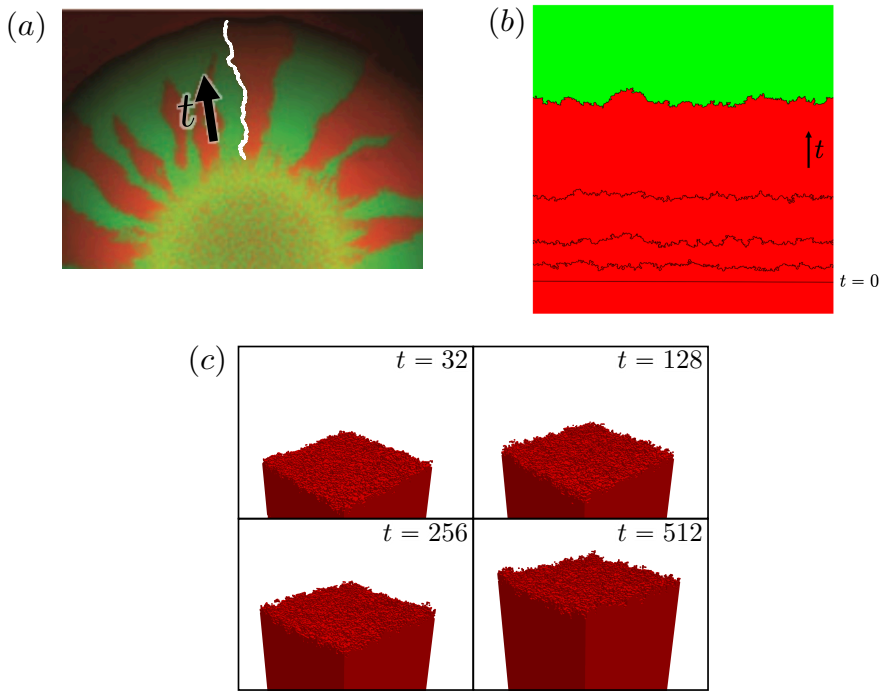

Figure 1.2: Invading population fronts in $d = 1,\ 2,$ and 3 spatial dimensions. In each of the subfigures, the time direction is denoted by the black arrow. **(a)** A micrograph of a microbial colony (*E. coli*) of two fluorescently labeled, selectively neutral strains, adapted from Ref. [2]. The colony grows outward in the direction of the arrow, with the actively-growing population confined to the colony rim. Thus, the growing population has effective spatial dimension $D = 1$ and we can understand the evolution of the system as a $d = 1+1$ dimensional process, with the $+1$ indicating the time-like direction in which the overall population grows. After starting from an initially mixed state, the two species tend to segregate and the boundary between two species is a single point whose time evolution is shown by the white line. **(b)** A computer simulation of a two-dimensional population on a lattice, where each lattice site contains an actively-dividing cell which displaces a neighboring cell after each cell division. The lattice is initialized with half green and half red cells, the latter enjoying a selective advantage. In this $D = 2$ spatial dynamics, we see a single snapshot of the advancing species (red) as it invades the green species due to the selective advantage. The black lines show the position of the interface between strains, starting at the initial state at $t = 0$ where the two species are separated by a straight line. **(c)** The same computer simulation in (b), implemented for a three-dimensional population on a lattice. This time, the two species are initially separated by a flat, two-dimensional interface. For these $D = 3$ dynamics, we see a time series for the invasion of the advancing species (red) as it invades the space initially occupied by the green species (not shown). The invasion front in each picture can be thought of as the surface defined by the furthest extent of the red cells into the green strain territory.

will be on unstable, mutation populations invading stable, 'bystander' populations. In Ch.3 I will compare the bystander model introduced in Ch.2 with biological experiments, with a focus on tumor growth and development. Next, in Ch.4, I will discuss the implementation of the bystander model using lattice simulations in C++, including a discussion of the algorithms used and optimization. Finally, I conclude with Ch.5, where I will review the most important aspects and results of the thesis.

1.1 Well-mixed population genetics

In population genetics, an invasion front can arise whenever one population grows into the territory of another population where both populations compete and fight for resources. Thus, our system will consist of a number of distinct "species", which for the concerns of this thesis will be assumed to represent distinct alleles of a single genetic locus. Each species will have its own growth rate, depending on the selective advantage of each species' phenotype. The spatial structure of the system being considered, along with the spatial distribution of the present populations, will determine which individuals compete with one another for space and resources. The dimensionality of the population's spatial structure will also determine the dimensionality of the invasion front; in general, an invading population initially distributed over D spatial dimensions will give rise to a $D-1$ dimensional invasion front, as shown in Fig. 1.2. For example, in a microbial colony grown on a Petri dish, the actively-dividing population is confined to the effectively one-dimensional population rim as the interior of the population will typically not receive enough nutrients to grow. In this case, strains within that actively-growing population will develop point-like interfaces. As these populations grow, these point-like interfaces will wander, creating the "genetic sector" pattern shown in Fig. 1.2(a). If, on the other hand, the population on the Petri dish is constantly supplied with nutrients, then the effectively two-dimensional actively growing populations will develop line-like (one-dimensional) interfaces between strains, as shown with a computer simulation in Fig. 1.2(b). For three-dimensional populations such as actively growing tumors, strains within such populations will develop more complex interfaces, such as undulated (effectively two-dimensional) surfaces. As we will see in Ch. 2 and Ch. 3, the spatial structure and

dimensionality of the invasion front will strongly affect how that front evolves over time, thus having an effect on the determination of whether the invasion was successful. It is clear that there will be a wide range of possibilities for both the dimensionality and geometry of the population, as various factors such as the nutrient conditions, local environment, and growth substrate will all have an effect on the spatial population structure.

Before I can discuss the influence of spatial structure on evolutionary dynamics in general, it is first necessary to understand the dynamics of systems with no spatial structure, i.e., well-mixed systems. I will then move on to discuss spatially structured systems, with a particular focus on the behavior of invading fronts. In well-mixed populations, it is understood that each individual is equally likely to interact and compete with every other individual in the medium, as any cell within the population is equally likely to be near any other one when the population is well-stirred. Due to the typically large number of actively-dividing cells in such populations, the population densities of each species type in such well-mixed experiments evolve in a deterministic manner according to a particular set of coupled differential equations which accurately capture all the dynamics of the evolution. As an illustrative example, consider the simple case of a well-mixed test tube initially containing equal quantities of some species A and another species B with three possible "reactions" representing the outcomes of strain-strain competition and possible mutation:

$$\begin{cases} A + B & \xrightarrow{k_A} A + A \\ A + B & \xrightarrow{k_B} B + B \\ A & \xrightarrow{\mu} B \end{cases} \tag{1}$$

The reaction rates k_A, k_B represent the relative growth rates, respectively, of species A and species B, while μ is the rate at which species A mutates into species B. Suppose we know that cells of species A will produce another cell of the same species at the rate Γ_A, and that species B will similarly reproduce at the rate Γ_B. Thus, the relative growth rates k_A, k_B can be written as:

$$k_A = \frac{\Gamma_A}{\Gamma_A + \Gamma_B};\ k_B = \frac{\Gamma_B}{\Gamma_A + \Gamma_B} \tag{2}$$

This type of setup might represent two species of bacteria suspended in the same liquid medium, competing with one another for resources.

If we let ρ_A, ρ_B represent the densities of species A and B, respectively, the coupled differential equations governing the evolution of this system will be given by

$$\begin{cases} \frac{d}{dt}\rho_A &= k_A\rho_A\rho_B - k_B\rho_A\rho_B - \mu\rho_A \\ \frac{d}{dt}\rho_B &= -k_A\rho_A\rho_B + k_B\rho_A\rho_B + \mu\rho_A \end{cases} \tag{3}$$

These coupled differential equations can be decoupled into two homogenous, second-order differential equations. Thus, given the reaction rates k_A, k_B, μ and appropriate initial conditions, the coupled differential equations in Eq.(3) have unique solutions and so the evolution is deterministic. An important assumption here is that the number of A and B cells remains large so that individual cell deaths and births do not substantially alter the densities $\rho_{A,B}$. As we will see, this determinism becomes lost when we look at spatially-distributed populations [such as the colony shown in Fig. 1.2(a)] because such populations will have locally small population sizes, enhancing the stochastic effects of individual birth and death events.

1.2 Spatial population genetics

We return our attention now to spatially structured populations. Spatially structured populations, such as the neoplastic cells of a tumor, have more complex dynamics than the deterministic model described in the previous section: the geometry of the population will influence which individuals compete and the population will develop spatial heterogeneities as different species will occupy spatially distinct portions of the total population. Perhaps the most striking difference between spatially structured evolutionary dynamics and the well-mixed case is that the spatial heterogeneities introduced in the former give rise to small number fluctuations, or genetic drift. That is, since each individual in the population now only interacts with a finite number of other individuals in the local vicinity, we can no longer assume that the local densities evolve according to deterministic differential equations, but instead their dynamics are inherently stochastic.

One way to look at a spatially structured system is to place each cell with diameter a at the vertices of some lattice, and to assign some dynamical rules for cell division and death, which here would correspond to the replacement of a cell in the lattice. A schematic of the

dynamical rules implementing such spatial dynamics is shown for a one-dimensional lattice in Fig. 1.3, where species A is represented by the red cells and species B is represented by the blue cells, and where we have set $a = 1$ so that lengths will be measured in cell diameters. For the purposes of this thesis, we will consider the situation where each site can only contain a single individual. Alternatively, we could have allowed more than a single individual, say N individuals, to occupy each site on our lattice; in this case, the well-mixed case would be recovered in the limit as $N \longrightarrow \infty$. The case of finite N is known as a stepping-stone model, reviewed in detail in Ref. [3]. Note that the lattice shown in Fig. 1.3 is in fact a two-dimensional lattice; with one dimension running along the length of the actively-dividing population, and another dimension representing the "time-like" direction of overall population growth. This could be an actual spatial dimension when the population is invading virgin territory, or it could represent the time-series of a dividing population. In the example of Fig. 1.3, each row represents a single generation of cells who compete with their nearest neighbors to survive into the next generation, which is displaced approximately one cell diameter in the growth direction. Beyond growing edges of microbial colonies as in Fig. 1.2(a), we might find these effectively one-dimensional spatially structured populations in ecology, such as an animal species that lives along a coast line. If we let τ_g represent the average time it takes for a single generation (row) to produce the next generation (row) of cells, then we expect the system to evolve in multiples of τ_g; it is usually beneficial to set $\tau_g = 1$ and treat the time as being measured in generations. In general, a spatially structured system with D spatial dimensions can be described by a $d = D + 1$-dimensional model. This is particularly useful for systems with $d < 3$ since the entire evolution of the system can be embedded in 3D space.

To find an analytical description of the evolutionary dynamics, it is useful to move to a coarse-grained model. We define $f_A(\mathbf{x}, t), f_B(\mathbf{x}, t)$ to respectively represent the fraction of cells of species A, B, in a small patch surrounding a point $\mathbf{x}$ on the population frontier at time t, as illustrated in Fig. 1.4. To illustrate the stochastic behavior of this setup, let us focus on a single local patch in Fig. 1.4. If we set the time of the system in the figure to be t_0, the state of the local patch labelled $\mathbf{x}_i$ is given by the set of variables $\{f_A(\mathbf{x}_i, t_0) = 1,\ f_B(\mathbf{x}_i, t_0) = 0\}$. Now we may ask, what is the probability that in the next generation, say at time t_1, that

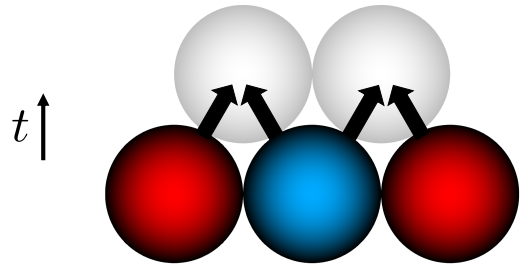

Figure 1.3: To give our two-species model spatial structure, we may assign an individual cell to each site on a single row of an 2−dimensional hexagonal lattice. A single row of the 2−dimensional lattice represents a single generation; the next generation is obtained by allowing each cell to compete with its neighbor for the spot in the adjacent site in the next generation, as shown by the thick arrows in the figure. Such an evolution will be a $d = 1 + 1$ dimensional process, with the $+1$ time-like direction (thin arrow) representing the direction of the overall population growth.

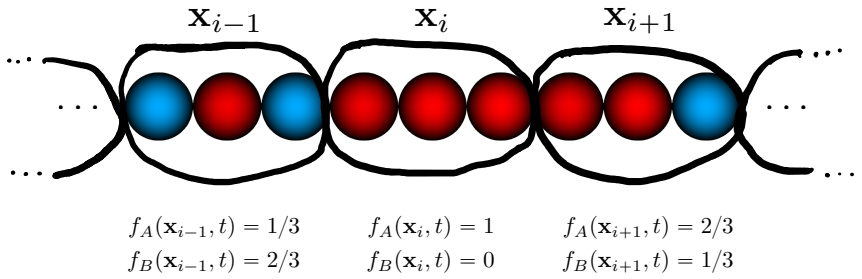

Figure 1.4: In order to find an analytical description of the evolutionary dynamics of the lattice model, we move to a coarse-grained model in which we define the fractions of each species as functions over coarse-grained "patches" of the lattice in each generation. Each patch is indexed by the pair $(\mathbf{x}_i, t)$ where $\mathbf{x}_i$ is the index of the patch and t is the generation. In the figure, we show values for the fractions of species A (red) and B (blue) in each patch to illustrate how this is done in practice.

$f_B(\mathbf{x}_i, t_1) = 1/3$? The only possibility for a blue cell to enter the patch $\mathbf{x}_i$ in the next generation, is for the blue cell at the edge of patch $\mathbf{x}_{i-1}$ to win in competition with the red cell at the edge of the patch $\mathbf{x}_i$, which will occur with probability $P(B|B+A) = k_B$. Thus, the probabilities for the future state of patch $\mathbf{x}_i$ are given by

$$P(\{f_A(\mathbf{x}_i, t_1) = 1, f_B(\mathbf{x}_i, t_1) = 0\} \mid \{f_A(\mathbf{x}_i, t_0) = 1,\ f_B(\mathbf{x}_i, t_0) = 0\}) = k_A,$$

$$P(\{f_A(\mathbf{x}_i, t_1) = 2/3, f_B(\mathbf{x}_i, t_1) = 1/3\} \mid \{f_A(\mathbf{x}_i, t_0) = 1,\ f_B(\mathbf{x}_i, t_0) = 0\}) = k_B,$$

where the line indicates that we condition on the particular state shown to the right of the line. Similarly, the probabilities of the next generation at time t_2 will depend on the state of the system at time t_1 and we may generate the conditional probabilities $P(\{f_A(\mathbf{x}_i, t_2), f_B(\mathbf{x}_i, t_2)\} \mid \{f_A(\mathbf{x}_i, t_1), f_B(\mathbf{x}_i, t_1)\})$. As the stochastic rules only depend on the current state of the system, we have a Markov stochastic process. Of course, other coarse-graining procedures are possible and the precise nature of the coarse-grained stochastic rules will depend on the details of the coarse graining [4].

To find the equations for the time-evolution of our system, we would write down a master equation for the Markov process just described and then derive the appropriate Langevin equations. This method is out of the scope is this thesis, so I will simply state the resulting stochastic differential equations, derived from an analogous coarse-graining procedure:

$$\begin{cases} \partial_t f_A &= D\nabla^2 f_A + k_A f_A f_B - \mu f_A + \sqrt{2\tau_g^{-1}\Delta f_A}\eta_A(\mathbf{x}, t) \\ \partial_t f_B &= D\nabla^2 f_B + k_B f_A f_B + \mu f_A + \sqrt{2\tau_g^{-1}\Delta f_B}\eta_B(\mathbf{x}, t) \end{cases} \tag{4}$$

where $D \approx a^2/\tau_g$ is the diffusion constant and measures the rate of cell rearrangement at the invasion front; τ_g is the generation time; $\Delta = N^{-1}$ is the measure of the strength of genetic drift and is the inverse of the number of cells N in each patch – note that the effect of genetic drift, then, is strongest when $N = 1$, the limit in which we recover our lattice model; ∇^2 is the usual Laplace operator for the dimension and geometry of our system – for the 1D model being discussed we have $\nabla^2 = \frac{\partial^2}{\partial x^2}$; and $\eta(\mathbf{x}, t)$ are Gaussian noise functions with $\langle\eta(\mathbf{x}, t)\rangle = 0$ and $\langle\eta(\mathbf{x}, t)\eta(\mathbf{x}', t')\rangle = \delta(\mathbf{x} - \mathbf{x}')\delta(t - t')$.

The spatially structured model described by Eq. (4) admits travelling wave solutions. These travelling wave solutions represent invading population fronts between our two species.

Returning for a moment to the lattice model, imagine a set-up in which we have an infinite one-dimensional lattice with site indices given by $i \in \mathbb{Z}$; suppose we place cells of type A on all sites such that $i < 0$ and cells of type B on all sites such that $i \geq 0$. At time $t = 0$ there will be a population front between the cells at sites $i = -1, i = 0$, and this front will travel either left or right, depending on the relative values of Γ_A, Γ_B. The overall shape of the front may be described by Eq. (4) on sufficiently large length scales.

I will discuss travelling wave fronts in more detail later, after which I will discuss the main focus of this thesis: roughening invasion fronts. But first, I am going to briefly review the motion of the interface between two species in a one-dimensional population on a lattice. As we shall see, in this case, the boundary between species performs a random walk.

1.3 Random walks

In the previous section I proposed a setup for our lattice model for competition between two species which would produce a single travelling front with the direction of propagation determined by the relative values of the growth rates Γ_A, Γ_B. Since the spatial dimension of our lattice is $D = 1$, the travelling front will have dimension $D = 1 - 1 = 0$ and thus will be a single point. In a more realistic population, the interface would be some localized region which may represent a group of cells where the two strains intermix. This travelling front/point might represent the invasion of a coastal animal species as it takes new territory along its one-dimensional habitat from another species. This type of invasion front, being a single point, can alternatively be represented as a walker on a lattice. Treating this 0-dimensional invasion front as a walker on a lattice will allow us to find analytical expressions for some properties of interest, namely the wandering of the front due to individual cell birth and death. I will briefly introduce the concept of a random walker, and then show that the invasion front will wander an average distance $w > 0$ away from its initial position according to a power law in time,

$$\langle w(t) \rangle \sim t^{\nu}, \tag{5}$$

where ν is an exponent whose value is to be determined, and in general will be insensitive to the details of the model.

We define a one-dimensional random walk by placing a walker on a 1D lattice. Over each time step Δt of the simulation, the walker can move either to the left or to the right by an amount Δx with probabilities k_B, k_A, respectively. This corresponds to displacements of a single A cell by a B cell and vice-versa. The question of interest is to determine the probability of the random walker to be at some location x after some large number N of time steps: $t = N\Delta t$. This probability is written as $P(x,t)$, and can be found by writing out the master equation for the system and solving it with the appropriate initial condition representing a single initial walker at the origin, say. In the continuum limit where we take $\Delta x, \Delta t \to 0$, the probability to find a random walker at some point (x,t) is given by

$$P(x,t) = \frac{1}{\sqrt{4\pi D t}} \exp\left[-\frac{(x-vt)^2}{4Dt}\right] \tag{6}$$

where D is a parameter that determines the diffusivity, and v is the "velocity" and determines the bias of the walker which, as already mentioned determined by the relative values of k_A, k_B. In Fig. 1.5 we see examples of our $1D$ lattice model behaving as a random walker in three cases: $v = 0$, $v < 0$, and $v > 0$.

Note that Eq. (6) is the specific probability distribution given the condition that the random walker started out at $t = 0$ as a delta function centered at the origin. With the probability distribution we may now calculate any property $f(x)$ of the random walker that is a function of the position by

$$\langle f(x) \rangle = \int dx f(x) P(x,t). \tag{7}$$

In particular, we can find the average distance $\langle w(t) \rangle$ of the walker from the origin, which would represent the distance a single domain wall wanders away from its starting point in the evolutionary dynamics. Calculating this distance yields:

$$\begin{aligned} \langle w(t) \rangle &\equiv \sqrt{\langle (x(t))^2 \rangle - \langle x(t) \rangle^2} \\ &= \sqrt{\int dx x^2 P(x,t) - \left(\int dx x P(x,t) \right)^2} \\ &= \sqrt{2Dt + v^2t^2 - v^2t^2} = \sqrt{2Dt} \end{aligned} \tag{8}$$

Thus, we see that the exponent ν in Eq. (5) should be $\nu = \frac{1}{2}$ for our one-dimensional lattice model. This type of process, with $\nu = \frac{1}{2}$, is referred to as a *diffusive* process, in that the

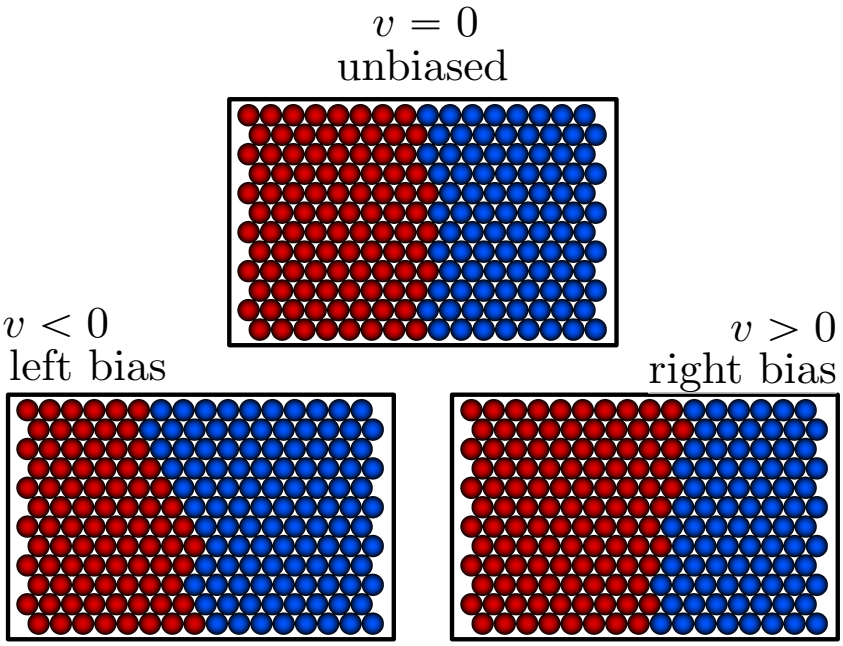

Figure 1.5: We see, using three special cases, that the 1D lattice model with appropriate initial conditions really does behave as a random walker on a lattice. The bias of the random walker is determined by the relative values of k_A, k_B.

random walker representing the interface between the two species diffuses. A random walker with wandering exponent $\nu > \frac{1}{2}$ is called *super-diffusive*, while a walker with $\nu < \frac{1}{2}$ is called *sub-diffusive*. Note that a more realistic one-dimensional population, such as the edge of microbial colonies like the one in Fig. 1.2(a), the overall population shape also roughens over time, developing undulations. These shape undulations may couple to the motion of our interfaces between species. In this case, the population frontier roughness typically enhances the motion of the domain walls, leading to a super-diffusive behavior with $\nu > 1/2$ [5].

1.4 Fisher waves

In general, travelling wave solutions to Eq. (4) are known as Fisher waves and are solutions to the Fisher-Kolmogorov-Petrovsky-Piskunov equation, which we will consider in this section. The noiseless, non-stochastic version of the Fisher-KPP equation is similar to the stochastic versions, but without stochastic terms. The Fisher equation, or the Fisher-KPP equation, was first studied in order to explain the spread of advantageous genes throughout a population. Continuing with the example proposed in the previous sections as a simple spatially distributed population, let us consider an animal species uniformly distributed along a one-dimensional coastal habitat. Suppose there arises some spontaneous mutation within the species that happens to provide the species with a selective advantage. In terms of the growth rate factors, suppose the advantageous effect of the mutation is such that $\Gamma_\mu = \Gamma_0 + s$ where Γ_μ is the growth rate of the mutant type of the animal species, Γ_0 represents the "*wild-type*" growth rate, and s, the *selection parameter*, is a paremeter that measures the degree of selective advantage gained by the mutant species. It is beneficial here to normalize the growth rates so that the wild-type growth rate is set to unity, so we have

$$\Gamma_0 = 1, \ \Gamma_\mu = 1 + s \ . \tag{9}$$

The advantageous gene will sweep through the population as a travelling wave according to the Fisher-KPP equation. The equation may be readily derived from Eq. (4) by assuming that the total fraction of the wild-type and mutant strains add up to 1. So, letting $f \equiv f(x,t)$

represent the fraction of the mutant species at some location x and time t along the one-dimensional population, the Fisher-KPP equation reads

$$\partial_t f = D\frac{\partial^2 f}{\partial x^2} + sf(1-f) \ , \tag{10}$$

where s is the selection parameter, and D is the diffusion constant, and is a measure of local rearrangements of the population on scale much smaller than that of the travelling wave. Note that $1-f$ is necessarily the local fraction of the wild-type strain. Since we have for the moment chosen to ignore the noise, this type of Fisher wave will evolve deterministically. A primary question of interest for the Fisher wave is how quickly will the mutant type sweep over the entire population; or, put differently, what is the speed, v, of the travelling wave solution to Eq. (10)? As it turns out, the travelling wave solution to Eq. (10) will have a characteristic speed dependent on the rate of diffusion D and the selection parameter s. I will follow the analysis from Fisher's original paper[6] to find the necessary bounds on the wave speed.

Let us assume a travelling wave form for the solution of Eq. (10)

$$f(x,t) = f(x-vt) \equiv f(z) \ ,$$

which have the following partial derivatives:

$$\frac{\partial f}{\partial t} = -v\frac{df}{dz}, \text{ and } \frac{\partial f}{\partial x} = \frac{df}{dz}.$$

Substituting this travelling wave ansatz into Eq. (10) yields

$$D\frac{d^2 f}{dz^2} + v\frac{df}{dz} + sf(1-f) = 0 \ . \tag{11}$$

Since we assume the advantageous mutation arises in a localized area inside a population filled with the wild-type strain, then at each snapshot in time we have $f = 0$ in front of the wave (where the wild-type strain dominates) and behind the wave we have $f = 1$ (in the region the mutant population has already swept). An example of the travelling wave front is illustrated in Fig. 1.6 for successive time snapshots. The interesting stuff happens at the wave front, where f is changing with respect to the wave front position. When working within a single snapshot in time we may talk about the "wave front position" in terms of the variable x, but it is important to understand the general wave-front position is given for all

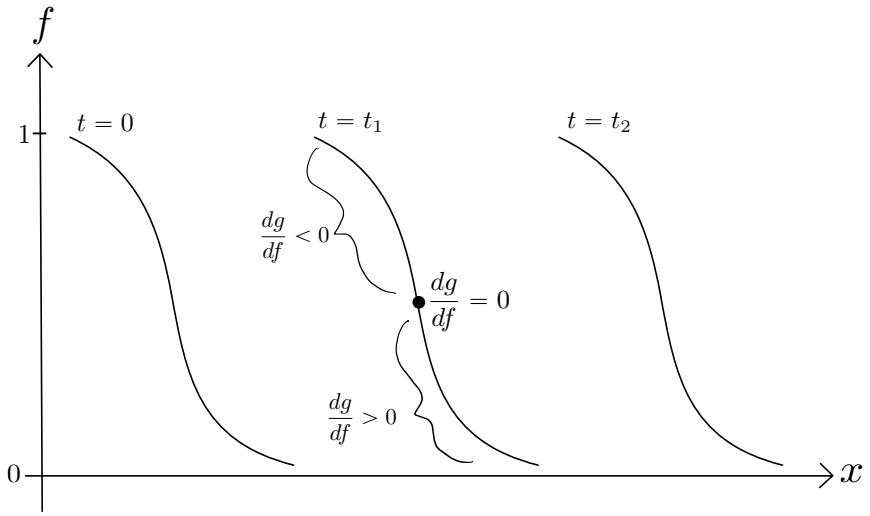

Figure 1.6: Successive time snapshots of a one-dimensional travelling wave. In the middle waveform, I have labelled the inflection point where $\frac{dg}{df} = 0$ (black dot) as well as the regions in front of and behind the inflection point where $\frac{dg}{df} > 0$ and $\frac{dg}{df} < 0$, respectively.

times in terms of the variable $z = x - vt$. Also note that in this coarse-grained model, the wave front has a width, given roughly by the region over which f transitions from 0 to 1.

It is useful to rewrite Eq. (11) in terms of the gradient of f, which we denote

$$g = -\frac{df}{dz} \ . \tag{12}$$

Then, using the chain rule,

$$\begin{aligned} \frac{d^2 f}{dz^2} &= \frac{d}{dz}\left(-g(f(z))\right) \\ &= -\frac{dg}{df}\frac{df}{dz} = g\frac{dg}{df} \ , \end{aligned} \tag{13}$$

where $\frac{dg}{df}$ is a measure of how the gradient of f changes due to changes in f. We expect there to be some value for the fraction, f, of the mutant gene for which the gradient, g, is maximal. This maximal point is marked with a dot in Fig. 1.6. Also shown in Fig. 1.6 are the regions on the wave front for which we expect $\frac{dg}{df}$ to be positive or negative. Using Eq. (13), Eq. (11) becomes

$$Dg\frac{dg}{df} - vg + sf(1-f) = 0 \ . \tag{14}$$

At the maximal gradient, $\frac{dg}{df} = 0$, Eq. (14) gives

$$vg = sf(1-f) \ . \tag{15}$$

In front of the wave, where $\frac{dg}{df} > 0$ we see that $g \longrightarrow 0$ as $f \longrightarrow 0$, and it is not clear what happens to $\frac{dg}{df}$ as $f \longrightarrow 0$. If we assume that g, f both approach zero at the same rate, then it follows that

$$\lim_{f\to 0} \frac{g}{f} \longrightarrow u \ ,$$

where u is some yet to be determined constant. Then, by employing L'Hôpital's rule, and using the chain rule, namely

$$\frac{dg}{dz} = \frac{dg}{df}\frac{df}{dz} \Longrightarrow \frac{dg}{df} = \frac{dg/dz}{df/dz} \ ,$$

we also get

$$\lim_{f\to 0} \frac{dg}{df} \longrightarrow u \ ,$$

so Eq. (14) becomes, in the limit of vanishing fraction f of the mutant,

$$\lim_{f \longrightarrow 0} \left(D \frac{g}{f} \frac{dg}{df} - v \frac{g}{f} + s(1-f) \right) = Du^2 - vu + s = 0 \ . \tag{16}$$

The quadratic equation in Eq. (16) has real solutions for u only for wave speeds, v, such that

$$v \geq 2\sqrt{Ds} \ . \tag{17}$$

Note that the assumption that $\frac{g}{f} \longrightarrow u$ as $f \longrightarrow 0$, where u is some constant, works since we can rule out the other possibilities, $\frac{g}{f} \longrightarrow 0$ and $\frac{g}{f} \longrightarrow \infty$, by looking at constraints on v due to Eq. (14). In the first case, since we are in front of the wave where $\frac{dg}{df} > 0$, if $\frac{g}{f} \longrightarrow 0$, then, dividing out a factor of g, for the first term of Eq. (14) we have $D\frac{dg}{df} > 0$, and so for the full equation to be equal to zero, we must have $v > s/(\frac{g}{f})$. So if $\frac{g}{f} \longrightarrow 0$ with vanishing f, then we must have $v \longrightarrow \infty$, which we can rule out as a possibility.

Alternatively, if $\frac{g}{f} \longrightarrow \infty$ (and so, by once again applying L'Hôpital's rule, $\frac{dg}{df} \longrightarrow \infty$), again dividing out a factor of g from Eq. (14), the final term $s/(\frac{g}{f})$ must go to zero from the positive side, $s/(\frac{g}{f}) \longrightarrow 0^+$, and so in order for the full equation to be equal to zero, we must have $v > D\frac{dg}{df}$. So if $\frac{g}{f} \longrightarrow \infty$ with vanishing fraction, v must grow without bound, as well. Thus, we find that for physical travelling wave solutions to Eq. (10), we must have $\frac{g}{f} \longrightarrow u$ as $f \longrightarrow 0$ in front of the wave, and so the wave speed of these physical travelling waves must be greater than or equal to $v_{\min} = 2\sqrt{Ds}$.

Although we have found a minimal velocity for the Fisher wave, one may show that this minimal speed is approached for any sufficiently sharp initial interface between strains [7]. The result from Eq. (17) for the speed of a Fisher also holds in systems with spatial dimension $D > 1$.

1.4.1 Noisy Fisher Waves

Recall that a population distributed over D spatial dimensions will give rise to $D-1$-dimensional invasion fronts. So far we have only concretely discussed Fisher waves arising in one-dimensional systems, where we expect the travelling wave solution to Eq. (10) to have a localized region representing the interface between strains, while the corresponding lattice

model will have just a single point representing such an interface. The astute reader might wonder, then, why the arguments in this section have treated the wave not as a single point but as being spread out over a finite region of space, as depicted by the waveforms in Fig. 1.6. Truly, the lattice model we investigated in the previously has a 0-dimensional, point-like, boundary between the two populations, as can be seen clearly in Fig. 1.5. The apparent contradiction is resolved when we recall that we first arrived at Eq. (4) via course-graining our lattice model (Fig. 1.4); this course-graining of our spatial dimension into "patches" allowed us to track the states of our lattice in terms of a real-valued function $f : \mathbb{R}^{1+1} \longrightarrow \mathbb{R}$, but is also required us to treat these patches as representing well-mixed populations. Recall that the contribution of the noise term in Eq. (4) is determined by the quantity $\Delta = N^{-1}$; so, by ignoring the noise in Eq. (11) we have essentially taken the limit of the stochastic version as $N \longrightarrow \infty$, taking us into the continuous, deterministic case where the invasion front has become smeared out.

One way to connect the discrete/stochastic and continuous/deterministic cases is to introduce to Eq. (10) a new variable, ϵ, which we can think of as the value of $f(\mathbf{x}, t)$ when there is just a single cell at the site indexed by $(\mathbf{x}, t)$. Thus, when $f(\mathbf{x}, t) < \epsilon$ we set $f(\mathbf{x}, t) = 0$, introducing a cutoff to the density. In this way, the continuous/deterministic case is obtained in the limit $\epsilon \longrightarrow 0$ which is equivalent to the limit $N \longrightarrow \infty$. A reasonable choice is $\epsilon = 1/N$. It turns out that introducing this ϵ – and thus approaching the discrete/stochastic case – introduces corrections to the velocity of the traveling waves [8, 9] . That is, for a fixed ϵ, the travelling wave solution of (10) will have a wave velovity v_ϵ which converges to $v_0 = 2\sqrt{Ds}$ as $\epsilon \longrightarrow 0$. This correction to the velocity, for $\epsilon = 1/N$, has the form [10]

$$v_\epsilon - v_0 \sim \sqrt{Ds}\frac{1}{\log^2 N}. \tag{18}$$

In other words, the fluctuations due to discrete cell birth/death events will serve to increase the wave speed by a an amount proportional to $1/\log^2 N$. Note that this correction varies very slowly with N, so we may expect it to be significant even if N is large.

The effects of small number fluctuations/genetic drift on the dynamics of spatial reaction-diffusion systems is of great interest to a wide range of scientific fields. In addition to introducing corrections to invasion front velocities, fluctuations have been shown to have

other effects on reaction diffusion systems such as introducing new states that are not present in the mean-field description[11], or introducing new spatial correlations which can lead to macroscopic phenomena that are absent in the mean-field description[12]. As we will see in the next section, one such example of fluctuation-induced macroscropic phenomena is the enhanced roughening effect of a population front as the invading population approaches the critical point of an active-to-inactive phase transition – an example of a phase diagram for such a phase transition is shown in Fig. 1.7, where the phase transition critical line is indicated by the dashed black line.

1.5 Roughening population fronts

Returning for a moment to the Fisher-KPP equation, in the small N limit, we can insert a stochastic term into Eq. (10) to obtain the noisy Fisher-KPP equation

$$\partial_t f = D\nabla^2 f + sf(1-f) + \sqrt{f(1-f)}\eta, \tag{19}$$

which captures aspects of the stochastic birth/death dynamics. Details on this derivation are given in, e.g., Ref. [4]. Here, $\eta(\mathbf{x},t)$ is a Gaussian noise with mean $\langle\eta(\mathbf{x},t)\rangle = 0$ and correlations $\langle\eta(\mathbf{x},t)\eta(\mathbf{x}',t')\rangle \propto \delta(\mathbf{x}-\mathbf{x}')\delta(t-t')$. Since we are now allowing for fluctuations in our system, one might wonder what affect these fluctuations might have on an invading population front. We have already argued that such fluctuations will modify the wave speed. We might also correctly guess that the front itself will develop undulations that can be observed on a macroscopic scale.

Since we are now talking explicitly about invading population fronts and their shape, we should take a moment to pause and talk about dimensionality. Let us focus on lattice models in particular, ignoring for the moment the coarse-grained models leading to the Fisher-KPP equation. In the previous sections we discussed an invading front using a lattice model in which the populations were distributed along a single dimension. It was already mentioned that such a system might model the evolutionary dynamics of a coastal animal species. As we saw in the previous sections, an invasion front in a one-dimensional system will be represented by a single point; This is generalized to higher dimensions in the following way:

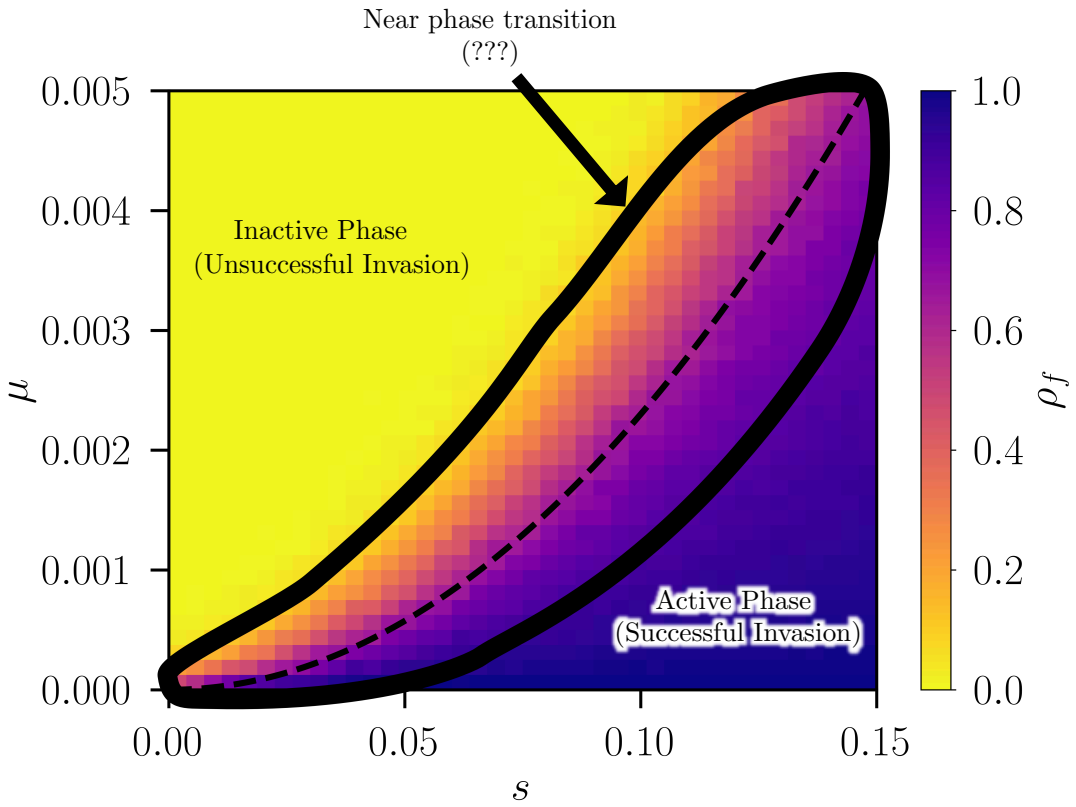

Figure 1.7: A phase diagram for an active-to-inactive phase transition with selection and mutation. In the active (purple) or inactive (yellow) regions we might expect an invasion front to behave as a travelling wave. Near the phase transition when no species has a clear advantage over the other, the behavior of such a front is uncertain.

For invading population fronts, a system with D spatial dimensions will give rise to a $D-1$-dimensional front. For the single point, the idea of "roughness" is represented by the large spread in values for the position of the front over many runs. We already saw some of these aspects by studying the average deviation $\langle w(t) \rangle$ of the domain wall away from its initial position (which we argue moves diffusively). In larger dimensions, the idea of roughness will be seen in the geometry of the front itself. So, for a population distributed over two spatial dimensions, such as a bacteria colony growing on a Petri dish (with nutrients continually supplied so that cells continue dividing everywhere within the population), an invasion front will be a one-dimensional line, as we see for the lattice model example of a fast-growing species (red) invading a slow-growing species (green) in Fig. 1.8. We will focus on two-dimensional systems for the remainder of this section since they give rise to the simplest invasion fronts with geometric fluctuations.

To quantify these fluctuations, it will be helpful to first come up with a useful coordinate system. One such coordinate system for a one-dimensional population front (this describing the boundary between populations distributed over two spatial dimensions) is depicted in Fig. 1.8. In the figure I have shown a snapshot in time say, at time t, of an invading population (red) as it colonizes the territory of another population (green). The details of the lattice model and update rules for the system shown in the figure are described in the next chapter and I will not discuss it here, as the details are not important for our general arguments. It will suffice to say that we have a 2-dimensional hexagonal lattice with appropriate update rules that are applied each generation and the time is advanced by the generation time τ_g. By coarse-graining the lattice in the usual way, we have density functions $f_R(\mathbf{x}, t), f_G(\mathbf{x}, t) : \mathbb{R}^{2+1} \longrightarrow \mathbb{R}$ describing the distribution of red and green cells, respectively, over the two-dimensional space. The initial conditions are such that at time $t = 0$, the fractions are

$$(f_R, f_G) = \begin{cases} (1,0) & , \ y \leq y_0 \\ (0,1) & , \ y > y_0 \end{cases} \tag{20}$$

This describes two blocks of red and green cells, respectively, separated by the line $y = y_0$.

If the red population enjoys a selective advantage over the green one, we expect for the red population to sweep over the green population, establishing a Fisher wave (in the

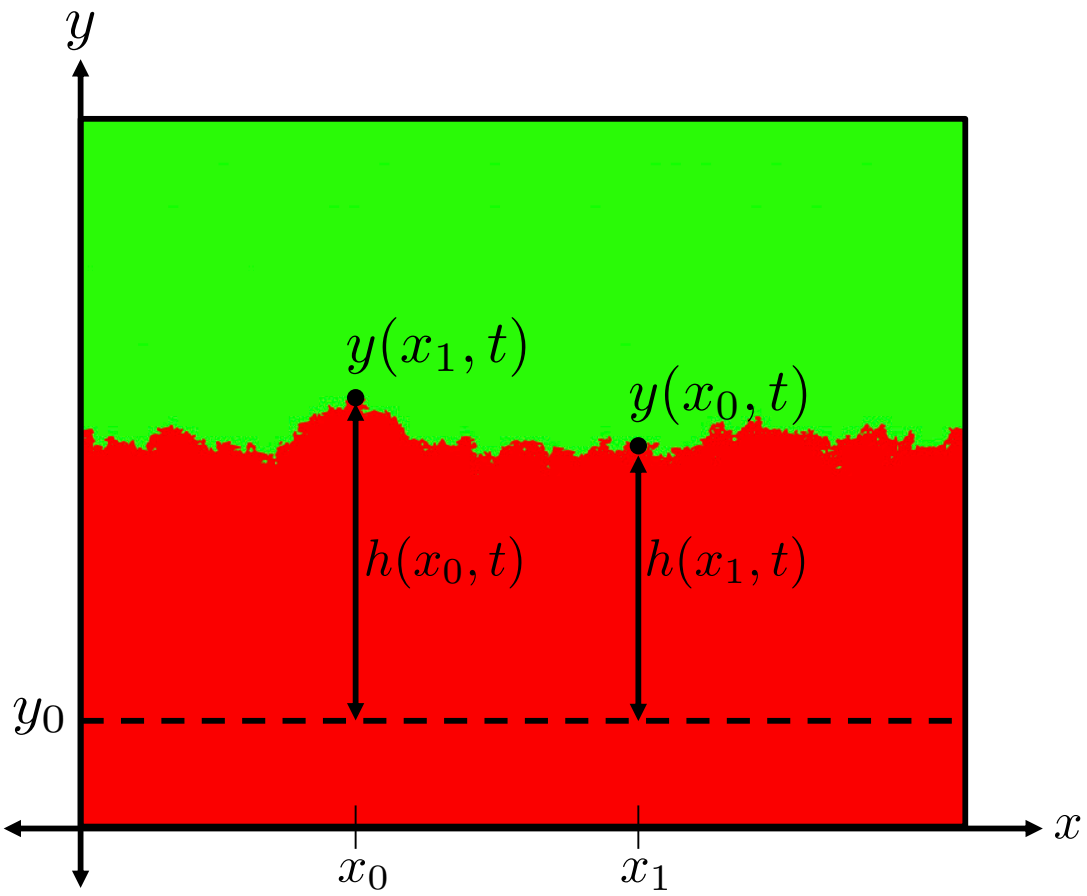

Figure 1.8: A picture of the evolution of a two-species lattice model with competition where species A (red) is invading species B (green). Initial conditions are such that at $t = 0$ $f_A(\mathbf{x}, 0) = 1, f_B(\mathbf{x}, 0) = 0,\ y < y_0;\ f_A(\mathbf{x}, 0) = 0, f_B(\mathbf{x}, 0) = 1,$ otherwise; and so the height function $h(x, t)$ for the population front is defined as $h(x, t) = y(x, t) - y(x, 0) = y(x, t) - y_0$

appropriate coarse-grained picture of the lattice evolution). Now, we may track the position of the population front as a height function $h(x,t) : \mathbb{R}^{1+1} \longrightarrow \mathbb{R}$, where x is the position along the front. This is possible because we have set up our coordinate system and initial condition so that one of the spatial dimensions, which we have taken to be the x-direction, is parallel to the dimension of the front. This way, the height function, $h(x,t)$ is defined as the distance travelled by the front in the y-direction, $y(x,t)$:

$$h(x,t) = y(x,t) - y_0, \tag{21}$$

with $y(x,t)$ the y-coordinate of the boundary between green and red regions at a fixed x and time t.

Now, we expect the red cells to sweep through the green cells like a Fisher wave, but since we are now allowing fluctuations we may now ask what is happening along the front itself? We first note that our height function holds the same information about the shape of our front regardless of the chosen reference point, y_0 in Fig. 1.8. So, we might imagine moving to a reference frame that is moving at the same speed as the Fisher wave. In such a reference frame, we expect that the front will grow and roughen due to localized fluctuations along the front. Such growth, influenced by small number fluctuations, is a well-studied phenomenon and occurs in, for example, the growth of biofilms or in a chemical species depositing on a catalyst. If the growth of the interface can be described by the height field $h(x,t)$, then under a wide range of conditions, the height field $h(x,t)$ will obey the so-called Kardar-Parisi-Zhang (KPZ) equation [13]:

$$\frac{\partial}{\partial t}h(x,t) = \nu\nabla^2 h(x,t) + \frac{\lambda}{2}\left(\nabla h(x,t)\right)^2 + \eta(x,t), \tag{22}$$

where ν represents the tension in the front which tends to smooth the front out, λ represents interface growth normal to the front (which tends to increase "roughness"), and $\eta(x,t)$ is the usual spatiotemporal Gaussian noise. Note that the equation is unchanged when we add a total constant to our height function $h(x,t) \longrightarrow h(x,t) + C$, as expected. It has been shown that for simple population genetics models on a lattice, population fronts do indeed roughen according to the KPZ equation [14]. An example of what this roughening might look like visually is shown in Fig. 1.9.

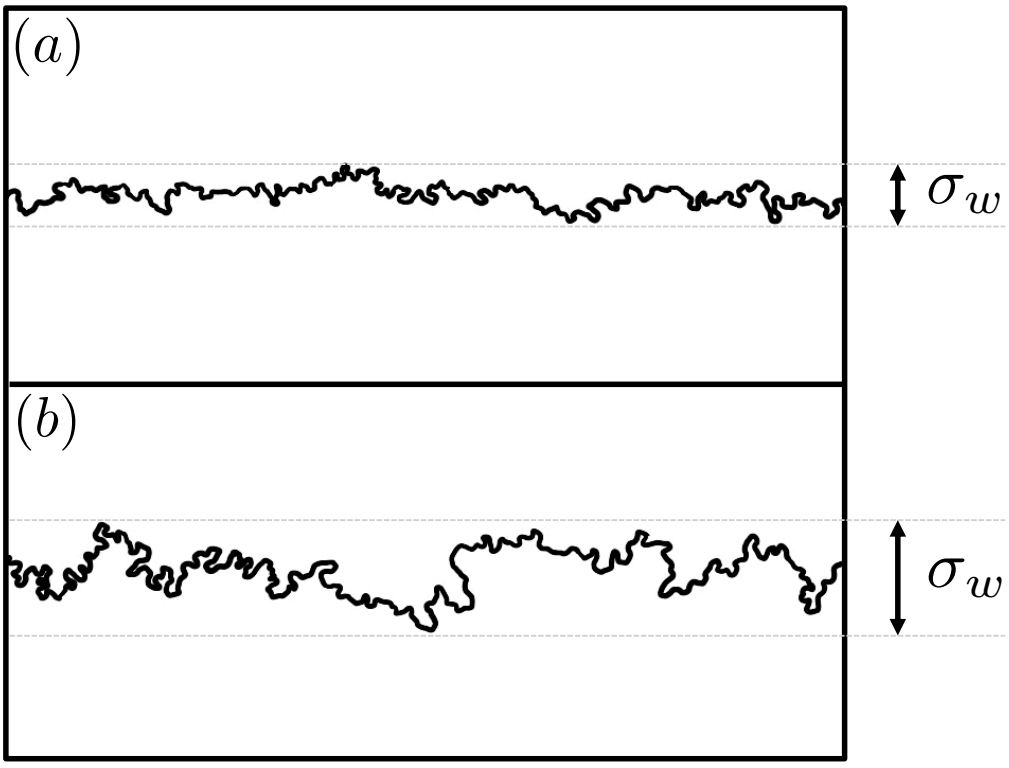

Figure 1.9: An example of a roughening effect for an invading population front. Referring to the type of active-to-inactive phase transition shown in Fig. 1.7, in (a) the invading population is in the active phase and evolves like a travelling Fisher wave; and in (b) the invading population is near the phase transition producing a visible enhancement in the roughness of the front.

This kind of roughening effect is of primary importance to the remainder of this dissertation. In particular, I sought to determine whether such a roughening effect might be produced by the internal dynamics of an invading population. This investigation is detailed in Chapter 2. It should be noted that the three-species model introduced in the next chapter cannot be readily described using the Fisher equation as introduced in this chapter. As I will show, the interface between an unstable population consisting of two separate species and a bystander population consisting of a separate third species will develop an enhanced roughness as the internal dynamics of the unstable population approaches mutational instability. Thus, to describe this more complicated model, the Fisher wave equation would need to be modified in order to take into account the internal dynamics of the unstable population. Such an equation was derived for a mutationally unstable population with many mutations in [15]. To derive analogous equations for the three-species model introduced in the next chapter, the equations in [15] would need to be further modified to take into account the existence of a neutral bystander species.

Chapter 2

Invasion of unstable populations

The following text in this chapter is taken from an article, titled *Shape of population interfaces as an indicator of mutational instability in coexisting cell populations*, which was accepted for publication by the Journal of Physical Biology in August, 2020. The figure titles were revised from the original text so that they follow the proper figure caption guidelines as detailed in the UTK Guide to the Preparation of Theses and Dissertations, 2020-2021.

Invasion and competitive exclusion is a common phenomenon in biology, with examples spanning a wide range of length and time scales: An invasive land animal species may compete with the species already present in the ecological habitat [16], microbial strains may compete and invade each other within a growing biofilm [17, 18], or virus strains may compete for host resources [19]. Such competitions also exist *within* the tissues of various organisms, during development and in cancerous growth: a tumor which starts out as a small cluster of rapidly growing and mutating cells must compete with surrounding healthy tissue [20]. In all of these examples, the spatial structure of the population may have a significant impact on the strain competition and evolution.

Spatially-distributed populations are markedly different from their well-mixed counterparts. Because local population sizes are small compared to the population size in a well-mixed test tube, *genetic drift* or small number fluctuations become more important: Strains within spatially-distributed populations are more likely to locally fix. Also, deleterious mutations more readily accumulate at leading edges of growing populations compared to well-mixed populations where natural selection would eliminate such variants [21, 22]. There

may be mitigating factors that reduce this mutational load, however, such as the presence of an Allee effect due to strain cooperation, for example [23]. These considerations are particularly important for invading cancerous populations which exhibit genomic instability [24, 25] and are typically spatially heterogeneous, consisting of a wide distribution of strains [26, 27, 28, 1]. It is becoming increasingly clear that spatial evolutionary models are necessary to understand the evolutionary dynamics of cancer cell populations [29, 30].

The mutations that drive uncontrolled growth in cancerous populations are the so-called *driver mutations*. However, the majority of mutations are *passenger mutations* which have a neutral or slightly deleterious effect on the cancer cells. Such mutations are ubiquitous in cancerous populations, although their importance for cancer progression has only recently been recognized [31]. Weakly deleterious passenger mutations can rapidly accumulate at the edges of spatially-distributed populations, and the combined deleterious effect can lead to a cancer population collapse. Therefore, the elucidation of the impact of the passenger mutations may lead to new cancer therapies and a better understanding of the efficacy of existing therapies [32, 33]. Indeed, an effective cancer treatment may involve increasing the mutation rate such that the passengers overwhelm the drivers or increasing the deleterious effect of the passengers such that the drivers are no longer able to sustain tumor growth. The accumulation of deleterious mutations leading to population collapse is termed "mutational meltdown" [34, 35]. Already there is evidence that cancer therapies may be developed that target passenger mutations to expose vulnerabilities in cancer growth [36] and that cancer cell lines are particularly vulnerable to mutational meltdown [37].

In this paper, we develop a simple spatial model of the invasion of a cellular population (i.e., a "bystander") by another population (i.e., an "invader") that is acquiring deleterious mutations. We show that when the mutating invader is near a mutational meltdown, the interface between the invader and bystander becomes rougher and more undulated. Such interface shapes and physical cues are important as advances in medical imaging allow us to probe the spatial structure of cancerous growth with unprecedented detail [38]. Tumor shape is increasingly being used for diagnostic purposes. For example, the shape of a tumor boundary is used as a diagnostic tool in breast cancers where a rougher tumor edge may indicate a more malignant growth [39]. Spatial heterogeneities also influence the timing

of the cancer progression [40]. It is therefore useful to build explicitly spatial models to understand what to look for in clinical images and to better understand the spatial signatures of particular evolutionary dynamics.

Although these spatial evolutionary aspects have only recently been explored in cancerous populations, many of the predictions of spatial models have been borne out in studies of microbial range expansions where a population of microbes grows into virgin territory (e.g., as a colony on a Petri dish). Here, small number fluctuations and local fixation yield a sectoring phenomenon where initially mixed strains demix into uniform sectors containing a single strain over time [5, 2]. The previously mentioned enhancement of deleterious mutations near population edges has also been verified via DNA sequencing of bacterial range expansions [22]. Also, the mutational meltdown we will consider in this paper has been demonstrated in yeast cell colonies, where a simple lattice model of the kind employed in this work successfully predicted the effects of the increased genetic drift in spatial populations [41]. Increasingly, results from such microbial populations and simple spatial evolutionary models are yielding insights into what may happen in cancerous populations [42, 29]. The results presented here are also applicable to the microbial range expansions.

The evolutionary dynamics explored here (e.g., the motion and coarsening of the sectors of strains) has universal properties tying together a large class of systems including tumor growth, reaction-diffusion processes, granular material avalanches, and epidemic spreading [43]. For example, tumor shapes have been shown to have the same fractal boundary properties as films deposited by molecular beam epitaxy [44]. Therefore, many of the techniques originally developed to understand physical phenomena, such as the phase ordering of deposited binary films [45, 46], may be employed to understand the spatial evolutionary dynamics of microbial populations [47]. The universal properties of all of these systems include coarse-grained properties such as the scaling of interface roughness with time [48], a quantity we will explore here for the interface between the bystander and the invader. We may thus reasonably expect our results to hold generally, as we will be concerned with such coarse-grained properties.

In cancerous tissue, current sequencing techniques have a limited ability to probe the spatial structure of the cancer cell population. Adaptation of sequencing techniques

to spatially-distributed populations is important as spatial effects have been shown to significantly impact DNA sequencing data of cancerous cell populations [49]. Our study presents a complementary approach where we show that physical cues such as the shape of an interface between competing cellular strains may indicate certain properties of the evolutionary dynamics of the tissue (e.g., the proximity to a mutational meltdown transition). Such heuristic measures are useful in conjunction with DNA sequence information, which is often difficult to interpret and does not typically take into account the spatial structure of the cancerous population [49].

In this paper, we build a model for how a mutating strain invades a non-mutating strain in both one- and two-dimensional habitats, which we call $d = 1+1$ and $d = 2+1$ evolutions, respectively. The $+1$ indicates the time dependence. Our focus here is on the competition between multiple strains within a population, so for simplicity we consider *flat* habitats which do not change shape as the population evolves. For $d = 1 + 1$, such a habitat may be a coast, a river bank, or a thin duct. For $d = 2 + 1$, the strains may be in a microbial population growing on a flat surface or in an epithelial tissue. Another possibility is that the population in which the strains compete is the leading edge of a range expansion. In this case, we assume that the population growth is confined to a thin region at the edge, which remains flat during the range expansion. This approximation will hold as long as there is a sufficient surface tension keeping the population edge uniform which occurs in yeast cell colonies, for example [50]. However, if the population edge roughens over time, the roughening will generically change the genetic sector motion [14], an analysis of which is beyond the scope of the current work.

Note that for a cellular population at the edge of a range expansion, the $+1$ dimension represents the direction of the range expansion. So, in other words, for $d = 1+1$, the strains we study may live along a thin, effectively one-dimensional edge of a two-dimensional range expansion (e.g., a thin microbial colony grown on a Petri dish). For $d = 2+1$ evolution, the population may be the effectively two-dimensional, flat edge of a three-dimensional range expansion. A more realistic scenario is perhaps the $d = 3 + 1$-dimensional case where a population embedded in three dimensions evolves in time with various strains within the population competing for the same space. Although we do not study this case specifically

here, the lower dimensional cases provide intuitions for considering this scenario. Also, if the geometry of the three-dimensional population has a large aspect ratio, then our one and two (spatial) dimensional models will describe the behavior of cross sections of the population. A similar kind of dimensional reduction was recently employed for describing bacterial competition in three-dimensional channels [51].

Previous work has shown that range expansions develop frontiers with enhanced roughness when the population is near a phase transition in its evolutionary dynamics (e.g., at a mutational meltdown transition [14] or near the onset of mutualistic growth [52]). In this work, we consider invasion frontiers which are markedly different as the invader grows into a surrounding population which may *reinvade* if the invader growth rate decreases. So, the invasion front speed v will depend on the relative growth rates of the two populations and may vanish or change signs. In other words, the competition interfaces studied here have a variable speed, unlike a range expansion in which a population grows into a virgin territory with some particular growth rate. In this sense, the competition interface studied here is more similar to a *range shift*, in which the population growth is limited by the environment [53]. We will see that in the case of $d = 2 + 1$-dimensional expansions, the average speed v of the invasion front will influence the interface roughness.

We will also show here that, like the range expansion, the invasion frontier develops an enhanced roughening at the mutational meltdown transition of the invader population. However, unlike a range expansion frontier, the roughening is more subtle, especially in the $d = 2 + 1$-dimensional case where the relative growth difference between the invader and bystander populations (and consequent invasion front speed v) also influences the roughening dynamics. The invasion frontier does not maintain a compact shape, and isolated pieces of the invading population may pinch off and migrate into the surrounding "bystander" population, especially when the growth rates of the two populations are nearly equal. In this paper we will discuss these issues and connect the shape of the undulating frontier to the evolutionary dynamics of the invading population.

The paper is organized as follows: In the next section, we present our lattice model for invasion by a mutating population for $d = 1 + 1$ and $d = 2 + 1$-dimensional cases. In Section 2.2 we briefly review the nature of the mutational meltdown transition that may

occur in the unstable invading population. In Section 2.3 we study the survival probability of the invading strain and construct phase diagrams characterizing whether or not the invasion is successful as a function of the internal mutation rate μ and the selective advantage of the various cellular strains. In Section 2.4, we analyze the roughening invasion front near the mutational meltdown transition for the invading population. Finally, we conclude with a discussion of the implications of our results in Section 2.5.

2.1 Model

We consider a simple lattice model, in the spirit of the Domany-Kinzel cellular automaton [4, 54], of invasion of a stable population by a mutating invader consisting of two species, a fast-growing and a slow-growing strain into which the fast-growing one can mutate. We set the fast-growing strain growth rate to unity $\Gamma_f = 1$ without loss of generality, so that time is measured in generation times τ_g of the fast-growing strain. The slow-growing strain within the invader population will have growth rate $\Gamma_s = 1 - s$, where $0 \leq s < 1$ is a measure of the deleterious effect of the mutation. In a tumor or microbial colony, we know that the initial cluster of growing cells may encounter other cells (e.g., surrounding healthy tissue or competing microbial strains). So, we have a third species representing the "bystander" population. The bystander will not mutate, but will be able to displace the mutating population via cell division. We set the bystander growth rate to $\Gamma_b = 1 + b - s$, with b the selective advantage of the bystander over the slow-growing invader strain.

The internal dynamics of the invading population (i.e., the mutation rate μ and selection parameter s) will influence how the invader interacts with the bystander strain, with increasing μ or s leading to an overall fitness decrease for the invading population, as might happen in a cancerous tissue during a course of therapy that increases the deleterious mutation rate of the cancerous cells. We focus on the region between the mutating population and the bystander, which we call the invasion front. As we will show, when the invader is close to mutational meltdown, the invasion front develops an enhanced "roughness."

The specific lattice model rules are as follows: In both one- and two-dimensional population scenarios, we consider a three-strain model in which a single "bystander" strain

(yellow cells in Fig. 2.1) grows in the presence of a fast-growing invading strain (red cells in Fig. 2.1) that can mutate to a more slowly growing strain (black cells in Fig. 2.1). These cells occupy a single lattice location, as shown in Fig. 2.1. During each generation (cell division) time τ_g, the lattice of cells is regenerated by allowing for adjacent cells to compete and divide into empty sites representing the next generation of cells. In a range expansion context, these empty sites would represent unoccupied territory at the frontier. Alternatively, these updates can represent a turnover of cells due to birth and death within the population. After all empty sites in the next lattice have been filled (rows for $d = 1+1$ and sheets for $d = 2+1$ as shown in Fig. 2.1(a) and (b), respectively), the process can be repeated, generating a sequence of successive generations of the population (or, alternatively, a moving frontier of a range expansion). Note that each time a red, fast-growing cell is chosen to occupy an empty site, it mutates to a slower-growing black strain with probability μ.

So, our model contains just three parameters: the deleterious effect of the mutation s, the selective advantage b of the bystander population over the slow-growing strain, and the mutation rate μ. We will be interested in the regime $0 < \Gamma_s \leq \Gamma_b \leq \Gamma_f$, for which $0 \leq s < 1$ and $0 \leq b \leq s$. In this case, the bystander either invades the slow-growing strain or is invaded by the fast-growing strain, as illustrated in Fig. 2.2 for a $d = 1 + 1$ simulation. Note that this reinvasion by the bystander population makes the invasion frontier markedly different from, say, a range expansion frontier. In a range expansion frontier, the range expansion always moves in one direction according to the growth rate of the total population. Here, the interface between the bystander and invader can move in different directions or even remain, on average, stationary. We will see that this aspect will be important when studying the roughness of the interface.

Our parameterization allows us to tune the dynamics of the black/red mutating invader population separately. As we will analyze in the next section, the invader has an internal "mutational meltdown" at which the fast-growing red strain is removed from the population due to mutation. This occurs for $\mu \gtrsim s^2$ in $d = 1 + 1$ dynamics and $\mu \gtrsim s \ln s$ in $d = 2 + 1$ dynamics ($\mu > s$ in well-mixed populations). Note that an important limitation of our model is that we assume cells divide into adjacent spots on the lattice so that cell motility is essentially absent (apart from the short-range cell rearrangements occurring due to the

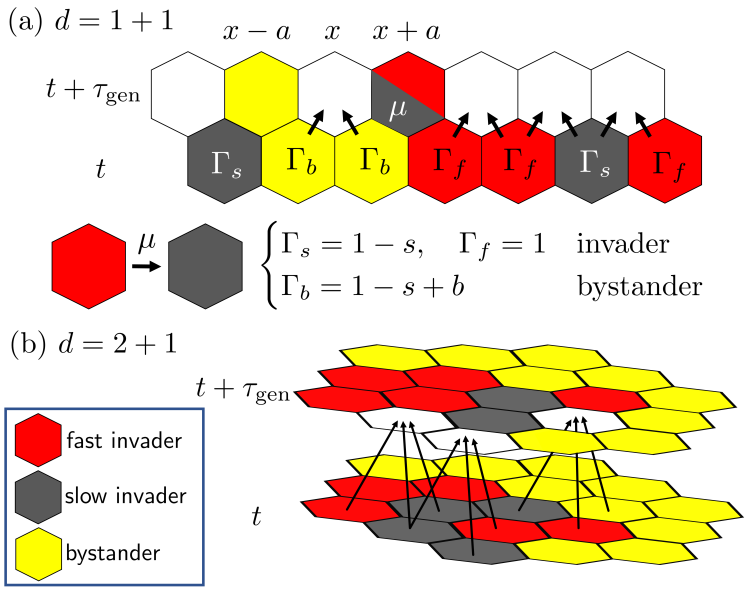

Figure 2.1: Update rules for the bystander model. (a) In $d = 1 + 1$ dimensions each generation is evolved by allowing for two cells from the previous generation to compete for an empty lattice site, as shown by the arrows. The probability of occupation by a cell of a type $i = s, f, b$ is proportional to its growth rate Γ_i, where s is the slow growing black strain, f is the fast growing red strain, and b is the yellow bystander. If a red (fast-growing) cell is placed in the empty spot, then it in addition has a probability μ of mutating to the slower-growing black strain. (b) For a two-dimensional population ($d = 2 + 1$) the generations are evolved on staggered triangular lattices, as shown. This time, *three* cells compete to divide into empty lattice sites. Otherwise, the dynamics are the same as the $d = 1 + 1$ case.

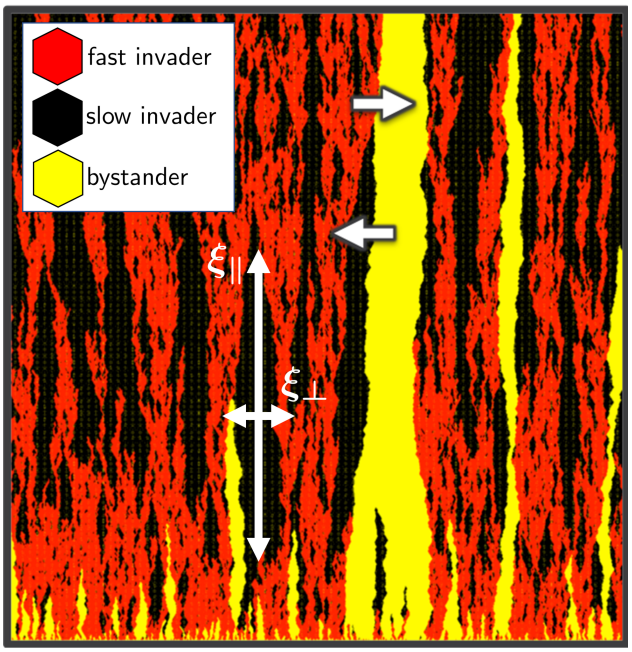

Figure 2.2: A $d = 1{+}1$ simulation of a red/black mutating population invading a bystander yellow one. Here, the yellow strain grows faster than the black strain but slower than the fit red strain. The invasion front between the black/red population and the bystander strain can be characterized by a random walk with alternating bias. The yellow strain invades the black patches and is invaded by the red patches. The sizes of the red and black patches are controlled by the internal dynamics of the black/red strain. We illustrate the characteristic sizes $\xi_{\perp}$ and lifetime $\xi_{\parallel}$ of the black patches.

cell division). This is a reasonable approximation for certain microbial populations such as yeast cell colonies [41] or small, avascular solid tumors where cells primarily proliferate [55].

2.2 Mutational meltdown

Let us first focus on the invading population and perform a simple analysis of the internal dynamics. The invader consists of two strains: one fast-growing red strain and a second slow-growing black strain into which the fast-growing strain mutates with rate μ per cell per generation. In the parameter space (μ, s), we find two distinct phases [4]: In one phase, the fraction of the fast-growing strain remains positive after many generations, $\rho_f(t \to \infty) > 0$; we call this phase the *active phase*. In the other phase, called the *absorbing* or *inactive phase*, the fast-growing strain eventually completely dies out and $\rho_f(t \to \infty) = 0$. There is a line of continuous phase transitions (μ^*, s^*) which defines the boundary between the two phases. Examples of these phases, and the critical region ($\mu \approx \mu^*$, $s \approx s^*$), are shown in Fig. 2.3 where we have removed the bystander population in order to see the internal dynamics of the invader.

We can understand this transition in a well-mixed population (a mean-field approach). Consider just the invading, mutating population. The fraction $\rho_f \equiv \rho_f(t)$ of the fast-growing strain within the mutating population changes according to:

$$\frac{d\rho_f}{dt} = s\rho_f(1 - \rho_f) - \mu\rho_f, \tag{1}$$

which approaches $\rho_f(t \to \infty) = 1 - \mu/s$ for $\mu < s$, and $\rho_f(t \to \infty) = 0$ for $\mu > s$. The line $\mu = s$ is our set of critical points $(\mu^*, s^* = \mu^*)$. For a spatially distributed population, small number fluctuations or "genetic drift" dramatically alters the shape of the phase boundary: The phase transition occurs for $\mu^* \sim (s^*)^2$ in one-dimensional populations (such as at the edge of a growing biofilm [41]) and $\mu^* \sim s^* \ln(s^*)$ for two-dimensional populations [15]. This phase transition, called a "mutational meltdown," is known to belong to the directed percolation (DP) universality class [43]. For the particular lattice model we consider here, this has been explicitly verified [4] by mapping the model to the well-studied Domany-Kinzel cellular automaton [54]. The efficacy of this simple model has been demonstrated in

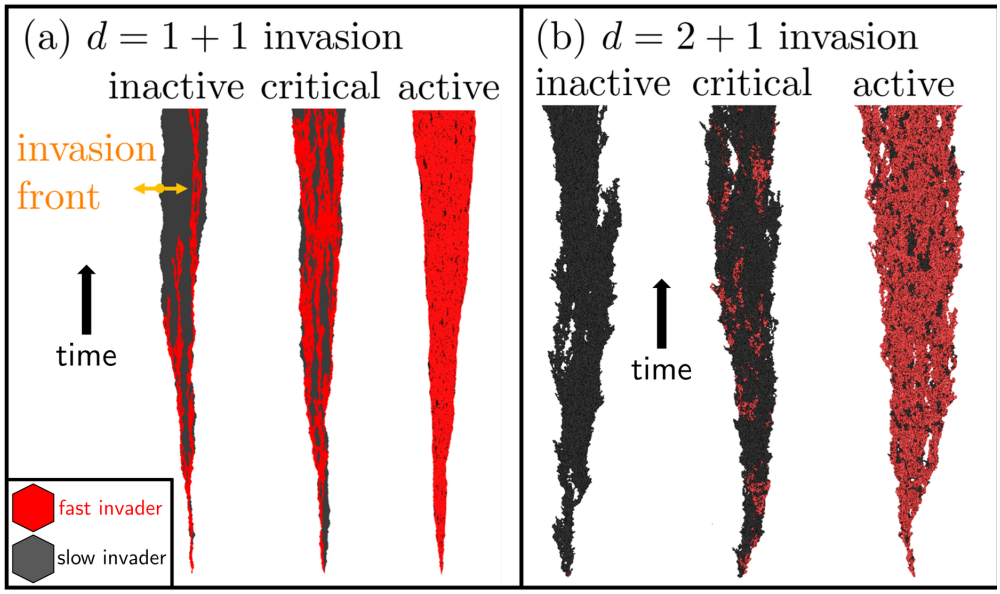

Figure 2.3: Simulated sectors of a black/red mutating population invading a bystander population (surrounding white area). The sectors in the figure were initialized by a single red cell in a (a) one-dimensional and (b) two-dimensional population. The populations are evolved for about 100 generations, with the time direction indicated. In (a), we indicate the motion of the invasion front (which in this case is a point) between the two populations. In (b), the invasion front would be the complicated boundary between the black/red population and the white space at each time slice t along the vertical direction. The phases of the internal dynamics of the invading population (inactive, critical, and active phases) are indicated. In the inactive phase, the red, fast-growing mutant is lost from the population over time. As the invader population transitions from the inactive to the active phase in which the red strain is maintained, the invasion front exhibits enhanced undulations.

a synthetic yeast strain, for which the parameters μ and s could be tuned over a broad range encompassing the DP phase transition [41].

Note that when the population approaches the mutational meltdown transition, the slow-growing black strain within the population begins to take over. In the active phase in Fig. 2.3, the black strain makes finite-sized, small patches within the red population. Then, as we approach the transition, the black strain patch sizes diverge. In the critical regime, the average patch size becomes infinite. Then, in the "inactive" phase, the red strain will eventually die out completely, leaving behind just the slowly growing black strain. We shall see that it is this divergence of the black strain patch size near the transition which is responsible for enhanced invasion front roughening.

The two-species model may be generalized to include an arbitrary number of possible mutations, and such models have been shown to exhibit critical behavior that deviates from the DP universality class, but the loss of the fittest mutant in the population is still well-described by DP [15]. The multi-species generalization has many additional interesting phenomena such as multi-critical behavior [56], which would allow for interesting extensions of the work presented here. In this paper, for simplicity, we shall focus on the fittest mutant in an invading population with just two species. The fittest strain could represent, for example, a driver mutation which has swept through a cancerous tissue. The driver strain could then acquire deleterious mutations over time with rate μ. We will focus here on just the initial loss of fitness, characterized by a single mutation to the slower growth rate $\Gamma_s = 1 - s$.

2.3 Invasion probabilities

We now construct a phase diagram for successful invasion of the bystander strain by the mutating invader (Fig. 2.4). We initialize a well-mixed population of equal parts of the mutating red and bystander yellow strains ($\rho_b = \rho_f = 1/2$) on the lattice, and then calculate the density $\rho_m = \rho_f + \rho_s$ of the mutating red/black population at long times t. If $\rho_m \to 0$ and the red/black population dies out at long times, then the "invasion" is unsuccessful. Otherwise, ρ_m approaches a non-zero value and the invasion succeeds. The results are shown in Fig. 2.4 for $d = 1 + 1$ and $d = 2 + 1$, where we see the two distinct phases. We

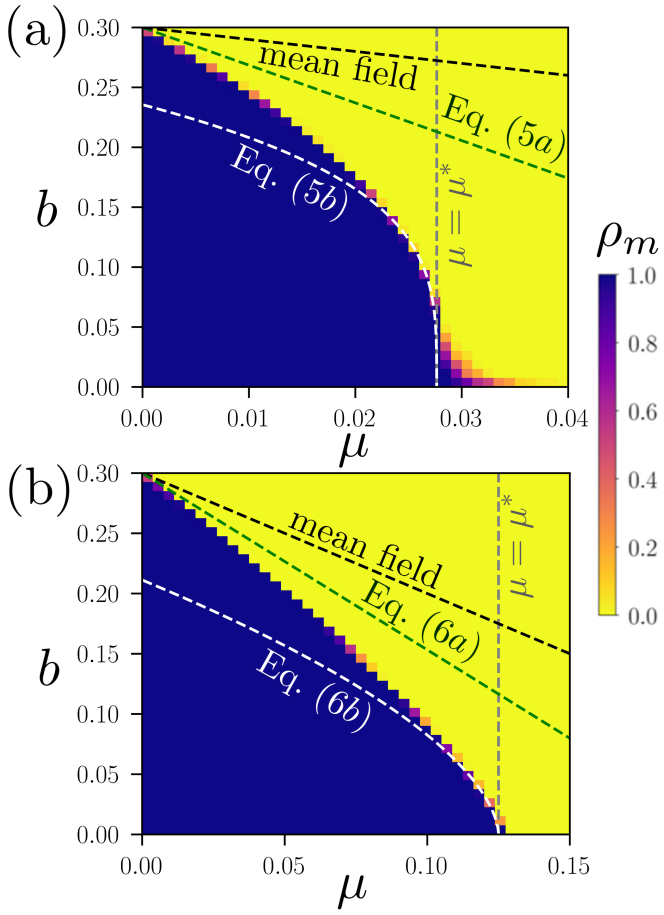

Figure 2.4: Phase diagrams for (a) the $d = 1+1$ case and (b) the $d = 2+1$ case, calculated by initializing a well-mixed population of the red and yellow strains and evolving the whole population for $t \approx 10^6$ generations. In (a) we use a one-dimensional population of $L = 5000$ cells and average over 256 runs of the evolution. In (b) we have a two-dimensional population with L^2 cells, where $L = 500$. Here we average over 40 evolution runs. In both cases we set $s = 0.3$. After evolving for 10^6 generations, we calculate the red/black mutating fraction of the total population: $\rho_m = \rho_f + \rho_s$. In each phase diagram, the black dashed line corresponds to the mean field prediction $\mu = s - b$. The green and white dashed lines correspond to the improved predictions [see Eqs. (5*a*), (5*b*), (6*a*), (6*b*)] for $\mu \ll \mu^*$ and $\mu \approx \mu^*$, respectively, that take into account the spatial structure of the population. We also indicate the line $\mu = \mu^*$ along which we find a mutational meltdown transition within the invading red/black population.

also include the phase boundary $\mu = s - b$ for the well-mixed population (blacked dashed line in Fig. 2.4), which we derive in the next subsection. Note how far away the well-mixed population transition line is from the actual transition in a spatial population. The genetic drift associated with the spatial populations suppresses the invasion by the red/black mutating population.

We also know that as we approach this mutational meltdown transition at $\mu = \mu^*$ for a fixed s [vertical dashed lines in Fig. 2.4], the characteristic sizes $\xi_\perp$ and the characteristic lifetimes $\xi_\parallel$ of black, slow-growing strain clusters diverge according to $\xi_\perp \sim \Delta^{-\nu_\perp}$ and $\xi_\parallel \sim \Delta^{-\nu_\parallel}$, where $0 < \Delta < 1$ is the distance from the phase transition in the (μ, s) plane and $\nu_\perp$ and $\nu_\parallel$ are critical exponents associated with directed percolation ($\nu_\perp \approx 1.097$, $\nu_\parallel \approx 1.734$ for $d = 1 + 1$ and $\nu_\perp \approx 0.734$, $\nu_\parallel \approx 1.295$ for $d = 2 + 1$ [43]). We illustrate the sizes $\xi_{\perp,\parallel}$ in Fig. 2.2. The black patches will interact differently with the bystander than with the red patches of the fast-growing strain. As the patch sizes $\xi_{\parallel,\perp}$ diverge (when $\Delta \to 0$), they would have a more pronounced effect on the invasion dynamics. In particular, there will be larger regions over which either the yellow strain invades a black patch, or a red patch invades the yellow bystander. This will increase the amount of "wiggliness" of the invasion front between the bystander and the mutating red/black population. We will see in the following that there is a significant enhancement of the roughness as we approach the mutational meltdown transition.

2.3.1 Mean field analysis

To understand the behavior of this three-species model, we first briefly describe what happens in a well-mixed (mean-field) context. Consider the time-evolution of the fractions ρ_f, ρ_s, ρ_b of the fast-growing, slow-growing, and bystander strains, respectively. For a fixed total population size, we have $\rho_f + \rho_s + \rho_b = 1$. Given our growth rates $\Gamma_f = 1$, $\Gamma_s = 1 - s$, and $\Gamma_b = 1 - s + b$, we may define corresponding selection coefficients characterizing the

competition between pairs of strains:

$$\begin{aligned} s_{bs} &= \frac{\Gamma_b - \Gamma_s}{(\Gamma_b + \Gamma_s)/2} = \frac{2b}{2 - 2s + b} \\ s_{fb} &= \frac{\Gamma_f - \Gamma_b}{(\Gamma_f + \Gamma_b)/2} = \frac{2(s-b)}{2 - s + b} \\ s_{fs} &= \frac{\Gamma_f - \Gamma_s}{(\Gamma_f + \Gamma_s)/2} = \frac{2s}{2 - s}, \end{aligned} \tag{2}$$

with selection parameters $0 < s < 1$ and $0 < b < s$. In terms of these selection coefficients, the equations for the time-evolution of the bystander and fast-growing strain fractions $\rho_{b,f} \equiv \rho_{b,f}(t)$ in a well-mixed population are

$$\begin{cases} \partial_t \rho_b = s_{bs} \rho_b \left(1 - \rho_b - \rho_f\right) - s_{fb} \rho_f \rho_b \\ \partial_t \rho_f = s_{fs} \rho_f \left(1 - \rho_b - \rho_f\right) + s_{fb} \rho_f \rho_b - \mu \rho_f \end{cases} . \tag{3}$$

If $\rho_b = 0$, we recover the two-species dynamics of the invader population with a directed percolation-like process between the fast-growing and slow-growing strains. We can also verify that there is no sensible stable fixed point where both the bystander population and the invader coexist. Instead, if $\mu > s - b$, then the bystander will sweep the total population and $\rho_b(t) \to 1$ with increasing time t. Otherwise, if $\mu < s - b$, the invasion by the mutating population is successful and we find $\rho_b(t) \to 0$ over time. Moreover, if $\mu > s$, we get a collapse of the fast-growing strain $[\rho_f(t) \to 0]$, and then the bystander strain will win out since $\Gamma_s < \Gamma_b$. So, the mutational meltdown transition of the invader population occurs when $\mu = s$ in this mean field analysis.

The mean-field analysis tells us that we should expect a critical *surface* in the (μ, s, b) parameter space given by $\mu = s - b$ separating a region of successful ($\mu < s - b$) or failed ($\mu > s - b$) invasion of the bystander strain by the mutating invader (which itself may undergo a mutational meltdown when $\mu > s$). As we have already seen, the spatially-distributed populations also have this critical surface but the enhanced genetic drift *suppresses* the phase space for successful invasion. To add the effects of genetic drift and the spatial distribution of the population to Eq. (3), we would have to incorporate a spatial diffusion term $\nabla^2 \rho_{b,f}$ in each of the equations and stochastic noise terms describing the birth/death dynamics (see [15] for a more detailed description). These additional terms significantly modify our

mean field equations and introduce different phenomena, such as propagating waves (moving population interfaces) which we will analyze in Section 2.4.

We can get a better approximation to the critical line for the $d = 1 + 1$ case than that given by the mean field theory by considering a single domain wall. We expect the total length along the domain wall to be split into sections of average length ℓ_s where the slow-growing strain competes with the bystander and sections of average length ℓ_f where the fast-growing strain competes with the bystander. During each generation time τ_g the domain wall can move by a single cell length a. So, in the fast-growing strain segments, we expect the fast-growing clusters to out-compete the bystander and protrude by an amount $\tau_g \ell_f (s - b)/a$, with τ_g the generation time. Similarly, we expect the slow-growing clusters to be out-competed by the bystander and recede by an amount $\tau_g \ell_s b/a$. At the phase transition, we expect these competitions to cancel each other out as the mutating invading population is *barely* able to invade the bystander in this case. Hence, we should have $\ell_f(s - b) \approx \ell_s b$ so that

$$b \approx \frac{\ell_f s}{\ell_s + \ell_f} \approx \phi_f \, s, \tag{4}$$

where ϕ_f is the red-cell (fast-growing) fraction of the mutating invader population. We now can use Eq. (4) to predict the critical line in (μ, b)-space for a fixed s in two limiting cases: $\mu \ll \mu^*$ and $\mu \sim \mu^*$, where μ^* is the critical value for μ for the specific fixed value of s at which we get the mutational meltdown transition within the red/black invading population.

To complete the derivation, we just need an estimate for the fraction of fast-growing strain ϕ_f. First, when $\mu \ll \mu^*$, the invader population is in the "active" phase, and the patches of black slow-growing strain are small and rarely collide, as shown in Fig. 2.3. In $d = 1 + 1$-dimensions, the boundaries of these black patches are well-described by pairs of random walkers, yielding an estimate $\phi_f \approx 1 - A_1 \mu/s^2$ [4, 57, 2]. The amplitude A_1 is model-dependent, and we have $A_1 \approx 0.5$ for this Domany-Kinzel-type model, consistent with previous results [15]. As $\mu \to \mu^*$, however, the fast-growing strain vanishes ($\phi_f \to 0$), and we have to make another approximation. From the random walk model, we expect that ϕ_f vanishes when $\mu = \mu^* \sim s^2$. Then, when $\mu \approx \mu^*$, we would be near a directed percolation (DP) phase transition where ϕ_f serves as an order parameter. The order parameter vanishes

according to $\phi_f \approx A_2(\mu^* - \mu)^\beta$, where A_2 is an amplitude that will depend on s and $\beta \approx 0.276$ is a DP critical exponent [43]. We may now use Eq. (4) to make an estimate of the critical value of b for $d = 1 + 1$:

$$b = s(1 - A_1\mu/s^2) \qquad (\mu \ll \mu^*) \tag{5a}$$

$$b = sA_2(\mu^* - \mu)^\beta \qquad (\mu \approx \mu^*) \tag{5b}$$

where A_1 and A_2 can be calculated numerically from separate simulations of the two-species model. These improved estimates are plotted onto the phase diagrams in Fig. 2.4(a) (green dashed line for the $\mu \ll \mu^*$ case and white dashed line for the $\mu \approx \mu^*$ case).

For $d = 2 + 1$-dimensional evolutions, the situation is more complicated because the patches of the invader strain no longer have a compact shape describable by a simple random walk [see Fig. 2.3(b)]. However, we expect that the bystander population may reinvade the invading mutating population when $b > s\phi_f$ because, much like in the $d = 1 + 1$ case, the average growth rate of the invader strain is approximately $\overline{\Gamma} \approx \phi_f\Gamma_f + (1-\phi_f)\Gamma_s = 1 - s + \phi_f s$. The bystander strain has growth rate $\Gamma_b = 1 - s + b$, so we see that the growth rates are equal when $b = \phi_f s$. We now just need estimates for ϕ_f for $d = 2 + 1$. When $\mu \ll \mu^*$, previous work [15] has shown that $\phi_f \approx 1 - A_3\mu \ln(s/s_0)/s$, with $A_3 \approx 0.3$ and $s_0 \approx 40$ some model-dependent parameters. Conversely, when $\mu \approx \mu^*$, we again find a DP transition with $\phi_f \approx A_4(\mu^* - \mu)^\beta$ with critical exponent $\beta \approx 0.584$ for $d = 2 + 1$ [43]. The corresponding estimates are for $d = 2 + 1$:

$$b = s[1 - A_3\mu \ln(s/s_0)/s] \qquad (\mu \ll \mu^*) \tag{6a}$$

$$b = sA_4(\mu^* - \mu)^\beta \qquad (\mu \approx \mu^*) \tag{6b}$$

These two approximations are plotted in Fig. 2.4(b), with $\mu \ll \mu^*$ in the green dashed line and $\mu \approx \mu^*$ in the white dashed line.

The phase diagrams in Fig. 2.4 were constructed with simulations using mixed initial conditions; the first generation of cells on the lattice were populated by an even mixture of fast-growing (red) cells and bystander (yellow) cells. These phase diagrams are heat maps corresponding to the density ρ_m of the mutating population (red/black strains) after many

generations. Our simulations were performed for $t \approx 10^6$ generations, which yields the steady state solution for the mutating population fraction ρ_m for the vast majority of points on the phase diagram in Fig. 2.4, except for points very near the phase transition line. Note that our improved mean field estimates based on directed percolation and the random walk theory (white and green dashed lines, respectively) do a reasonable job of approximating the shape of the phase boundary, especially when $\mu \approx 0$ and our system reduces to a simple competition between fast-growing red cells and bystander yellow cells. The directed percolation approximation works better near $\mu \approx \mu^*$, where we find the mutational meltdown transition of the invader population which is in the directed percolation universality class.

Another biologically interesting quantity to look at is the survival probability P_{surv} of the progeny of a single red cell invader in a population of yellow bystander cells as $t \to \infty$. Such a survival probability would represent the probability of tumor invasion, for example, from a single mutated cell (i.e, a cell with a newly-acquired driver mutation) within an otherwise healthy population. If the bystander is replaced by the slow-growing strain and we have a two-species evolution, then the evolution will be exactly the same as a directed percolation with a "single-seed" initial condition. We would then have $P_{\text{surv}} \propto \rho_f$ due to the rapidity reversal symmetry of directed percolation [43]. In other words, the survival probability tracks the behavior of the fraction ρ_f of the fast-growing strain in a *different* simulation where the initial condition is a *well-mixed* population (or a population of just the mutating, fast-growing strain). In the three-species model we consider, there is no rapidity symmetry due to the presence of the bystander strain. Nevertheless, we expect that the survival probability P_{surv} vanishes on the same critical surface as the fraction ρ_m (plotted in Fig. 2.4) because the invader strain will not be able to invade if the fast-growing strain is lost from the population. We show in Fig. 2.5 the survival probability P_{surv} at long times, which does indeed vanish at approximately the same place as ρ_m in Fig. 2.4. So, the approximations we used to estimate where ρ_m vanishes serve as good predictors of the transition of the survival probability, as well. We also show the phase boundary at a smaller values of s ($s = 0.1$) in the right panels of Fig. 2.5. Note that our estimates work for the phase boundary in this case, also.

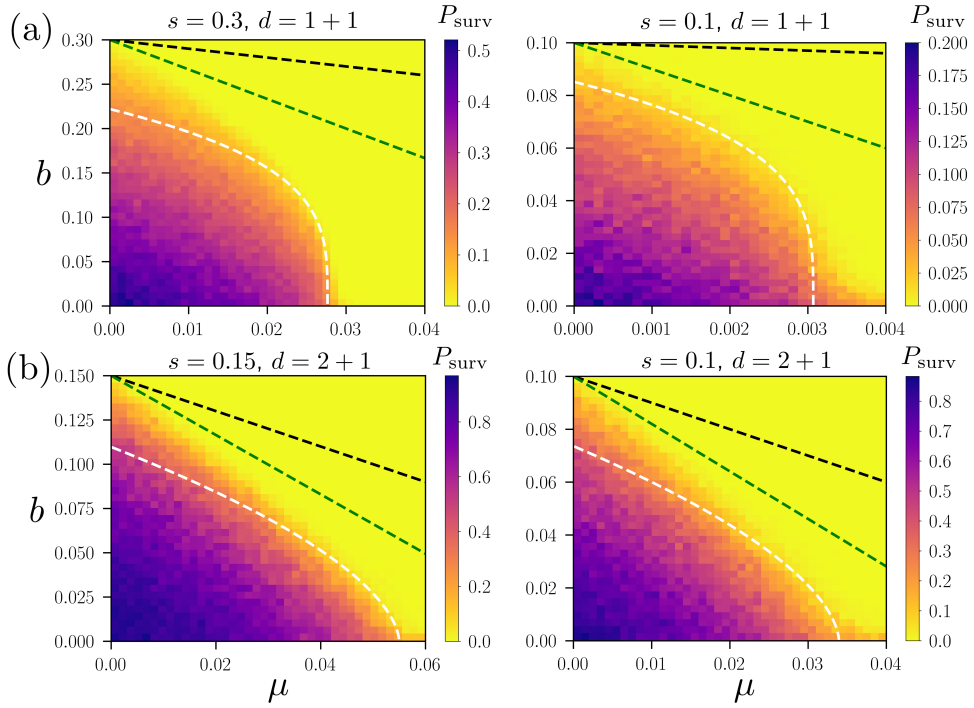

Figure 2.5: Long-time survival probabilities P_{surv} of clusters generated from a single mutating red cell in a yellow bystander population for (a) $d = 1 + 1$-dimensional range expansions for $s = 0.3$ (left panel) and $s = 0.1$ (right panel), after $t = 10^6$ generations on a lattice with size $L = 5000$ cells averaged over 128 runs; and (b) $d = 2 + 1$-dimensional range expansions for $s = 0.15$ (left panel) and $s = 0.1$ (right panel), after $t = 2 \times 10^3$ generations on a lattice with size L^2 cells with $L = 500$, averaged over 256 runs. We show the expected transition shape near the $\mu \approx \mu^*$ DP transition in the white dashed line [Eqs. (5*b*),(6*b*)]. The black dashed line is the transition position for a well-mixed population. The green dashed line is an improved mean-field estimate of the transition discussed in the main text [Eqs. (5*a*),(6*a*)].

2.4 Roughening invasion fronts

We now study the shape of the interface between the mutating and bystander populations. When either of the populations is invading the other, the invasion front behaves as a noisy Fisher-Kolmogorov-Petrovsky-Piscounov wave [6, 58]. Most previous studies of such waves have focused on competition between two homogeneous populations or the range expansion of a population into virgin territory. The noise plays a crucial role here [59], strongly modifying, for example, the wave speed. Also, in the (exactly soluble [60]) $d = 1 + 1$ case, there is a diffusive wandering of the front around its average position.

For $d = 2 + 1$, the situation is more complicated, but generally the noisy wave front will have a characteristic roughening. This roughening falls in the Kardar-Parisi-Zhang (KPZ) universality class [13], although observing the predicted scaling behavior of this class is challenging for noisy Fisher waves [10, 61]. For example, for the KPZ class of interfaces, the characteristic size σ_w of the interface should grow as $\sigma_w \propto t^{1/3}$. However, a basic analysis of the noisy Fisher waves [62] is more consistent with $\sigma_w \propto t^{0.272}$, which is also what we observe in our model. Although this apparent discrepancy has been explained, the proper recovery of the KPZ exponents requires a deeper analysis outside the scope of the current paper [10]. So, for our simulations, we will find consistency with previous analyses of noisy Fisher waves and leave the more extensive analysis of the interface shape scaling for future work. Also, as the speed of the invasion goes to zero, we expect a transition to a different, "voter-model" [63] interface coarsening behavior as both the mutating and the bystander populations become stable and do not invade each other (on average). The interface roughens in a different way in this "critical" case, with the characteristic size σ_w of the interface increasing diffusively as $\sigma_w \propto t^{1/2}$. We will observe such crossovers in our simulation results.

We discuss these issues in more detail below and show that our model exhibits a range of behaviors depending on the invasion speed and the proximity of the mutating population to the meltdown transition. These invasion waves are examples of "pulled" wavefronts [64], which are driven by the growth (invasion) at the leading edge of the wave. Various aspects of such wave fronts are reviewed in, e.g., Ref. [65]. We shall see in the following that adding mutations to one of the populations significantly modifies the expected pulled front wave

behavior and, in the $d = 1 + 1$ case, introduces a super-diffusive wandering of the interface. The $d = 2 + 1$ case presents an even richer set of behaviors depending on the mutation rate and relative fitness of the mutating and bystander populations. Our purpose here will not be the particular value of scaling exponents, but rather general features of the roughening dynamics such as a change in roughening due to the internal evolutionary dynamics of the invading strain.

2.4.1 $d = 1 + 1$-dimensional invasions

In $d = 1 + 1$, domain walls between the bystander and the invading populations can be characterized by a random walk with alternating bias (when $\Gamma_s < \Gamma_b < \Gamma_f$) as the bystander will invade the slow-growing species and be invaded by the fast-growing species within the mutating population. As we approach the mutational meltdown transition, the average size of clusters of the slow-growing strain will diverge as expected from the directed percolation transition. In Fig. 2.6 we see a comparison of two domain walls for $d = 1+1$. At the bottom of the figure, Fig. 2.6(b), we see a domain wall where the mutating red/black population is far from the two-species phase transition. In this case, the black patches in the population are small and do not influence the motion of the invasion front much. Conversely, in Fig. 2.6(a), we see a domain wall with the mutating population near a mutational meltdown. In this case, there is an enhancement of the "roughness" of the domain wall as the large black patches create more regions of alternating bias in the domain wall between the yellow bystander and the red/black invading population.

To obtain a more quantitative estimate of this roughening effect, we set up a simulation with initial conditions that include a sharp boundary between the bystander and the mutating population: the bystander occupies lattice sites $i \leq L/2$, and all other lattice sites $i > L/2$ are occupied by the mutating population (taken to be all red, fast-growing cells initially). We then track the position $x(t)$ of the invasion front over time. We measure the roughness of the front by calculating the variance of the position:

$$\langle [w(t)]^2 \rangle = \langle [x(t) - \langle x(t) \rangle]^2 \rangle = \langle [x(t)]^2 \rangle - \langle x(t) \rangle^2, \tag{7}$$

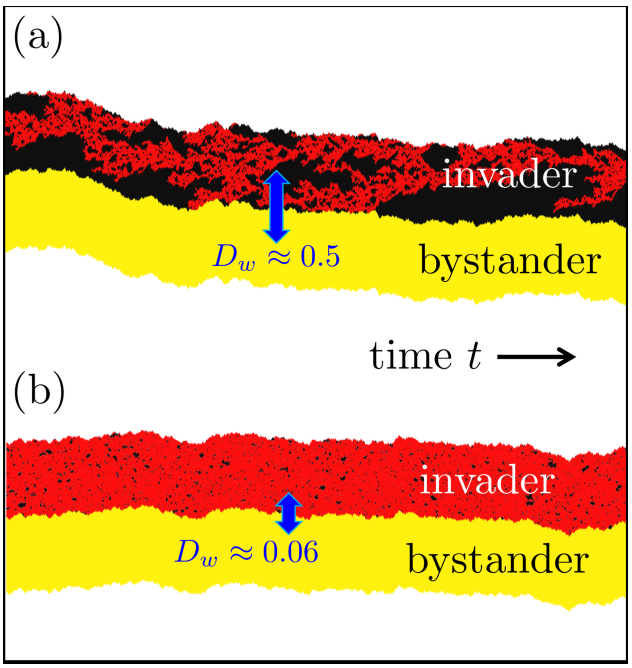

Figure 2.6: A picture of the domain wall in a $d = 1 + 1$-dimensional evolution between the invading black/red population and the bystander yellow population (a) near the two-species (DP) phase transition ($\mu = 0.02765$, $b = 0.045$, $s = 0.3$) and (b) far from the mutational meltdown DP phase transition ($\mu = 0.005$, $b = 0.2775$, $s = 0.3$). We see that the roughness of the domain wall becomes enhanced for (a) with a domain wall diffusion coefficient $D_w \approx 0.5$ as compared to $D_w \approx 0.06$ for (b) (calculated using the time-averaged squared displacements of the wandering interface in the corresponding figures [66]). The width of the population is $L = 300$ cells, and we show the evolution over 10^4 generations. This long evolution time allows for an observation of the domain wall wandering. However, as a result, the cells are compressed along the time direction in this figure.

where we average over sufficient runs to ensure convergence. In the case of a domain wall between just two strains, perhaps with a difference in growth rates, the domain wall performs a biased random walk [4]. Therefore, we may expect that our position $x(t)$ also performs a diffusive motion in time. The number fluctuations at the boundary introduce a stochasticity to the motion, while the difference in growth rates provides a deterministic bias. So, for a boundary between two strains, we expect the variance $\sigma_w(t) \equiv \sqrt{\langle [w(t)]^2 \rangle}$ to satisfy

$$\sigma_w(t) \approx \sqrt{D_w t}, \tag{8}$$

with D_w a diffusion coefficient for the domain wall. Indeed, $x(t)$ itself should perform a biased random walk and we may extract the diffusion constant D_w from a time series of the position $x(t)$ performing a time average of the squared displacements of the interface [66]. We did this for the simulations shown in Fig. 2.6. We see that when the population is near a mutational meltdown at $\mu \approx \mu^*$ [Fig. 2.6(a)], the observed diffusivity is much larger than for a population far away from this transition [Fig. 2.6(b), with $\mu \ll \mu^*$]. However, a proper measurement of D_w requires an ensemble averaging over many simulation runs and also a longer time series.

We shall see in the following that a more detailed analysis of the boundary motion will show that $x(t)$ actually performs a *super-diffusive* motion near the mutational meltdown $\mu \approx \mu^*$, with displacements satisfying $\sigma_w(t) \propto t^\nu$, with $\nu > 1/2$. Super-diffusivity is not uncommon in spatial population dynamics: In a range expansion, for example, the roughness of the expansion front may contribute to the motion of sectors of strains, introducing super-diffusivity to the sector boundary motion [5]. However, this super-diffusivity depends on the conditions of the growth, and a diffusive motion often serves as a reasonable approximation [67, 68]. Moreover, if we are just thinking about populations living in a fixed one-dimensional geometry, then we expect sector boundary motion to be diffusive. We will find diffusive motion of our sector boundaries everywhere in the (s, b, μ) parameter space, except near the mutational meltdown $\mu \approx \mu^*$ where the sector motion becomes super-diffusive.

Let us now analyze the dynamics in more detail. For a domain wall or invasion front between our mutating, heterogeneous invader population and the homogeneous bystander, the slow- and fast-growing strain patches of the invader will interact differently with the

bystander. We can analyze how this impacts the domain wall motion by studying the standard deviation $\sigma_w(t) = \sqrt{\langle [w(t)]^2 \rangle}$, averaged over an ensemble of simulation runs. We sample our evolved population at times $t = t_i$ and calculate the effective exponent associated with the interface width:

$$\nu(t = t_i) \equiv \frac{\ln[\sigma_w(t_i)/\sigma_w(t_{i-1})]}{\ln[t_i/t_{i-1}]}, \tag{9}$$

where we choose $t_i/t_{i-1} \approx 2$. The effective exponent $\nu(t)$ approaches a limiting value at long times. Moreover, any super-diffusive enhancement to the roughness would be seen as a limiting value $\nu > 1/2$. The exponent is plotted for various values of (s, b, μ) in Fig. 2.7. We find that there is an enhanced, super-diffusive roughness ($\nu > 1/2$) whenever the mutating population is close to the mutational meltdown (DP) transition at $\mu = \mu^*$ [along the vertical line in the phase diagram in Fig. 2.4(a)]. The enhanced value of ν near the DP transition may be understood by considering the limiting case $b = 0$. In this case, the bystander population and slow-growing strain within the mutating population will grow at the same rate, so then an initial condition with a single red fast-growing cell in a population of yellow bystander cells will expand as it would in a standard DP process with a single seed initial condition. Hence, the standard deviation $\sigma_w(t)$ scales with the DP dynamical critical exponent: $\sigma_w(t) \sim t^{\nu_{\rm DP}}$, with $\nu_{\rm DP} = 1/z \approx 0.6326$ for $d = 1 + 1$ [43]. This exponent is indicated with the upper dashed line in Fig. 2.7. Introducing a non-zero $b > 0$ should not change the situation much; we would only expect a difference in the bias of the domain wall motion.

Away from the DP transition, the invasion front has a diffusive behavior, with $\sigma_w(t) = \sqrt{D_w t}$. The diffusion constant D_w may be measured and serves as a good indicator of the mutational meltdown transition because D_w should diverge as $\mu \to \mu^*$ for fixed b and s. This is illustrated in Fig. 2.8 for $s = 0.3$ and values of b along the phase transition boundary. In this $d = 1 + 1$-dimensional case, the value of b, according to our simplified analysis, does not change the wandering behavior of the domain walls as it only serves to change the domain wall *bias*. This hypothesis is consistent with the data shown in Fig. 2.7, where the red squares and purple crosses, despite having very different b values, both exhibit super-diffusive exponents $\nu(t) > 1/2$ at long times because both points are near the mutational meltdown transition at $\mu = \mu^*$. We do see small b-dependent differences, but these may be due to the

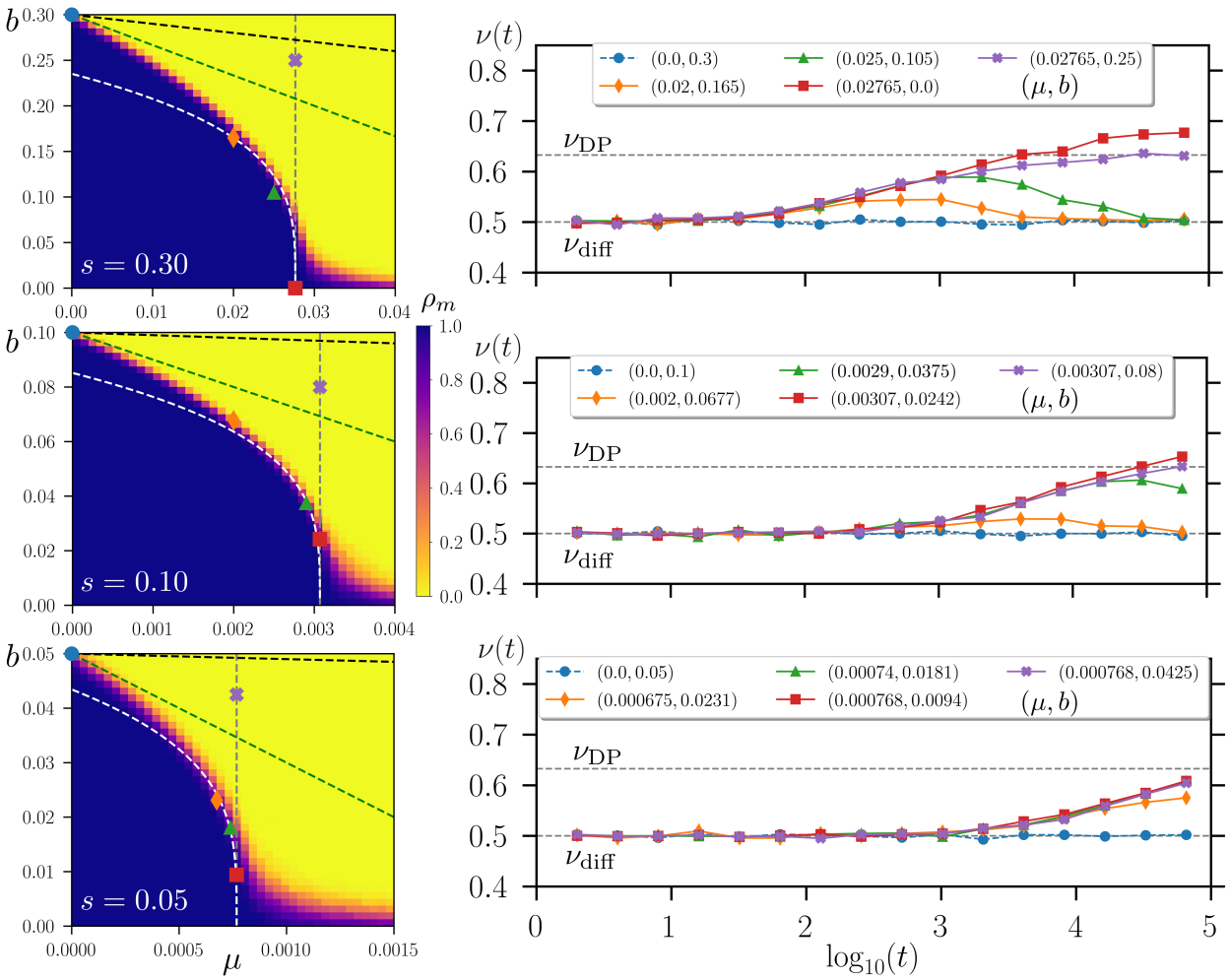

Figure 2.7: For varying values of $s = 0.3$ (top row), 0.1 (middle), and 0.05 (bottom), we show how, as we move along the critical line starting from $(\mu, b) = (0, s)$ towards the two-species critical point at $(\mu, b) = (\mu^*, 0)$, the domain walls separating a bystander population from a mutating one acquire super-diffusive behavior. The phase diagrams on the left illustrate where we calculate the roughness exponent $\nu(t)$ on the right. Two limiting values of the exponent are indicated with dashed lines in each plot: a diffusive $\nu = 0.5$ (lower lines) and a super-diffusive, directed percolation value $\nu \approx 0.6326$ (upper lines). The phase diagrams were created from simulations with $L = 5000$, $t = 10^5$ generations, and averaged over 256 runs. The exponent curves on the right were created from simulations with $L = 5 \times 10^4$, $t = 10^5$ generations, and 400 runs.

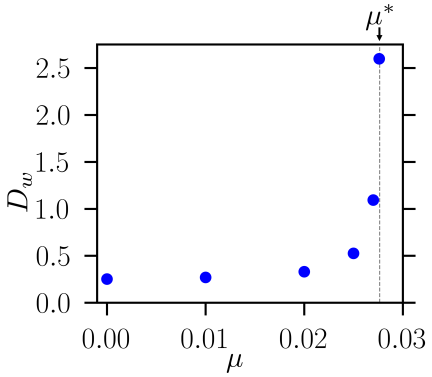

Figure 2.8: The diffusion coefficient, D_w, of the boundary between healthy and cancerous populations, measured for various values of μ approaching μ_c with $s = 0.3$ and b adjusted so that the mutating and bystander populations remain relatively neutral. We see that as $\mu \to \mu^* \approx 0.02765$, D_w diverges as the domain wall becomes super-diffusive. The coefficients were calculated from simulations with $L = 3 \times 10^4$, $t \approx 2 \times 10^6$ generations, and averaged over $N = 1280$ runs.

finite time of our simulations. Indeed, the (super-)diffusive scaling of the interface motion only holds at sufficiently long times. It is clear that, especially for the smaller values of s in Fig. 2.7, that the exponent $\nu(t)$ has not yet saturated to its long-time, limiting value within the simulation time. In any case, it is clear that we find super-diffusive motion whenever μ approaches μ^* in the various panels of Fig. 2.7. As we shall see in the next subsection, the situation changes dramatically for the $d = 2 + 1$-dimensional case where the interface between the bystander and mutating populations will behave differently depending on the value of b.

2.4.2 $d = 2 + 1$-dimensional invasions

For a two-dimensional population, the invasion front is no longer a simple point, but rather an undulating *line* between the bystander and the mutating red/black population. Moreover, this line can *thicken* as pieces of the invader population pinch off and migrate into the bystander population due to rearrangements induced by the cell division. This dissolution of the front is more prominent when the bystander and the invader have approximately the same growth rates. An example of the complicated frontier shapes are shown in Figs. 2.9 and 2.10. In Fig. 2.9 an initially circular mutating population gets reinvaded by the yellow bystander population in (a) and is approximately neutral with respect to yellow population in (b). We see that in (b) the initially circular interface dissolves. In (a), the dissolution is less prominent, but still has an effect. This difference in roughness properties between Fig. 2.9(a) and (b) indicates that the overall growth speed v of the interface between the bystander and the invader will influence the interface roughness. This is in marked contrast to the $d = 1 + 1$-dimensional case where the average difference in growth rates between the invader and bystander only changes the bias of the (super-)diffusive motion of the interface. Thus, for $d = 2+1$, we will have to consider the dynamics at particular values of the average interface speed v.

We begin with some qualitative observations of the dynamics. For relatively neutral populations with an average interface speed $v = 0$, we again expect to see an enhanced roughening of the interface over time when the invader population approaches a mutational meltdown, much like in the $d = 1+1$-dimensional case. We can see the enhanced roughening

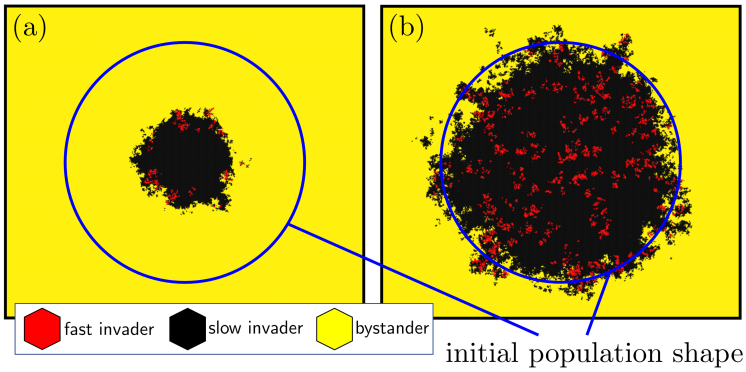

Figure 2.9: Snapshots at $t = 8192$ generations of an initially circular cluster of the fast-growing strain (initial diameter of 400 cells). In (a), the internal parameters (μ, s) of the cancerous population are set such that there is an average bias of the interface so the growth speed $|v| > 0$. In (b), we have $|v| = 0$, and thus we are on the critical line of the 3-species phase diagram. In this case, the interface between populations dissolves, thus our characterization of the interface roughness becomes more complicated in the $d = 2 + 1$-dimensional case than it was for the $d = 1 + 1$-dimensional case, where the boundary was a single point performing a biased (super)diffusive random walk.

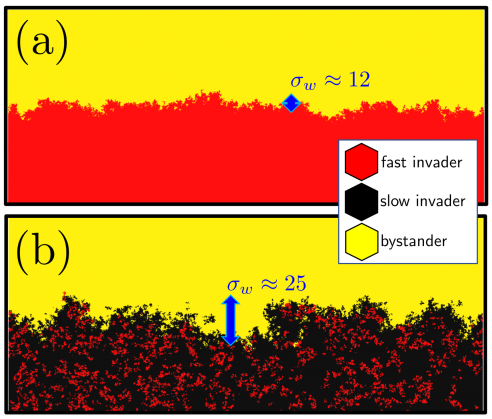

Figure 2.10: A comparison of the invasion front (in a population with a total of $L^2 = 100^2$ cells) along the critical line (equal growths for the bystander and invader) at $t \approx 4000$ generations, starting from an initially uniform, flat interface. In (a), a non-mutating invader is in neutral competition with the bystander ($\Gamma_f = \Gamma_b = 1$ and $\mu = 0$) and the interface remains overall stationary and dissolves over time. In (b), we are just below the mutational meltdown transition with $\mu \approx 0.053$ (for $s = 0.15$) for the invader. We set $b = 0$ so that the interface is stationary. We see dissolving of the interface, but there is an enhanced roughness due to the mutational meltdown. The roughness may be quantified directly for these simulation snapshots via the width σ_w as defined in Eq. (11). Note that width (given in cell diameters) in (b) is twice that in (a).

qualitatively in Fig. 2.10 for this special case where the invader and bystander have the same growth rate [i.e., we are on the phase transition boundary in Fig. 2.4(b) and $v = 0$]. In Fig. 2.10(a) we have a non-mutating invader and in Fig. 2.10(b) we have an invader near a mutational meltdown transition. The frontier is more undulated in Fig. 2.10(b) near the meltdown transition. The increased undulation may be quantified by studying the average interface width σ_w, which we now describe.

In order to partially mitigate the effects of the "fuzzing out" of the interface, we quantify the roughening by looking at the average location of the interface at each time t during the evolution. To do this, we set up a coordinate system where we orient a linear population interface along the x-direction of our lattice and we let x_i represent the zigzagged columns of our hexagonal lattice along this direction, as shown in Fig. 2.11. Then, for each column x_i, we define the interface location by averaging over all N_u locations of red/black cells within a certain range:

$$\overline{y}(x_i, t) = \frac{1}{N_u} \sum_{y=y_{\min}}^{y_{\max}} y(x_i, t), \tag{10}$$

where $y_{\min} = y_{\min}(x_i, t)$ is the location of a black/red cell on the lattice at the point x_i (and time t) such that all cells with $y < y_{\min}$ are also black or red. Similarly, $y_{\max}$ is the location of the red/black cell for which all cells with $y > y_{\max}$ are all yellow. The scheme is illustrated in Fig. 2.11.

Using the average location $\overline{y}(x_i, t)$ allows us to define an interface width $\sigma_w(t)$ by averaging over all x_i along the interface:

$$\sigma_w(t) = \sqrt{\left\langle \frac{1}{L} \sum_{x_i=0}^{L} \left[\overline{y}(x_i, t) - \overline{\overline{y}}(t)\right]^2 \right\rangle}, \tag{11}$$

where

$$\overline{\overline{y}}(t) = \frac{1}{L} \sum_{x_i=0}^{L} \overline{y}(x_i, t). \tag{12}$$

The angular brackets in Eq. (11) indicate an ensemble average over many population evolutions. However, we may also use $\sigma_w(t)$ as an indicator of the front roughness for a single snapshot of a population at a particular time, as shown in Fig. 2.10. Examples of the calculated $\sigma_w(t)$ (averaged over many simulation runs) for various values of selection

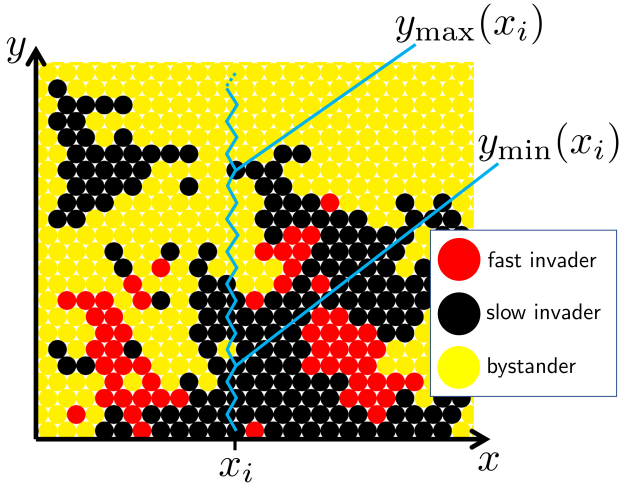

Figure 2.11: Schematic for finding an average location of the interface between the yellow bystander and red/black mutating populations. The interface runs along the x direction. We identify columns x_i in the hexagonal lattice as shown with the blue zigzagged line. At each column x_i, the average position $\overline{y}$ is calculated by averaging over all red/black cell locations between the red/black cell which is the furthest into the mutating region [at $y_{\min}(x_i)$] and the black/red cell which is the furthest into the bystander population [at $y_{\max}(x_i)$].

parameter b and mutation rate μ are shown in Fig. 2.12. For example, in the case where the invader and bystander populations are relatively neutral and there are no mutations, the roughening of the interface illustrated in Fig. 2.10(a) is shown with blue circles (connected by a dashed line) in Fig. 2.12. The interface in Fig. 2.10(b) approximately corresponds to the red squares in Fig. 2.12. Note that, as expected, $\sigma_w(t)$ increases significantly faster in time for the latter case compared to the former.

Note that it is possible to define the interface width σ_w in other ways, including estimating the interface position using the location $y_{\min}$ or $y_{\max}$ (see Fig. 2.11). Alternatively, one might use the difference $y_{\max} - y_{\min}$ as a measure of the "fuzziness" of the interface, which we might also expect to roughen near a mutational meltdown. We have verified that using other definitions of the interface roughness does not change the long-time scaling properties of the interface roughness or the relative enhancement of the roughness near a mutational meltdown. It would be interesting, however, to more systematically study the consequences of using alternative definitions of the roughness.

We will now focus our quantitative analysis on the $v = 0$ case of a stationary (on average) interface, since it is along the critical line where we find a predictable roughening effect. We will then take a closer look at the cases $|v| > 0$ where either the mutating population or the bystander has an overall selective advantage. This introduces complications as the roughening behavior of a moving front is different from a stationary one. Indeed, whenever $|v| > 0$, the invasion becomes a noisy Fisher wave which has its own particular roughening properties. We shall see that a non-zero velocity v will suppress the interface roughness at long times, but signatures of the roughening due to mutational meltdown persist at shorter times.

Voter model coarsening, $v = 0$

Along the 3-species critical surface, where the invader and bystander are relatively neutral, we expect to see an enhancement of the interface roughening as we approach the mutational meltdown transition $\mu \to \mu^*$ for the invader population [the bottom terminal end of the phase boundary in Fig. 2.4(b)]. To quantify the roughening, we can calculate the effective exponent $\nu(t)$ [see Eq. (9)] from the interface width $\sigma_w(t)$ defined in Eq. (11). Without a bias,

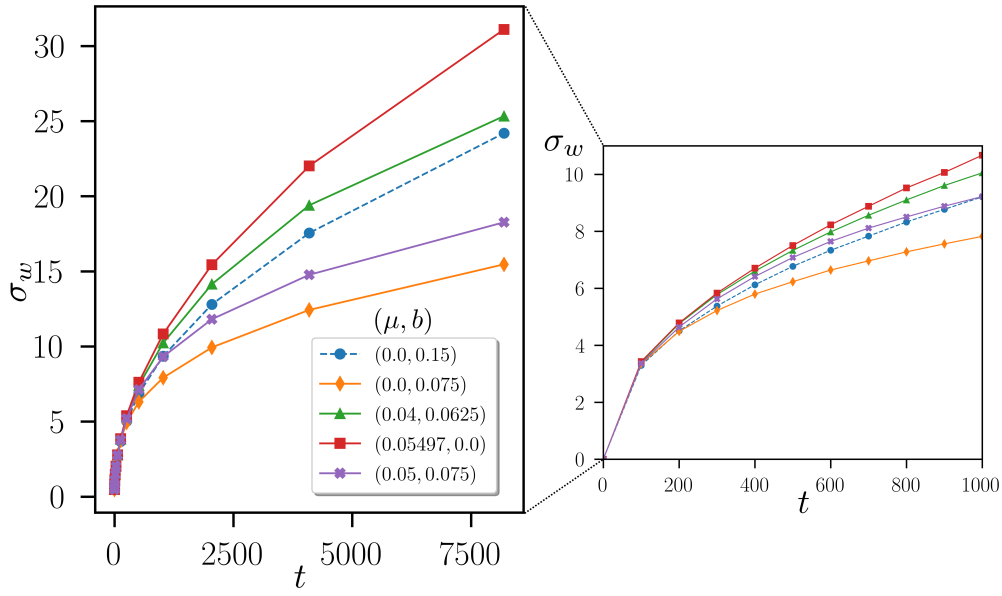

Figure 2.12: The interface width σ_w [see Eq. (11)] in units of cell diameters of a $d = 2 + 1$-dimensional invasion, starting from an initially flat interface between the mutating and bystander population ($\sigma_w = 0$) 4000 cells long and with $s = 0.15$ for various values of selection parameter b and mutation rate μ. [Note that it is helpful to consult the phase diagram in the top panel of Fig. 2.13 for identifying the locations of these points in the (μ, b) plane.] The interface is evolved for $t = 10^4$ generations, and we average over 160 runs. Lines connect the points to guide the eye. Note that the value of b strongly influences the behavior of σ_w, as seen by comparing the red squares and the purple crosses, both of which have the mutating population near meltdown ($\mu \approx \mu^*$). In general, we find a suppressed roughness when the mutating and bystander populations are not relatively neutral (compare blue dashed line to orange diamonds and purple crosses). Otherwise, for (on average) stationary interfaces, we see the enhanced roughness due to population meltdown (green triangles and red squares). The smaller plot shows the roughness at short times.

we expect that the interface coarsening should be described by voter-model-like dynamics [69] because the invader and bystander populations divide into each other without a surface tension. We generally expect a diffusive behavior $\sigma_w \propto \sqrt{t}$ in this case.

In Fig. 2.13 we see an enhanced roughening as $\mu \to \mu^*$ as we move along the phase transition boundary ($v = 0$): The limiting value ν of the exponent increases as we move along the phase transition line towards the mutational meltdown at $\mu = \mu^*$. Interestingly, near mutational meltdown, the width σ_w seems to grow approximately diffusively with $\sigma_w \propto t^{0.5}$ (red squares in Fig. 2.13), whereas the non-mutating case $\mu = 0$ coarsens according to the power law $\sigma_w \propto t^{0.4}$ (blue circles in Fig. 2.13). We might have expected larger values for these exponents as the non-mutating case should be closest to the voter model dynamics where interfaces dissolve diffusively, similarly to the dynamics of σ_w in the $d = 1 + 1$ case away from the meltdown transition [69]. However, generalizations of the voter model can yield different results for interface coarsening and determining the value of the exponent ν can be subtle [70]. Another possibility is that ν is suppressed due to our particular choice of lattice update rules. It would be interesting to study the behavior with simulations with overlapping generations (independently dividing cells).

Although the behavior for $d = 2+1$ is different from the $d = 1+1$ case where the domain wall roughening was clearly super-diffusive near mutational meltdown and diffusive away from it (see Fig. 2.7), we also find here that the mutational meltdown enhances the interface undulations by modifying the exponent ν associated with the interface width $\sigma_w \propto t^{\nu}$, increasing ν from a value of approximately 0.4 to 0.5. A more dramatic difference is found when we move away from the $v = 0$ critical line and have either the mutating invader population or the bystander grow with an overall selective advantage. We then find a moving Fisher wave with a suppressed exponent ν, as we will see in the next section.

Fisher wave roughening, $|v| > 0$

The comparison between $|v| > 0$ and $v \approx 0$ dynamics can be seen prominently if we consider an initially disc-like population of the invader strain. Then, any non-zero velocity will either shrink or grow the initial disc. An example of an $v < 0$ evolution is shown in Fig. 2.9(a) where the bystander strain reinvades the invader, which eventually dies out. Conversely, when $v \approx$

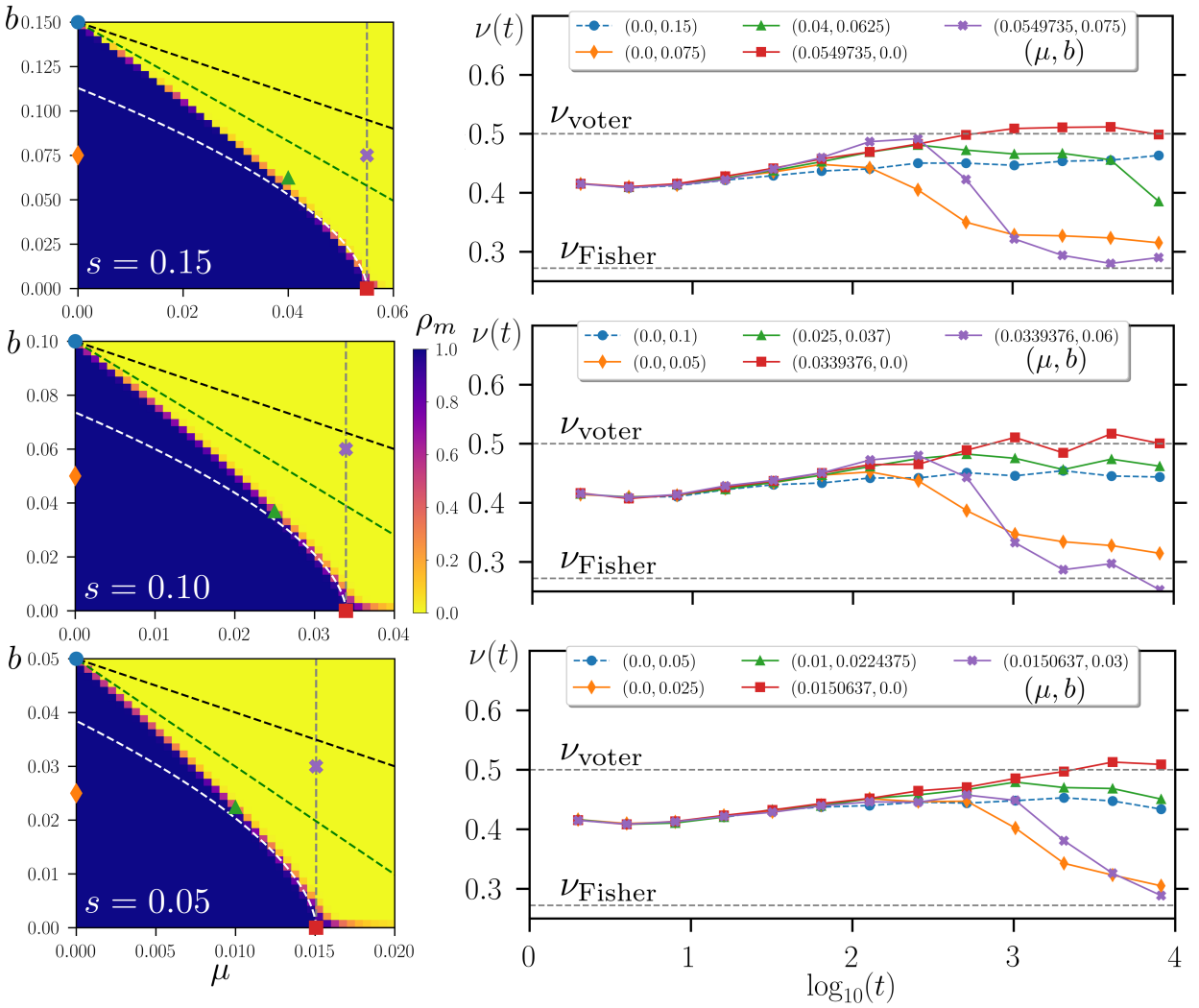

Figure 2.13: Interface roughening exponents $\nu(t)$ are calculated on the right plots for different combinations of (b, μ) indicated on the phase diagrams on the left, for varying values of $s = 0.3$ (top row), 0.1 (middle), and 0.05 (bottom). As we move along the critical line (blue circles, green triangles, and red squares), we show the enhancement of the boundary roughness [from $\sigma_w \propto t^{0.4}$ to $\sigma_w \propto t^{0.5}$]. Away from the critical line (purple crosses and orange diamonds), we see the effects of Fisher wave dynamics. Here either the mutating population (orange diamonds) or the bystander (purple crosses) has a selective advantage, and the moving interface has a suppressed roughness at long times, approaching $\sigma_w(t) \propto t^{0.272}$ (bottom dashed lines in plots on the right), consistent with previous Fisher wave simulation results [62]. The phase diagrams have the same simulation parameters as in Fig. 2.5. The exponent curves on the right use interfaces that are initially 4000 cells long, and we average over 160 runs.

0, we can see in Fig. 2.9(b) that the boundary between the invader and bystander gradually dissolves. This illustrates the key feature that makes $|v| > 0$ different from the critical line: one of the populations (either the mutating population or the bystander) becomes *unstable* and will deterministically shrink in the presence of the other population.

Let us consider first the simplest case when $\mu = 0$ and we have an interface between a (non-mutating) fast-growing red strain and the yellow bystander. The orange diamond point data in Fig. 2.13 show what happens in this case. The interface behaves as a noisy Fisher-Kolmogorov-Petrovsky-Piskunov wave [6, 58] describing the invasion of the bystander. Without fluctuations (in the mean field limit), these waves admit stationary shapes and we have no roughening over time. However, fluctuations prevent the formation of stationary wave fronts for the $d = 1 + 1$ and $d = 2 + 1$-dimensional cases. For $d = 2 + 1$, previous simulations [62] show that the interface width is expected to grow as t^{ν} with $\nu \approx 0.272$. This coarsening is consistent with our results for σ_w, as the orange diamond data points in the right panels of Fig. 2.13 approach the $\nu \approx 0.272$ limiting value at long times, indicated by the lower dashed line. The time until convergence, however, is quite long as the effective exponent $\nu(t)$ continues to decrease over the course of the entire simulation run time.

The case of a non-mutating invader is interesting for $d = 2+1$ because we would naively expect our system to fall into the KPZ universality class. The average interface position $\overline{y}(x_t, t)$ could be interpreted as a kind of "height function" and the interface width σ_w should scale like $\sigma_w \propto t^{1/3}$ at early times, consistent with $d = 1 + 1$-dimensional KPZ dynamics. A broad class of systems fall into this universality class (see [48] for a review) as the KPZ equation includes the most relevant nonlinearity associated with lateral growth. However, we see here that the behavior is more subtle as the fuzzing out of the interface will contribute to the measured roughness. This complication in measuring the interface roughness was discussed and analyzed in previous work [10]. Our focus here, however, is not the particular exponent associated with the roughening but rather the effects of adding mutations. We will see that adding mutations does enhance the roughness, but only at short/intermediate times while the fast-growing, mutating strain maintains a significant fraction within the population.

Consider the portion of the phase diagram where the bystander can reinvade the mutating population due to fitness loss at a non-zero mutation rate μ (purple crosses in Fig. 2.13). Here, the evolution of our system begins at first as biased competition between two species (between *fast-growing* and *bystander* species) but as the *fast-growing* cells mutate and die off, the bystander population begins to reinvade the *slow-growing* species, and eventually we should find a Fisher wave of the bystander invading the less fit, mutating population. On the right side of Fig. 2.13 we see that the exponent $\nu(t)$ for the purple crosses at first is enhanced as we would expect near mutational meltdown ($\mu \approx \mu^*$). At later times, however, once a Fisher wave is established, the exponent eventually dips down and is consistent with a Fisher-wave-like coarsening [62]. One can track this especially easily in the $s = 0.15$ case (top row of Fig. 2.13) where we see that the purple cross data points follow the critical roughening points (red squares) and then transition to a slower roughening more consistent with a regular Fisher wave (orange diamonds). The evolution of $\sigma_w(t)$ for this case is also shown in Fig. 2.12. One sees here that at times $t < 1000$ (smaller plot), the purple cross data points have a larger width $\sigma_w(t)$ due to the mutational meltdown dynamics, but $\sigma_w(t)$ then crosses over to smaller values for longer times when the Fisher wave behavior dominates.

2.5 Conclusion

We have now analyzed a simple model of invasion of a stable, homogeneous population by a population acquiring deleterious mutations at a rate μ. We examined this invasion in both one- and two-dimensions as a function of the mutation rate μ, the selective advantage s of the fast-growing strain within the mutating population, and the selective advantage b of the bystander population. We have shown that the effectively small local population sizes (compared to a well-mixed population) suppress the probability that the invasion succeeds. This suppression can be understood by analyzing the motion of the boundary between the mutating population and the bystander population it is invading. We find a reasonable estimate of the phase transition position in the (μ, b, s) phase space, as shown in Figs. 2.4,2.7, and 2.13. Our model assumed that cell motility within our population is suppressed, with the only cell rearrangements occurring due to cell division and local competition for space. It

would be interesting to consider the effects of a spatial motility as it has been demonstrated that some of the expected features of spatial dynamics, such as spatial heterogeneity and local fixation of strains is partially mitigated by increased cell motility [71].

Next, we considered the properties of the invasion front and showed that this front undulates more when the mutating population is near the meltdown transition at which it loses the fittest strain. For $d = 1 + 1$ dimensions, this transition is well-characterized by the directed percolation universality class, and we used properties of this class to understand the enhancement of the roughening. In the future, it would be interesting to compare our results to experiments. One possibility is to use microbial populations such as bacteria or yeast where one may design strains with varying (μ, b, s). Another possibility would be to examine such invasions in cancers. For instance, it would be interesting to monitor the edges of a tumor over time as it either grows or shrinks. We predict that if the tumor begins losing fitness due to accumulated deleterious mutations (during treatment, for example), then we should be able to observe this transition to "mutational meltdown" as a roughening of the tumor edges.

For $d = 2+1$ dimensions, we find a range of behaviors for the roughening interface. When the speed of the invasion front approaches zero, the interface roughens more significantly due voter-model-like coarsening. We also find an enhancement of the roughening exponent as the mutating population approaches meltdown. On the other hand, when either the mutating or the bystander population has a selective advantage and the population interface develops an overall velocity, the roughening is suppressed, and we find roughening exponents consistent with those observed for noisy Fisher waves at long times. Therefore, the long-time behavior of the population interface roughness serves as an indicator of whether or not a selective sweep is occurring within the population: Moving population fronts will be smoother than stationary ones in which the invader and bystander populations are relatively neutral.

At intermediate times, we see signatures of the meltdown as the interface roughens more rapidly when the mutating population is near the meltdown transition, even in the case when there is an overall bias to the interface motion (see smaller plot of Fig. 2.12). Also, we focused here on just one aspect of the roughening, namely the early time behavior $\sigma_w \propto t^{\nu}$. For long times t, the interface undulation size will eventually saturate due to the finite system size L,

and we might expect a general scaling form $\sigma_w = t^\nu f(t/L^\beta)$, with $f(x)$ a scaling function and β a new critical exponent. The scaling properties of this saturation should also depend on the proximity to the mutational meltdown transition. It would also be interesting to consider a $d = 3 + 1$-dimensional evolution such as the invasion of surrounding tissue by a compact cluster of cancerous cells. In this case, the invasion front would be an entire *surface* which could also pinch off and coarsen. Previous simulations of the noisy Fisher wave dynamics suggest that the situation in this case is similar to the $d = 2+1$ case considered here [62]. We would again expect to find some enhancement of the interface roughening when a mutating invader is near a mutational meltdown transition.

Interestingly, increased roughening is typically an indicator of more malignant cancerous growths, and the roughness of tumor edges has been a useful prognostic indicator in a wide variety of cancers [72]. Also, in general, increased heterogeneity results in a worse clinical prognosis [73]. While our results point to the possibility of an opposite correlation, our model does not take into account tumor vasculature or cancer cell motility. Conversely, most of the clinical studies focus on more mature tumors which have developed a vasculature. Hence, we expect our model to be relevant for early, small avascular tumors or regions of larger tumors lacking vasculature. These small tumors are not easily detected as they are typically just a few millimeters in size. Nevertheless, small spheroidal avascular tumors are good *in vitro* models for early cancer growth [74]. It would thus be interesting to study the edges of such cultured tumors under a large mutational load. We may also verify some of our results in microbial range expansions (e.g., in yeast cell colonies grown on Petri dishes) where there is little cell motility. A promising experimental realization of a $d = 2+1$-dimensional expansion may be a growing cylindrical "pillar" of yeast cells, as realized in Ref. [75].

Chapter 3

Experimental motivations for the bystander model

Although the model introduced in the previous chapter is an interesting statistical model for the study of roughening phenomena in its own right, I should acknowledge that I have thus far motivated the study of this model by indicating that it could offer insight into the mutational meltdown of cancerous tumors. I have yet in this thesis to discuss experimental data about how tumors grow, mutate, and die, nor what the shape of a tumor looks like as it does these things. I attempt to remedy that in this chapter, where I will seek to justify the study of this model, and my identification of the model as one that is useful for the study of tumor growth.

3.1 Driver and passenger mutations

Although it is not possible to directly observe the earliest stages of human cancer growth, advances in genetic sequencing and genotyping have allowed researchers to investigate the genetic history of large tumors; these studies have found that cancerous tissues contain a large number of genetic alterations. Very few of these genetic alterations lead to the development of a cancerous tumor. The few that drive cancer growth are called "driver" mutations, since they confer the cells in which they arise a selective advantage which drives cancer growth. The various other alterations are called "passenger" mutations, and they

mostly have a negligible impact on cell fitness. Some passenger mutations, however, may have a deleterious affect on the cell fitness. Despite their deleterious affect on cell fitness, these passenger mutations can accumulate in the overall cell population enough to have an effect on the tumor's long-term survival, leading, for example, to mutational meltdown of the tumor over time [32, 31].

In my bystander model introduced in the previous chapter, I assume that the existence of the fast-growing species is due to a driver mutation which confers to the cell a selective advantage. As discussed in [31], as a tumor grows, there is always a chance for a new driver mutation or a passenger mutation to occur. When a new driver mutation occurs, the selective advantage of the mutation causes it to sweep through the population contributing to tumor proliferation; after some time, however, deleterious passenger mutations begin to accumulate which again slows the tumor growth until the next driver mutation occurs. This is demonstrated in Fig. 3.1, taken from [31]. Thus, since I am considering only a single driver mutation, my model can be thought of as simulating only one period of the saw-toothed pattern in Fig. 3.1, where accumulations of passenger mutations cause the growth rate of the tumor population to slow down and possibly lead to mutational meltdown.

In my model, every time a fast-growing cell divides it will mutate at a rate of μ into the slow-growing strain. In my simulations, the mutation rate typically has a range of $\mu \sim 0.0 - 0.015$ division^{-1}. The simulations of mutational meltdown of tumor populations in [31] used ranges for the mutation rate, μ, and selective advantage (s and b in my model) which were centered on values obtained from experimental data. The values explored correspond to a range of $\mu = 0.00005 - 50.0$ division^{-1} based on an estimate from literature of $\mu = 0.05$ division^{-1}. The literature-based estimate is slightly larger than the typical values used in my simulation, though on the same order of magnitude. Note that we also assume a one-step fitness decrease of the mutant, whereas a more realistic model would take into account the gradual fitness decrease from an accumulation of multiple deleterious mutations. It would be interesting to incorporate this more detailed fitness landscape, but we can think of our model as describing the fate of the initial, most-fit strain within the invading tumor population [15].

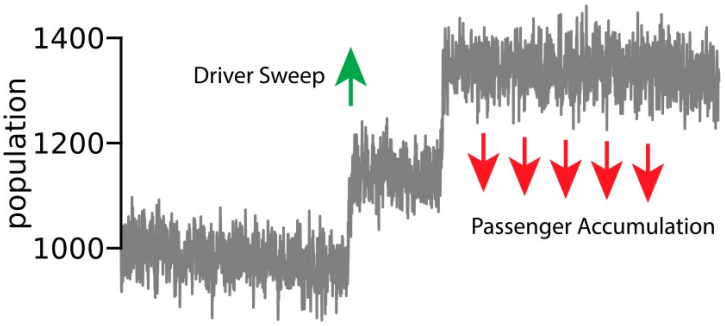

Figure 3.1: A schematic of cancer evolution taken from [31]. This characteristic saw-tooth pattern in the population size of a tumor is due to the initial sweep of an advantageous driver mutation through the population which causes the tumor to proliferate. After the accumulation of passenger mutations, however, the growth begins to slow down and perhaps even decrease until the next driver mutation occurs.

The range of values for μ used in my simple competition-mutation model was constrained by the fact that the mutational meltdown phase transition for the fast-growing population occurs when $\mu = \mu_c \sim s^2$ in $1+1$-dimensions and $\mu = \mu_c \sim s/\ln(s)$ in $2+1$-dimensions. In $3+1$-dimensions, μ would be a bit higher than that estimated from literature in [31] since the phase transition will occur at $\mu = \mu_c \sim s$. Thus the value used to the selection parameter, s, is equally important as the mutation rate; this was estimated in [31] to be $s \sim 10^{-3}$ and the range explored in the mutational meltdown simulations was $10^{-1} - 10^{-4}$. In my simulations, I used values in the range $10^{-2} - 1.5 \times 10^{-1}$. I used slightly larger values, since simulations require much more simulation time for smaller s since it takes longer for the total population of bystander + mutating population to fix. However, as we can see, these values still correspond to biologically relevant situations.

So far I have discussed the selective advantages or disadvantages a driver or passenger mutation may confer to the cell in which they occur, without any discussion of the mechanisms through which these advantages are realized through competition between cells. I will discuss this briefly in the next section, and then I will discuss roughening phenomenon in tumors.

3.2 Dynamics of cell-to-cell competition

The competition that occurs between cells in the bystander model is simply due to a difference in growth rate. This is, of course, a very simplified version of what happens in cell-to-cell competition in real tissues. Although the specific mechanisms of cell-to-cell competition is still an open question in the biology, and there are many mechanisms that seem to play a role, including growth rate signaling and nutrient absorption, the end result remains the same: cells with higher growth rates tend to proliferate, and cells with lower growth rates tend to die out [76]. Some possible mechanisms for competition between cells is illustrated in Fig. 3.2 along with an illustration of the resulting modes of competition: proliferation of faster-growing cells, or elimination of unfit cells; the illustrations in the figure are taken from [76].

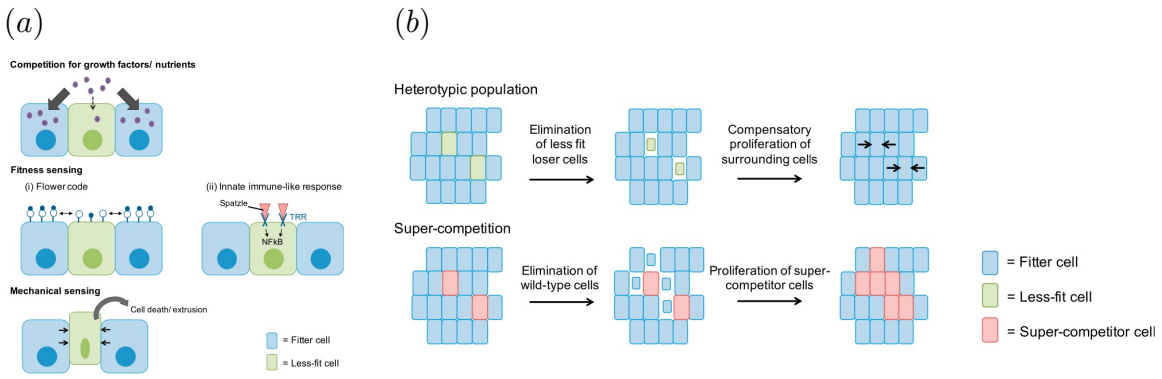

Figure 3.2: **(a)** Schematics, taken from [76], of some proposed mechanisms for cell-to-cell competition in cell tissues. Note that a wide variety of biological mechanisms may be involved in interactions between different cell strains. Despite the variety of mechanisms through which cell-competition occurs, the results of cell competition occurs in two main modes as depicted in **(b)**: elimination of less fit cells, or proliferation of more fit, or faster-growing, cells.

Since the purpose of the bystander model is to quantify the roughening behavior at an invasion front as one population approaches mutational instability, we have modelled competition as occurring only due to the relative difference in growth rates, ignoring the possible mechanisms for cell competition to occur. This is sufficient for the purpose of my model, since the basic result of fitter cells proliferating and less fit cells dying out is indeed the main result of cell-to-cell competition mechanisms in real cell populations. More biologically relevant computational models exist, which take into account, for example, nutrient diffusion, cell proliferation/quescience/necrotization, or fitness signaling between cells [77, 78]. It would indeed be interesting to incorporate these more complex dynamics into a future model of roughening invasion fronts.

I have opted to keep cell competition in my model as a simple difference between growth rates, since I am less concerned with the specific internal dynamics of the "cancerous" population (other than whether or not the population is close to mutational meltdown) and am instead interested in what happens at the boundary between the bystander and cancerous populations. Of course, the macroscopic dynamics of the two populations arises due to the microscopic dynamics of the competition between these two populations, and indeed certain aspects of the bystander model oversimplify with how a tumor really grows. For example, a real avascular tumor will only grow to a certain size before it stops expanding; this is due to the necrotization and quiecense of cells at the tumor interior, which results in a saturation of tumor size after which time the proliferating tumor cells at the edge begin to grow inward rather than outward, likely due to an accumulation of chemical signals released by the necrotizing cells at the center of the tumor [78, 79]. Moreover, sufficiently large tumors will develop a vasculature which allows for nutrients to penetrate the tumor interior. This results in a complex distribution of dividing cells within the tumor population.

Thus, my model should not be understood as an attempt to model the growth of a realistic tumor, but rather as an attempt to quantify how interfaces between populations evolve as a function of one of the population's internal dynamics. Indeed, we may think of a small patch of a tumor which is receiving enough nutrients to grow and which contains some distribution of competing strains. The roughening effect shown to occur in the bystander model in the previous chapter should be understood as a general behavior that one might

expect to observe in the shape of a tumor that is close to mutational meltdown. One signature of this meltdown would be a roughening of the interface between the growing tumor and the surrounding healthy tissue. As imaging technologies evolve, it will be possible to test these predictions via detailed, spatially-resolved imaging of cancer progression.

3.3 Roughening in cancerous tumors

Although imaging a time sequence of *in vivo* tumors is difficult, much progress has been made in the quantitative analysis of *in vitro* tumor shapes. Indeed, it is possible to culture cancerous cells in various geometries. A particular geometry of interest is a "microspheroid" of cells, which has been increasingly appreciated as a more faithful representation of real tumors than, say, a well-mixed culture of cells [80].

A quantitative analysis of tumor roughness, however, can be found in [81], in which the roughness of cultured brain tumor cells in a two-dimensional medium was analyzed. The typical profile for the growing brain tumor, taken from the paper, is shown in Fig. 3.3 and the power-law behavior that was extracted for the width function of the tumor is shown in Fig. 3.4. Interestingly, the authors find that the initial roughness of the tumor grows as $\langle w(t) \rangle \sim t^{0.38}$, which was confirmed for a variety of tumor types [82]. This value is close to our simulated value in the bystander model (see Fig. 2.13) at early times, but this cultured tumor is not competing with health tissue, so our model does not strictly apply. Another study with cultured cancer cells found a KPZ scaling with $\langle w(t) \rangle \sim t^{0.32}$ for both linear and radial tumor interfaces [83]. The linear profiles are shown in Fig. 3.3(a). Examples of some radial profiles are shown in Fig. 3.3(b). The scaling of the tumor interface roughness may be measured in these cultured tumors and a robust scaling regime may be measured, as shown in Fig. 3.4.

The experiments presented here have limitations as they are not direct tests of our model. For example, it would be interesting to perform an experiment where such a cultured tumor coexists with another cellular population which may compete with the tumor cells, as in our "bystander" strain described in the previous chapter. The analysis of the tumor roughness in [81, 83] also did not look at tumors nearing mutational meltdown. This would also be

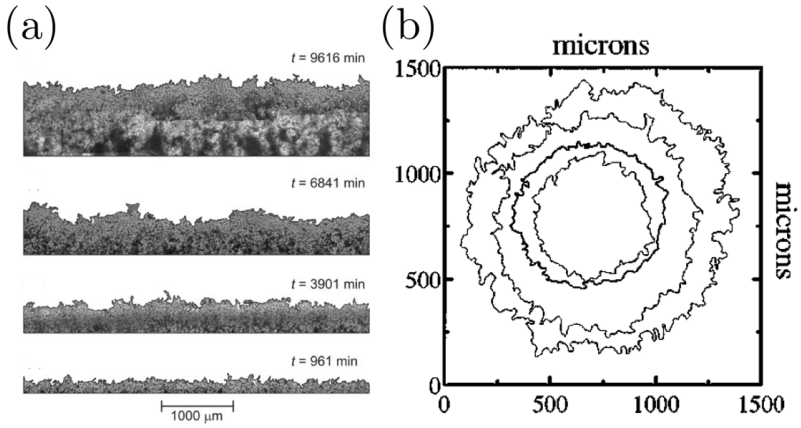

Figure 3.3: (a) Taken from [83], profiles of a growing, cultured tumor (Hela cells). Note the roughening of the front over time. (b) Taken from [81], typical profiles for the growth of brain tumors (a rat astrocyte glioma) cultured on a two-dimensional medium in radial configurations.

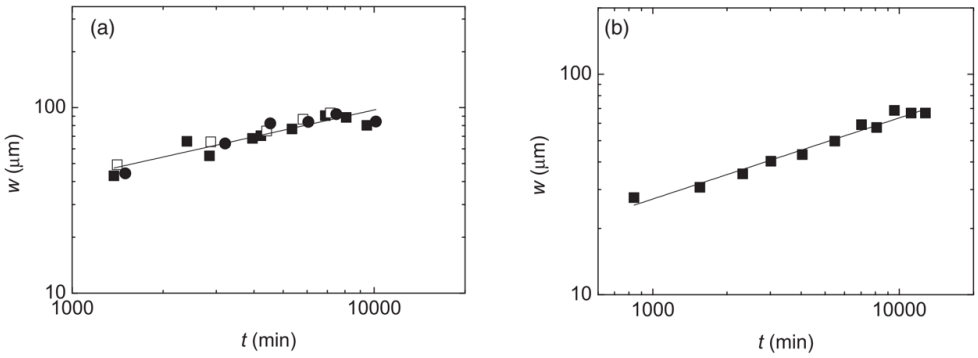

Figure 3.4: Taken from [83], the average tumor profile roughness as a function of time t. The straight lines correspond to the scaling $w \sim t^{0.32}$. The profiles are calculated for both linear tumor shapes in (a) and for radial ones in (b). Examples of these two different geometries are shown in Fig. 3.3.

an interesting condition which would be an explicit test of the predictions made in the previous chapter. These previous experimental results, however, do provide good evidence that my model is on the right track with thinking about roughness and power-law behavior for tumors. Indeed, tumor edges exhibit power-law scaling of their roughness, as shown explicitly in Fig. 3.4, and this roughness may provide signatures of the internal dynamics within the tumor.

Chapter 4

Implementation of lattice simulations in C++

In the previous chapters I have shown the results of simulations of evolving cellular populations, with a special focus on the bystander model for cancer growth introduced in Chapter 2. In this chapter I will give a brief overview of how I implemented these model using Monte Carlo simulations in C++.

All the simulations I will describe are of stochastic lattice models. The general set-up for the bystander model is as follows:

1. Start with a lattice containing N sites. The connections between sites will be determined by the geometry of the lattice. In the following I will use triangular lattices.
2. Assign to each site on the lattice a cell belonging to one of three strains: fast-growing, slow-growing, or bystander.
3. Define update rules. In the following I use two kinds of update rules: parallel updates, or random-sequential updates.
4. Update the lattice according to the update rules and increment the time accordingly.
5. Repeat step 4 until the time reaches the pre-determined total simulation time.

For the remainder of this chapter, I will go through each of the steps laid out above to show my implementation of the bystander model using C++.

4.1 Step 1 - Setting up the lattice

Setting up the lattice is the most straightforward step. A lattice of any dimension and geometry is just a collection of N sites, and so can be implemented as a simple 1-dimensional vector of integers:

```
vector<int> lattice(N);
```

I have opted to keep all vectors flattened (that is, 1-dimensional) for optimization purposes. Doing the multi-dimensional vector-indexing manually allowed me to speed up the simulation by storing certain values that are repeatedly calculated, like $L \times L \times 3$ (as will become clear later) say, as a global parameter. This speeds up vector access, which is beneficial since vector access takes up a significant part of the simulation (particularly for the random sequential case, as we will see). Doing the vector indexing calculations manually also allows me to place certain calculations in an optimized location to further prevent needless calculations being repeated such as in nested for-loops.

The number of sites on the lattice will be determined by the dimensionality of the system being modelled. It is useful to introduce a length variable, say L, so that the number of sites will be given as $N = L^d$ where d is the dimensionality of the system:

```
int L = //size of the lattice;

//one dimensional lattice
vector<int> lattice(L);

//two dimensional lattice
vector<int> lattice(L*L);

//three dimensional lattice
vector<int> lattice(L*L*L);
```

4.2 Step 2 - Assigning cells to each site

To initialize the lattice, I assign an integer value to each site. In the bystander model I used "1" to indicate the bystander strain, "2" to indicate the fast-growing strain, and "3" to indicate the slow-growing strain. In a typical simulation, I start with an initial lattice with

each site randomly assigned either "1" or "2". For the random generator I used the Marsene Twister pseudorandom number generator via the Boost C++ library:

```
#include <boost/random.hpp>

int seed = time(0);
boost::mt19937 generator(seed);
boost::random::uniform_int_distribution<int> disint(1,2);
void latt_init(vector<int> &latt, int Nsites)
{
  for (int i=0; i<Nsites; ++i) {
    latt[i] = disint(generator);
  }
}

//1D lattice
vector<int> lattice(L);
latt_init(lattice, L);

//2D lattice
vector<int> lattice(L*L);
latt_init(lattice, L*L);

//3D lattice
vector<int> lattice(L*L*L);
latt_init(lattice, L*L*L);
```

4.3 Step 3 - Defining update rules

The geometry of the lattice will be defined by the connections between sites on the lattice. The way this is done in practice will vary slightly depending on the type of update rules used. For 1- and 2-dimensional lattices, I used parallel update rules; whereas for the 3-dimensional case I used random-sequential update rules, which I will discuss later.

4.3.1 Parallel update rules

For parallel update rules (for 1- and 2-dimensional lattices), I make use of two vectors, one representing an "even" lattice, and the other representing an "odd" lattice. These two

identical lattices will represent subsequent generations of the evolving population. At each update, I will populate one lattice based on the current state of the other. The way this is done will vary depending on the dimension of the lattice.

1-dimensional lattice

The one-dimensional lattice is just a straight line with neighboring sites connected to one another. At every update, neighboring sites on the lattice will compete with one another to occupy a single site on an identical lattice. This is pictured in Fig. 4.1.

Using Fig. 4.1 as a reference, it is clear which sites will compete with one another during the update. To run the update, I will cycle through the next lattice (odd or even) and determine which cells from the previous generation will compete to occupy each site on the new lattice. Additionally, if the mutation parameter for the simulation is positive, and the winner of the competition has identity "2" (representing the fast-growing strain), then I also check to see if a mutation event occurs in which case the current site will ultimately occupied by identity "3" (representing the slow-growing strain). I repeat this process for a pre-determined number of generations:

```
int N_generations = //number of generations to simulate
int L  = //size of lattice

vector<int> latt_even(L);
vector<int> latt_odd(L);

boost::random::uniform_real_distribution< double > dis(0,1);

latt_init(latt_even, L);

//do the updates
for (int t = 1; t < N_generations; ++t) {

  //update odd lattice
  for (int i = 0; i < L; ++i) {
    int rannum = dis(generator);
    latt_odd[i] = compete(latt_even[mod(i-1,L)], latt_even[i], rannum);

    if (mu > 0.000001 && latt_odd[i] == 2) {
```

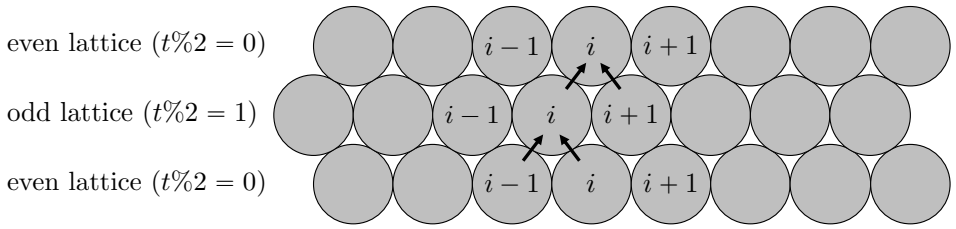

Figure 4.1: In a 1-dimensional lattice, closest neighbors compete with one another to populate a single site in the next generation, represented as an identical lattice offset by 0.5 (given a cell-size of $a = 1$). In the figure, $x\%y$ represents x modulo y, or the remainder of the operation x/y. The labels in the figure show how the indices of competing partners depend on whether the current generation is represented by an "even" or "odd" lattice.

```
      int rannum2 = dis(generator);
      if (rannum2 < mu) ++latt_odd[i];
    }
  }
  ++t;

  //update even lattice
  for (int i = 0; i < L; ++i) {
    int rannum = dis(generator)
    latt_even[i] = compete(latt_odd[i], latt_odd[mod(i+1,L)], rannum);

    if (mu > 0.000001 && latt_even[i] == 2) {
      int rannum2 = dis(generator);
      if (rannum2 < mu) ++latt_even[i];
    }
  }
}
```

where the `compete` function is yet to be defined, and I have made use of periodic boundary conditions via the following mod function:

```
int mod(int a, int b)
{
  return (a%b + b)%b
}
```

to ensure the result is always positive, since vectors in C++ do not support negative indices.

The `compete` function takes in two integers (1, 2, or 3) and a random number and returns an integer (1, 2, or 3) depending on the relative growth and/or mutation rates. This function is implemented as follows:

```
int compete(int A, int B, double RN)
{
  double g_A, g_B;

  g_A = 1.0 - s_By*(double)((3-A)*(2-A))/2.0 - s_S*(double)((A-2)*(A-1))/2.0;
  g_B = 1.0 - s_By*(double)((3-B)*(2-B))/2.0 - s_S*(double)((B-2)*(B-1))/2.0;

  if (RN < gA / (gA + gB)) {
    return A;
  } else {
    return B;
  }
}
```

Where `s_By` and `s_S` are the selection paremeters for the bystander and slow-growing strains, respectively. So `g_A`,`g_B` represent Γ_A, Γ_B, or the growth rates for the competing species `A`, `B` $\in \{1, 2, 3\} = \{\text{bystander}, \text{slow-growing}, \text{fast-growing}\}$. If the drawn random number is less than the relative growth rate of Γ_A as compared to Γ_B then species `A` wins; otherwise, species `B` wins. Thus, if the relative growth rate of Γ_A is very large, there is a higher chance of it winning, or if it is very small there is a lower chance, as expected.

2-dimensional lattice

The 2-dimensional lattice will be updated in the same way as the 1-dimensional lattice, with an odd and even lattice that are alternately updated based on the state of the other. Each lattice will now be two-dimensional, which can easily be obtained from the lattice shown in Fig. 4.1 but now treating the time direction as a spatial direction. Then, the even and odd lattices will be stacked on top of one another in layers which will be the new time direction. This schematic, with appropriate index labels, is depicted in Fig. 4.2.

Using Fig. 4.2 as a reference for the appropriate indices of competing neighbors, we can write down the update rules for the 2-dimensional lattice which will be almost identical to the 1-dimensional case. The differences of course will be in the competing neighbors, and also the compete function will need to be adjusted since now there are three competitors instead of just two, so a simple comparison of a random number against a single relative growth rate will no longer suffice.

```
int N_generations = //number of generations to simulate
int L  = //size of lattice

vector<int> latt_even(L*L);
vector<int> latt_odd(L*L);

boost::random::uniform_real_distribution< double > dis(0,1);

latt_init(latt_even, L*L);

//do the updates
for (int t = 1; t < N_generations; ++t) {
```

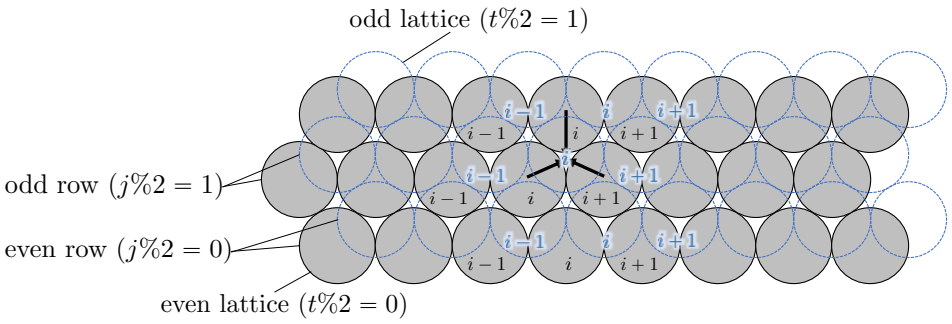

Figure 4.2: The 2-dimensional lattice is obtained by treating the time-like direction of the 1-dimensional lattice as a spatial dimension. The time direction is then obtained by stacking offset copies of the 2-dimensional lattice on top of one another. Now, three neighbors will compete with one another to populate a single site in the next generation. As before, $x\%y$ represents x modulo y. The labels in the figure show how the indices of competing partners depend on whether the current generation is represented by an "even" or "odd" lattice in addition to whether the new site exists on an even or odd row..

```
//update odd lattice
for (int j=0; j<L; ++j) {
  for (int i=0; i<L; ++i) {

    int currindex = j*L + i;

    //get neighboring indices
    int i_left = mod(i-1,L), i_right = mod(i+1,L);
    int j_down = mod(j-1,L), j_up = mod(j+1,L)

    if (j%2==0) {
      // even j
      int competitor_a = latt_even[j*L + i];
      int competitor_b = latt_even[j*L + i_right];
      int competitor_c = latt_even[j_up*L + i_right];
    } else {
      //odd j
      int competitor_a = latt_even[j*L + i];
      int competitor_b = latt_even[j*L + i_right];
      int competitor_c = latt_even[j_up*L + i];
    }

    int rannum = dis(generator);
    latt_odd[i] = compete(competitor_a, competitor_b, competitor_c, rannum);

    if (mu>0.000001 && latt_odd[i]==2) {
      int rannum2 = dis(generator);
      if (rannum2 < mu) ++latt_odd[i];
    }
  }
}
++t;

//update even lattice
for (int j=0; j<L; ++j) {
  for (int i=0; i<L; ++i) {

    int currindex = j*L + i;

    //get neighboring indices
    int i_left = mod(i-1,L), i_right = mod(i+1,L);
    int j_down = mod(j-1,L), j_up = mod(j+1,L);

    int rannum = dis(generator)
    if (j%2==0) {
```

```
        //even j
        int competitor_a = latt_even[j*L + i];
        int competitor_b = latt_even[j*L + i_left];
        int competitor_c = latt_even[j_down*L + i];
      } else {
        //odd j
        int competitor_a = latt_even[j*L + i];
        int competitor_b = latt_even[j*L + i_left];
        int competitor_c = latt_even[j_down*L + i_left];
      }
      latt_even[currindex] = compete(competitor_a, competitor_b, competitor_c, rannum);

      if (mu > 0.000001 && latt_even[i] == 2) {
        int rannum2 = dis(generator);
        if (rannum2<mu) ++latt_even[i];
      }
    }
  }
}
```

The new `compete` algorithm will need to be able to pick between three competitors weighted by their respective growth rates. Since the random number is a uniform real distribution between $(0, 1)$, and there are only three possible outcomes (1, 2, or 3 wins); an easy way to accomplish this is to divide the interval $(0, 1)$ up into three sections whose lengths will be determined by how many of each species (1, 2, 3) are participating in the competition and their relative growth rates. Then, wherever in the range $(0, 1)$ the drawn random number lands will determine the winner. This algorithm is implemented as follows:

```
int compete(int A, int B, int C, double RN)
{
  vector<int> counts(3,0);
  counts[A-1]++;
  counts[B-1]++;
  counts[C-1]++;

  vector<double> gammas(3,0.0);
  vector<double> factors = {1.0-s_By, 1.0, 1.0-s_S};

  double denom = 0.0;
  for (int i=0; i<3; ++i) denom += factors[i]*(double)counts[i];
  for (int i=0; i<3; ++i) gammas[i] = factors[i]*(double)counts[i]/denom;
```

```
  double sum = 0.0;
  for (int i=0; i<3; ++i) {
    sum += gammas[i];
    if (sum >= RN && gammas[i]>0.000001) return (i+1);
  }
}
```

4.3.2 Random sequential update rules

The 1- and 2-dimensional lattices were easy to implement using parallel update rules since I was able to utilize an unused spatial direction as the time-direction which allowed me to create the rules for which cells compete with one another for a spot in the next lattice. For a 3-dimensional lattice, it is no longer possible to do so, since there is no unused spatial direction into which I could project a dual lattice. So, for the 3-dimensional case I found it necessary to switch to random sequential update rules instead of parallel. I should note that random sequential update rules could have been used from the beginning, but as we will see, parallel updates are much faster and cheaper computationally.

Before I get into the actual update rules, I'll go over the structure of the lattice and how I determine which neighbors compete.

Just as I was able to construct the 2-dimensional lattice from the 1-dimensional update rules, the 3-dimensional lattice can easily be constructed from the 2-dimensional update rules with the time-direction now becoming a spatial direction. So the full lattice will consist of layers of offset, two-dimensional hexagonal lattices stacked on top of one another. Now, any site in the lattice will have a total of 12 nearest neighbors. I make use of a function I call `get_neighbors` to collect all the neighbors of a given set of indices and store the neighbors into a vector. The proper indices for nearest neighbors will depend both on whether or not the j-index and the k-index are even or odd.

```
void get_neighbors(int i, int j, int k, vector<int> &latt, vector<int> &neighbors)
{
  int lxindex = mod(i - 1, lattsize);
  int rxindex = mod(i + 1, lattsize);
  int byindex = mod(j - 1, lattsize);
  int fyindex = mod(j + 1, lattsize);
```

```
int uzindex = mod(k + 1, lattsize);
int dzindex = mod(k - 1, lattsize);

if (k%2==0) {
  if (j%2==0) {
    neighbors[0] = latt[k*L*L + j*L + lxindex];
    neighbors[1] = latt[k*L*L + j*L + rxindex];
    neighbors[2] = latt[k*L*L + fyindex*L + i];
    neighbors[3] = latt[k*L*L + fyindex*L + rxindex];
    neighbors[4] = latt[k*L*L + byindex*L + i];
    neighbors[5] = latt[k*L*L + byindex*L + rxindex];
    neighbors[6] = latt[uzindex*L*L + j*L + lxindex];
    neighbors[7] = latt[uzindex*L*L + j*L + i];
    neighbors[8] = latt[uzindex*L*L + byindex*L + i];
    neighbors[9] = latt[dzindex*L*L + byindex*L + i];
    neighbors[10] = latt[dzindex*L*L + j*L + lxindex];
    neighbors[11] = latt[dzindex*L*L + j*L + i];
  } else if (j%2 == 1) {
    neighbors[0] = latt[k*L*L + j*L + lxindex];
    neighbors[1] = latt[k*L*L + j*L + rxindex];
    neighbors[2] = latt[k*L*L + j*L + rxindex];
    neighbors[3] = latt[k*L*L + byindex*L + i];
    neighbors[4] = latt[k*L*L + fyindex*L + lxindex];
    neighbors[5] = latt[k*L*L + fyindex*L + i];
    neighbors[6] = latt[uzindex*L*L + j*L + lxindex];
    neighbors[7] = latt[uzindex*L*L + j*L + i];
    neighbors[8] = latt[uzindex*L*L + byindex*L + lxindex];
    neighbors[9] = latt[dzindex*L*L + j*L + lxindex];
    neighbors[10] = latt[dzindex*L*L + j*L + i];
    neighbors[11] = latt[dzindex*L*L + byindex*L + lxindex];
  }
} else if (k%2 == 1) {
  if (j%2 == 0) {
    neighbors[0] = latt[k*L*L + j*L + lxindex];
    neighbors[1] = latt[k*L*L + j*L + rxindex];
    neighbors[2] = latt[k*L*L + byindex*L + i];
    neighbors[3] = latt[k*L*L + byindex*L + rxindex];
    neighbors[4] = latt[k*L*L + fyindex*L + i];
    neighbors[5] = latt[k*L*L + fyindex*L + rxindex];
    neighbors[6] = latt[uzindex*L*L + j*L + i];
    neighbors[7] = latt[uzindex*L*L + j*L + rxindex];
    neighbors[8] = latt[uzindex*L*L + fyindex*L + rxindex];
    neighbors[9] = latt[dzindex*L*L + j*L + i];
    neighbors[10] = latt[dzindex*L*L + j*L + rxindex];
    neighbors[11] = latt[dzindex*L*L + fyindex*L + rxindex];
```

```
    } else if (j%2 == 1) {
      neighbors[0] = latt[k*L*L + j*L + lxindex];
      neighbors[1] = latt[k*L*L + j*L + rxindex];
      neighbors[2] = latt[k*L*L + byindex*L + lxindex];
      neighbors[3] = latt[k*L*L + byindex*L + i];
      neighbors[4] = latt[k*L*L + fyindex*L + lxindex];
      neighbors[5] = latt[k*L*L + fyindex*L + i];
      neighbors[6] = latt[uzindex*L*L + j*L + i];
      neighbors[7] = latt[uzindex*L*L + j*L + rxindex];
      neighbors[8] = latt[uzindex*L*L + fyindex*L + i];
      neighbors[9] = latt[dzindex*L*L + j*L + i];
      neighbors[10] = latt[dzindex*L*L + j*L + rxindex];
      neighbors[11] = latt[dzindex*L*L + fyindex*L + i];
    }
  }
}
```

This `get_neighbors` function effectively defines the geometry of the lattice by defining the nearest neighbors of any given site. From here, the task is to determine which sites need to be updated, to collect the nearest neighbors of that site, and implement a competition algorithm that chooses a winner out of the 12 competitors, and update the lattice and the time. Determining which site needs to be updated and updating the time are accomplished using Gillespie algorithm, which will be outlined in the next section.

The Gillespie Algorithm

The current task to evolve the 3-dimensional system is to determine at each time-step which site is to be updated, then the task is to perform the update and to determine how the time gets advanced; this is accomplished using the Gillespie algorithm. For the purpose of my particular model, the algorithm will follow the following steps:

1. Initialize the lattice and compile a list of all active sites; For the purposes of my model, a site is considered active if the identity of the cell at that site is different than the identity of a cell at any adjacent cell (additionally, if $\mu > 0$, all red cells are considered active at all times), as illustrated in Fig. 4.3.

2. Randomly pick one of the sites in the active list, using growth the rates of the cells at each site as weights.

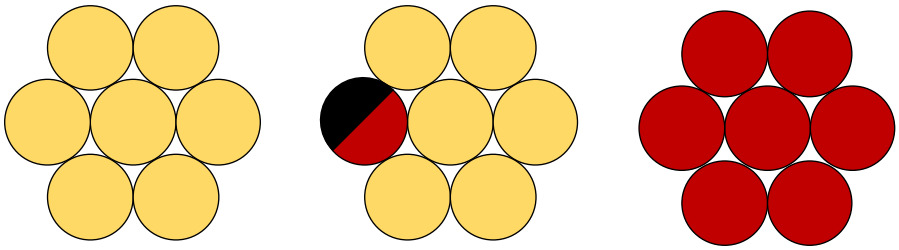

Figure 4.3: Any site on the lattice is considered active if the identity of the cell occupying that site is different from the identity of a cell at any adjacent cell. Shown in the figure is a 2-dimensional lattice, with the center site in each image the site being determined to be active or inactive; a 3-dimensional hexagonal lattice will have 12 nearest neighbors (the 6 shown in the figure on the same slice along the z-axis, as well as three cells in the z-slice above and three cells below). Additionally, all red cells are considered active if $\mu > 0$.

3. Do competition between the cells at adjacent sites of the chosen site, and replace the cell at the chosen site with the winning cell.

4. Determine if the chosen site remains active or has become inactive as a result of the update and do the same check on all adjacent sites, removing or adding these sites from the list of active sites as appropriate.

5. Calculate the time increment dt by drawing a normally distributed random number, $\mathrm{RN} \in (0, 1)$ and calculating $\mathrm{dt} = -\log(1 - \mathrm{RN})/N_A$, where N_A is the total number of sites in the list of active sites. Then, increment the time $t = t + \mathrm{dt}$.

6. Check if the time has exceeded the predetermined simulation time. If it has, terminate the simulation; if the time has not yet exceeded the predetermined simulation time, go back to Step 2 and repeat the process.

Keeping and updating the list of all active sites in the lattice can easily be the most computationally expensive part of the algorithm, particularly if the removal and addition of sites to the list results in a resizing of the list container. To get around this issue, I implement an algorithm by which I create a vector, called `active`, with size equal to the maximum possible number of active sites, which is just the total number of sites. I keep track of the number of active sites via a separate parameter, which I call `active_meta`. I then implement `add` and `delete` algorithms as follows:

```
void add(int new_element, vector<int> &active, int &active_meta)
{
  active[active_meta] = new_element;
  ++active_meta;
}

void delete(int delete_index, vector<int> &active, int &active_meta)
{
  active[delete_index] = active[active_meta - 1];
  --active_meta;
}
```

Using this algorithm, the size of the container never needs to change and adding and removing elements from my list is just a matter of copying values and keeping track of the `active_meta` parameter.

Of course, for the actual simulation the `active` list and `add` and `delete` functions will not be quite as simple as the algorithms above. With the parallel update rules, it was sufficient to keep track of the lattice via a vector with size exactly equal to the number of sites. Now, however, there is more information to keep track of at each site:

1. The identity of the cell occupying the site
2. Whether or not the site is active
3. If the site is active, where the site resides in `active`, the list of active sites.

Thus, the lattice vector will need to have a size of $N \times 3$, with $N = L \times L \times L$ for the 3-dimensional case.

Similarly, the active list will need to hold 3 values for each site: the x-, y-, and z-indices. When referring to vector indices, I will always represent x-, y-, and z-indices by the lower-case letters i, j, k respectively. Thus, the size of the active vector is also $N \times 3$. As it turns out, implementing Step 2 of my Gillespie Algorithm will be simpler if I have three separate active lists, one for each identity of cell. If I combine all three of these lists into one list with size $N \times 3 \times 3$, so that the first third of the list contains only bystander cells ($\mathtt{m} = 1$); the next third will contain only fast-growing cells ($\mathtt{m} = 2$); and the final third will contain only slow-growing cells ($\mathtt{m} = 3$). Thus, the parameter `active_meta` will now also have to be a vector. I implement `active_meta` as a vector with size 4: the first element will be the total number of active sites of any identity, the second element will be the number of active bystander cells, the second element will be the number of active fast-growing cells, and the third element will be the number of active slow-growing cells. Thus, the last element in any of the three active lists can be accessed via the formula:

$$\mathtt{active\_index} = (\mathtt{m} - 1) \times L \times L \times L \times 3 + (\mathtt{active\_meta[m]} - 1) \times 3 \qquad (1)$$

An index pointing to a site in the active list will always point to the i-index. The j and k indices can be obtained by adding 1 or 2 to the index, respectively.

Given a set of indices (i, j, k), the lattice index in my flattened vector is calculated via the formula:

$$\texttt{latt\_index} = k \times L \times L \times 3 + j \times L \times 3 + i \times 3. \tag{2}$$

Given any set of indices, and thus having obtained `latt_index`, the additional information of the site is obtained as follows:

1. The index `latt_index` $+ 1$ will point to an integer that is 0 if the site is inactive, and 1 if the site is active.

2. If the site is active, the index `latt_index` $+2$ will point to the site's position in `active`, the list of active sites.

This structure of the `active` and `lattice` vectors are illustrated pictorially in Fig. 4.4. Given the structure thus defined, the `add` and `delete` algorithms are implemented as follows:

```
void add_to_active(int i, int j, int k, int m, vector<int> &active, vector<int> &active_meta
    )
{
  //get the index pointing to the next unused position in the appropriate active list
  int aMx3 = (m - 1)*L*L*L*3 + active_meta[m]*3;

  //set the values for i,j,k in the new position
  active[aMx3] = i;
  active[aMx3 + 1] = j;
  active[aMx3 + 2] = k;

  //get the index pointing to the position of the site in the lattice vector
  int latt_index = k*L*L*3 + j*L*3 + k*3;

  //update the lattice by indicating the site is now active and where the site can be found
      in the active list
  lattice[latt_index + 1] = 1;
  lattice[latt_index + 2] = aMx3;

  //update the number of active sites
  ++active_meta[0];
  ++active_meta[m];
}
```

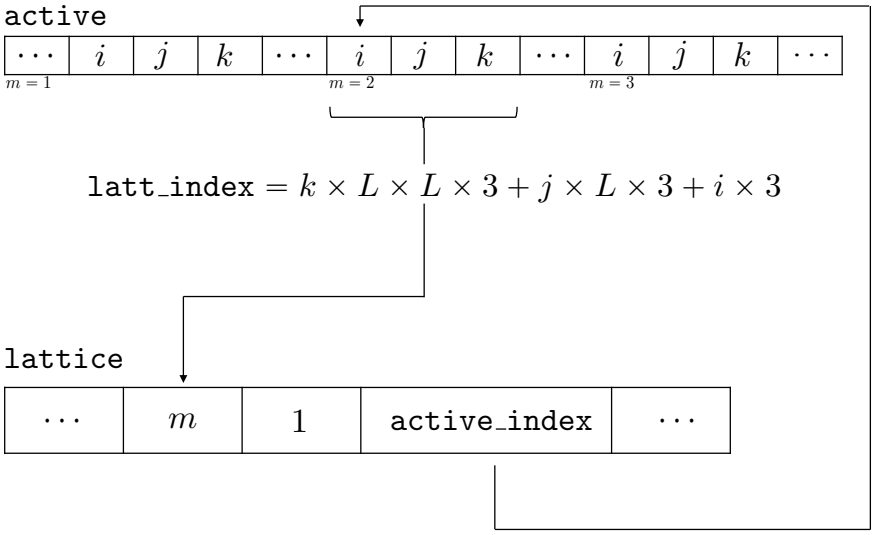

Figure 4.4: The structure of the active list and the lattice are illustrated with arrows indicating how these vectors point to one another.

```
void remove_from_active(int i, int j, int k, int m, vector<int> &active, vector<int> &
    active_meta)
{
  //get index pointing to this sites position in the lattice vector
  int latt_index_to_remove = k*L*L*3 + j*L*3 + k*3;

  //get the index pointing to the sites position in the active vector
  int active_index = lattice[latt_index_to_remove + 2];

  //get the index pointing to the last site in the appropriate active list
  int last_index = (m - 1)*L*L*L*3 + (active_meta[m] - 1)*3;

  //copy the values for the last site in the appropriate active list to the position that is
       to be deleted
  active[active_index] = active[last_index];
  active[active_index + 1] = active[last_index + 1];
  active[active_index + 2] = active[last_index + 2];

  //update the lattice to reflect the site is now inactive
  lattice[latt_index_to_remove + 1] = 0;

  //get the lattice index pointing to the site which was just moved in the active list
  int latt_index_moved = active[last_index + 2]*L*L*3 + active[last_index + 1]*L*3 + active[
      last_index]*3;

  //update the lattice with the correct index pointing to the sites new position in the
      appropriate active list
  lattice[latt_index_moved + 2] = active_index;

  //update the number of active sites
   --active_meta[0];
   --active_meta[m];
}
```

Now that I have a list of the active sites and the appropriate algorithms to keep the list updated, I can move on with the additional steps of my Gillespie Algorithm. First, however, I should initialize the lattice and the list of active sites:

```
int seed = time(0);
boost::mt19937 generator(seed);
boost::random::uniform_int_distribution<int> disint(1,2);
boost::random::uniform_real_distribution<double> dis(0,1);
```

```
int L = //size of lattice
int N = L*L*L;

vector<int> lattice(N*3);
vector<int> active(N*3*3);
vector<int> active_meta(4, 0);
vector<int> neighbors(12);

int LxL3 = L*L*3;
int Lx3 = L*3;

//initialize the lattice
for (int k = 0; k < L; ++k) {
  int tmpindexk = k*LxLx3;
  for (int j = 0; j < L; ++j) {
    int tmpindexj = tmpindexk + j*Lx3;
    for (int i = 0; i < L; ++i) {
      int tmpindex = tmpindexj + i*3;
      lattice[tmpindex] = disint(generator);
    }
  }
}

//initialize the active list
for (int k = 0; k < L; ++k) {
  int tmpindexk = k*LxLx3;
  for (int j = 0; j < L; ++j) {
    int tmpindexj = tmpindexk + j*Lx3;
    for (int i = 0; i < L; ++i) {
      int tmpindex = tmpindexj + i*3;
      int m = lattice[tmpindex];
      bool active_bool = false;
      get_neighbors(i, j, k, &lattice, &neighbors);
      //check if the current site has any neighbors occupied by cells of different
          identities
      for (int l = 0; l < 12; ++l) {
        if (m != neighbors[l]) {
          active_bool = true;
          break;
        }
      }
      //if the previous check returns true, add the site to the active list
      if (active_bool) add_to_active(i, j, k, m, active, active_meta);
    }
```

```
  }
}
```

Choosing a site from the list of active sites is done by splitting up the real interval $(0, 1)$ into three sections, each weighted by the number of active cells of each species and their respective growth rates. Then, a random number is drawn and wherever the number falls in this interval will determine the winning species. Then, another random number is drawn to choose a random site from the appropriate list. A vector I call `active_rand` is passed into the function and is populated the winning cell type and the index pointing to the position of the randomly pointed site in the active list. This algorithm is implemented as follows:

```
void choose_active(s_DP, s_BY, vector<int> &active_meta, vector<int> &active_rand, double
    rannum1, double rannum2)
{
  int winner;
  int winner_index = std::numeric_limits<int>::infinity();
  double sum = 0.0;
  vector<double> sums(3, 0.0);

  double BY_factor = (1.0 + s_BY) * (double)active_meta[1];
  double DP_factor = (1.0 + s_DP) * (double)active_meta[3];
  double denom = BY_factor + active_meta[2] + DP_factor;

  sums[0] = BY_factor / denom;
  sums[1] = (double)active_meta[2] / denom;
  sums[2] = DP_factor / denom;

  for (int i = 0; i < 3; ++i)
  {
    sum += sums[i];
    int m = i + 1;
    if (sum >= rannum1 && active_meta[m] > 0) {
      winner = m;
      break;
    }
  }

  int ran_index = round(rannum2 * (double)(active_meta[winner] - 1));
  active_rand[0] = (winner - 1)*L*L*L*3 + ran_index*3;
  active_rand[1] = winner
}
```

Note that the "growth rates" used in the `choose_active` function have the form $1+$`s_By` and $1+$`s_DP`. It would be more active to call these rates "death rates" since I am choosing a site whose cell is going to die, and the cell's neighbors will compete to see who will occupy the now empty site.

Now that an active site has been chosen, I move on to Step 3 of my Gillespie Algorithm. I collect the nearest neighbors of the chosen active site via the `get_neighbors` function, and check if the neighboring cells all have the same identity or not. If the neighboring cells all have the same identity, then no competition needs to occur and I simply replace the chosen site with a cell matching the identity of its neighbors. Otherwise, I do competition between the neighbors in the same way as I did competition between 3 cells: I split up the interval $(0, 1)$ into three regions whose size is determined by the number of cells of each identity doing competition and their relative growth rates, then I draw a random number and determine the winner based on where the random number falls in this range.

To avoid having to do the same calculation many times over, I make use of a function called `counts_to_gammas` that stores for each possible combination of cells doing competition the appropriate splitting up of the interval $(0, 1)$. First, I create a new vector called `competitors` with size 12, and I copy the values of the `neighbors` vector into the `competitors` vector. This vector then gets passed into the `counts_to_gammas` function. The results for each combination of competitors is stored into the `gammas` vector which is initiallized with all values equal to infinity. First, I check to see if the value at the appropriate index for `gammas` is equal to infinity; if this check returns true, then I do the appropriate calculation, otherwise I skip the calculation and populate a vector called `win_factors` with the appropriate stored values. The `win_factors` vector is then passed into another function, called `win_strain` which determines the winner based on the same trick of splitting up the interval $(0, 1)$ and determining the winner based on where a random number falls in the interval.

```
void counts_to_gamma(s_BY, s_DP, vector<int> &competitors, vector<int> &gammas, vector<int>
    &win_factors)
{
  int gindex;
```

```
  vector<int> counts(3, 0);
  for (int i = 0; i < 12; ++i) {
    ++counts[competitors[i]-1];
  }
  gindex = counts[0] * 432 + counts[1] * 36 + counts[2] * 3;

  if (this->gammas[gindex] == std::numeric_limits<double>::infinity()) {
    double denom = (double)counts[0] * (1.0 - s_BY) + (double)counts[1] * 1.0 + (double)
        counts[2] * (1.0 - s_DP);
    gammas[gindex] = (double)counts[0] * (1.0 - s_BY) / denom;
    gammas[gindex + 1] = (double)counts[1] / denom;
    gammas[gindex + 2] = (double)counts[2] * (1.0 - s_DP) / denom;
  }
  win_factors[0] = gammas[gindex];
  win_factors[1] = gammas[gindex + 1];
  win_factors[2] = gammas[gindex + 2];
}

int win_strain(vector<int> &win_factors, double rannum)
{
  double sum = 0.0;
  int winner;

  for (int s = 0; s < 3; s++ ) {
    sum += win_factors[s];
    if (sum >= rannum && win_factors[s] > 0.000001) {
      return winner;
    }
  }
}
```

Once the winner of competition is determined, I check if the winning cell belongs to the fast growing strain. If so, and if $\mu > 0$, I draw an additional random number to determine if that cell mutates into the slow-growing strain. Finally, I update the lattice, check the current site and all neighboring sites to see if it is still active or if it has become inactive during the update, and add or remove sites from the active list as necessary. Finally, I draw an additional random number and calculate the time advancement using the total number of active cells: `active_meta[0]`. I then advance the time and check if the time has exceeded the predetermined simulation time. If so, I terminate the simulation; otherwise, I go back to Step 2 and pick a new site to update.

When the simulation is terminated, I scan through the lattice and count the number of cells belonging to each identity. I then calculate the relative densities of each cell type and print these to an output file. For the construction of phase diagrams, I do this in a loop, changing the values for `s_BY`, and μ in each iteration.

I have outlined here only the most important machinery in my implimentation of the Gillespie Algorithm. The full version of my code with the entire update algorithm can be found on github at the following link:

So far I have only covered one type of initial condition, where the lattice is initialized with sites randomly populated by cells of either the bystander strain and the fast-growing strain. In the next and final section I'll say a few words about other possible initial conditions.

4.4 Initial conditions

Aside from the mixed initial state I've already discussed, I make use of two other kinds of initial conditions:

1. All sites are populated with the bystander strain except for a small central cluster of fast-growing cells

2. All sites are populated with the bystander strain except for a layer of fast-growing cells.

These initial conditions are illustrated for 2-dimensions in Fig. 4.5, the 1- and 3- dimensional counterparts are easily visualized as generalizations of the initial conditions in the figure.

The cluster initial condition is useful for constructing phase diagrams where instead of recording the relative densities of each cell type after the simulation terminates, I calculate the probability that the cluster survives after the predetermined simulation time. This kind of simulation requires multiple runs to get a good probability phase diagram. These initial conditions are also useful for creating figures from the final state after the simulation has terminated. These figures can give a qualitative insight into what is happening at the edge of the "cancerous" population (fast-growing + slow-growing cells); since the primary interest of these kinds of simulations is to look at roughening phenomenon, these kinds of simulations

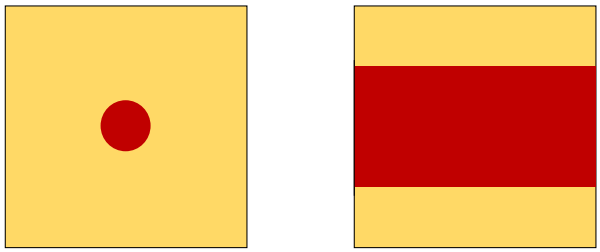

Figure 4.5: For 2-dimensional simulations, two types of initial conditions that differ from the mixed case are illustrated.

and the resulting figures can be quite valuable. I've included some examples of such figures for the 3-dimensional case in Fig. 4.6.

It would be interesting to extract quantitative information on the interfacial roughening for the cluster initial conditions. Whereas I have not done this for this thesis, it would be interesting to look at this kind of quantitative analysis in future work.

The second type of initial conditions are what I have called *domain wall* initial conditions, since the primary object of interest is the behavior of the initially flat domain walls between the bystander strain and the "cancerous" population (fast-growing and slow-groing strains). As discussed in 3, this type of initial condition was used to extract quantitative analysis of the roughening phenomena. This kind of quantitative analysis of the roughening phenomena has not yet been done for the 3-dimensional case; this would also make a very interesting future project.

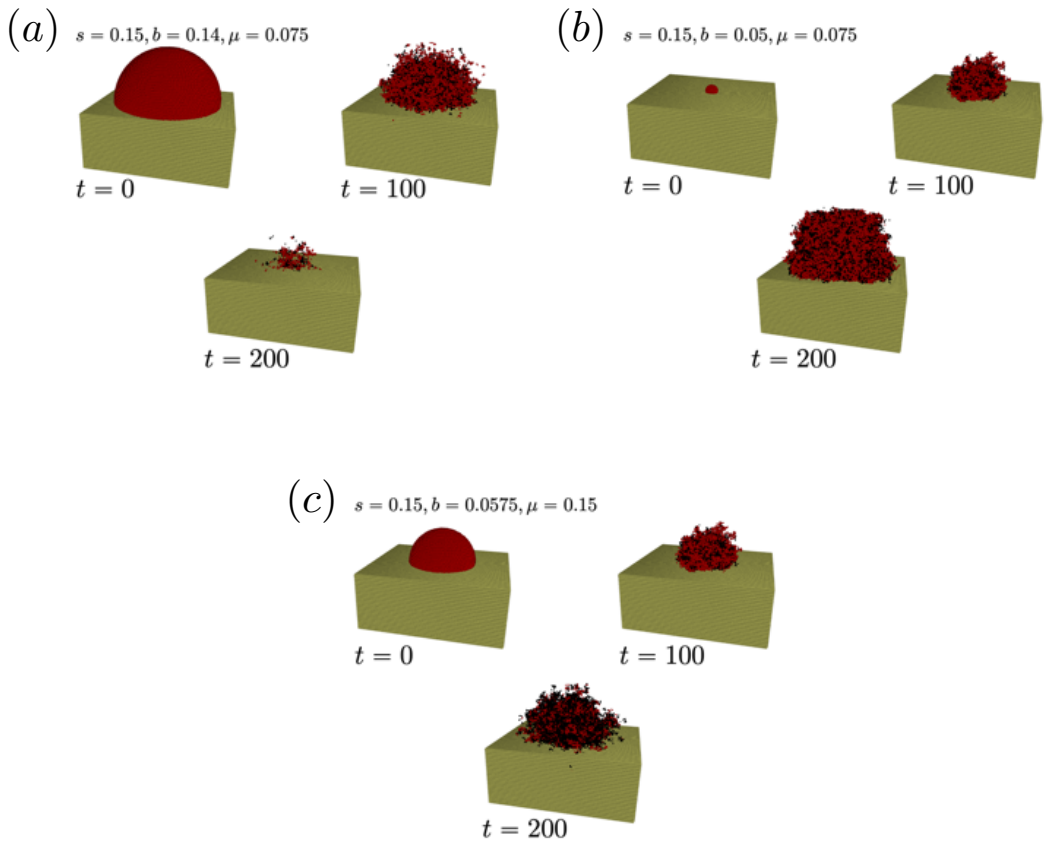

Figure 4.6: Pictures constructed from simulations with total simulation time of either $t = 100$ or $t = 200$. **(a)** In the inactive phase, the red/black population always dies out. **(b)** in the active phase, the red/black population always successfully invades the bystander population. **(c)** Near the phase transition, the edges of the invading population will begin to roughen.

Chapter 5

Conclusions

In this thesis I have studied and explored roughening invasion fronts in population dynamics. In Chapter 1, I introduced the main concepts of invasion fronts and roughening phenomena. Specifically I introduced the stochastic Fisher equation and dicussed moving invasion fronts as characterized by Fisher waves. In Chapter 2, I introduced my bystander model for cancer growth and presented the results of my investigation into invasion fronts between an unstable invading population and a neutral bystander. Namely, I showed that when an unstable, mutating population invades a healthy bystander, the internal dynamics of the mutating population has a qualitative effect on the shape of the invasion front. That is, the invasion front will develop enhanced roughness as the mutating population nears mutational meltdown. I quantified this by defining a width function on the interface between bystander and unstable populations and extracted a power-law in the time-evolution of the width and roughness of the interface was characterized by the exponent associated with the power-law behavior of the width as a function of time. This analysis was accomplished for interfaces in $1+1$-dimensional and $2+1$-dimensional systems. Though the $2+1$-dimensional system had much richer dynamics, namely the potential of cross-over between two limiting regimes, both the $1+1$-dimensional and $2+1$-dimensional systems gave rise to enhanced roughening behavior as the unstable population approached mutational meltdown, as evidenced in the crossover in power-law behavior of the width function.

Because of the more complicated details of invasion fronts between an unstable invader and a neutral bystander, the evolution of these fronts represents a departure from the

Fisher equation introduced in Chapter 1. For the three-species model, constructing an analogous equation to the Fisher equation would require taking into account not only the interaction between the two separate populations, but also the internal dynamics of the unstable population. Such equations have already been developed for an mutationally unstable population in isolation[15], but not yet for the case with the addition of a neutral bystander.

In Chapter 3 I took a brief look at some of the literature regarding cancer growth and modelling of cancerous tumors to consider if my model makes sense as one describing tumor growth. In particular, I looked at an experiment on brain tumor cultures in which power-law behavior for the roughness of the tumors were extracted and I compared these results to those of my model.

Finally, in Chapter 4 I outlined some of the most important machinery for my implementation of my model in C++ with particular attention paid to my attempts at modelling a $3+1$-dimensional system, and outlined some of the challenges there, as well as suggestions for future research projects.

Bibliography

[1] Andrea Sottoriva, Haeyoun Kang, Zhicheng Ma, Trevor A. Graham, Matthew Salomon, Junsong Zhao, Paul Marjoram, Kimberly Siegmund, Michael Press, Darryl Shibata, and Christina Curtis. A Big Bang model of human colorectal tumor growth. *Nat. Genet.*, 47:209–216, 2015. 2, 28

[2] O. Hallatschek and D. R. Nelson. Life at the front of an expanding population. *Evolution*, 64:193–206, 2010. 3, 29, 42

[3] K. S. Korolev, M. Avlund, O. Hallatschek, and D. R. Nelson. Genetic demixing and evolution in linear stepping stone models. *Rev. Mod. Phys.*, 82:1691–1718, 2010. 7

[4] M. O. Lavrentovich, K. S. Korolev, and D. R. Nelson. Radial Domany-Kinzel models with mutation and selection. *Phys. Rev. E*, 87(012103):012103, 2013. 10, 20, 32, 36, 42, 49

[5] O. Hallatschek, P. Hersen, S. Ramanathan, and D. R. Nelson. Genetic drift at expanding frontiers promotes gene segregation. *Proc. Natl. Acad. Sci. U.S.A.*, 104(50):19926–80, 2007. 14, 29, 49

[6] R. A. Fisher. The wave of advance of advantageous genes. *Ann. Eugen.*, 7:355–369, 1937. 15, 46, 62

[7] U. Ebert and W. van Saarloos. Front propagation into unstable states: universal algebraic convergence towards uniformly translating pulled fronts. *Physica D*, 146:1–99, 2000. 18

[8] Eric Brunet and Bernard Derrida. Shift in the velocity of a front due to a cutoff. *Phys. Rev. E*, 56:2597–2604, Sep 1997. 19

[9] Leonid Pechenik and Herbert Levine. Interfacial velocity corrections due to multiplicative noise. *Phys. Rev. E*, 59:3893–3900, Apr 1999. 19

[10] E. Moro. Internal fluctuations effects on Fisher waves. *Phys. Rev. Lett.*, 87(23):238303, 2001. 19, 46, 62

[11] J. Garcia-Ojalvo and J. Sancho. Noise in spatially extended systems. 1999. 20

[12] John L. Cardy and Uwe C. Täuber. Field theory of branching and annihilating random walks. *Journal of Statistical Physics*, 90(1):1–56, 1998. 20

[13] M. Kardar, G. Parisi, and Y.-C. Zhang. Dynamic scaling of growing interfaces. *Phys. Rev. Lett.*, 56(9):889–892, 1986. 24, 46

[14] J. T. Kuhr, M. Leisner, and E. Frey. Range expansion with mutation and selection: dynamical phase transition in a two-species Eden model. *New J. Phys.*, 13:113013, 2011. 24, 30, 31

[15] M. O. Lavrentovich. Critical fitness collapse in three-dimensional spatial population genetics. *J. Stat. Mech. Theory Exp.*, 2015(5):P05027, 2015. 26, 36, 38, 41, 42, 43, 67, 104

[16] K. Petren and T. J. Case. An experimental demonstration of exploitation competition in an ongoing invasion. *Ecology*, 77(1):118–132, 1996. 27

[17] J. B. Xavier and K. R. Foster. Cooperation and conflict in microbial biofilms. *Proc. Natl. Acad. Sci. U.S.A*, 104(3):876–881, 2007. 27

[18] C. D. Nadell, K. Drescher, and K. R. Foster. Spatial structure, cooperation and competition in biofilms. *Nat. Rev. Microbiol.*, 14:589–600, 2016. 27

[19] J. Bahl, D. vijaykrishna, E. C. Holmes, G. J. D. Smith, and Y. Guan. Gene flow and competitive exclusion of avian influenza A virus in natural reservoir hosts. *Virology*, 390:289–297, 2009. 27

[20] M. M. Merino, R. Levayer, and E. Moreno. Survival of the fittest: Essential roles of cell competition in development, aging, and cancer. *Trends Cell Biol.*, 26:776–788, 2016. 27

[21] Y. Willi, M. Fracassetti, S. Zoller, and J. Van Buskirk. Accumulation of mutational load at the edges of a species range. *Mol. Biol. Evol.*, 35:781–791, 2018. 27

[22] L. Bosshard, I. Dupanloup, O. Tenaillon, R. Bruggmann, M. Ackermann, S. Peischl, and L. Excoffier. Accumulation of deleterious mutations during bacterial range expansions. *Genetics*, 207:669–684, 2017. 27, 29

[23] Félix Foutel-Rodier and Alison Etheridge. The spatial muller's ratchet: surfing of deleterious mutations during range expansion. *bioRxiv*, 2019. 28

[24] D. P. Cahill, K. W. Kinzler, B. Vogelstein, and C. Lengauer. Genetic instability and darwinian selection in tumours. *Trends Cell Biol.*, 9:M57–M60, 1999. 28

[25] S. Negrini, V. G. Gorgoulis, and T. D. Halazonetis. Genomic instability - an evolving hallmark of cancer. *Nat. Rev. Mol. Cell Biol.*, 11:220–228, 2010. 28

[26] A. Marusyk and K. Polyak. Tumor heterogeneity: Causes and consequences. *Biochim. Biophys. Acta Rev. Cancer*, 1805:105–117, 2010. 28

[27] L. G. Martelotto, C. K. Y. Ng, S. Piscuoglio, B. Weigelt, and J. S. Reis-Filho. Breast cancer intra-tumor heterogeneity. *Breast Cancer Res.*, 16:R48, 2014. 28

[28] J. Yoo, S. Chong, C. Lim, M. Heo, and I. G. Hwang. Assessment of spatial tumor heterogeneity using CT growth patterns estimated by tumor tracking on 3D CT volumetry of multiple pulmonary metastatic nodules. *PLoS ONE*, 14:e0220550, 2019. 28

[29] C. Gidoin and S. Peischl. Range expansion theories could shed light on the spatial structure of intra-tumour heterogeneity. *Bull. Math. Biol.*, 81:4761–4777, 2019. 28, 29

[30] Kirill Korolev, Joao Xavier, and Jeff Gore. Turning ecology and evolution against cancer. *Nat. Rev. Cancer*, 14:371, 2014. 28

[31] C. D. McFarland, K. S. Korolev, G. V. Kryukov, S. R. Sunyaev, and L. A. Mirny. Impact of deleterious passenger mutations on cancer progression. *Proc. Natl. Acad. Sci. U.S.A.*, 110(8):2910–2915, 2013. 28, 67, 68, 69

[32] Christopher D. McFarland, Julia A. Yaglom, Jonathan Wojtkowiak, Jacob Scott, David Morse, Michael Sherman, and Leonid Mirny. The damaging effect of passenger mutations on cancer progression. *Cancer Res.*, 77(18):4763–4772, 2017. 28, 67

[33] Christopher D. McFarland, Leonid A. Mirny, and Kirill S. Korolev. Tug-of-war between driver and passenger mutations in cancer and other adaptive processes. *Proc. Natl. Acad. Sci. U.S.A.*, 111(42):15138–15143, 2014. 28

[34] H. J. Muller. The relation of recombination to mutational advance. *Mutat. Res.*, 1:2–9, 1964. 28

[35] W. Gabriel, M. Lynch, and R. Bürger. Muller's ratchet and mutational meltdowns. *Evolution*, 47:1744–1757, 1993. 28

[36] F. L. Muller, S. Colla, E. Aquilanti, V. E. Manzo, G. Genovese, J. Lee, D. Eisenson, R. Narurkar, P. Deng, L. Nezi, M. A. Lee, B. Hu, J. Hu, E. Sahin, D. Ong, E. Fletcher-Sananikone, D. Ho, L. Kwong, C. Brennan, Y. A. Wang, L. Chin, and R. A. DePinho. Passenger deletions generate therapeutic vulnerabilities in cancer. *Nature*, 488:337–342, 2012. 28

[37] Y. Zhang, Y. Li, T. Li, X. Shen, T. Zhu, Y. Tao, X. Li, D. Wang, Q. Ma, Z. Hu, J. Liu, J. Ruan, J. Cai, H.-Y. Wang, and X. Lu. Genetic load and potential mutational meltdown in cancer cell populations. *Mol. Biol. Evol.*, 36:541–552, 2019. 28

[38] R. A. Gatenby, O. Grove, and R. J. Gillies. Quantitative imaging in cancer evolution and ecology. *Radiology*, 269:8–15, 2013. 28

[39] R.-F. Chang, W.-J. Wu, W. K. Moon, and D.-R. Chen. Automatic ultrasound segmentation and morphology based diagnosis of solid breast tumors. *Breast Cancer Res. Treat.*, 89:179–185, 2005. 28

[40] Erik A. Martens, Rumen Kostadinov, Carlo C. Maley, and Oskar Hallatschek. Spatial structure increases the waiting time for cancer. *New J. Phys.*, 13:115014, 2011. 29

[41] M. O. Lavrentovich, M. E. Wahl, D. R. Nelson, and A. W. Murray. Spatially constrained growth enhances conversional meltdown. *Biophys. J.*, 110:2800–2808, 2016. 29, 36, 38

[42] D. Fusco, M. Gralka, J. Kayser, A. Anderson, and O. Hallatschek. Excess of mutational jackpot events in expanding populations revealed by spatial Luria–delbrück experiments. *Nat. Commun.*, 7:12760, 2016. 29

[43] H. Hinrichsen. Non-equilibrium critical phenomena and phase transitions into absorbing states. *Adv. Phys.*, 49(7):815–958, 2000. 29, 36, 40, 43, 44, 50

[44] A. Brú, S. Albertos, J. L. Subiza, J. L. Garcá-Asenjo, and I. Brú. The universal dynamics of tumor growth. *Biophys. J.*, 85:2948–2961, 2003. 29

[45] B. Drossel and M. Kardar. Phase ordering and roughening on growing films. *Phys. Rev. Lett.*, 85(3):614–617, 2000. 29

[46] B. Drossel and M. Kardar. Model for growth of binary alloys with fast surface equilibration. *Phys. Rev. E*, 55(5):5026, 1997. 29

[47] J. M. Horowitz and M. Kardar. Bacterial range expansions on a growing front: Roughness, fixation, and directed percolation. *Phys. Rev. E*, 99:042134, 2019. 29

[48] T. Halpin-Healy and Y.-C. Zhang. Kinetic roughening phenomena, stochastic growth, directed polymers and all that. aspects of multidisciplinary statistical mechanics. *Phys. Rep.*, 254:215–414, 1995. 29, 62

[49] K. Chkhaidze, T. Heide, B. Werner, M. J. Williams, W. Huang, G. Caravagna, T. A. Graham, and A. Sottoriva. Spatially constrained tumour growth affects the patterns of clonal selection and neutral drift in cancer genomic data. *PLOS Comput. Biol.*, 15:e1007243, 2019. 30

[50] B. Nguyen, A. Upadhyaya, A. van Oudenaarden, and M. P. Brenner. Elastic instability in growing yeast colonies. *Biphys. J.*, 86:2740–2747, 2004. 30

[51] T. Shimaya and K. A. Takeuchi. Lane formation and critical coarsening in a model of bacterial competition. *Phys. Rev. E*, 99:042403, 2019. 31

[52] M. O. Lavrentovich and D. R. Nelson. Asymmetric mutualism in two- and three-dimensional range expansions. *Phys. Rev. Lett.*, 112:138102, 2014. 31

[53] K. J. Gilbert, S. Peischl, and L. Excoffier. Mutation load dynamics during environmentally-driven range shifts. *PLoS Genet.*, 14:e1007450, 2018. 31

[54] Eytan Domany and Wolfgang Kinzel. Equivalence of cellular automata to Ising models and directed percolation. *Phys. Rev. Lett.*, 53:311–314, 1984. 32, 36

[55] A. Giese, R. Bjerkvig, M. E. Berens, and M. Westphal. Cost of migration: invasion of malignant gliomas and implications for treatment. *J. Clin. Oncol.*, 21:1624–1636, 2003. 36

[56] U. C. Täuber, M. J. Howard, and H. Hinrichsen. Multicritical behavior in coupled directed percolation processes. *Phys. Rev. Lett.*, 80(10):2165–2168, 1998. 38

[57] J. Otwinowski and J. Krug. Clonal interference and Muller's ratchet in spatial habitats. *Phys. Biol.*, 11(5):056003, 2014. 42

[58] A. Kolmogorov, N. Petrovsky, and N. Piscounov. Étude de l'équation de la diffusion avec croissance de la quantité de matière et son application a un problème biologique. *Moscow Univ. Math. Bull.*, 1:1, 1937. 46, 62

[59] O. Hallatschek and K. S. Korolev. Fisher waves in the strong noise limit. *Phys. Rev. Lett.*, 103:108103, 2009. 46

[60] C. R. Doering, M. A. Burschka, and W. Horsthemke. Fluctuations and correlations in a diffusion-reaction system: exact hydrodynamics. *J. Stat. Phys.*, 65:953–970, 1991. 46

[61] G. Tripathy and W. van Saarloos. Fluctuation and relaxation properties of pulled fronts: a scenario for nonstandard Kardar-Parisi-Zhang scaling. *Phys. Rev. Lett.*, 85(17):3556–3559, 2000. 46

[62] J. Riordan, C. R. Doering, and D. ben Avraham. Fluctuations and stability of Fisher waves. *Phys. Rev. Lett.*, 75(3):565–568, 1995. 46, 61, 62, 63, 65

[63] J. T. Cox and D. Griffeath. Diffusive clustering in the two dimensional voter model. *Ann. Prob.*, 14(2):347–370, 1986. 46

[64] A. N. Stokes. On two types of moving front in quasilinear diffusion. *Math. Biosci.*, 31:307–315, 1976. 46

[65] W. van Saarloos. Front propagation into unstable states. *Phys. Rep.*, 386:29–222, 2003. 46

[66] V. Tejedor, O. Bénichou, R. Voituriez, R. Jungmann, F. Simmel, C. Selhuber-Unkel, L. B. Oddershede, and R. Metzler. Quantitative analysis of single particle trajectories: mean maximal excursion method. *Biophys. J.*, 98:1364–1372, 2010. 48, 49

[67] B. T. Weinstein, M. O. Lavrentovich, W. Möbius, A. W. Murray, and D. R. Nelson. Genetic drift and selection in many-allele range expansions. *PLOS Comput. Biol.*, 13(12):e1005866, 2017. 49

[68] K. S. Korolev, J. B. Xavier, D. R. Nelson, and K. R. Foster. A quantitative test of population genetics using spatiogenetic patterns in bacterial colonies. *Am. Nat.*, 178(4):538–552, 2011. 49

[69] I. Dornic, H. Chaté, J. Chave, and H. Hinrichsen. Critical coarsening without surface tension: the universality class of the voter model. *Phys. Rev. Lett.*, 87(4):045701, 2001. 60

[70] C. M. Bordogna and E. V. Albano. Study and characterization of interfaces in a two-dimensional generalized voter model. *Phys. Rev. E*, 83:046111, 2011. 60

[71] B. Waclaw, I. Bozic, M. E. Pittman, R. H. Hruban, B. Vogelstein, and M. A. Nowak. A spatial model predicts that dispersal and cell turnover limit intratumour heterogeneity. *Nature*, 525:261–264, 2015. 64

[72] F. Davnall, C. S. P. Yip, G. Ljungqvist, M. Selmi, F. Ng, B. Sanghera, B. Ganeshan, K. A. Miles, G. J. Cook, and V. Goh. Assessment of tumor heterogeneity: an emerging imaging tool for clincial practice? *Insights Imaging*, 3:573–589, 2012. 65

[73] L. G. T. Morris, N. Riaz, A. Desrichard, Y. Şenbabaoğlu, A. A. Hakimi, V. Makarov, J. S. Reis-Filho, and T. A. Chan. Pan-cancer analysis of intratumor heterogeneity as a prognostic determinant of survival. *Oncotarget*, 7:10051–10063, 2016. 65

[74] L.-B. Weiswald, D. Bellet, and V. Dangles-Marie. Spherical cancer models in tumor biology. *Neoplasia*, 17:1–15, 2015. 65

[75] C. Vulin, J.-M. Di Meglio, A. B. Lindner, A. Daerr, A. Murray, and P. Hersen. Growing yeast into cylindrical colonies. *Biophys. J.*, 106:2214–2221, 2014. 65

[76] A. Di Gregorio, S. Bowling, and T. A. Rodriguez. Cell competition and its role in the regulation of cell fitness from development to cancer. *J. Developmental Cell*, 38:621–634, Sep 2016. 69, 70

[77] D. Alemani, F. Pappalardo, M. Pennisi, S. Motta, and V. Brusic. Combining cellular automata and lattice boltzmann method to model multiscale avascular tumor growth coupled with nutrient diffusion and immune competition. *Journal of Immunological Methods*, 376(1-2):55–68, February 2012. 71

[78] S. Dormann and A. Deutsch. Modeling of self-organized avascular tumor growth with a hybrid cellular automaton. *In Silico Biol.*, 2:393–406, 2002. 71

[79] J. Freyer. Role of necrosis in regulating the growth saturation of multicellular spheroids. *Cancer Research*, 48:2432–2439, 1988. 71

[80] C. R. Thoma, M. Zimmermann, I. Agarkova, J. M. Kelm, and W. Krek. 3D cell culture systems modeling tumor growth determinants in cancer target discovery. *Adv. Drug Deliv. Rev.*, 69-70:29–41, 2014. 72

[81] A. Brú, J. M. Pastor, I. Fernaud, I. Brú, S. Melle, and C. Berenguer. Super-rough dynamics on tumor growth. *Phys. Rev. Lett.*, 81:4008–4011, 1998. 72, 73

[82] A. Brú, S. Albertos, J. L. Subiza, J. L. García-Asenjo, and I. Brú. The universal dynamics of tumor growth. *Biophys. J.*, 85:2948–2961, 2003. 72

[83] M. A. C. Huergo, M. A. Pasquale, P. H. González, A. E. Bolzán, and A. J. Arvia. Growth dynamics of cancer cell colonies and their comparison with noncancerous cells. *Phys. Rev. E*, 85:011918, 2012. 72, 73, 74

[84] T. Halpin-Healy. $(2 + 1)$-dimensional directed polymer in a random medium: scaling phenomena and universal distributions. *Phys. Rev. Lett.*, 109:170602, 2012.

[85] B. G. Barreales, J. J. Meléndez, R. Cuerno, and J. J. Ruiz-Lorenzo. Kardar-Parisi-Zhang universality class for the critical dynamics of reaction-diffusion fronts. *J. Stat. Mech. Theory Exp.*, 2020:023203, 2020.

[86] David A. Kessler, Zvi Ner, and Leonard M. Sander. Front propagation: Precursors, cutoffs, and structural stability. *Phys. Rev. E*, 58:107–114, Jul 1998.

Vita

Daniel Castillo is originally from Los Angeles, California. After graduating high school, Daniel moved to Tuscaloosa, Alabama where he attended The University of Alabama and studied Chemistry and Physics. After graduating with his Bachelor of Chemistry and Bachelor of Science in Physics degrees, Daniel moved to Knoxville, Tennessee to begin his career as a graduate student at The University of Tennessee for his Doctorate of Pholosophy in Physics. For his first few years, Daniel studied quantum information and computation under George Siopsis until he ultimately joined the research group of Maxim Lavrentovich, studying spatial phenomena in population dynamics. Daniel is grateful for his family who have given him so much support and love through the long and arduous process of obtaining his PhD.