**Tumor Metastasis Simulation via Lattice-Gas Cellular Automata**

**Abstract**

Cancer remains a devastating disease even in the modern era of medicine. While the molecular characteristics of cancer growth have been studied for some time, it is only relatively recently that methods within computational biology have been applied to this aspect of oncology. As computational capabilities progress, tumor growth and metastasis may prove to be pragmatically studied *in silico*. One such method study leverages the conceptual foundation of the cellular automata model and extends it using physical concepts, including the Cellular Potts model and gaseous fluid theory as applied to Euclidean lattices of cells.

Purpose: Implementation of a cellular automata model that is modular with respect to neighborhoods, dimensions, plotting, and physical constants.

**§1. Introduction**

This cellular automata simulation leverages existing models first developed to simulate gaseous particles and applies the same computational logic to growing tumor cells. The so-called Lattice Gas Cellular Automata (LGCA) uses a finite Euclidean lattice of cell “sites”, such that each site has interactivity with neighboring sites via predefined neighborhood bounds (Von Neumann; this particular choice is further explained). Additionally, proxies for cell-to-cell interaction *within* neighborhoods are modeled via like-type proximity. This approach was first implemented computationally and plotted in two dimensions, and then implemented in three dimensions. Furthermore, metastasis via intravasation and extravasation of single or cluster of cancer cells was simulated in 2D cellular automata using parameters and probabilities described in a previous literature[[1]](#footnote-1).

**§2. Methods**

**§2.1 Two-Dimensional LGCA**

Lattice-Gas Cellular Automata (in two dimensions) leverages existing principles of cellular automata[[2]](#footnote-2) and adds the movement of particles – cells with states this case – through discrete velocity vectors on a two-dimensional matrix. Depending on its local Von Neumann neighborhood, a given cell’s likely state in the next timestep of the simulation is computed (the “reactive step”.). Then, a cell’s propensity to move or propagate is applied via stochastic movement along discrete vectors to neighboring sites (Von Neumann neighborhoods, whereby each site on the lattice is associated with five possible channels of movement).

To this end, we wrote a *Go* program that declares a two-dimensional lattice of cell structs, whereby each cell bears a state in the form of a string, a location in the form of an ordered pair of integers, a velocity direction (for movement and propagation purposes), and an array of pointers to cells of its local neighborhood.

**§2.2 Reactive Step**

Cells may transition from state to state[[3]](#footnote-3)[[4]](#footnote-4). That is, we define “cells” within the lattice as either cancerous, healthy, or necrotic. Cancerous cells can either be in a state of quiescence (non-propagative, but not necrotic either) or in a proliferative state (whereby cell division will take place and a new cancer cell will propagate at the next timestep). Necrotic cells are defined as previously cancerous cells that died due to a lack of available resources (space, in this case). Healthy cells are simulated as sites on the lattice not occupied by other cells, or sites upon which other cells can invade and spread. Apoptotic cells were additionally modeled though not implemented in the runnable simulation for simplicity, though modular, workable code is included as “commented” code.

Cell state transitions are computed probabilistically via a simplified adaptation of Lattice-Boltzmann Energy theory. Three cell coupling coefficient[[5]](#footnote-5) parameters are employed to this effect: , , and . These represent modeling constants proportional to the strength of membrane coupling between cancer cells (quiescent included), necrotic-necrotic cell interaction, and cancerous-necrotic interaction.

Next, these constants are imputed into Lattice-Boltzmann energy factor equations defined for cells of type proliferative, quiescent, and necrotic, together with the count of cells of each type in the current neighborhood (*C*, *N*, for cancerous and necrotic, respectively):

In other words, one of the following reactions is possible for between any two given timesteps at site of the lattice [[6]](#footnote-6):

1. Proliferation: .
2. Quiescence: (no change).
3. Necrosis: .

Note that these Boltzmann models are highly related, proportional abstractions of the Hamiltonian energy model employed in the generalized Cellular Potts[[7]](#footnote-7) automata model[[8]](#footnote-8).

Next, these computed Boltzmann energies are imputed into a simplified[[9]](#footnote-9),[[10]](#footnote-10) lattice-gas probability model, such that proxies for proliferative, quiescence, and necrosis likelihoods are obtained by comparing ratios of Boltzmann factors[[11]](#footnote-11):

States of each cell on the lattice are then updated according to these probabilities at each timestep.

**§2.3 Movement Step**

Following the computational “reactive step” above, each cell is primed for potential movement as a function of their probabilistic states. Propagative cancerous cells will divide, with nascent cells invading the adjacent neighborhood least dense in cancerous cells (modeling invasion), with parental cells remaining in a current site. Modeling principles of chemotaxis[[12]](#footnote-12), necrotic cells will move toward regions *most* dense in other necrotic cells (the simulation plots a path of movement). Finally, quiescent cancer cells will not move. If a tie is obtained between cell-type densities in adjacent neighborhoods (e.g., if two or more adjacent neighborhoods contain the name number of *C* or *N* cells), one of these equivalent neighborhoods is chosen at random for cell movement and/or propagation.

Cells of relevant types are then moved synchronously (i.e., a single timestep change results in all cells updated across the lattice and within neighborhoods) at the given timestep, resulting in an updated lattice. Note that a Von Neumann cellular automata neighborhood (see Figure 1.) was chosen in this instance due to the practicality of implementation[[13]](#footnote-13) as well as reasons of physical verisimilitude[[14]](#footnote-14).

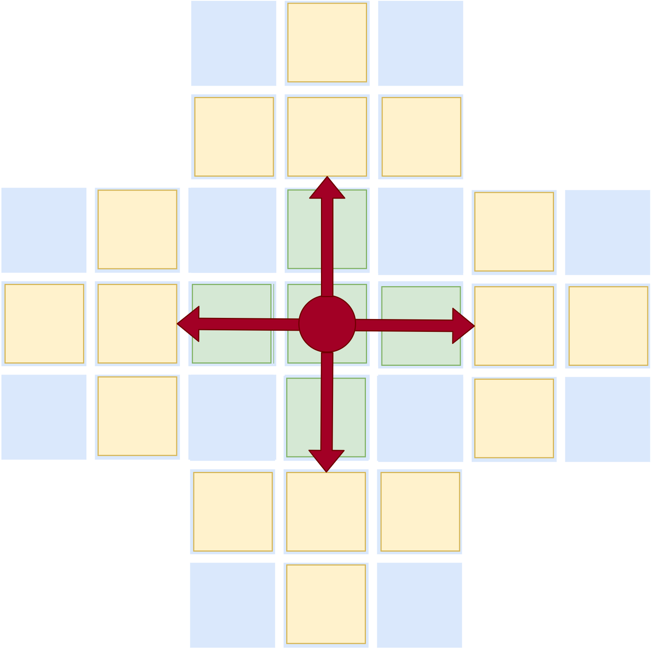


Figure 1. 2-D Von Neumann neighborhood with adjacent lattice sites. Velocity vectors superimposed.

**§2.5 Metastasis**

When cancer cells propagate, they also generate new blood vessels, which is called angiogenesis. The new blood vessels not only provide necessary chemicals and nutrients for tumor growth, it also provides the path for cancer cells to invade into the blood vessel (intravasation). If they survive the physical stress of blood flow and attacks from immune system, they may invade into a secondary site (extravasation) and metastasize (see Figure 2.). For the simulation of this particular form of metastasis, we had to be more specific of the cancer cell type, as the probability of metastasis to secondary sites differs for many primary tumors[[15]](#footnote-15). The probabilities we used in this simulation were derived from past literatures and are described in Table 1.

A picture containing text, map

Description automatically generated

Figure 2. The overview of the invasion-metastasis cascade.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Description | Value | Reference |
| Ps | Single Cancer cell survivability | 5 × 10-4 | Luzzi et al. (1998) |
| Pc | Cancer cell Cluster survivability | 2.5 × 10-2 | Luzzi et al. (1998),  Aceto et al. (2014) |
| E1 | Extravastation probability to bones | 0.5461 | Kuhn Laboratory (2017) |
| E2 | Extravastation probability to lungs | 0.2553 | Kuhn Laboratory (2017) |
| E3 | Extravastation probability to liver | 0.1986 | Kuhn Laboratory (2017) |

Table 1. Metastatic probabilities used in the simulation[[16]](#footnote-16)[[17]](#footnote-17)[[18]](#footnote-18)

**§2.5 Plotting**

Each generation of the updated boards was exported as .csv file, with x, y, coordinates and the state of the cell, in order to facilitate read files into *R*. The csv files exported from *Go* with appropriate numerical suffix, and data frame for each generation was generated. We used ggplot2[[19]](#footnote-19), due to the versatility the package provides in plotting. The PNG files generated in *R* was read back into *Go* for GIF generation.

**§2.6 Beyond 2 Dimensions**

The implementation of this simulation is highly modular, and the code easily lends itself to expansion. One such expansion that was implemented was an extension to three-dimensional space. That is, a three-dimensional lattice of cells was defined on a row, column, and “aisle” basis. All computational and logistical two-dimensional functions were then altered to accommodate the new spatial arrangement of the simulation.

**§2.7 Three-Dimensional Reactive Step**

This step is logically equivalent to that of the two-dimensional implementation (see §2.2), only that cell states at a given timestep are defined by a three-dimensional Von Neumann neighborhood (see Figure 2.)

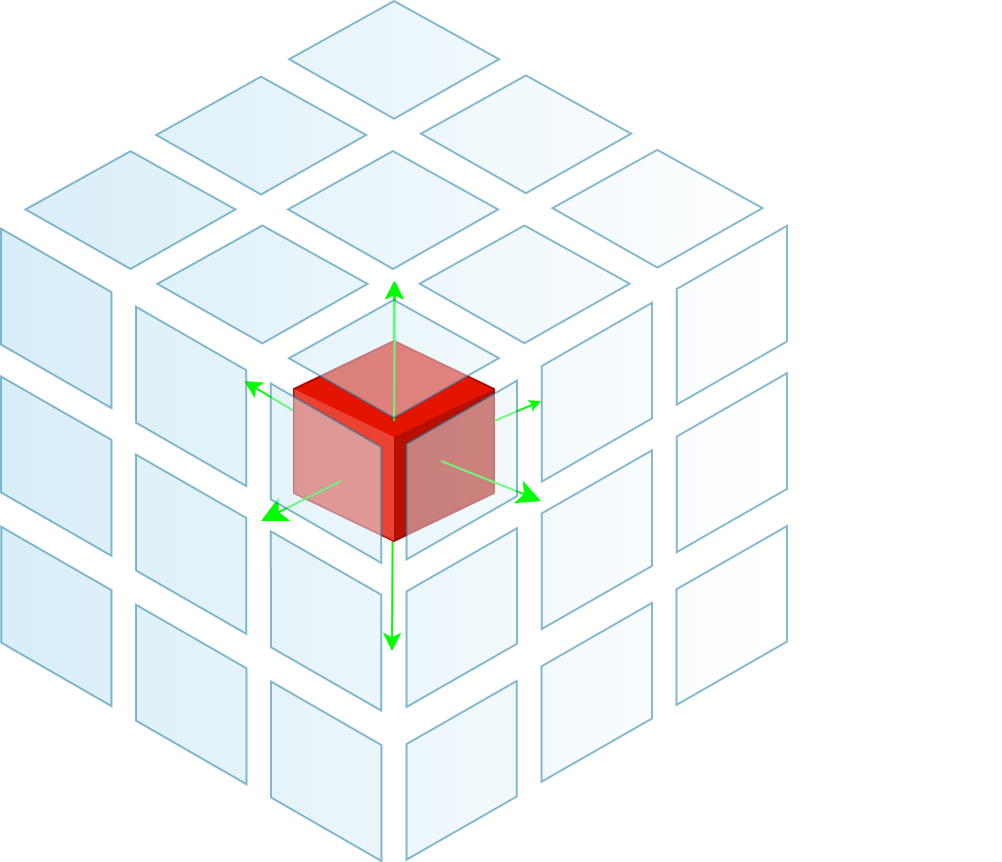


Figure 3. The 3-D Von Neumann arrangement, with velocity vectors of the center cell superimposed.

**§2.8 Propagation Step in Three Dimensions**

Cell movement and propagation are again equivalent logically to the two-dimensional implementation (see §2.3), only that the implementation required an extension into the “aisle” space, accommodating the extra dimension. Cell movement has two extra potential axes (see Figure 2.) for two additional discrete velocity vectors.

**§2.9 Plotting in Three Dimensions**

Similar to the two-dimensional plotting approach, a csv file for each iteration of updated “cube” was exported, with the additional column representing the “aisle”. A rgl package[[20]](#footnote-20) in *R* was used for visualization in 3D. Screenshots were captured to export the 3D image in PNG files, and the GIF was generated in *Go*.

**§3. Results**

**§3.1 Two-Dimensional Plot**

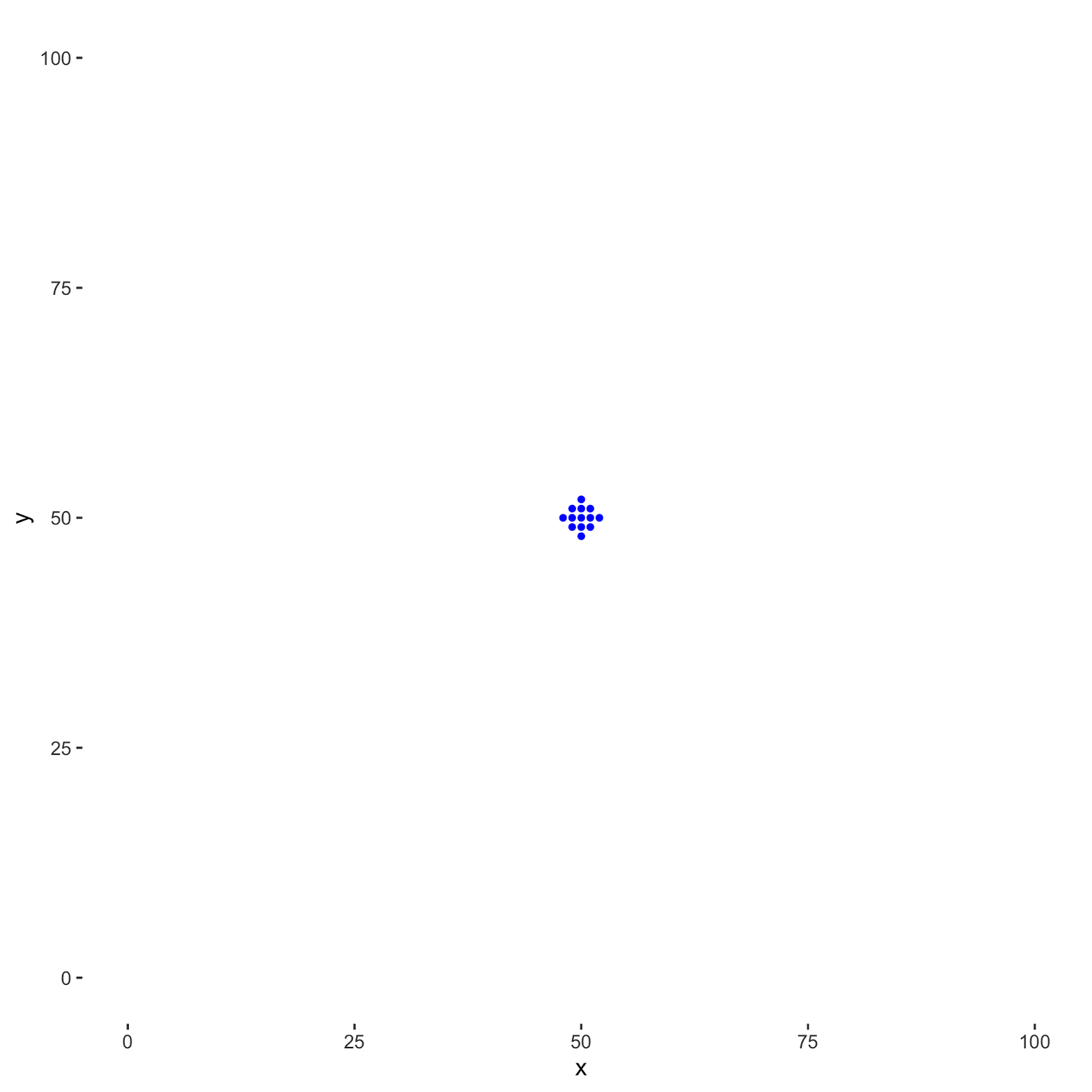


Figure 4. Initial two-dimensional lattice seeded with a cancerous central neighborhood and adjacent sites. Units in cells (pixels).

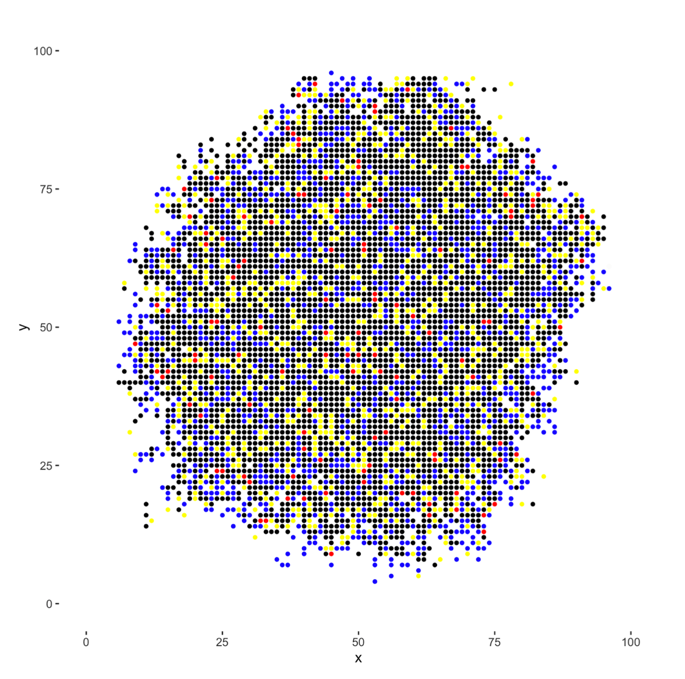


Figure 5. Lattice after 50 generations of growth.

**§3.2 Three-Dimensional Plot**

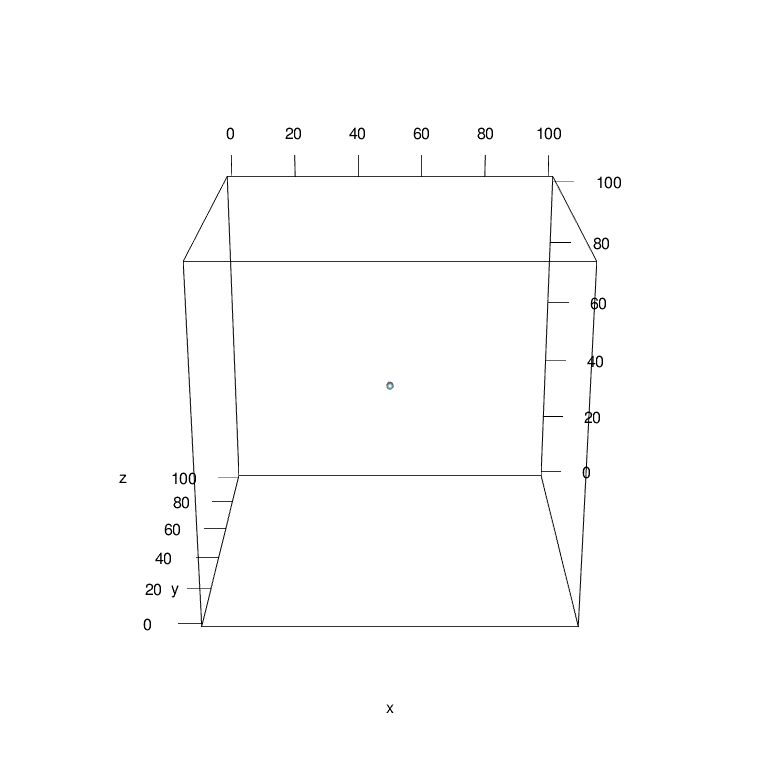


Figure 6. Three-dimensional lattice similarly seeded with cancerous cells.

A close up of a map

Description automatically generated

Figure 7. Three-dimensional lattice after 35 generations of the simulation (only cancerous cells plotted for simplicity).

**§3.3 Metastasis**

**A picture containing sky, water

Description automatically generated**

Figure 8. Arbitrarily placed ruptured vessel coordinates(red) for metastatic simulation

**A screenshot of a map

Description automatically generated**

Figure 9. Graph showing the number of metastasized cells when simulated for 3000 generations

**§3.3 Discussion**

The model was simulated on a two-dimensional lattice for 50 generations (see Figures 3,4.), and a three-dimensional model was simulated for 35 generations of growth (see Figures 5,6.). Additionally, for this simulation, coupling constant parameters were set to ; per the advice of previous two-dimensional investigation[[21]](#footnote-21).

Though the rules employed in these automata and metastasis simulation are somewhat simple compared to true cellular dynamics, nonetheless tumor “cells” are modeled somewhat realistically. Competition for resources (spatial constraints) here between healthy (background) cells and proliferative cancer cells is apparent, as is the complex interplay between cells of different states.

The ruptured vessels were arbitrarily placed (see Figure 8.) and metastasis was simulated on the two-dimensional lattice for 3000 generations (see Figure 9.)

**§4. Conclusions**

This implementation represents a successful, novel implementation of lattice-gas cellular automata as applied to principles of tumor growth. Furthermore, this implementation can be run in virtually infinitely many ways simply by altering existing constraints. Additionally, the model is sufficiently modular to allow for simple expansion, such as the addition of apoptotic cells, healthy cell behavior, additional coupling constraints, and more complex neighborhood geometry.

Some limitations of this model are apparent, however. The implementation is compute-intensive and memory-heavy; more complex investigations would need to better utilize principles of parallelism to remain practical. Additionally, it is unclear how a fundamentally bitmap-based model such as this would scale to much larger numbers of cells. Perhaps vector-drawn “cells”, with internal environments, would be apropos in a future implementation.

In addition, we could also improve this model more realistically by considering space steps and timesteps Δx, Δy, Δz, and Δt that correspond to real-world values in centimeters and seconds.

Future expansion upon this model could additionally incorporate real-time rendering rather than precomputation of lattices for plotting. In *Go,* this is conceivable via rendering into two images at alternate timepoints, and plotting one of them, such that one image is always being “painted” and one displayed. This could be further optimized via the use of the additional CPU cores (or a GPU) if local rendering is required.

In conclusion, this model, while simple, is an arguably powerful multifaceted tool for simulating cancerous cell growth *in silico*.

**§5. Works Cited**

Franssen, Linnea C., Tommaso Lorenzi, Andrew EF Burgess, and Mark AJ Chaplain. "A mathematical framework for modelling the metastatic spread of cancer." *Bulletin of mathematical biology* 81, no. 6 (2019): 1965-2010.

Doyle, Barry, Karol Miller, Adam Wittek, and Poul M.F. Nielsen, eds. *Computational Biomechanics for Medicine*. New York, NY: Springer New York, 2014. https://doi.org/10.1007/978-1-4939-0745-8.

Ghaemi, Mehrdad, and Amene Shahrokhi. “Combination of the Cellular Potts Model and Lattice Gas Cellular Automata for Simulating the Avascular Cancer Growth.” In *Cellular Automata*, edited by Samira El Yacoubi, Bastien Chopard, and Stefania Bandini, 297–303. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006.

“Glauber’s Dynamics | Bit-Player.” Accessed November 29, 2019. http://bit-player.org/2019/glaubers-dynamics.

Graner, null, and null Glazier. “Simulation of Biological Cell Sorting Using a Two-Dimensional Extended Potts Model.” *Physical Review Letters* 69, no. 13 (September 28, 1992): 2013–16. https://doi.org/10.1103/PhysRevLett.69.2013.

Roussos, Evanthia T., John S. Condeelis, and Antonia Patsialou. “Chemotaxis in Cancer.” *Nature Reviews. Cancer* 11, no. 8 (July 22, 2011): 573–87. https://doi.org/10.1038/nrc3078.

Wolf-Gladrow, Dieter A. *Lattice-Gas Cellular Automata and Lattice Boltzmann Models An Introduction*. 1st ed. 2000. Lecture Notes in Mathematics, 1725. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000. https://doi.org/10.1007/b72010.

Budczies, Jan, Moritz von Winterfeld, Frederick Klauschen, Michael Bockmayr, Jochen K. Lennerz, Carsten Denkert, Thomas Wolf et al. "The landscape of metastatic progression patterns across major human cancers." *Oncotarget* 6, no. 1 (2015): 570.

Luzzi, Keith J., Ian C. MacDonald, Eric E. Schmidt, Nancy Kerkvliet, Vincent L. Morris, Ann F. Chambers, and Alan C. Groom. "Multistep nature of metastatic inefficiency: dormancy of solitary cells after successful extravasation and limited survival of early micrometastases." *The American journal of pathology* 153, no. 3 (1998): 865-873.

Aceto, Nicola, Aditya Bardia, David T. Miyamoto, Maria C. Donaldson, Ben S. Wittner, Joel A. Spencer, Min Yu et al. "Circulating tumor cell clusters are oligoclonal precursors of breast cancer metastasis." *Cell* 158, no. 5 (2014): 1110-1122.

Kuhn Laboratory (2017) Breast Cancer Progression Models. Accessed December 5, 2019. http://kuhn.usc.edu/breast\_cancer/.

H. Wickham. ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York, 2016.

Adler, Daniel & Murdoch, Duncan. (2017). Rgl: 3D Visualization Using OpenGL.

Wolfram, Stephen. “Statistical Mechanics of Cellular Automata.” *Reviews of Modern Physics* 55, no. 3 (July 1, 1983): 601–44. https://doi.org/10.1103/RevModPhys.55.601.

1. Franssen, Linnea C., Tommaso Lorenzi, Andrew EF Burgess, and Mark AJ Chaplain. "A mathematical framework for modelling the metastatic spread of cancer." *Bulletin of mathematical biology* 81, no. 6 (2019): 1965-2010. [↑](#footnote-ref-1)
2. Stephen Wolfram, “Statistical Mechanics of Cellular Automata,” *Reviews of Modern Physics* 55, no. 3 (July 1, 1983): 601–44, https://doi.org/10.1103/RevModPhys.55.601. [↑](#footnote-ref-2)
3. This simulation has roots in the Ising Modelof ferromagnetic dipole moments as applied to the Glauber Algorithm, except here we consider +1 and -1 spin states as proportional to cancerous or necrotic cells. [↑](#footnote-ref-3)
4. “Glauber’s Dynamics | Bit-Player,” accessed November 29, 2019, http://bit-player.org/2019/glaubers-dynamics. [↑](#footnote-ref-4)
5. Dieter A. Wolf-Gladrow, *Lattice-Gas Cellular Automata and Lattice Boltzmann Models An Introduction*, 1st ed. 2000., Lecture Notes in Mathematics, 1725 (Berlin, Heidelberg: Springer Berlin Heidelberg, 2000), https://doi.org/10.1007/b72010. [↑](#footnote-ref-5)
6. Mehrdad Ghaemi and Amene Shahrokhi, “Combination of the Cellular Potts Model and Lattice Gas Cellular Automata for Simulating the Avascular Cancer Growth,” in *Cellular Automata*, ed. Samira El Yacoubi, Bastien Chopard, and Stefania Bandini (Berlin, Heidelberg: Springer Berlin Heidelberg, 2006), 297–303. [↑](#footnote-ref-6)
7. Graner and Glazier, “Simulation of Biological Cell Sorting Using a Two-Dimensional Extended Potts Model,” *Physical Review Letters* 69, no. 13 (September 28, 1992): 2013–16, https://doi.org/10.1103/PhysRevLett.69.2013. [↑](#footnote-ref-7)
8. That is, , where *i,j* are lattice sites, *J* is the boundary coefficient determining the adhesion between two cells of types τ(σ), τ(σ') (*C*,*N* in the current model), *σi* is the cell at site *i*, τ(σ) is the cell type of cell σ, J is a boundary coefficient determining the adhesion between two cells of types τ(σ),τ(σ'), δ is the Kronecker Delta, *v*(σ) the volume (area in two dimensions) function, *V*(σ) the target volume (area in two dimensions, but not applicable to the current model since spatial constraints per neighborhood are implicitly enforced), and λ a Lagrange multiplier to determine the optimized strength of the volume constraint. [↑](#footnote-ref-8)
9. Boltzmann factors as defined in literature are simplified in this model to ensure no 64-bit floating point issues given the magnitude of the constants involved. That is, , where *x* is cell state. [↑](#footnote-ref-9)
10. Barry Doyle et al., eds., *Computational Biomechanics for Medicine* (New York, NY: Springer New York, 2014), chap. 2, https://doi.org/10.1007/978-1-4939-0745-8. [↑](#footnote-ref-10)
11. Note that these computational steps are highly modular. Additional cell types and constraints are easily implemented at the level of code if desired. [↑](#footnote-ref-11)
12. Evanthia T. Roussos, John S. Condeelis, and Antonia Patsialou, “Chemotaxis in Cancer,” *Nature Reviews. Cancer* 11, no. 8 (July 22, 2011): 573–87, https://doi.org/10.1038/nrc3078. [↑](#footnote-ref-12)
13. A Von Neumann cellular automata neighborhood was chosen in this instance mainly due to memory constraints. An original implementation for this project used a strictly three-dimensional Moore neighborhood, for a total of 26 neighbors. Since we aimed to preserve prior lattice states for plotting, this quickly became impractical when computing on local machines.

    Additionally, a square or cubic Von-Neumann neighborhood (in two or three dimensions) obeys the Platonic solid principle, such that lattice vectors are all of equal length (an important assumption imputed into the cell behavior). Other Platonic-solid neighborhoods include octahedral, dodecahedral, and icosahedral adjacency vectors (the modularity of this model would easily allow a future implementation to make use of these more complex neighborhoods). [↑](#footnote-ref-13)
14. Wolf-Gladrow, *Lattice-Gas Cellular Automata and Lattice Boltzmann Models An Introduction*, 106. [↑](#footnote-ref-14)
15. Budczies, Jan, Moritz von Winterfeld, Frederick Klauschen, Michael Bockmayr, Jochen K. Lennerz, Carsten Denkert, Thomas Wolf et al. "The landscape of metastatic progression patterns across major human cancers." *Oncotarget* 6, no. 1 (2015): 570. [↑](#footnote-ref-15)
16. Luzzi, Keith J., Ian C. MacDonald, Eric E. Schmidt, Nancy Kerkvliet, Vincent L. Morris, Ann F. Chambers, and Alan C. Groom. "Multistep nature of metastatic inefficiency: dormancy of solitary cells after successful extravasation and limited survival of early micrometastases." *The American journal of pathology* 153, no. 3 (1998): 865-873. [↑](#footnote-ref-16)
17. Aceto, Nicola, Aditya Bardia, David T. Miyamoto, Maria C. Donaldson, Ben S. Wittner, Joel A. Spencer, Min Yu et al. "Circulating tumor cell clusters are oligoclonal precursors of breast cancer metastasis." *Cell* 158, no. 5 (2014): 1110-1122. [↑](#footnote-ref-17)
18. Kuhn Laboratory (2017) Breast Cancer Progression Models. http://kuhn.usc.edu/breast\_cancer/. [↑](#footnote-ref-18)
19. H. Wickham. ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York, 2016. [↑](#footnote-ref-19)
20. Adler, Daniel & Murdoch, Duncan. (2017). Rgl: 3D Visualization Using OpenGL. [↑](#footnote-ref-20)
21. Doyle et al., *Computational Biomechanics for Medicine*, chap. 2. [↑](#footnote-ref-21)