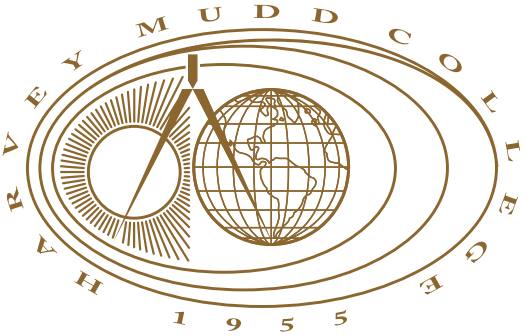
2004

Modeling Cancer Cell Response to Immunotherapy

Eric Harley
Harvey Mudd College



Mathematically Modeling Spatial Tumor Growth in Parallel

Eric Harley

Lisette de Pillis, Advisor

Jon Jacobsen, Reader

May, 2004

HARVEY MUDD
COLLEGE
Department of Mathematics

Abstract

Significant work has been done modeling cancerous tumor growth and response to therapy under certain simplifying assumptions, specifically, the assumption of spatial homogeneity. We have chosen a spatially heterogenous model for cancer cell growth using a hybrid Lattice-Gas Cellular Automata method. Cell mitosis, apoptosis, and necrosis are explicitly modeled along with the diffusion of nutrients and a necrotic signal. The model implementation is verified qualitatively and is modified to execute on a parallel computer.

Contents

List of Figures

Acknowledgments

I would like to thank the Math and Computer Science departments at Harvey Mudd College for allowing me to use their computing resources, without which this work would not be possible.

But I reserve my deepest thanks for my advisor, Professor Lisette de Pillis, without whose academic and personal guidance this work would not be possible.

Chapter 1

Introduction

A tumor is an abnormal mass of cells which possess no physiologic function, that is not inflammatory, and arises without obvious cause from cells of preexistent tissue. Tumors are categorized as benign or malignant according to their effect on the body.

If current trends continue, one out of three Americans will eventually develop a cancerous tumor. The Center for Disease Control reports that in 1995, an estimated 1,252,000 cases were diagnosed worldwide, with 547,000 deaths in the United States alone (U.S. Department of Health and Human Services (1995)). With new techniques for detection and treatment of cancer, the relative survival rate has now risen to 54 percent. Because the disease is so widespread it is vital to advance the state of knowledge in this field as rapidly as possible to understand what causes the appearance of tumors, how tumors grow, and how tumors respond to changes in the body either through the introduction of drugs or through natural bodily mechanisms. In this thesis we consider the second and third research objectives.

1.1 Treatment

Typically, physicians experiment in treating cancer with one of three modes of therapy: radiation therapy, chemotherapy, and immunotherapy. The first, and most widely used, involves applying radiation to the tumor cells which destroys them. Radiation therapy involves the use of highly sophisticated machines to administer treatments. Additionally, radiation therapy cannot treat all types of tumors. Tumors deep within the body cannot be treated because surrounding vital tissues may be destroyed. Drug based therapies like chemotherapy and immunotherapy, are used in those cases.

1.1.1 Chemotherapy

A more modest mode of treatment, and the most common after radiation therapy is through the administration of drugs into the body, chemotherapy. These drugs often are called *anticancer* drugs.

Normal cells grow and die in a controlled way. When cancer occurs, cells in the body that are not normal keep dividing and forming more cells without control. Anticancer drugs destroy cancer cells by stopping them from growing or multiplying. Healthy cells can also be harmed, especially those that divide quickly, such as hair, the stomach lining, and the immune system. Harm to healthy cells is what causes side effects such as hair loss and nausea. However, these cells usually repair themselves after chemotherapy (U.S. Department of Health and Human Services (2003)). Because some drugs work better together than alone, two or more drugs are often given at the same time. This is called combination chemotherapy.

1.1.2 Immunotherapy

The immune system is able to recognize tumor cells and destroy them. There are two main types of cells that respond to a tumor's presence. Nonspecific immune cells, such as NK cells, travel throughout the body's tissue and attack all foreign substances they find. Specific immune cells, such as CD8+ T-cells, attack only after a complex variety of mechanisms primes them to recognize similar threats (Chang et al. (2003)). The more white blood cells in the immune system, the more able an individual is able to fight infection. Although there are other factors affecting the strength of the immune system, the most common measure of health is the number of white blood cells, or circulating lymphocytes in the blood stream (Chang et al. (2003)).

Immunotherapy is similar to chemotherapy in that the patient is injected with a substance. However, the goal of immunotherapy is to boost the body's natural cancer fighting abilities. These substances can be made in the laboratory and given to patients to destroy cancer cells or change the way the body reacts to a tumor. They may also help the body repair or make new cells destroyed by chemotherapy. For example, interleukin 2 (IL-2) is a growth factor which causes CD8+ T cell proliferation (Chang et al. (2003)).

In both immunotherapy and chemotherapy, the tumor and the body can build resistance to the drugs being administered, diminishing their effect in time. This is another reason why multiple drugs are used in chemotherapy.

1.2 Problem Formulation

Planning the administration of drug based therapy into a treatment protocol is a complicated process. Physicians working with chemotherapy and immunotherapy have to decide which drugs to administer, when to administer them, and in what amounts. Often physicians employ heuristics in planning. The "Hit it Hard" method, where a large but non-lethal dose is given at the beginning of treatment with later doses dropping in amount, is popular among some. However the "slow and steady", a constant moderate dose, approach is popular among others. However, these are simply heuristics derived from experimenting with patients in clinical trials.

Clinical trials are expensive, costing about $12,000 to $15,000 for the treatment of a single patient. Developing new methods of treatment is expensive. Because of this novel ideas for treatment cannot easily be tested. There is a high financial cost, but there is also the cost of human lives. Because of this only the most promising new methods of treatment can be tested.

1.3 Problem Statement

One of the major stumbling blocks in testing drug therapies is the availability of a robust model of tumor growth. The problem we consider in this work is, how can we create an effective numerical simulation of tumor growth which incorporates the spatial heterogeneity of the tumor, and can accomodate the addition of chemotherapy and immunotherapy. In building this framework, we seek to create a model which can be simulated on a parallel computer, thus reducing the amount of time necessary to compute our results, or alternatively, increasing the size of the tumors that we can consider.

We now consider previous approaches to creating models of tumor growth.

Chapter 2

The Model

2.1 Assumptions and Constraints

2.1.1 Assumptions

In order to model cancer cell growth realistically, we must make a number of assumptions. These include the following:

- **Cells in the poopulation are either cancerous or necrotic.** There are many different populations of cells within the body: normal tissue cells, blood vessel cells, immune system cells, etc. Further, the state of a cell is so easily differentiated. But because we have concerned ourselves with modeling the growth of cancerous cells, we consider their growth in isolation.
- **Cells move about a regular 2D square lattice.** In reality, cells are not restricted to a finite set of positions. However, for efficieny in the simulation of our model we assume that a cell's motions are restricted. Additionaly configurations and dimensions could be accomodated, but for now a 2D lattice must do.
- **The size of each cancer cell is uniform across the tumor population and fixed over time.** In reality, cells come in different shapes and sizes. Further, different cell types have sizes on different scales.
- **Necrotic cells occupy less volume than cancerous cells.** A cell becomes necrotic by dying and bursting its contents into the tumor. The effect of this is that only one third of the cell is left over.

- **Cells interact locally: the movement and state of a cell can only be a function of cells adjacent on the lattice.** Because there does not appear to be a background signally network between cells which are separated by large distances in the body, we assume that the only interactions that affect the movement and state of a cell are local interactions among adjacent cells on the lattice.
- **The local concentration of cells in a specific area affect growth and movement.** We must explicitly model the effect that adjacent cells have upon each other at the local level.
- **Cells are memory less: the probability a cell expires is not dependent on the lifetime of the cell.** In reality, as a cell ages its internal machinery deteriorates and becomes unable to function properly. We approximate this by simply assuming each cell has an equal baseline probability of dying p_d which is augmented by the presence of nutrient, other cells, and necrotic signal.
- **Cell growth is dependent only on the number of cells in a local neighborhood, the concentration of nutrient, and the presence of necrotic signal.**
- **Tumor mass is avascular.** Blood vessels are not part of the tumor. We are simulating a *in vitro* experiment and vascularization has not been observed *in vitro*. Adding a blood vessel network is addressed in 4.
- **Nutrient distribution is applied uniformly outside the tumor population.** After each step in the tumor simulation we re-fresh the concentration of nutrient outside the tumor population while leaving untouched the concentration within the tumor. This corresponds to the *in vitro* growth protocol of pouring nutrient solution over the tumor.
- **There are no other forces affecting cell growth or death.** While there are a number of other factors which effect cell growth and death, we only concern ourselves with those discussed above.

2.1.2 Constraints

- **The model must encompass mitosis, apoptosis, and necrosis.** These basic cellular functions (division, death, and unnatural death) must be explicitly addressed by the model. Necrosis is when a cell swells and bursts forming a necrotic site. The presence of a necrotic site inhibits neighboring cells from entering mitosis (**?**).

- **Cells must respond to the presence of other cells.** Cells divide, move, and die because of the presence of other cells. If there are too many cells at a lattice site, then there is no room for a new cell to be placed. If there is room for a new cell, we would expect because of the mechanical pressure of neighboring cells that the new cell to move towards the direction of least cellular concentration. These physical realities must be addressed in our model.
- **Cells must be both added and removed from the system.** It is a fact that biological systems both grow and die. Our model must incorporate both cell division, and death. It must also deal with the disposal of dead cells from the system through some sort of purging process.
- **The model must address specifically address spatial heterogeneity**. The main drawback with the ODE models is that they do not consider the spatial distribution of tumor cells or nutrients. This is the advantage of PDE models. We hope to sidestep some of the drawbacks of the PDE models while still embracing spatial heterogeneity.
- **The model must be extendable to include chemotherapy and immunotherapy.** The whole purpose of creating this model is so that we have a suitable framework to explore new treatment protocols. If we cannot guarantee that chemotherapy and immunotherapy can be accomodated, then we are heading down the wrong path.

2.2 Hybrid Lattice-Gas Cellular Automata Model

The model we choose to work with is a hybrid Lattice-Gas Cellular Automata model first considered in Dormann and Deutsch (2002). In the rest of this work, we summarize their model, and make improvements in the implementation to accomodate larger tumors than those considered by Dormann and Deutsch (2002).

2.2.1 Lattice-Gas Methodology

Lattice-Gas models are concerned with modeling microscopic phenomena in a way which provides a macroscopic reality. In Lattice-Gas models, particles are modeled explicitly as moving along a fixed lattice with some velocity and fixed methods of interaction amongst particles.A very thorough treatment of the history and application of Lattice-Gas models to a diverse set of physical problems is Chopard et al. (1998).

Our model adds a little bit to the regular Lattice-Gas automaton. Our model consists of a regular lattice which is populated by some number of particles. Each particle contains both a state and a velocity. Each particle corresponds to a biological cell. The state of each particle indicates whether or not the cell is cancerous or necrotic, and if it is cancerous whether it is dividing (mitosis) or not (apoptosis).

At each time step the model is updated in two ways. First, the velocity of each cell determines where on the lattice, if space is available, the cell moves next. Second, the state of the cell is updated according to the state transition diagram shown in 2.1.

Each lattice site $\vec{x}$ in our model contains four transport channels $\hat{c}_i$ and one rest channel, $\hat{c}_0$. Cells in the rest channel have no velocity. While cells in the four transport channels have unit velocity in the direction corresponding to their transport channel. Each channel can accomodate at most one cell. The configuration of each lattice site is depicted in 2.2.1.

At each time step the particles in the model move in the direction of their velocity vector. When a particle encounters a boundary, it has its velocity reversed. This is depicted in 2.3.

When particles collide with each other, they are redirected along an orthogonal axis. This is depicted in 2.4.

As cells need food to live, our model incorporates a background nutrient gradient which is evolved according to the diffusion equation on a lattice of the same size and dimension as the lattice of tumor cells. Additionally, it has been noted (**?**, **?**) that there is a necrotic signal gradient created by the presence of necrotic cells. Our model incorporates this additional background necrotic signal in the same manner as the nutrient gradient.

Let the probability that a quiescent cell enters mitosis be p_m. Then in light of our modeling assumptions and constraints, the probability that at time t a cell at lattice site $\vec{x}$ divides $\bar{p}_m(\vec{x}, t)$ is a function of p_m, the number of and type of neighboring cells, and the presence of nutrient and necrotic signal.

Let the number of cancerous cells at a lattice site be defined as $n_C(\vec{x}, t)$, and the number of necrotic cells at a lattice site $n_N(\vec{x}, t)$. The total number of cells at a lattice site is $n(\vec{x}, t) = n_N(\vec{x}, t) + n_C(\vec{x}, t)$. The amount of nutrient at a lattice site is $c_{nut}(\vec{x}, t)$, while the minimum amount of nutrient necessary to sustain cellular life is t_{nut}.

Then the probability that a quiescent cell enters mitosis depends first upon the presence of necrotic cells and the concentration of nutrient. If necrotic cells are present, then mitosis in inhibited. Otherwise, if the amount of nutrient is sufficient to support the current amount of cells at the min-

imum level, then we divide as the function defined below from Dormann and Deutsch (2002),

$$\bar{p}_m(r) = \begin{cases} \frac{p_m}{n_C(\vec{x},t)} \left(\frac{C_{nut}(\vec{x},t) - t_{nut}}{1 - t_{nut}} \right), & \text{if } n_N(\vec{x},t) = 0 \text{ and } C_{nut}(\vec{x},t) > n_C(\vec{x},t) t_{nut}, \\ 0, & \text{otherwise.} \end{cases}$$

Similarly, a cell enter apoptosis with probability p_d augmented if there is not enough nutrient available. The probability of death is hastened by the presence of other cells, all competing for available nutrient, which gives the function,

$$\bar{p}_d(\vec{x},t) = \begin{cases} p_d n_C(\vec{x},t), & \text{if } C_{nut}(\vec{x},t) > n_C(\vec{x},t) t_{nut}, \\ 0, & \text{otherwise.} \end{cases}$$

A cell becomes necrotic if there is no longer enough nutrient to sustain the cells at the lattice site. There is a chance that a cell becomes necrotic if there are other cells at the lattice site, but there is still enough nutrient available. This probability is p_n, and we compute the probability for transition as,

$$\bar{p}_n(\vec{x},t) = \begin{cases} 0, & \text{if } n_N(\vec{x},t) = 0 \text{ and } C_{nut}(\vec{x},t) > n_C(\vec{x},t) t_{nut}, \\ p_n, & \text{if } n_N(\vec{x},t) > 0 \text{ and } C_{nut}(\vec{x},t) > n_C(\vec{x},t) t_{nut}, \\ 1, & \text{otherwise.} \end{cases}$$

In this way our model explicitly addresses the issue of spatial heterogeneity. We are considering the local presence of cells and their intercompetition for resources. Additionally, at each time step, a fraction of dead cells and necrotic cells are removed from the lattice, thus making room for new cells.

This model diverges from traditional cellular automata models in that cells move about a lattice and can populate other cells in the lattice. In a cellular automata model, cells remain fixed on their lattice. Additionally, this model diverges from traditional Lattice-Gas automata models in that each particle contains a state variable which indicates whether or not the cell is necrotic.

To summarize, the main features of our model are:

- **State.**

  This is the idea we are borrowing from traditional cellular automata. The state is simply a value from a finite set. In our model they will

indicate whether a cell is cancerous and dividing, cancerous and quiescing, or necrotic.

- **Transport channels.**

  The channels, depicted in 2.2.1 are central to moving the cells along the lattice. This is where the model diverges from traditional cellular automata. Cells have velocity and enter and exit between channels on the lattice. This allows cells to populate other areas of the lattice without necessarily growing a stream of cells.

2.3 Advantages and Disadvantages

Here we consider the relative merits of our hybrid LGCA model.

2.3.1 Advtanages

- **Flexible.** The lattice size can be increased to accomodate larger tumors.
- **Simple to understand.**

  There is a direct correspondance between cells in the model and biological cells in the computer simulation. Qualitative understanding of the models behavior can be immediately interpreted.

- **Easy to simulate.**

  The execution of the model consists of a series of simple identical steps. While the computations may be many, their individual complexities are small.

2.3.2 Disadvantages

- **Not easy to perform traditional analysis.** Unlike the work done with ODE and PDE models, there is no phase space. We cannot analytically determine the long term behavior of the system given the initial conditions. We can only take a statistical average of the simulations. This is somewhat unsatisfactory.
- **Parallelization requires a little bit of effort.** Nothing in life is free.

2.3.3 Preference Weighting

When a cell divides we do not simply place its offspring in a random transport channel. Instead, we consider a preference weighting. In light of our assumptions and constraints, a cell should prefer to be in an area of high nutrient concentration, and low competition with other cells. That is, given the choice between two lattice sites, the new cell would on average choose the less crowded, more nutrient rich one.

It is still possible for a cell to move towards an area of high concentration, especially if there is the presence of necrotic signal.

What we seek is a probability distribution function to be computed for a lattice site each time a cell divides. The probabilities of each transport channel are computed, and a random sample is taken. In our model we consider a linear function which combines the amount of necrotic signal, and the number of cells in the neighboring lattice site.

$$pref(\vec{x}, t) = 5c_{sig}(\vec{x} - \hat{c}_i) - \sum_{i=0}^{4} n_i(\vec{x} - \hat{c}_i, t) \tag{2.1}$$

When the concentration of necrotic signal is maximal, the only thing that can dissuade a cell from moving towards that lattice site is the total occupation of that site. If the necrotic signal is constant among all the neighboring lattice sites then the sites are ordered according to their population.

2.3.4 Cellular Collision

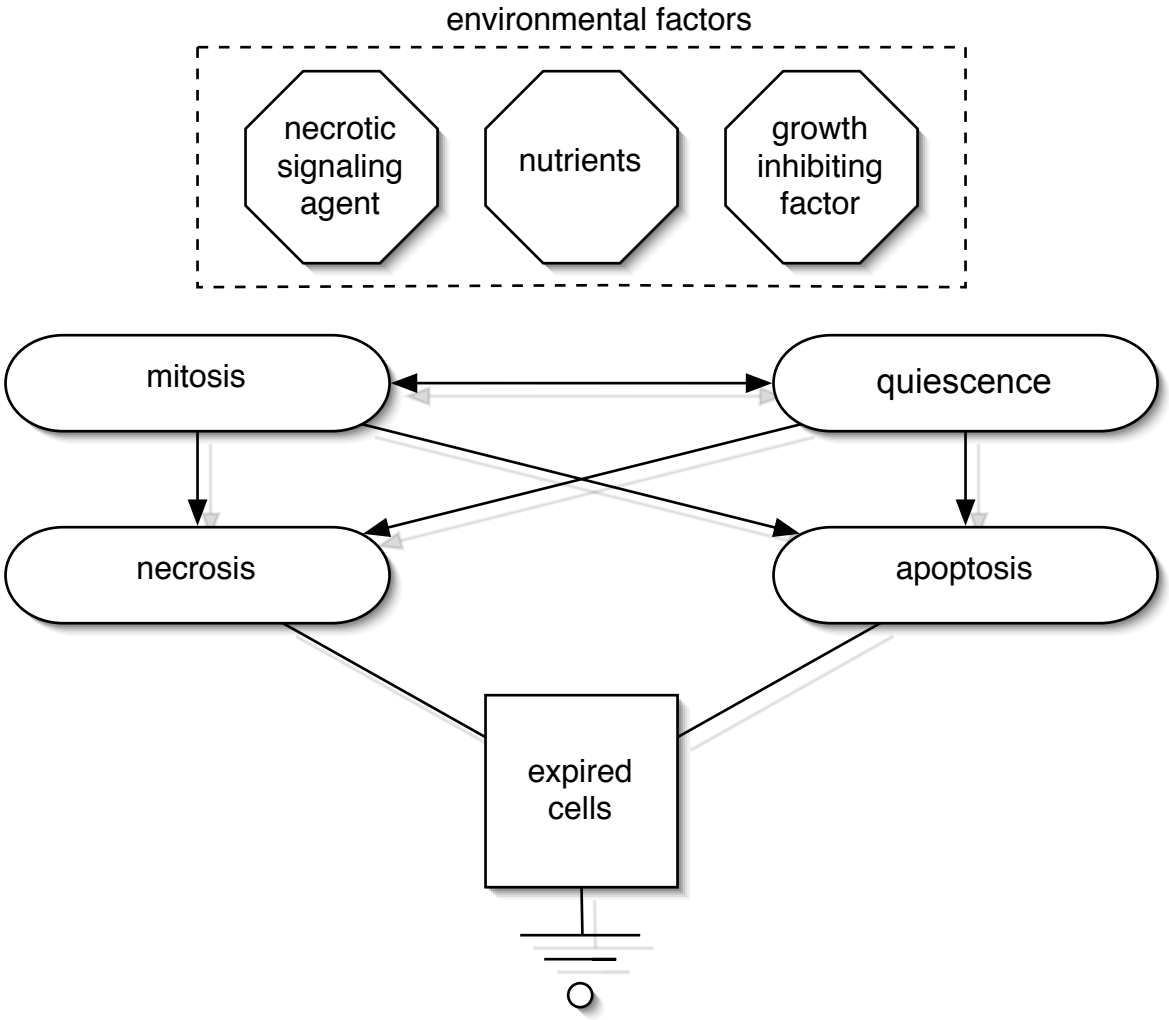


Figure 2.1: State Transition Diagram for Cancerous Cells

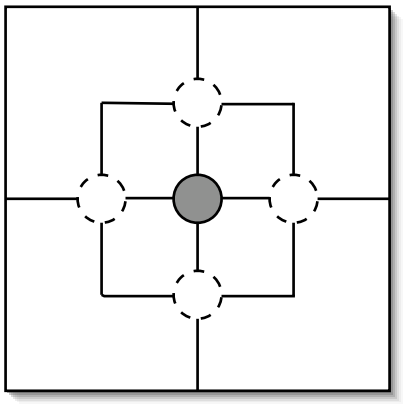


Figure 2.2: A 2D square lattice site with rest channel (center) and four transport channels: north, east, south, west.

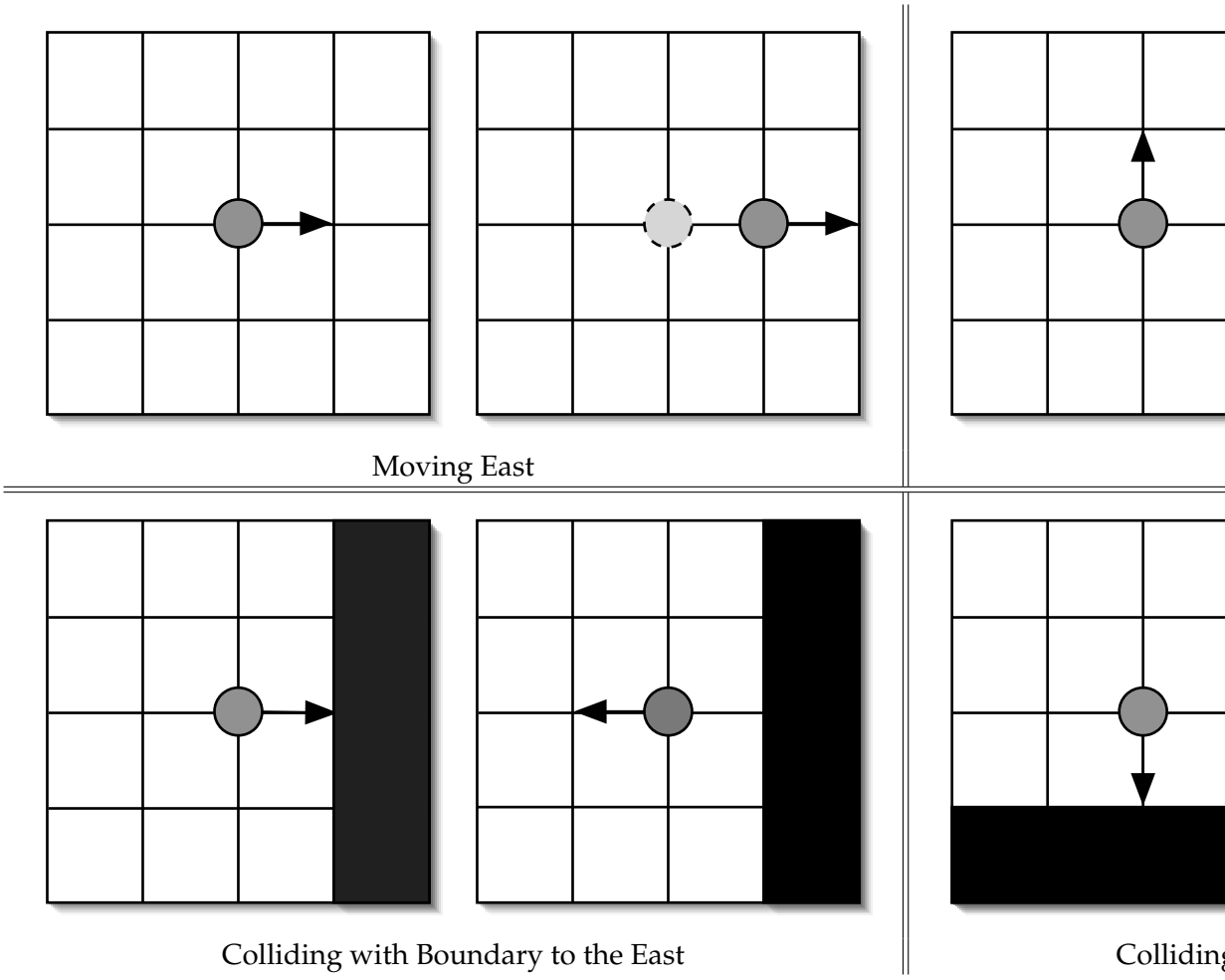


Figure 2.3: Dynamics of particles with respect to velocity

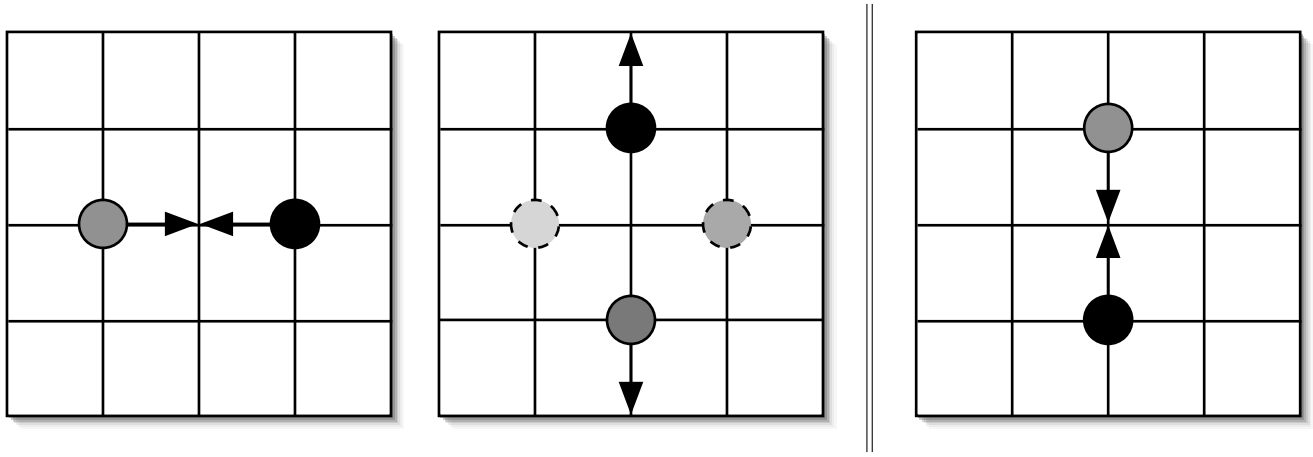


Figure 2.4: Dynamics of particles with respect to collisions

2.4 Diffusion of Signals

The model simulations investigated in Dormann and Deutsch (2002) were limited by the computational power available to a 200×200 lattice. On such a small lattice one cannot hope to simulate tumors which physicians could detect and identify as candidates for treatment with chemotherapy or immunotherapy. The principal computational limitation in Dormann and Deutsch (2002) was their use of the unconditionally stable Crank-Nicholson scheme in simulating the diffusion over time of the concentration nutrient and the necrotic signal $\rho(\vec{x}, t)$ using the diffusion equation with coefficient of diffusion D:

$$\frac{\partial \rho(\vec{x}, t)}{\partial t} = D\nabla^2 \rho(\vec{x}, t). \tag{2.2}$$

Such a scheme involves the inversion of a tridiagonal matrix equation at each time step and cannot easily be parallelized.

In this chapter we consider an alternative approach to the diffusion problem: using a similar lattice-gas automata, LGA, method as our model of tumor growth. The benefits are a simple, parallelizable implementation, and the potential re-use of code from our tumor model implementation.

2.4.1 Background of the Lattice-Gas Method

As discussed in **?**, LGA-like models have long been studied in pysics as theoretical models of molecular dynamics. The first LGA models were introduced for the theoretical study of fluid flow (**?**). It was later realized that the locality, uniformity and spatial regularlity of models based on LGA's make them ideal candidates for large-scale simulation on parallel hardware (**?**,**?**,**?**). Since then, LGA simulations have been used to study a variety of physical pehnomena, including fluid dynamics, chemical reactions, and phase changes (**?**, **?**, **?**, **?**, **?**), and diffusion (**?**).

The principal ideological advantage of LGA models is that the computer model is the same as the theoretical model. This is unlike models based on partial differential equations, for example, where an intervening stage of numerical analysis is needed. LGA models and related modeling techniques have both a theoretical and practical aspect, providing simple models that we can directly experiment with, visualize, and analyze mathematically.

2.4.2 Lattice-Gas Method Applied to Diffusion

The method we develop for modeling diffusion considers particles at the microscopic scale and recovers 2.2 in the limit of an infinitely fine lattice. The method we present was first introduced by **?**, the presentation and analysis is from Chopard et al. (1998) and D'Souza et al. (2001).

Consider a two-dimensional square lattice with particles hopping between adjacent sites of the lattice with unit velocity. Each lattice site has four *transport channels* $N_i(\vec{x}, t)$, along each lattice direction $\hat{c}_i$, at each site $\vec{x}$ at time t, where $i \in \{0, 1, 2, 3\}$, can either be occupied (with a particle to be moved) or empty. That is, $N_i(\vec{x}, t) = 1$ or 0 respectively. Since there are four channels at each lattice site, there can be up to four particles at each site. At every integer time, the particles can move in one of four directions corresponding to the four lattice directions: north, south, west, or east. Note that there is no rest channel here. Particles always have some velocity.

The dynamics of the model are decomposed into two phases: *interation* and *streaming*. Each unit time step in the simulation is composed of these two phases. We denote the state of the system at the fractional time step after interaction but before streaming as $N'_i(\vec{x}, t)$.

In the interaction phase, the velocity of particles are altered according to our local interaction rules. In the streaming phase, particles move along the lattice in the direction of their transport channels. If we were to remove the interaction phase, all particles would continue in their initial direction indefinitely.

The interaction rule considered in Chopard et al. (1998) had each channel at each site $\vec{x}$ at time t randomly permuted with each other channel at the site. This was done by a probabilistic function $r(\vec{x}, t) \in \{0, 1, 2, 3\}$ which rotated each channel by $r(\vec{x}, t) \cdot \pi/2$ radians. The introduction of $r(\vec{x}, t)$ allows the particles to execute simultaneous random walks on the lattice.

Thus after the permutation,

$$N'_i(\vec{x}, t) = N_{i+r(\vec{x},t)}(\vec{x}, t) \tag{2.3}$$

After the streaming phase, each transport channel is occupied by the particle from the adjacent lattice site in the transport channel in the direction opposite. That is, a north bound particle moves into the south transport channel of its northern neighbor, and vice versa. This can be summarized by the equation,

$$N_i(\vec{x}, t+1) = N'_i(\vec{x} - \hat{c}_i, t) = N_{i+r(\vec{x}-\hat{c}_i,t)}(\vec{x} - \hat{c}_i, t) \tag{2.4}$$

Note that $(\hat{c}_i = -\hat{c}_{i+2})$.

The principle advantage to using a LGA model in describing diffusion is that our computer implementation of the model is quite straight forward. At each time step we take each lattice point and randomly permute the particles present in its velocity channels. We then stream each particle in the direction of its velocity into adjacent lattice sites onto a new empty lattice. Once all the particles have been streamed, we copy the new empty lattice onto our old lattice and repeat the procedure.

By appropriately choosing the probability distribution for $r(\vec{x}, t)$, one can control the average number of steps in a particular direction a particle advances before it has its velocity altered. This, and the number of iterations we run our simulation for, allows us to control the coefficient of diffusion D.

Though Chopard et al. (1998) consider a more general model for particle interaction, we will show that when $r(\vec{x}, t)$ is uniformly distributed among the four transport channels the resulting model converges to the diffusion equation 2.2 in the limit of an infinitely fine lattice. We show this result through the lattice-Boltzmann statistical averaging method.

If we take the expected value of 2.4,

$$\begin{aligned} E[N_i(\vec{x}, t+1)] &= E[N_{i+r(\vec{x}-\hat{c}_i,t)}(\vec{x} - \hat{c}_i, t)] && (2.5) \\ &= E[\frac{1}{4}\sum_{j=0}^{3} N_j(\vec{x} - \hat{c}_i, t)] && (2.6) \\ &= \frac{1}{4}\sum_{j=0}^{3} E[N_j(\vec{x} - \hat{c}_i, t)]. && (2.7) \end{aligned}$$

We define the occupation number of each channel as $n_i(\vec{x}, t) = E[N_i(\vec{x}, t)]$, then the lattice-Boltzmann equation for our system is,

$$n_i(\vec{x}, t) = \frac{1}{4}\sum_{j=0}^{3} n_j(\vec{x} - \hat{c}_i, t). \quad (2.8)$$

We define the total density of a lattice site $\rho(\vec{x}, t)$:

$$\rho(\vec{x}, t) = \sum_{i=0}^{3} n_i(\vec{x}, t). \quad (2.9)$$

Combining 2.9 and 2.8,

$$\rho(\vec{x}, t+1) = \sum_{i=0}^{3} n_i(\vec{x}, t+1) \tag{2.10}$$

$$= \frac{1}{4}\sum_{i=0}^{3}\left[\sum_{j=0}^{3} n_j(\vec{x} - \hat{c}_i, t)\right] \tag{2.11}$$

$$= \frac{1}{4}\sum_{i=0}^{3}\rho(\vec{x} - \hat{c}_i, t). \tag{2.12}$$

We now take the limit as the time step Δt and the lattice spacing Δx both approach zero while $(\Delta x)^2/\Delta t$ approaches a constant.

We take the Taylor expansion of $\vec{x}$ in 2.10 to order $(\Delta x)^2$,

$$\rho(\vec{x}, t+1) = \frac{1}{4}\sum_{i=0}^{3}\left[\sum_{j=0}^{3} n_j(\vec{x}, t) + \Delta x(\hat{c}_i \cdot \nabla) n_j(\vec{x}, t) + \frac{(\delta x)^2}{2}(\hat{c}_i \cdot \nabla)^2 n_j(\vec{x}, t)\right] \tag{2.13}$$

$$= \rho(\vec{x}, t) + \frac{(\Delta x)}{4}\sum_{i=0}^{3}\sum_{j=0}^{3}(\hat{c}_i \cdot \nabla) n_j(\vec{x}, t) \tag{2.14}$$

$$+ \frac{(\Delta x)^2}{8}\sum_{i=0}^{3}\sum_{j=0}^{3}(\hat{c}_i \cdot \nabla)^2 n_j(\vec{x}, t) \tag{2.15}$$

Simplifying 2.13 we obtain,

$$\rho(\vec{x}, t+1) = \rho(\vec{x}, t) + \frac{(\Delta x)^2}{4}\nabla^2\rho(\vec{x}, t).$$

Which when Taylor expanded to order (Δt) we obtain the diffusion equation,

$$\frac{\partial\rho(\vec{x}, t)}{\partial t} = D\nabla^2\rho(\vec{x}, t).$$

where the diffusion constant, $D = (\Delta x)^2/(4\Delta t)$.

2.4.3 Results of Diffusion Simulation Using a Lattice-Gas Code

The simulation results for our method confirm qualitatively our belief that at the microscopic level the Lattice-Gas approximates the diffusion process.

In the first simulation, a 100×100 square is diffused for 3000 iterations. As the number of iterations increases the particles composing the rectangle execute simultaneous random walks with exclusion on the lattice. After 1

iteration, not every site on the lattice originally part of the rectangle is occupied. After 1000 iterations, the shape of the square is present, but ambiguous. After 3000 iterations, the shape of the square has all but disappeared.

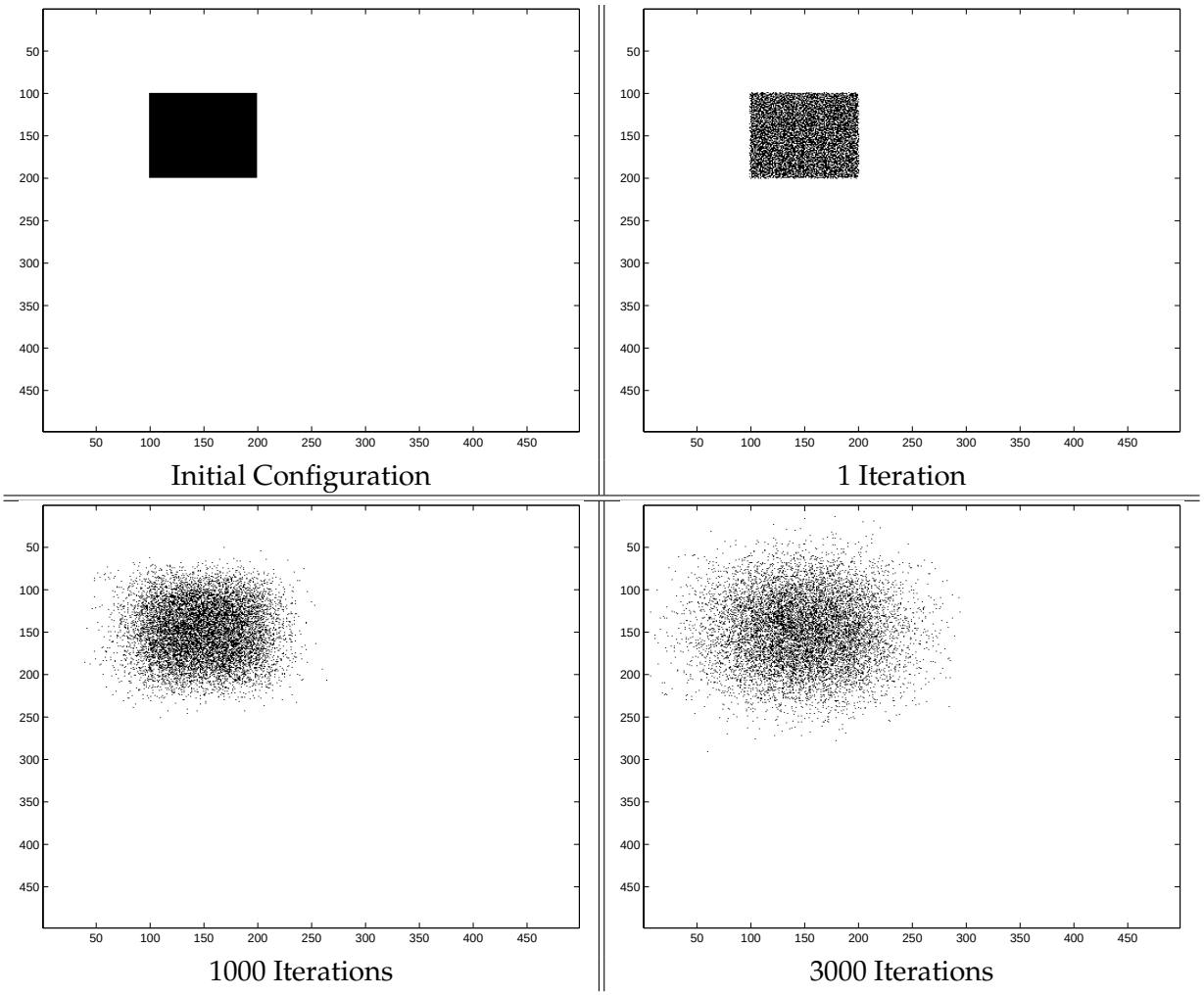


Figure 2.5: Lattice-Gas Diffusion on 100x100 Rectangle

In our second simulation, a triangle with base and height of length 100 is diffused for 3000 iterations. Again, as the number of iterations increases, the initial shape of the particles is made more ambiguous. After 3000 iterations, the shape of the distribution is indistinguishable, aside from the total quantity of cells, from the final shape in 2.5.

2.4.4 Conclusion on Diffusion

While the analysis and simulations are promising, we note that the LGA method does not exactly model diffusion, but that on the whole all the simulations will statistically average to diffusion.

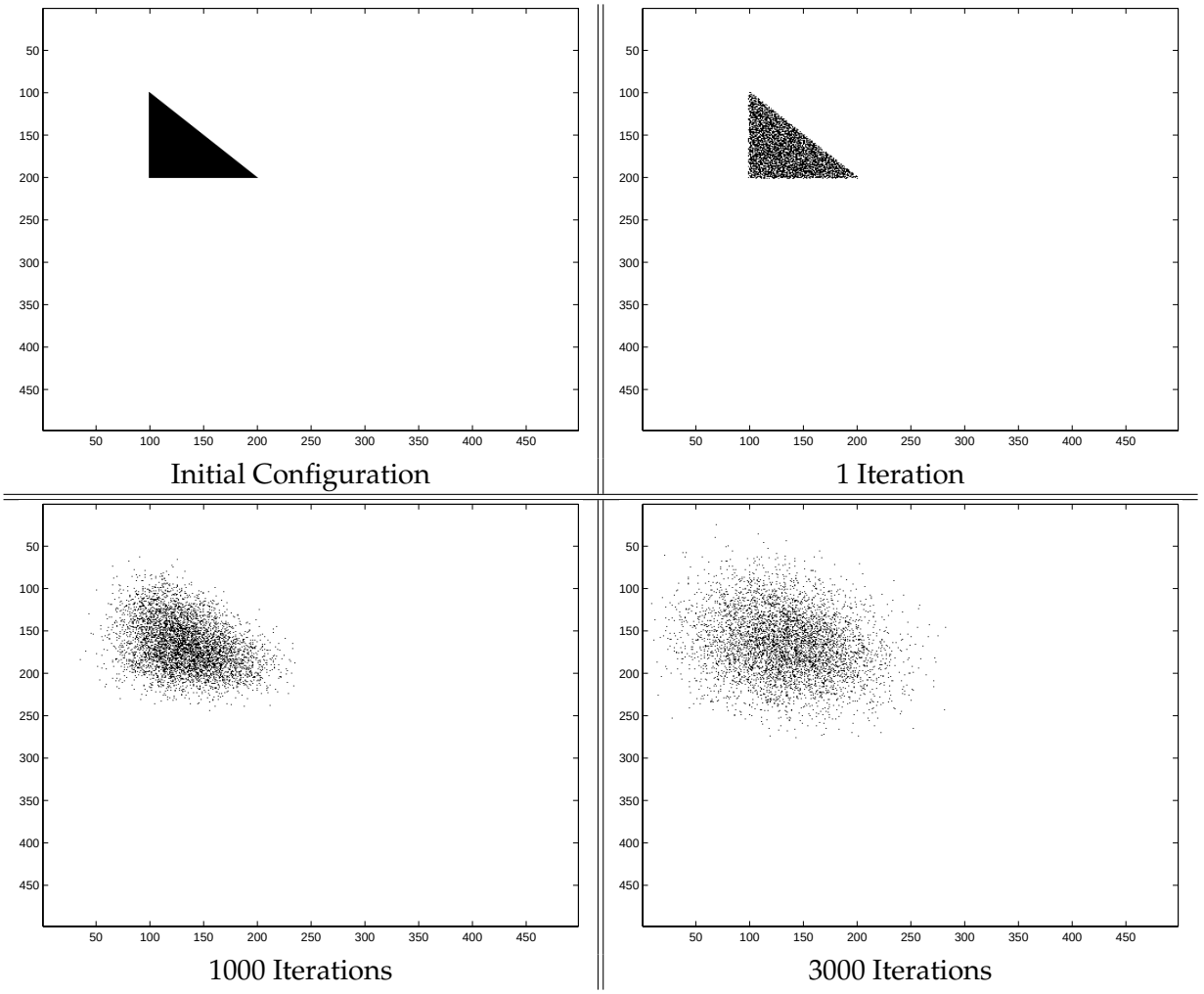


Figure 2.6: Lattice-Gas Diffusion on 100x100 Triangle

2.5 Parameter Estimation

Because this model is a re-implementation of the work in (Dormann and Deutsch (2002)), we calculate parameters as in their work.

- **Volume of tumor cells.** In our model we will consider V-79 cells. V-79 cells are commonly used as targets in toxicity studies, and other *in vitro* experiments.

  From the literature (**?**), our cells have a volume of $V_c = 3.351 \times 10^{-5}\text{mm}^3$.

- **Volume of necrotic cells.** We assume that necrotic cells occupy one third of the volume of a regular tumor cell, $V_n = 1.117 \times 10^{-5}\text{mm}^3$.

- **Length of lattice.** We follow the modeling assumption in (Dormann and Deutsch (2002)) that cells are packed in the volume of a cubic lattice node which is chosen to be twice the volume of one tumor cell, $6.7 \times 10^{-5}\text{mm}^3$ (**?**). Thus the length of a square lattice area is,

$$\Delta l = (2 \times V_c)^{1/3} = (6.7 \times 10^{-5}\text{mm}^3)^{1/3} = 0.04\text{mm}. \tag{2.16}$$

- **Scale of cellular dynamics.**

  Tumor cells have slower dynamics than the chemical signals. Cell dynamics are observed on the order of an hour, while chemical dynamics are observed on ther order of a minute (**?**). Time step for cell dynamics, $\Delta k = 1\text{hour}$. Time step for chemical diffusion, $\Delta k_d = 1\text{minute}$.

- **Diffusion coefficient of nutrient and necrotic signal.**

  We make the simplifying assumption that the coefficient of diffusion for the nutrient and the necrotic signal are the same.

$$D = 10^{-6}\frac{\text{cm}^2}{\text{s}} = 3.64\frac{\Delta l^2}{\text{min}} \tag{2.17}$$

- **Glucose uptake rate.** Studies from the literature (**?**) show that if the external glucose concentration is approximately,

$$1.15 \times 10^{-5}\frac{\text{mg}}{\text{mm}^3} = 7.7 \times 10^{-8}\frac{\text{mg}}{\Delta l^3},$$

then the consumption rate of glucose is,

$$7.2 \times 10^{-8} \frac{\text{mg}}{\text{cell} - \text{hour}}$$

So during a one hour period, a cell at a lattice site will consume all available nutrient.

$$\bar{c}_{nut} = 1 \frac{1}{\text{cell} - \text{hour}} \tag{2.18}$$

- **Critical glucose concentration** The amount of glucose necessary to sustain a cell is,

$$1.4 \times 10^{-4} \frac{\text{mg}}{\text{mm}}^3 = 9.38 \times 10^{-9} \frac{\text{mg}}{\Delta l^3}.$$

The critical glucose concentration, the percentage of normal consumption that can sustain a cell's life, is then,

$$t_{nut} = \frac{9.38 \times 10^{-9} \frac{\text{mg}}{\Delta l^3}}{7.7 \times 10^{-8} \frac{\text{mg}}{\Delta l^3}} = 0.12 \tag{2.19}$$

- **Doubling Times** V-79 cells divide between 10 and 19 hours (**?**, **?**). If we fix the doubling time at 16 hours, if we assume that the population initially grows exponetially, then the initial growth rate, the difference between the rate of mitosis and apoptosis (death), is,

$$p_m - p_d = \frac{\log 2}{16\text{hour}} = 0.04/\text{hour} \tag{2.20}$$

- **Rate of Mitosis** We take the rate of mitosis from Dormann and Deutsch (2002) with no further justification.

$$p_m = 0.05/\text{hour} \tag{2.21}$$

By 2.20, we compute the rate of apoptosis as,

$$p_d = p_m - 0.04/\text{hour} = 0.01/\text{hour} \tag{2.22}$$

- **Other parameters** We set the remaining parameters to the values published in Dormann and Deutsch (2002).

Necrosis rate,

$$p_n = 8 \times 10^{-3}. \tag{2.23}$$

The rate for dissolution of necrotic cells from the system,

$$q_d = 5 \times 10^{-4}. \tag{2.24}$$

Production rate for necrotic signal by necrotic cells,

$$\bar{c}_{sig} = 1. \tag{2.25}$$

The decay rate for necrotic signal from the system,

$$s_{sig} = 0.4. \tag{2.26}$$

2.6 Implementing the Model

In this section we discuss the implementation of the model's simulation: the proliferation of cells and the diffusion of nutrient and necrotic signal.

2.6.1 Procedural Outline

Our simulation follows these steps at each iteration:

1. Initialize the lattices for the tumor, nutrient, and necrotic signal.
2. First decay the necrotic signal, then remove any signal outside the tumor.
3. Refill nutrient on the lattice outside the tumor.

wing rules to each site on the lattice.

4. If the number of necrotic cell is greater than 0, remove each necrotic cell with rate q_d.
5. If the rest cell is in the proliferating state, and not all transport channels are occupied, create a new cell and place according to the preference weight.
6. Consume available nutrient if proliferating, or quiescent.
7. Remove all cells entering apoptosis.
8. For all cells entering necrosis, update state, add signal to necrotic signal gradient.

Stop parallel processing

9. Diffuse the necrotic signal and nutrient on their respective lattices.
10. Move cells according to their velocity vectors. Handle collisions where appropriate.

2.6.2 Implementing the Model

In this section we discuss the particulars of implementing our model.

- **Lattice Representation.** Because we are constrained to a 2D lattice, we represent by a 512×512 array of lattice points. Each lattice point is a structure containing five pointers to cells representing each transport channel: north, south, west, east, and rest.

  The nutrient and necrotic signal lattices are similar, but they do not compute with the rest channel.

- **cell**

  Each cell is a state and velocity pair. Each state is an integer representing the various states a cell may occupy: 0 - dividing, 1 - quiescing, 2-necrotic, 3-dead.

- **Rates of Diffusion** The nutrient diffuses at the same rate as the necrotic signal, however the cells moves at a much slower rate.

  We reconcile this by running the diffusion simulation for more time steps for the nutrient and necrotic signal than the cells. We run the nutrient diffusion for 1000 time steps, and we run the cellular *diffusion* for 300 time steps.

- **Handling Collisions.**

  Collisions between cells are handled by the routine which moves cells along the lattice in the direction of the transport channel. If two cells collide, they swap proceed to their neighboring lattice site, but exchange positions with an orthogonal particle, if present. That is, if a particle is heading north and is collided by a particle heading south, then the two particles are placed in the east and west transport channels instead of the rest channel.

- **probabilistic functions** One speed bottle neck is the implementation of the stochastic function $r(\vec{x}, t)$ which helps us to permute the particles in the nutrient and necrotic signal diffusion processes. We use the Unix rand(1) command to supply ourselves with a series of random numbers. rand(1) is useful because if we supply it with a fixed seed, we will always obtain the same series of random integers. This is useful if we would like to re-simulate an interesting run of our model.

  Alternatively, we could use pre-computed random numbers from rand(1) or a website. However, we do not currently handle this.

2.7 Parallelizing the Code

One of the hallmarks of lattice-gas methods is that they are easily ammenable to implementation on a parallel computing platform.

2.7.1 Choosing a computing platform

The question we ask ourselves is which computing platform should we use for our simulation.

1. **Vladymir** There exist in the public domain several codes to execute Lattice-Gas methods on parallel hardware. None is as promising as the recently released Vladymir library from Jonas Lätt (**?**). Vladymir can be parallelised in a completely implicit way: any program that runs on one processor will also run on a parallel machine, without any changes. However, you need to have PVM (discussed below) installed in order to use the parallel features.

   Also, since this library is only preliminary release, and most of the documentation is in French, the library remains on the border of suitability for our needs.

2. **MATLAB** MATLAB from The Mathworks is a general computing platform which offers some parallization features. However, the MATLAB compiler is an expensive addition, and in the author's experience, difficult to use.

3. **General computing platforms**

   The Parallel Virutal Machine system (PVM) is a portable message-passing programming system, designed to link separate host machines to form a *virtual machine* which is a single, manageable computing resource. The virtual machine can be composed of hosts of varying types, in physically remote locations. On the one hand, PVM is available free of charge, easy to target, and debug. On the other hand, PVM is difficult to maintain and requires the attention of a system administrator.

   MPI, the Message Passing Interface, is a bare-bones alternative to PVM. The programmer is responsible for configuring the system and ensuring that it is in working order before and during the computation. The trade-off is that MPI gives the programmer greater control over all aspects of parallelization.

Ultimately, we chose to use the PVM library to parallelize the code. This choice, in light of an able system administrator, allowed us to quickly leverage our exisiting serial code into a parallel code. We simply instruct the PVM to process each lattice interaction independently. We then reconstruct the solution on a single processor. The overhead involved in reconstucting the solution is minimal, but could be avoided if more thought was given to the matter.

The simulation scaled well to the four processors that were ultimately tested. From taking 3 minutes on a Pentium-4 to run a simulation of the model corresponding to 100 hours of cellular growth, the run took only 1 minute.

More parallelization is possible by cutting PVM out of the equation. However, doing so would introduce more overhead in the simulation code. The overhead involved in a parallel computation is governed by Ahmdal's Law, described below.

2.7.2 Overhead and Ahmdal's Law

The speed of a program is the time it takes the program to excecute. Speedup is defined as the time it takes a program to execute in serial (with one processor) divided by the time it takes to execute in parallel (with many processors). The formula for speedup is:

$$S = \frac{T(1)}{T(j)}$$

Where $T(j)$ is the time it takes to execute the program when using j processors. Efficiency is the speedup, divided by the number of processors used. Due to the cost of multiprocessor super computers, one must make sure that the marginal utility of the time and money spent building the computer is large enough, and not negative.

Amdahl's Law is a law governing the speedup of using parallel processors on a problem, versus using only one serial processor.

A Speedup Curve is simply a graph with an X-axis of the number of processors, compared against a Y-axis of the speedup. The best speed we could hope for, S = N, would yield a straight 45 degree curve. That is, if there were ten processors, we would realize a ten fold speedup. Anything better would mean that the program ran faster on a single processor than in parallel, which would not make it a good candidate for parallel computing. When B is constant Amdahl's Law predicts a speedup curve which is logarithmic and remains below the line S=N. This law shows that it is indeed

the algorithm and not the number of processors which limits the speedup. Also note that as the curve begins to flatten out, efficiency is drastically being reduced.

2.7.3 Implementation Conclusion

We first implemented the model on a serial computer using the C programming language. We then adapted the code to use the PVM system to execute the code on multiple computers in a cluster. We used only 4 computers, where each lattice (cellular, nutrient, necrotic) was quartered and simulated on each processor. The number of computers used in parallel could be increased by a factor of 2 or 4 with a little more work. The overhead involved in adding new processors is making sure that the interaction phase of the lattice-gas method correctly communicates the position of particles moving between the pieces of lattice each processor is assigned.

Alternatively, each lattice could be evolved separately on three computers. This would involve only minimal overhead in communicating the results of each evolution between the three computers. However, extending this approach beyond the number of lattices present produces the same concerns of complexity and overhead as in the approach considered above.

Chapter 3

Results

In this section we discuss the results of simulating the model under a number of initial conditions and analyze the performance of the model. While the simulation was generally stable, the performance of the model was not the holy grail of cancer simulation I had been looking for.

3.1 Experiments

In each of our experiments we consider the following protocol

We run the following three experiments to validate that this model qualitatively reflects clinically observed principals of biological growth.

Turning off the necrotic signal

Withholding nutrient after the tumor has grown a bit.

Seeing how the tumor responds to changing the diffusion parameter.

Results: 1) the tumor grows without bounds while this is not observed in other in vitro studies this is to be expected. there is nothing inhibiting growth except for a lack of nutrients, but nutrients are constantly being refreshed.

2) the tumor dies because it has no food. it would be disturbing if the tumor did otherwise.

3) for large value of nutrient diffusion, it should be like adding nutrient to the entire lattice, not just outside the tumor

for small values of nutrient diffusion, the core should die because the nutrient is not making it to the center of the tumor

for intermediate values, the growth should be similar to normal tumor growth

where are the thresholds between these regimes?

3.2 Performance

How well does the code scale up?

From 1 to 4 processors.

Again, Amdahl's law starts to come into play

If the dimension of the mesh is a lot larger than the number of processors then we can get away with lots of parallelization.

The valdymir library goes a long way to speed up this process, unfortunately it's in French and still in early releases.

Chapter 4

Possible Extensions to the Model

4.1 Other Geometries

Currently our hybrid LGCA simulation takes place on a 2D square lattice. Other Lattice-Gas models have been implemented on 2D hexagonal lattices, depicted in 4.1, and 3D square lattices. Because the diffusion process is implemented as a Lattice-Gas model, its underlying lattice can be extended with the tumor simulation. And because we are using the dimension splitting technique described in D'Souza et al. (2001), adding additional dimensions to our model of diffusion is simple and does not alter its convergence to the continuous diffusion equation in the limit of an infinitely fine lattice.

4.2 Additional Cellular Populations

In our current simulation we have limited ourselves to only cancerous cells. This restriction can be relaxed and we can add new states for the cells to accomodate a normal cell population, a competing population of cancer cells, or even immune cells.

Normal cells would consume nutrient, birth, live, and die at rates different from cancerous cells, but would occupy the same amount of space as cancerous cells.

4.3 Immune Cell Populations

Because immune cells interact with normal and cancerous cells they would need special consideration in the simulation. Their effectiveness would be diminished after each interaction with normal and cancerous cells. Additionally, the preference weighting function for immune cells would have to favor the cell moving towards areas of high cellular concentration instead to areas of low concentration. The transition probabilities for normal and cancrous cels would need to be changed to incorporate the presence of immune cells.

4.4 Blood Vessel Networks

Alarcon, Byrne, and Maini Alarcón et al. (2003) consider the effect a network of blood vessels which diffuse nutrient have on the growth of tumors. Their work considers the presence of a connected set of blood vessel cells on a 2D square lattice. Their model shows a dependence of the growth of tumors on the shape of the hemotopy, blood flow. A similar addition could be made to our hybrid LGCA model in either one of two ways.

First, add static background blood vessel lattice could be constructed that would diffuse nutrients. Nutrient would be replenished at some rate on the cellular lattice only where blood vessels were present in the background blood vessel lattice. This is in contrast to our current simulation's *in vitro* style application of nutrient everywhere on the lattice that is not currently occupied by a cell.

Alternatively, a static blood vessel cell could occupy the cellular lattice in addition to the other cells in the simulation. These cells would remain fixed on the lattice and thus not be subject to the streaming phase of the model. This would more closely model the physical reality of blood vessels having some volume.

In Alarcon, Byrne, and Maini's model of the blood vessel network Alarcón et al. (2003), the blood vessels are allowd to reorganize and change the distribution of nutrient between iterations of the model. If we require that the set of blood vessel cells remain connected, it is not entirely clear that our hybrid LGCA model could observe this constraint and still remain parallizable.

4.5 Chemotherapy Protocols

Perhaps most importantly, chemotherapy can be introduced easily. If we consider chemotherapy as another background signal like nutrient and necrotic signal, we simply add another diffusion computation to our model. We would have to modify the transition probabilities for all cells involved, as chemotherapy kills normal cells as well as cancerous cells.

Implementing chemotherapy in this way for our hybrid LGCA model makes for a straight forward testing of protocols for chemotherapy. After a number of iterations, we augment the background chemotherapy signal, and continue running iterations of the model while the chemotherapy signal is periodically diffused.

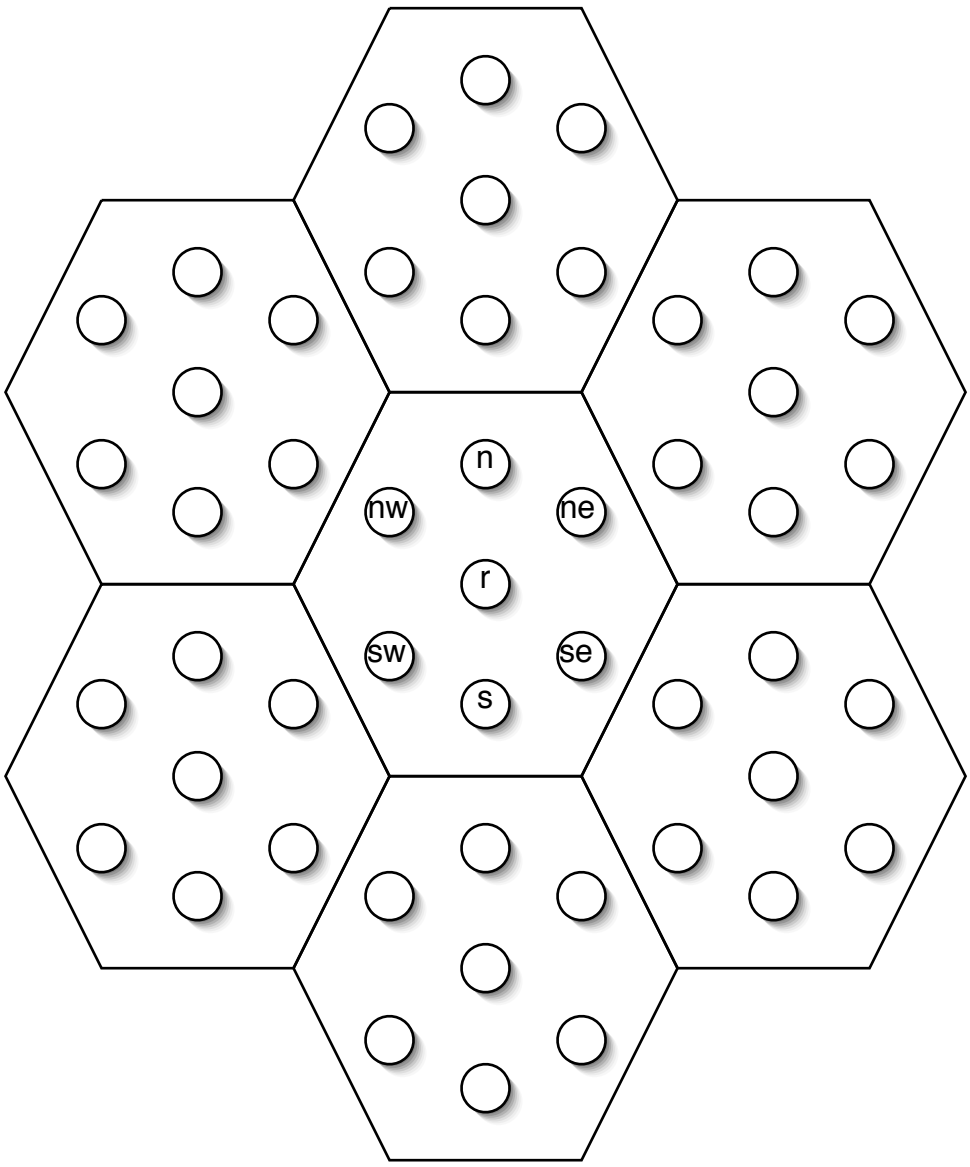


Figure 4.1: Hexagonal Lattice with Rest Channel and Six Velocity Channels

Chapter 5

Closing Remarks

We have found the following.

Parallel computing is an effective method for speeding up the simulation of our hybrid LGCA model. Because each iteration of our hybrid LGCA model executes many independent computations, each iteration can be broken up amongst many different processors in a parallel computing system.

PVM makes a good platform to test these methods. Though once a code has matured, it should be adapted to an explicitly parallel library, such as MPI, as that would give the most control over the details of parallelization and minimize the amount system overhead.

Parameters are sensitive to change. Accurate parameter estimation is necessary in order to have an accurate and predictive model.

There are many types of cancers. Biological research has shown that these different cancers depend upon a number of different factors. An important factor to breast cancer may be relatively unimportant in prostate cancer. Currently, the practice in mathemtical modeling is to accumulate values for parameters from disparate studies in the biological literature. While the rate of diffusion of nutrient may be taken from a mouse study, the rate of cellular division may be taken from a primate study. It is not clear whether this practice is appropriate.

Bibliography

Adam, J. A. (1987). A mathematical model of tumor growth: Effects of geometry and spatial nonuniformity on stability. *Mathematical Biosciences*, 86:183–211.

Alarcón, T., Byrne, H., and Maini, P. (2003). A cellular automaton model for tumor growth in inhomogeneous environment. *Journal of Theoretical Biology*, 225:257–274.

Brown, R. (1828). A brief account of microscopical observations made in the months on june, july, and august, 1827, on the particles contained in the pollen of plants; and on the general existence of active molecules in organic and inorganic bodies. *The Philosophical Magazine*, 4:161–173.

Byrne, H. and Chaplain, M. (1995). Growth of necrotic tumors in the presence and absence of inhibitors. *Mathematical Biosciences*, 135:187–216.

Byrne, H. M. and Chaplain, M. A. J. (1996). Growth of necrotic tumors in the presence and absence of inhibitors. *Mathematical Biosciences*, 135:187–216.

Chang, W., Crowl, L., Malm, E., Todd-Brown, K., Thomas, L., and Vrable, M. (2003). Analyzing immunotherapy and chemotherapy of tumors through mathematical modeling. Technical report, Harvey Mudd College.

Chaplain, M. and Sleeman, B. (1993). Modelling the growth of solid tumours and incorporating a method for their classification using nonlinear elasticity theory. *Journal of Mathematical Biology*, 31:431–473.

Chopard, B., Luthi, P., and Masselot, A. (1998). Cellular automata and lattice boltzmann techniques: An approach to model and simulate complex systems.

de Pillis, L. and Radunskaya, A. (2001). A mathematical tumor model with immune resistance and drug therapy: An optimal control approach. *Journal of Theoretical Medicine*, 3:79–100.

de Pillis, L. G. and Radunskaya, A. (2003). The dynamics of an optimally controlled tumor model: A case study. *Mathematical and Computer Modeling*.

Dormann, S. and Deutsch, A. (2002). Modeling of self-organized avascular tumor growth with a hybrid cellular automaton. *Silico Biology*, 35.

D'Souza, R., Margolus, N. H., and Smith, M. A. (2001). Dimension-splitting for simplifying diffusion in lattice-gas models. *Journal of Statistical Physics*, 107:401–422.

Düchting, W. and Vogelsaenger, T. (1985). Recent progress in modelling and simulation of three-dimensional tumor growth and treatment. *Biosystems*, 18:79–91.

Einstein, A. (1956). *Investigations on the Theory of Brownian Movement*. Dover, New York, New York.

Friedman, A. and Reitich, F. (1998). Analysis of a mathematical model for the growth of tumors. Technical Report 1546, Institute for Mathematics and Applications, Univ. MN.

Greenspan, H. (1972). Models for the growth of a solid tumor by diffusion. *Studies in Applied Math*, 52:317–340.

JACKSON, T. L. and BYRNE, H. M. (1999). A mathematical model to study the effects of drug resistance and vasculature on the response of solid tumors to chemotherapy. *Mathematical Biosciences*, 164:17–3.

Kansal, A., Torquato, S., IV, G. H., Chiocca, E., and Deisboeck, T. (2000). Simulated brain tumor growth dynamics using a three-dimensional cellular automata. *Journal of Theoretical Biology*, 203:367–382.

Liebling, T. M. and de Werra, D., editors (1997). *Recent progress in unconstrained nonlinear optimization without derivatives*, North-Holland. Mathematical Programming Society.

Petrovski, A. and McCall., J. A. W. (2001). Multi-objective optimisation of cancer chemotherapy using evolutionary algorithms. In *Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimisation*,

Zurich, Switzerland, Lecture Notes in Computer Science, Berlin. Springer-Verlag.

Qi, A., ZHENG, X., DU, C., and AN, B. S. (1993). A cellular automaton model of cancerous growth. *Journal of Theoretical Biology*, 161:1–12.

Savic, D. A. and Walters, G. A. (1995). Genetic algorithm techniques for calibrating network models. Technical Report 95/12, University of Exeter.

U.S. Department of Health and Human Services (1995). A national program of cancer registries. Technical report, Center for Disease Control and Prevention.

U.S. Department of Health and Human Services (2003). Fact sheet. Found at `http://www.cancer.gov`.

von Stryk, O. (1999). *User's Guide for A Direct Collocation Method for the Numerical Solution of Optimal Control Problems*. Simulation and Systems Optimization Group, Technische Universität Darmstadt.

Wasserman, R., Acharya, R., Sibata, C., and Shin, K. (1996). A patient-specific in vivo tumor model. *Mathematical Biosciences*, 136:111–140.

Winter, G., Periaux, J., Galan, M., and Cuesta, P. (1995). *Genetic Algorithms in Engineering and Computer Science*. John Wiley and Sons, Chickester, England.

Zitzler, E., Deb, K., Thiele, L., Coello, C. A. C., and Corne, D., editors (2001). *Evolutionary Multi-Criterion Optimization, First International Conference, EMO 2001, Zurich, Switzerland, March 7-9, 2001, Proceedings*, volume 1993 of *Lecture Notes in Computer Science*. Springer.