

Bacteria Growth & Survival Modeling Using Cellular Automata

Adeline Southard

1 Introduction

In this project, I introduce and further explore a cellular automata model to represent a competitive bacteria-food environment that simulates bacteria growth behavior in a space with a limited, albeit replenishing, food source. Originally a smaller project meant only to create the model itself, here I take a more thorough look at the system to better understand its survival/death behavior as a function of system food growth, while using a complexity measure to understand this. I take a closer look at how this behavior is influenced by different types of cellular automata neighborhoods and the introduction of randomness into an originally deterministic system.

2 Research Questions

This project investigates four main research questions concerning the competitive bacteria-food model:

1. How can I measure the spatial complexity of the bacteria growth system (how complex is it, and how can I characterize this)?
2. How does the total population of bacteria relate to food growth rate? Does the total bacteria population stabilize over time and survive, or does it die out?
3. As food growth rate increases, where does the transition from bacteria colony death to survival happen?
4. Does the complex behavior observed (during mini-project) occur with other types of cellular automata neighborhoods (von Neumann and Moore)?

Because research question one is associated with the methods used to measure complexity in the system, it is covered in the methods section of this report. Research questions two, three, and four are covered in the results section. While not one of the originally proposed research questions, the addition of randomness is also briefly discussed.

3 Methods

3.1 Cellular Automata System Rules

For this project I seek to model a two state system of bacteria and food in a cellular grid in two dimensions ($n \times n$ simulation space). In each distinct cell, a count for bacteria and food is kept. I use three major sets of rules to govern the system and how it changes over time: bacteria growth rules, food growth and depletion rules, and bacteria movement rules. These rules are deterministic and, hence, the simulations are reproducible.

3.1.1 Bacteria Growth Rules

If there is no food in a cell, all bacteria in that cell die. If there is sufficient food, then bacteria in the cell increase by a factor of `bacteria_growth_rate` (1-to-1 ratio of food to bacteria to reproduce a bacterium). If there is insufficient food to support a `bacteria_growth_rate` factor of increase, the bacteria count is increased by the amount of food in the cell (1-to-1 ratio).

3.1.2 Food Growth and Depletion Rules

Each cell has an initial food ‘count’. On each timestep, the food count is increased by a constant amount `food_growth` that is specified as a hyperparameter. The amount of food in each cell is limited to a `max_food` value that is specified as a hyperparameter. In each iteration of the simulation, the food count is decreased by the number of bacteria in a cell (1-to-1 ratio; one bacterium ‘eats’ one count of food). The amount of food cannot be negative (it is set to zero if bacteria exceeds food).

3.1.3 Bacteria Movement Rules

The amount of bacteria available to move out of each cell is the `bacteria_move_rate` times the amount of bacteria in each cell. To determine where each bacteria will move, I first calculate the excess amount of food in each cell (i.e. food - bacteria), then calculate the change in excess food when moving to each available cell using a matrix roll; moves where excess food increases are desirable, and a greater increase in excess food is more desirable than a lesser increase. The number of bacteria that move in each direction is proportional to the excess food increase in that direction. Moves in directions of zero or decreasing excess food do not occur.

3.2 Measuring the Complexity of the System

There are many methods to measure complexity [1]. For my purposes I wanted two specific measures of complexity:

1. A measure of the complexity at each time step of the simulation to see how the complexity of the system changes over time.
2. A measure of the overall complexity of a simulation to enable comparing different simulations with different parameters to each other.

After a limited amount of research, I chose to use Shannon Entropy as a measure of complexity in both these cases to keep things simple. Comparing the many measures of complexity in the literature could quickly become a full project (and then some) in itself!

Shannon Entropy is based on the information theory of Claude Shannon [2]. This measure of complexity is based on the idea that the amount of information needed to describe outcomes of a model provides a measure of how complex it is. In the context of my cellular automata simulation, the more information that is needed to describe the two states of the system, bacteria and food, the more complex it is. To define this measure of complexity we first need to describe what information is.

Shannon’s measure of information has the following properties:

1. Events that occur with high probability have low information. Events that occur with certainty provide no (zero) information because they are fully expected. For example, if the dynamics of a system result in the same thing happening every time, then it takes no information to describe the system.
2. Events that occur with low probability have high information. For example, if an event only occurs once in a million trials, it provides more information than an event that occurs in every trial since many trials are needed to observe the rare event and hence more information is needed to describe the system with rare events.
3. The information of two independent events is the sum of the information of the two events. For example, if a coin is flipped twice, the information of the outcome of the two flips is the sum of the information of the outcome of the first flip and the information of the outcome of the second flip. Finding out that a coin toss comes up twice as heads conveys twice as much information about the coin flipping system than finding out that a coin toss comes up heads once. Said differently, the more coin flips we have the more information we have to describe the coin flipping system (how fair or unfair the coin flipping system is, for example).

To satisfy property 1 and 2, we see that the inverse of the probability of an event x is a reasonable measure of information:

$$I(x) \sim \frac{1}{P(x)}$$

where $I(x)$ is the information of event x and $P(x)$ is the probability of event x occurring ($P(x)$ is the probability mass function of the random variable x).

This does not work as a definition of information though since it does not satisfy property 3. However, the logarithm of the inverse of the probability turns out to satisfy property 3, and because the logarithm is a monotonic function, it satisfies properties 1 and 2 as well. So the information of an event x is defined as:

$$I(x) := \log\left(\frac{1}{P(x)}\right) = -\log(P(x))$$

To see how this satisfies property 3, consider two independent events x_1 and x_2 that occur with probability $P(x_1)$ and $P(x_2)$. Because the events are independent, the probability of both events occurring is just the product of the probabilities of the two events:

$$P(x_1 \wedge x_2) = P(x_1)P(x_2)$$

By definition the information of the two events is:

$$I(x_1 \wedge x_2) = -\log(P(x_1 \wedge x_2)) = -\log(P(x_1)P(x_2)) = -\log(P(x_1)) + -\log(P(x_2))$$

Since the logarithm of a product is the sum of the logarithms of the factors, we see that we get the sum of the information of the two events, and hence satisfy property 3:

$$I(x_1 \wedge x_2) = I(x_1) + I(x_2)$$

Now that we have a definition of information, we can use it to define Shannon Entropy. Shannon Entropy is the expected value of the information of a random variable x with probability mass distribution $P(x)$:

$$H(x) = E_{x \sim P}[I(x)] = - \sum_{x \in X} P(x) \log(P(x))$$

where $H(x)$ is the Shannon Entropy of the random variable x with probability mass distribution $P(x)$. Note that the sum is over all possible outcomes of the random variable x , which is denoted by $x \in X$ where X is the set of all possible outcomes $X = \{x_1, x_2, \dots, x_n\}$ (in our case this will be histogram bins). However for a continuous random variable, the probability mass function is replaced with a probability density function and the sum is replaced by an integral (this is called differential entropy):

$$H(x) = - \int_{x \in X} p(x) \log(p(x)) dx$$

For my calculation purposes I will be using the discrete version of Shannon Entropy, where I create a histogram of the outcomes of the random variable (i.e. bacteria or food) into a finite number of bins and estimate the probability mass function as the probability of an outcome falling into a particular bin. For example, for a particular time step of the simulation, I can flatten the grid values into a single vector, quantize the bacteria population (or food amount) to a specified bin width and estimate the probability for each occurrence falling in a bin using the frequency of occurrences that fall in each bin (each bin corresponds

to the finite set of outcomes X). I can then use this estimated probability mass function to calculate the Shannon Entropy of the bacteria population (or food amount) at that time step.

Likewise, for the entire simulation I can calculate the Shannon Entropy of the bacteria population (or food amount) by just flattening the entire history of the bacteria population grids (or food amounts), over all time steps, into a single vector and calculate the Shannon Entropy.

If a base 2 logarithm is used, it can be shown that the Shannon Entropy gives a lower bound on the average number of bits that are needed to encode the symbols (e.g. bin number) of a random variable from a given probability distribution. So, the Shannon Entropy can be thought of as a measure of the minimum number of bits needed to describe the state of a system. If a logarithm with a different base is used, then the Shannon Entropy is measured in units of the base of the logarithm (e.g. nats for the natural logarithm).

Since I am using a histogram to estimate the probability mass function for the amount of bacteria (or food) that fall into a finite number of bins, the question arises what bin width should be used? I use Scott's rule [3] to estimate the optimal number of bins for a histogram. Scott's rule is based on the idea of minimizing the mean integrated square error of the histogram estimate of the probability density function. Assuming the data is normally distributed the rule is:

$$h = 2 \times 3^{\frac{1}{3}} \pi^{\frac{1}{3}} \frac{\sigma}{n^{1/3}} \approx \frac{3.49\sigma}{n^{1/3}}$$

where h is the optimal width of the bins, σ is the standard deviation of the data, and n is the number of data points. While this version of Scott's rule considers the data to be normally distributed, it is often used as a rule of thumb for other distributions as well [3, section 5], and I will use it as such here with the standard deviation calculated from the data.

3.3 Using Complexity to Characterize Bacteria Survivability

During the initial experiments of my bacteria-food cellular automata system, I observed behavior where the bacteria died out. This was expected for a food growth factor of 0 where the bacteria would exhaust all food in the simulation space and then die, but I *also* observed this same behavior even when there was a finite amount of food growth: the bacteria would grow outward in a wave, not moving back because of unfavorable conditions and ultimately dying out despite food regrowth.

In an effort to explore this behavior further, I use Shannon Entropy (my complexity measure) as an indicator of when this happens. This is because systems where bacteria populations grow outward and die out display lower complexity over time and, at the final timestep (assuming the simulation runs long enough), zero Shannon Entropy as there is no variation of bacteria population on the grid.

While rendering animations of the simulations takes a long time, running a simulation does not. This means that I can sequentially run many simulations with the same hyperparameters, varying the food growth and plotting simulation complexity as a function of the food growth rate to get a better idea of where this behavior is occurring.

3.4 Parametric Study

Because the observations that motivated this project were focused on behavior related to a change in the food regrowth factor of the system, I needed to do a parametric study to gain insight into this behavior. Thus, I fix all other hyperparameters, change food growth rate, and run many simulations to get a better look at what is going on. Reproduction rate is fixed at 2 since bacteria are assumed to only divide in half. Bacteria movement rate is fixed at 50% so that half of bacteria move out from their current cells into neighboring cells in each timestep. Max food is fixed at 100. All cells are initialized with a food count of 10. Finally, the initial bacteria population is always a single bacterium in the center cell of the simulation grid.

4 Results

4.1 How does bacteria population relate to food growth rate?

My first research question for this project was how the total population of bacteria relates to food growth rate. Does the behavior of the population over time change significantly as food growth rate is altered? Furthermore, how does this behavior manifest when the population fails to survive (the observation that motivated this project)?

To accomplish this, I plot the mean and standard deviation of the bacteria and food populations at each timestep. Using Shannon Entropy as discussed in the methods section, I also investigate how the complexity of the system changes over time and in turn use it as a method of observing population survivability.

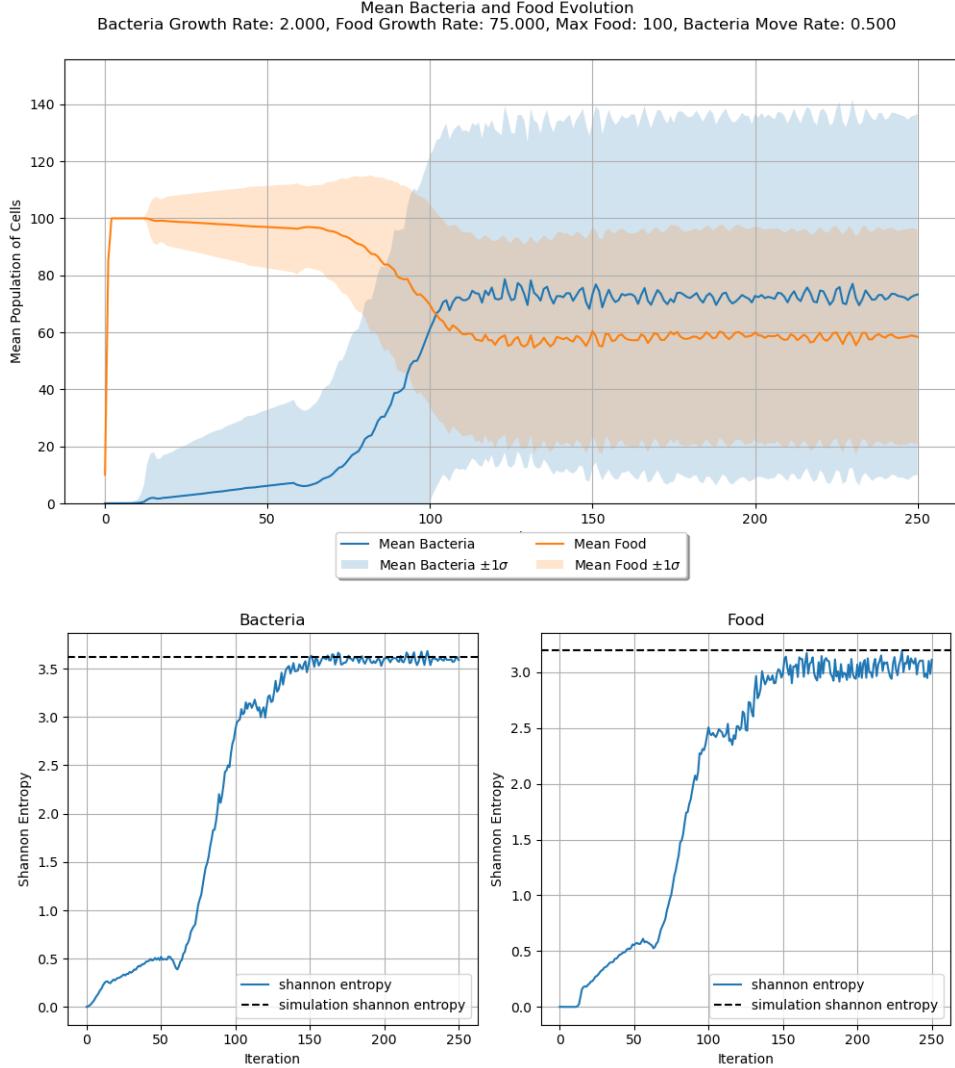


Figure 1. Bacteria and food mean cell populations and Shannon Entropy at each timestep. Shannon Entropy for the entire simulation is also shown as a dashed line. Bacteria growth rate was fixed at 2, food growth rate at 75, max food at 100, and bacteria move rate at 0.5. von Neumann neighborhood was used for bacteria movement.

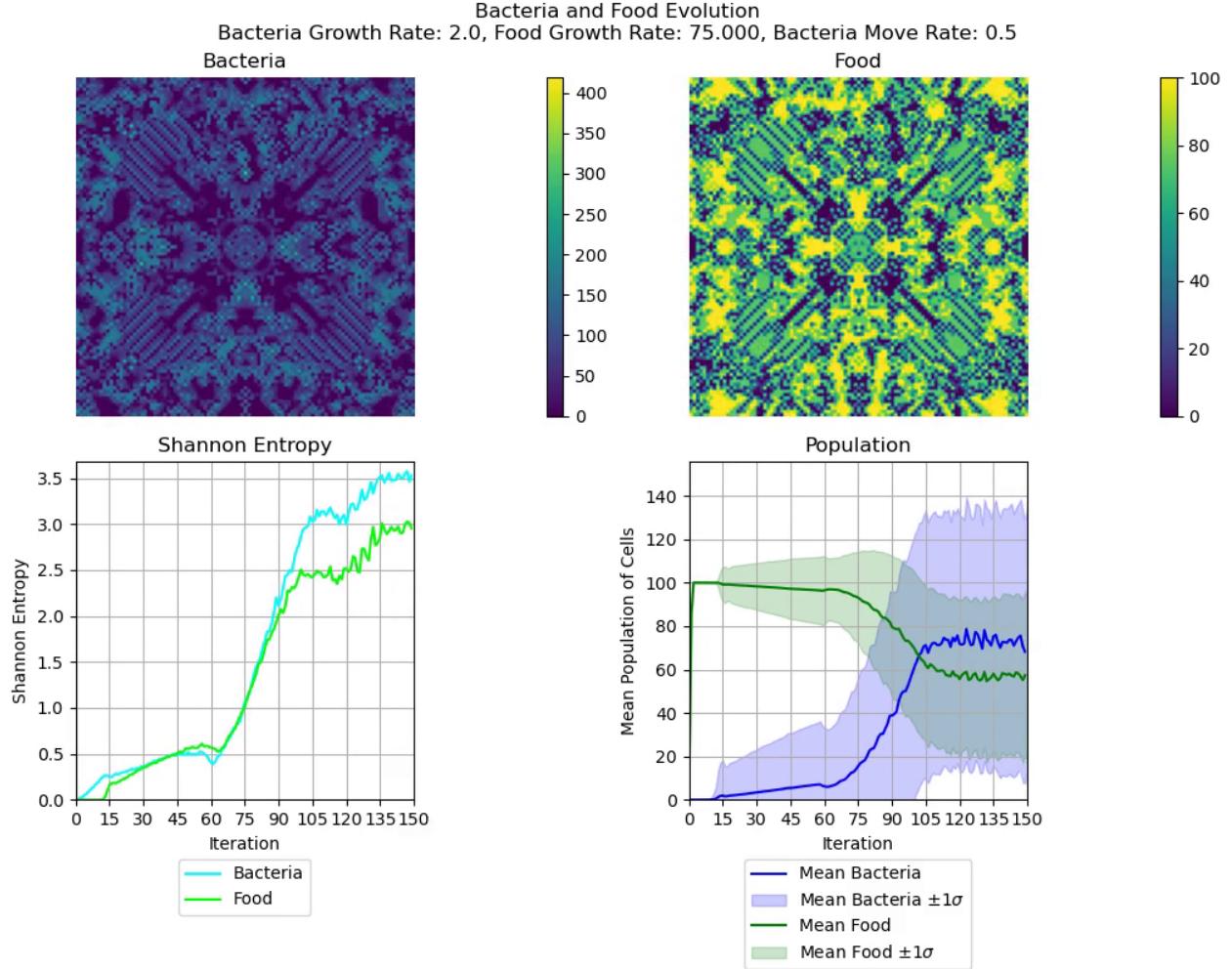


Figure 2. Animation of the population spatial distribution, the accompanying population statistics, and Shannon Entropy at each timestep of the simulation. Bacteria growth rate was fixed at 2, food growth rate at 75, max food at 100, and bacteria move rate at 0.5. von Neumann neighborhood was used for bacteria movement.

Figures 1 and 2 show results for the following: Bacteria growth rate: 2, food growth rate: 75, max food: 100, bacteria move rate: 0.5. As shown, the bacteria survives indefinitely with these parameters. Because of the high food growth, there is a steep increase from the starting food of 10 in each cell to the maximum of 100. The singular initial bacterium in the center of the simulation space grows in an outward wave as the population increases. Eventually, the food regrowth can only sustain a population of so many bacteria, so the mean bacteria population reaches a point of relative stability (as does the food population). Note that, in this population survival case, the standard deviation about the mean population is very high. This indicates that there are significant cell-to-cell population differences. As evidenced by the simulation animation in *Figure 2*, once the mean cell population reaches equilibrium, the bacteria in more bacteria-dense cells must regularly move from them into less bacteria-dense cells where food is regrowing. This creates a flux in the density of bacteria across cells despite the eventual bacteria and food populations reaching stability.

The Shannon Entropy provides support that my hypothesis of surviving populations having increasing complexity is true. As the bacteria population grows (and eventually must compete with itself for the limited food regrowth), the Shannon Entropy also grows. In addition, when the bacteria and food mean populations plateau, so does the Shannon Entropy.

This answers part of my second research question. As I originally hypothesized, the bacteria and food popu-

lations do eventually stabilize. This was already slightly evidenced by my original simulation visualizations, but measuring mean population sizes over time gives a more concrete measure of this.

Knowing that surviving cell populations do reach stability (in the sense that the mean cell population remains relatively constant), it's necessary to explore the other side of the second research question: how does the food growth factor affect this stability, and in addition the survivability of the bacteria population?

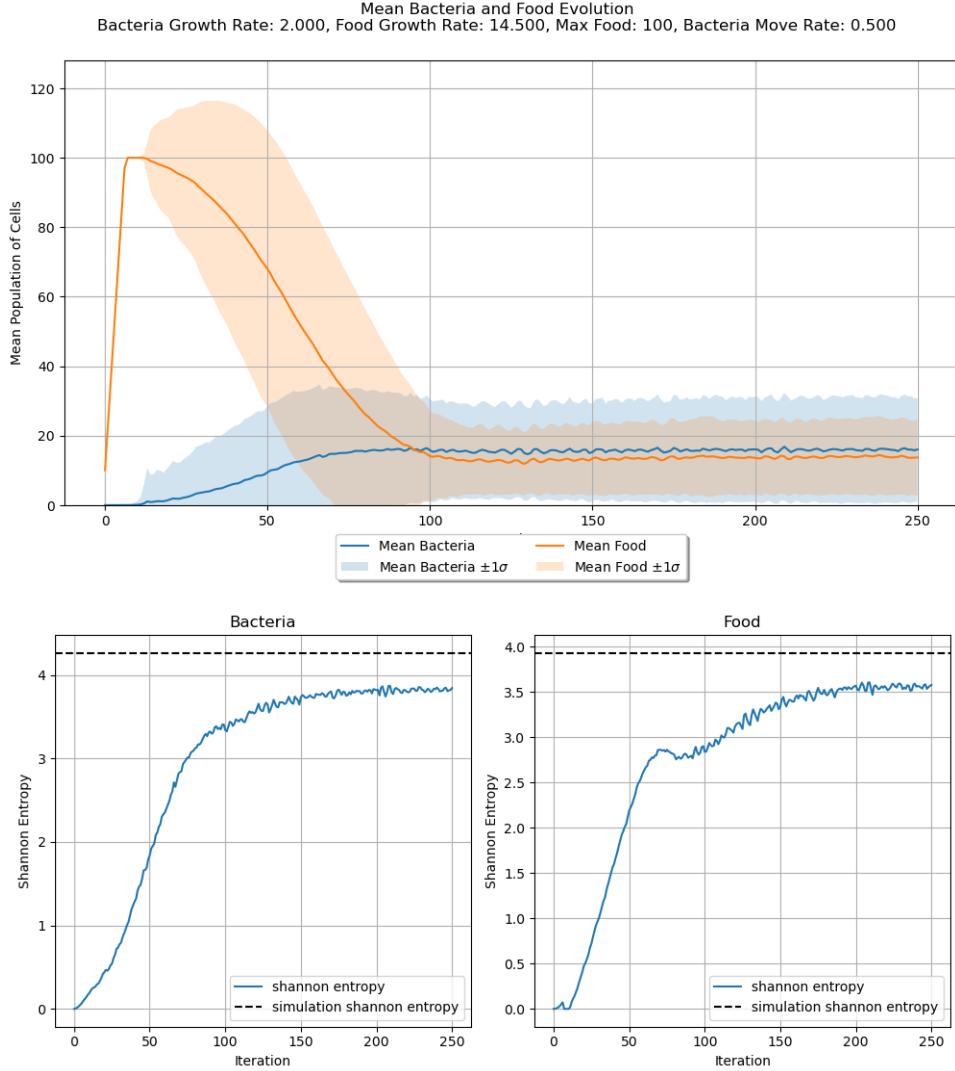


Figure 3. Bacteria and food mean cell populations and Shannon Entropy at each timestep. Shannon Entropy for the entire simulation is also shown as a dashed line. Bacteria growth rate was fixed at 2, food growth rate at 14.5, max food at 100, and bacteria move rate at 0.5. von Neumann neighborhood was used for bacteria movement.

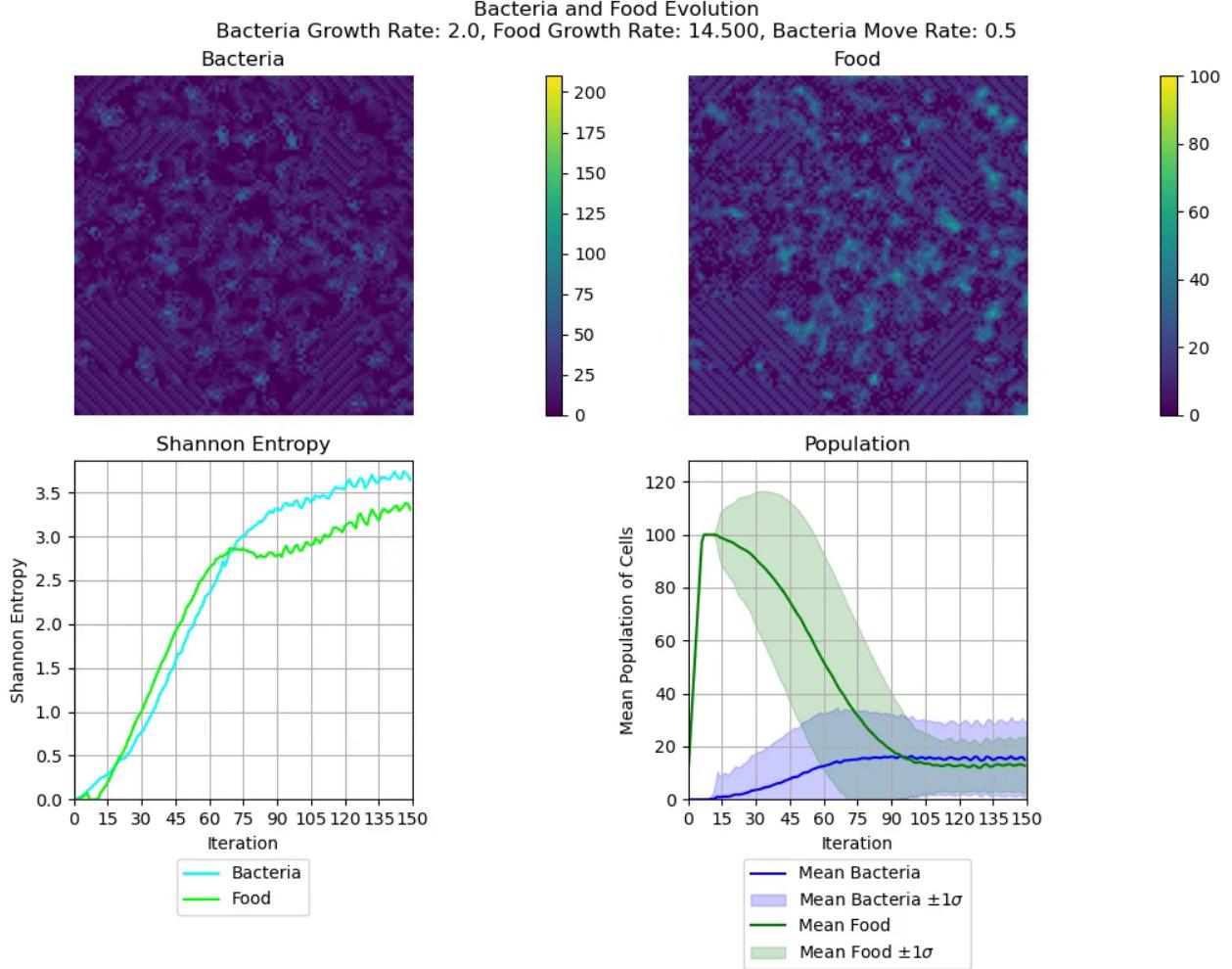


Figure 4. Animation of the population simulation, the accompanying population sizes, and Shannon Entropy at each timestep for the simulation. Bacteria growth rate was fixed at 2, food growth rate at 14.5, max food at 100, and bacteria move rate at 0.5. von Neumann neighborhood was used for bacteria movement.

In this experiment, I run a simulation with the same hyperparameters, *except* I use a dramatically different food growth factor of 14.5 instead of the previous 75. In this simulation, the mean cell population plateaus at a smaller population size than when the food growth factor was higher. This makes sense, since the food growth rate is much lower, it is unable to sustain as high a population of bacteria (see upper plots in Figures 1 and 3).

In this experiment, the Shannon Entropy plateaus for both the bacteria and food populations are greater than in the prior experiment with a much higher food growth factor (see Figures 1 and 3). The plateaus in this experiment also occur later than in the experiment with a much higher food growth factor (see lower plots in Figures 1 and 3). Perhaps with the greater restriction in food, there is more competition between bacteria in the simulation, and this results in more system complexity despite the bacteria populations being relatively smaller.

This answers the question of whether food growth affects the population size of the simulation. Food growth factor does have a significant impact, but in both of these experiments the population survives. Next, using the mean population and Shannon Entropy measures, I look at a case where the population does not survive.

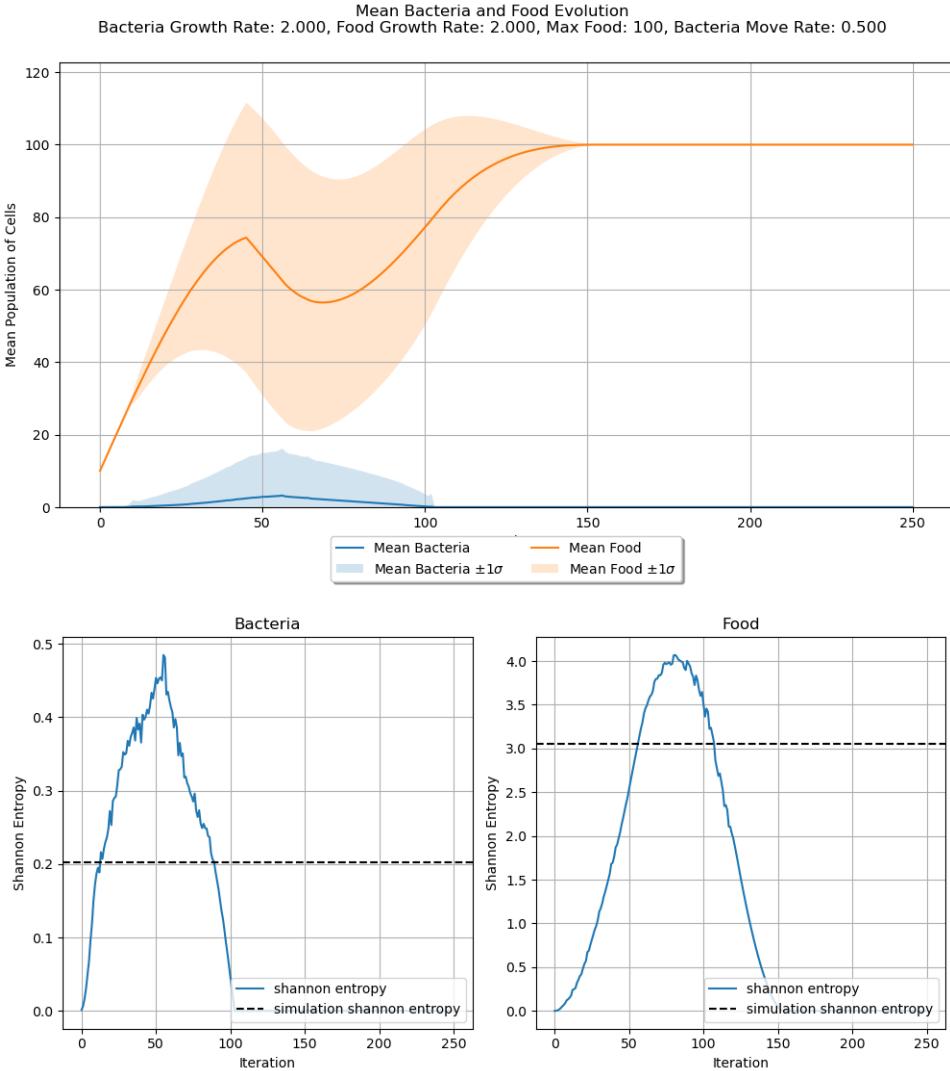


Figure 5. Bacteria and food mean cell populations and Shannon Entropy over each timestep. Shannon Entropy for the entire simulation is also shown as a dashed line. Bacteria growth rate was fixed at 2, food growth rate at 2, max food at 100, and bacteria move rate at 0.5. Note that this bacteria population does not survive. von Neumann neighborhood was used for bacteria movement.

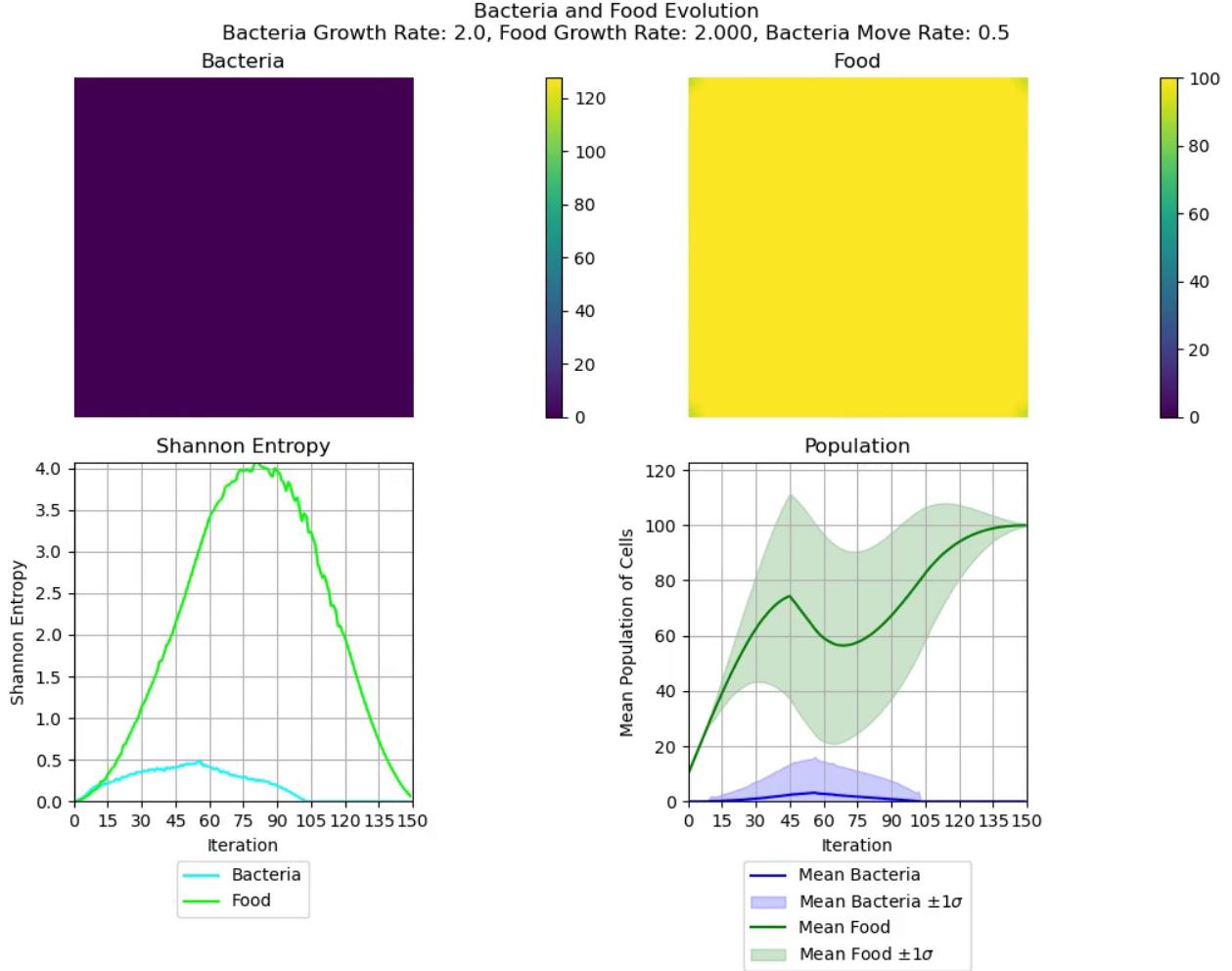


Figure 6. Animation of the population spatial distribution, the accompanying population statistics, and Shannon Entropy at each timestep of the simulation. Bacteria growth rate was fixed at 2, food growth rate at 2, max food at 100, and bacteria move rate at 0.5. Note that this population does not survive. von Neumann neighborhood was used for bacteria movement.

In this simulation, the bacteria grows outward in a wave but ultimately fails to grow behind the wave, dying out. This is interesting in that the food growth rate is non-zero, but nonetheless the bacteria population completely dies out.

Here, it is clear why Shannon Entropy can double as a measure of population survivability as well as complexity. When a bacteria population fully dies out, all cells in the simulation space drop to 0 bacteria, and thus the Shannon Entropy also drops to 0. Likewise, food eventually reaches a constant value of the max food (because all of the bacteria have died off), and hence the Shannon Entropy for it also becomes 0. This is helpful for the next part of the project, where having a simple method of seeing whether the population survived is valuable. There are certainly other, perhaps simpler methods of seeing whether a population survives, but using Shannon Entropy means that we can also get an idea of how the population behaved in terms of complexity *before* it died (which is something a basic sum of bacteria and food in each cell would not be able to capture). In this way, it serves two purposes.

4.2 Where does the transition from population survival to death happen?

Using mean population sizes and Shannon Entropy provides us information on how the system behaves as well as the complexity of the system, but the main motivation for this project was to explore where and why some

bacteria populations die out. My original hypothesis was that this process would happen gradually as food growth rate lessens (populations becoming less and less sustainable), but the results suggested something different.

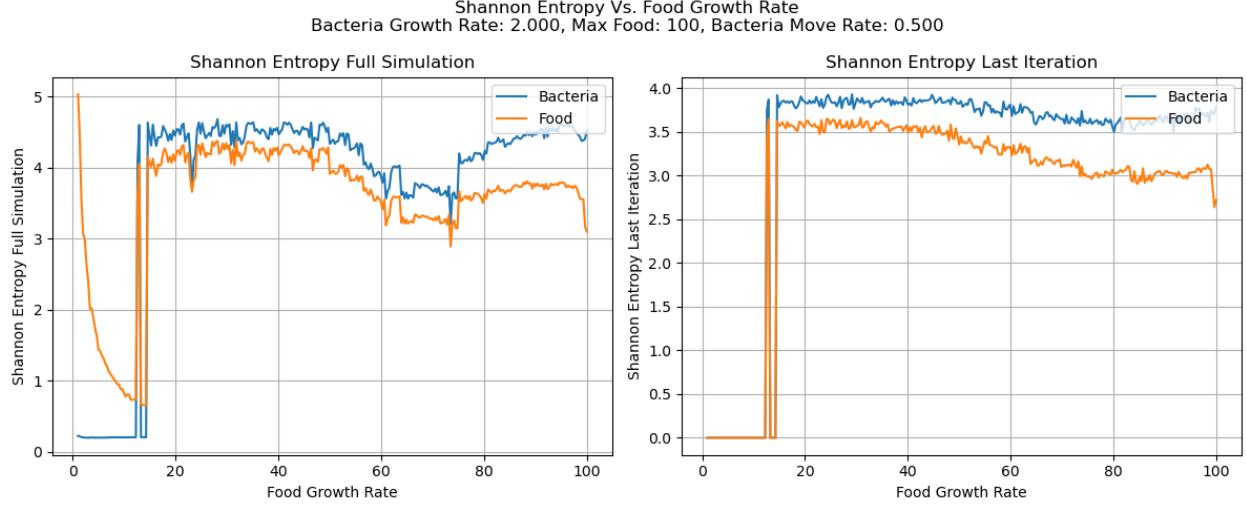


Figure 7. Shannon Entropy results for full simulation (left) and last timestep only (right) of 300 simulations. Food growth rate is varied linearly for each simulation from 0 to 100. Bacteria growth rate was fixed at 2, max food at 100, and bacteria move rate at 0.5. von Neumann neighborhood was used for bacteria movement.

As discussed prior, because Shannon Entropy provides a measure of whether the bacteria population survived or not, I use it here to get a rough idea of how the simulation behaved. This is most clear in the right plot, as the last iteration of the simulation will have a Shannon Entropy of 0 if the bacteria population died out and all food grew back to its maximum value.

As I had previously expected, at lower food growth rates the population fails to survive. However, contrary to my hypothesis, there is no gradual shift in Shannon Entropy from populations that die to ones that survive. As evident in the plots, populations with a lower food growth rate die out, but once the food growth rate becomes high enough, the population is immediately able to survive. In fact, as will continue to be clear later, the change in food growth rate needed to make this shift is incredibly sensitive. This answers the question of how food growth affects populations as it changes.

More notably, though (and unexpected), is the odd behavior right before the simulations make the jump from population death to survival. It appears as if populations begin surviving as food growth increases, but then go *back* to dying. To further investigate this behavior, I ran a more focused experiment of varying food growth rate, but only on this area of interest.

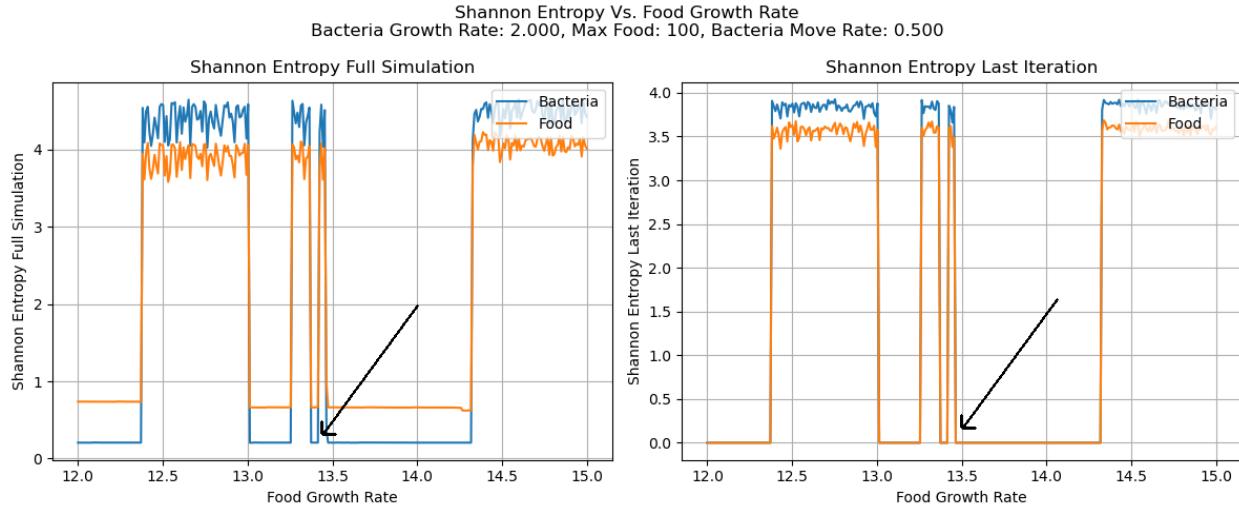


Figure 8. Shannon Entropy results for full simulation (left) and last timestep only (right) of 300 simulations. Food growth factor is varied linearly for each simulation from 12 to 15. Bacteria growth rate was fixed at 2, max food at 100, and bacteria move rate at 0.5. von Neumann neighborhood was used for bacteria movement. Note the sudden jumps in Shannon Entropy (and, by extension, population survivability) that indicate sudden changes in population behavior as a result of small changes in food growth rate.

In these plots, this odd behavior is more obvious. From around food growth rates of 12 to 15, there are multiple shifts from populations dying to being able to survive, and back. Eventually the populations continue being able to survive as food growth further increases, but this peculiar behavior is interesting because, intuitively, I would think that an increase in food growth rate should mean an increase in survivability. This is generally true, but this handful of exceptions break that rule, at least for the other hyperparameters used here.

Though answering my original research questions, this system behavior raises more questions than it answers! It seems that something in the deterministic nature of the system with certain values of food regrowth are prohibiting populations from properly growing.

After calculating the specific points at which these jumps in survivability occur to machine-level precision, I next look at two specific simulation cases right around where this jump in survivability occurs to get a better idea of the actual system behavior at these points.

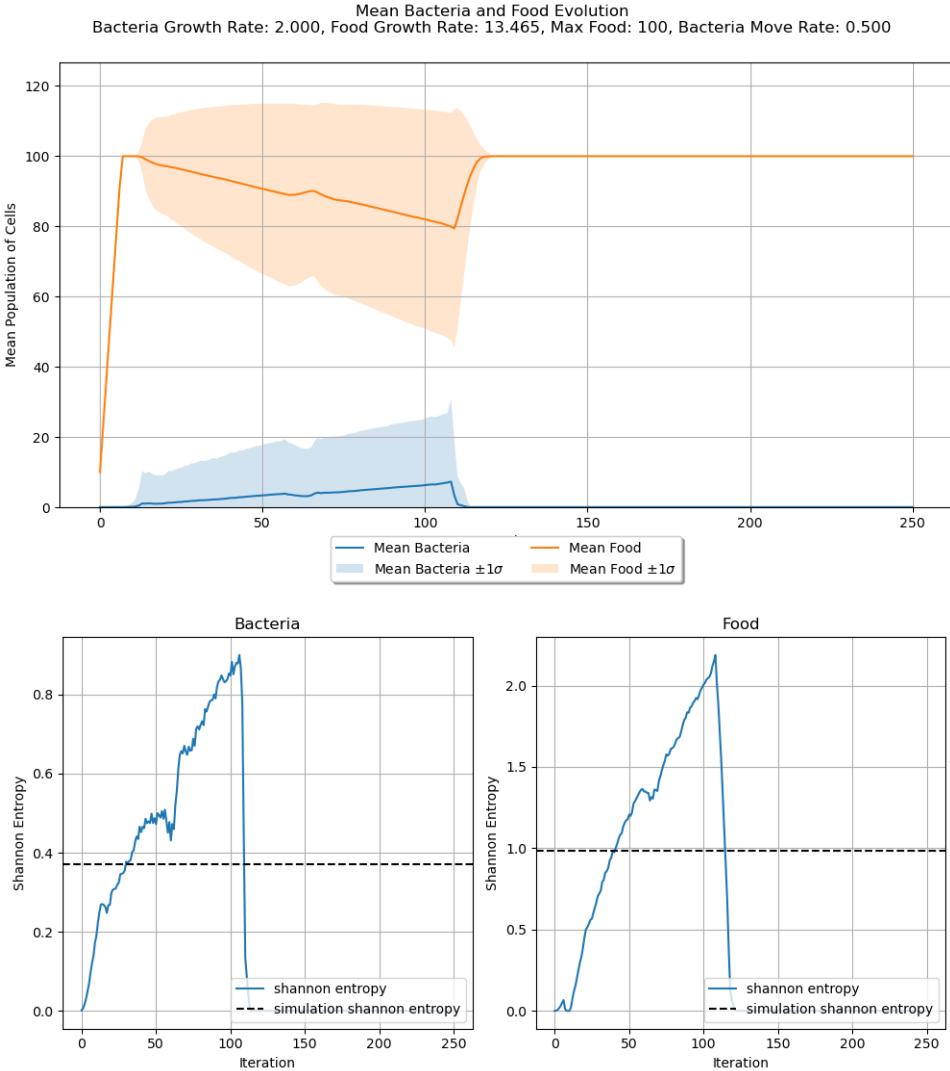


Figure 9. Bacteria and food mean cell populations and Shannon Entropy at each timestep in the simulation. Shannon Entropy for the entire simulation is also shown as a dashed line. Bacteria growth rate was fixed at 2, food growth rate at approximately 13.465, max food at 100, and bacteria move rate at 0.5. Note that this bacteria population does not survive. von Neumann neighborhood was used for bacteria movement.

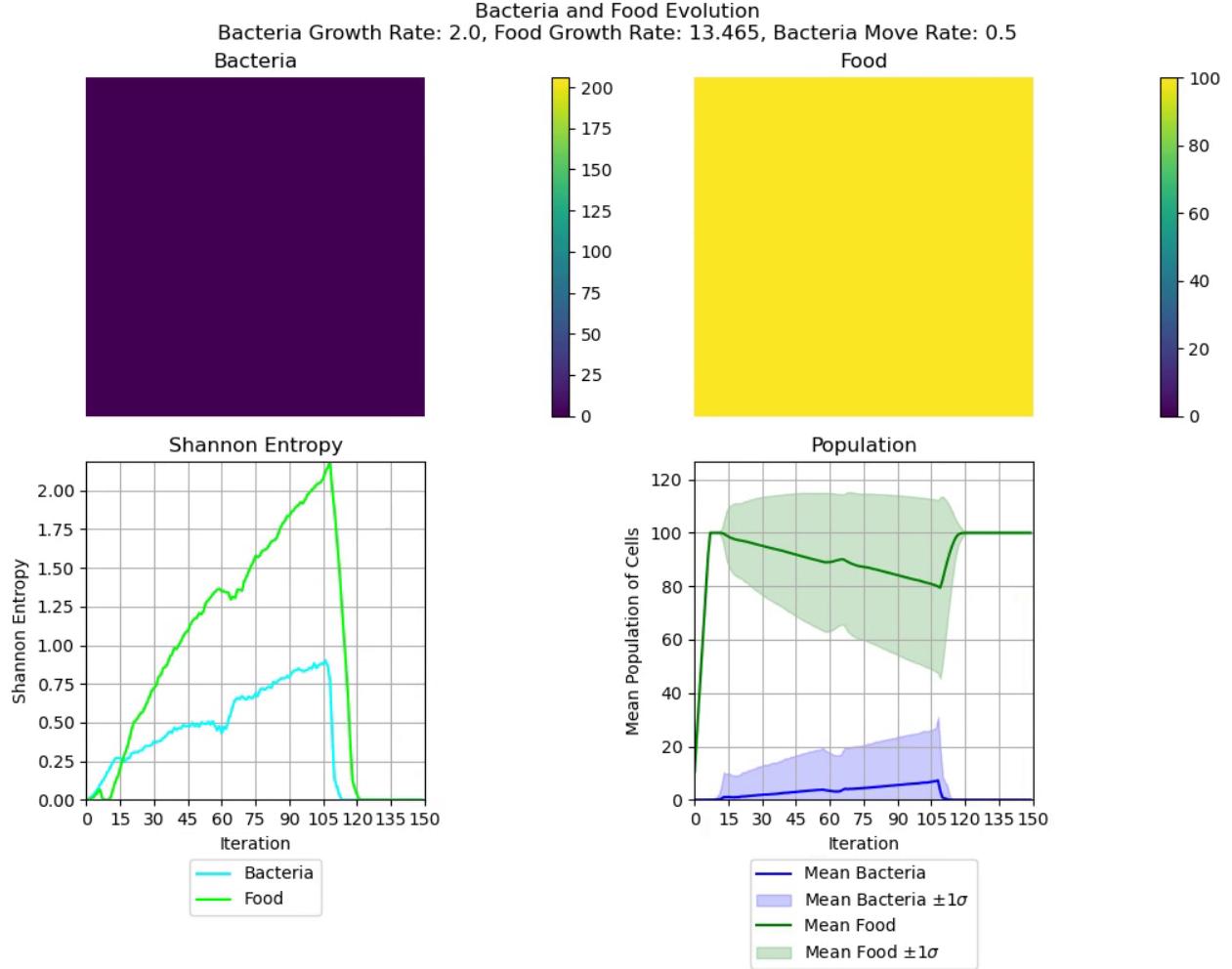


Figure 10. Animation of the population spatial distribution, the accompanying population statistics, and Shannon Entropy at each timestep of the simulation. Bacteria growth rate was fixed at 2, food growth rate at approximately 13.465, max food at 100, and bacteria move rate at 0.5. Note that this population does not survive. von Neumann neighborhood was used for bacteria movement.

Here I look at a simulation case at a food growth rate where a shift in survivability occurs. In this case, the change occurs around a food growth rate of 13.465.

As before, this population fails to survive. The bacteria grow outward in a wave, bounce back, but the waves of bacteria populations collide and ultimately collapse, fully dying out. This is odd as well because, while the population still dies, this is different behavior than observed in prior dying populations. There is symmetry to this system, and though the bacteria populations are able to grow back inward in the simulation space after their initial wave, they eventually collide with each other and still fail to survive.

While odd and something I want to further investigate, this behavior again suggests to me that there are consequences of the deterministic structure of this system (evident by the symmetry of the bacteria populations in the simulation visualization) that mean for specific food growth values, the populations inevitably collapse. This hypothesis was supported while looking at my last research question, which follows.

In the next case, I increase the food growth rate by only 0.05, and, as the Shannon Entropy food growth experiments suggested, the populations survive and reach stability like seen prior. The results shown in the following two figures further corroborate my prior findings.

Seeing as this may be a property of the deterministic system, my final research question gains more signif-

icance. Though odd (and not something I know why is happening), perhaps this is merely behavior that occurs with this specific cellular automata neighborhood (von Neumann).

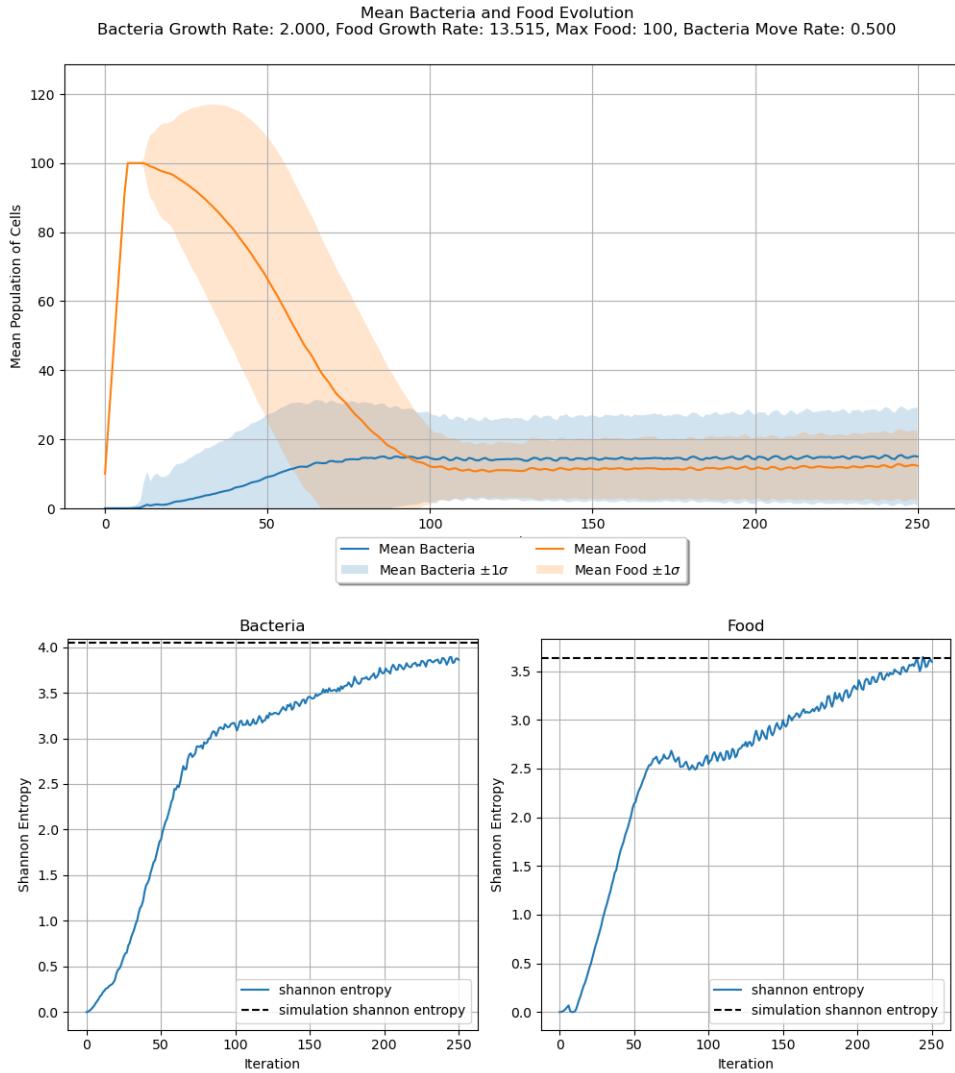


Figure 11. Bacteria and food mean cell populations and Shannon Entropy at each timestep. Shannon Entropy for the entire simulation is also shown as a dashed line. Bacteria growth rate was fixed at 2, food growth rate at approximately 13.515, max food at 100, and bacteria move rate at 0.5. von Neumann neighborhood was used for bacteria movement.

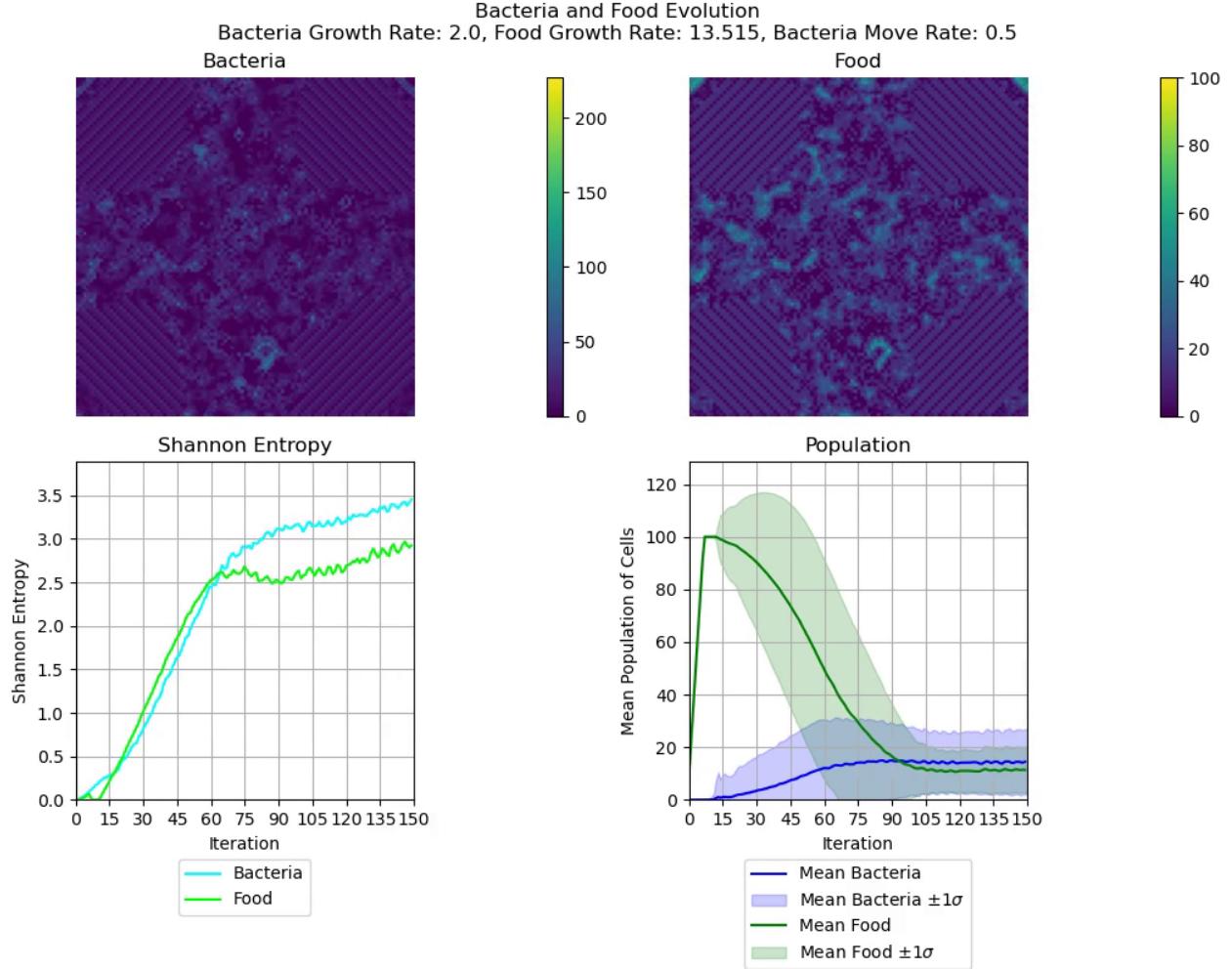


Figure 12. Animation of the population spatial distribution, the accompanying population statistics, and Shannon Entropy at each timestep of the simulation. Bacteria growth rate was fixed at 2, food growth rate at approximately 13.515, max food at 100, and bacteria move rate at 0.5. von Neumann neighborhood was used for bacteria movement.

4.3 Is this behavior consistent for other neighborhoods?

Throughout this project I used von Neumann neighborhoods, but it was also important for me to look at how differing neighborhoods affect the results that I'd seen so far. In this final experiment, I run the simulation using Moore neighborhoods for the bacteria movement instead of von Neumann neighborhoods. I run the same changing food growth factor experiment as before in order to compare to the odd behavior that I saw with the von Neumann neighborhood.

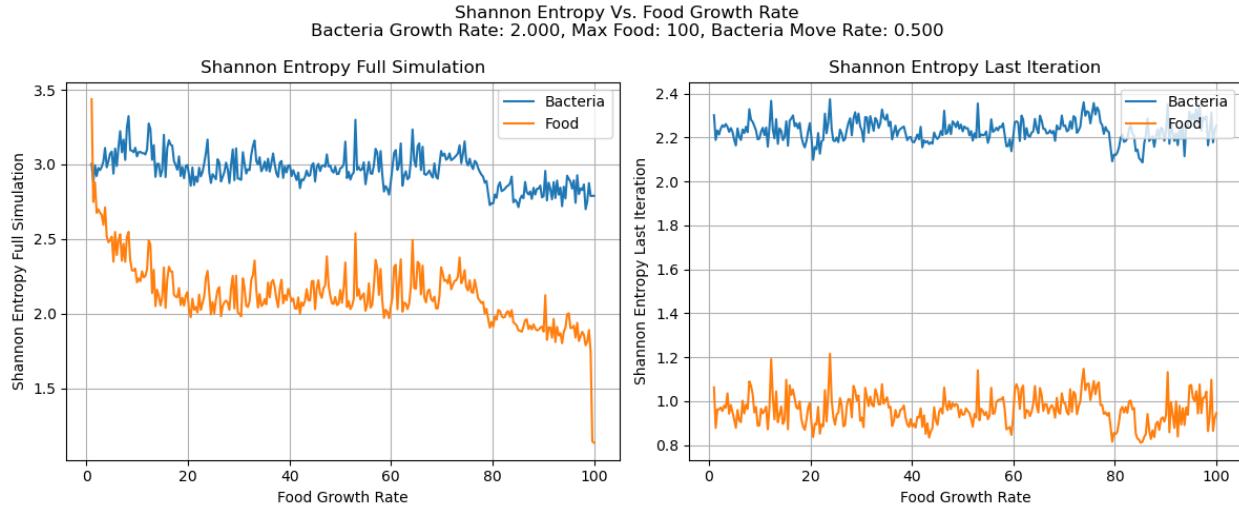
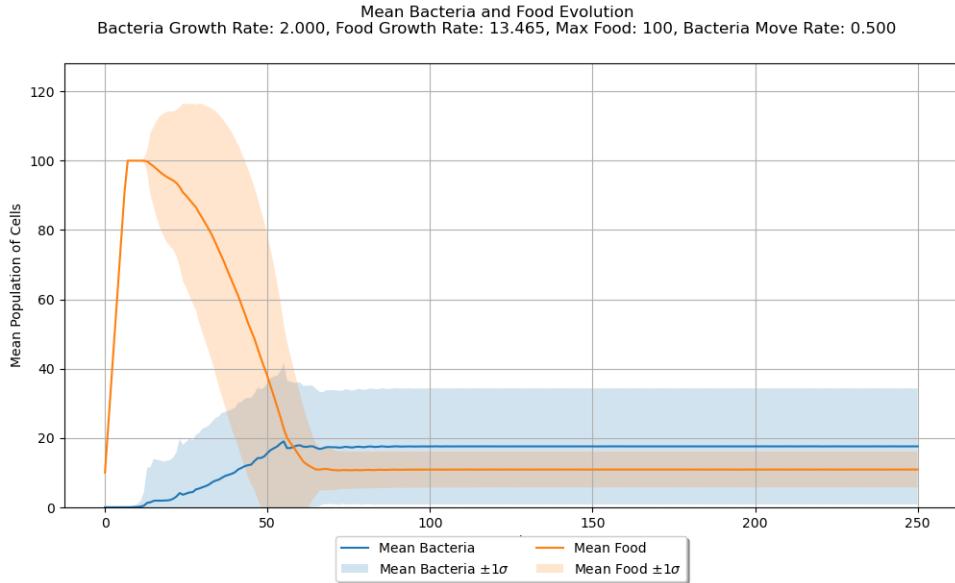


Figure 13. Shannon Entropy results for full simulation (left) and last timestep only (right) of 300 simulations. Food growth factor is varied linearly between each simulation from 0 to 100. Bacteria growth rate was fixed at 2, food growth rate at 2, max food at 100, and bacteria move rate at 0.5. Moore neighborhood was used for bacteria movement.

The difference in these results are stark. Firstly, there are no points with food growth more than 0 where the bacteria population fails to survive. This is significant, because it supports the idea that the behavior seen before is a deterministic product of the bacteria movement rules used. When bacteria are given more choices in their ability to move, they are no longer put in situations where they die out, even at low food growth factors that are more competitive.



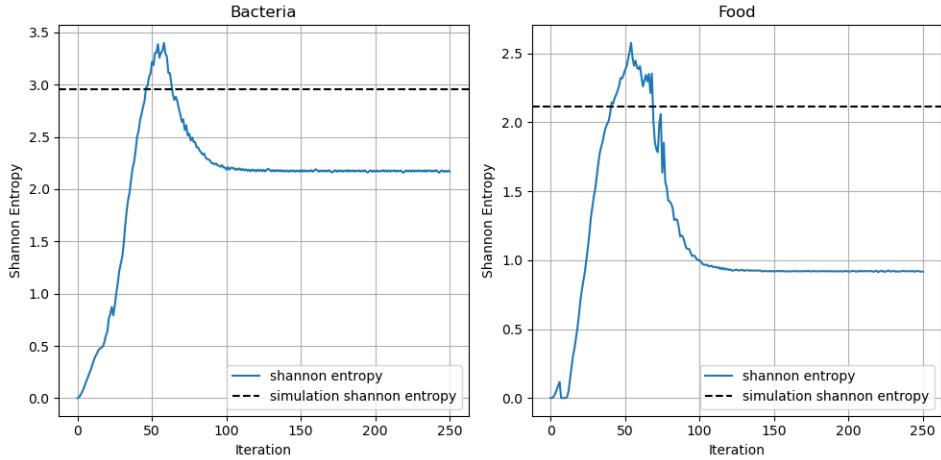


Figure 14. Bacteria and food mean cell populations and Shannon Entropy at each timestep. Shannon Entropy for the entire simulation is also shown as a dashed line. Bacteria growth rate was fixed at 2, food growth rate at approximately 13.465, max food at 100, and bacteria move rate at 0.5. Moore neighborhood was used for bacteria movement.

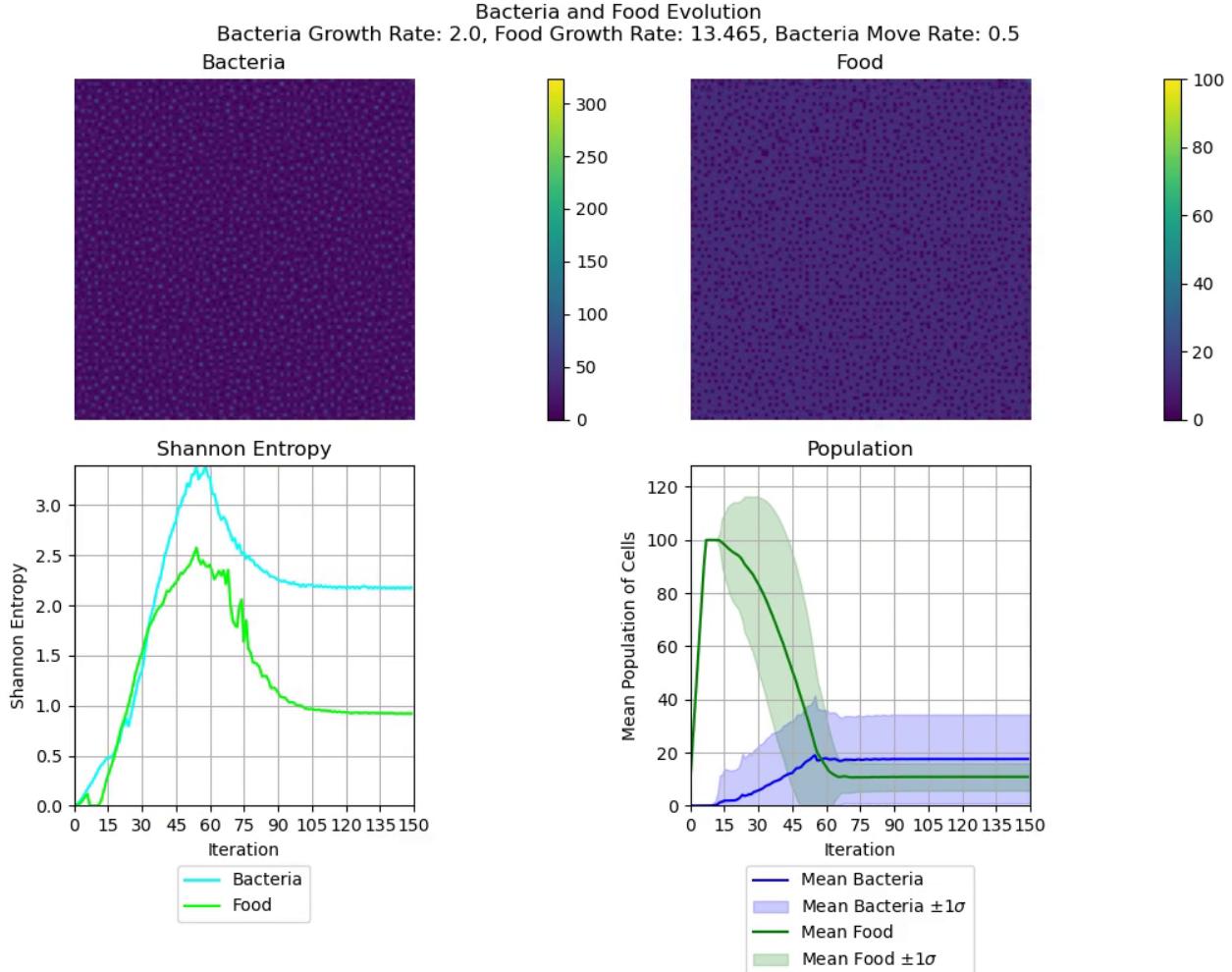


Figure 15. Animation of the population spatial distribution, the accompanying population statistics, and Shannon Entropy at each timestep of the simulation. Bacteria growth rate was fixed at 2, food growth rate

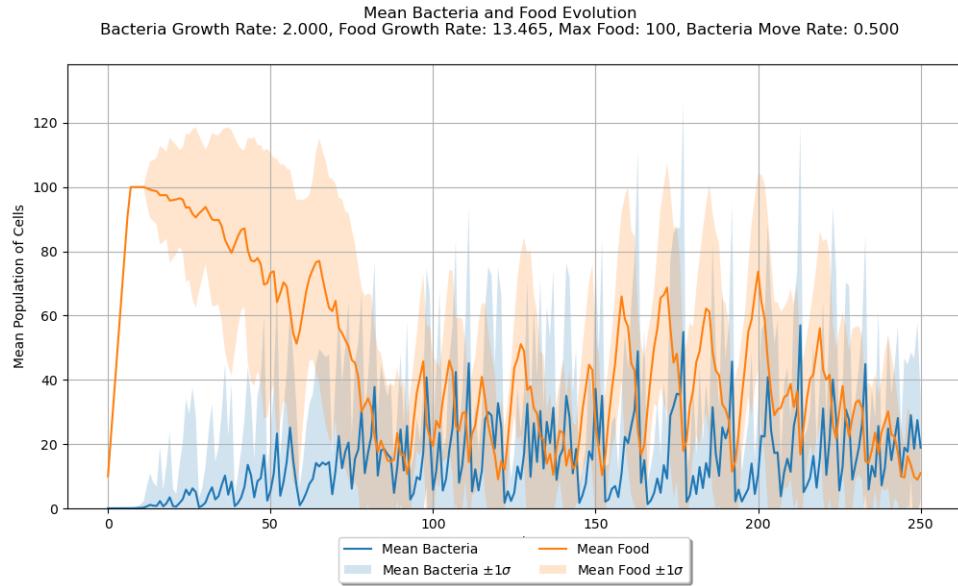
at approximately 13.465, max food at 100, and bacteria move rate at 0.5. Moore neighborhood was used for bacteria movement.

In this final specific test, I keep all other parameters identical as the prior von Neumann counterpart, use a food growth rate (approx. 13.465) that resulted in population death in the prior case, and only changed the neighborhood type to Moore.

In the simulation visualization, it's clear that this small change allows for the bacteria to move into cells it was previously unable to with the von Neumann neighborhood, allowing for survival. Interestingly, though, is that though this population stabilizes in its Shannon Entropy, it has much different behavior in the simulation space. Bacteria populations seem to oscillate back and forth between neighboring cells after reaching stability, settling into a repetitive pattern rather than the kind of population movement that was seen in the von Neumann cases that survived.

5 Introducing Randomness With Random Rule Order

While not part of the original research questions, I felt it was important to introduce randomness into the system to see how it would affect the bacteria populations. I wanted to introduce randomness while keeping all of the other hyperparameters as used before constant, therefore the approach used was to randomize *rule order*. This means that on each timestep of the simulation, the execution order of the three rules (bacteria growth, food growth/depletion, and bacteria movement) is randomized. This means that it is possible for two of each rule to happen sequentially across separate timesteps, and also can enable situations where bacteria can escape a cell before their update rule to evade death in a cell with empty food. Here, the von Neumann neighborhood is used with the prior used food growth rate that resulted in population death before (see Figures 9 and 10).



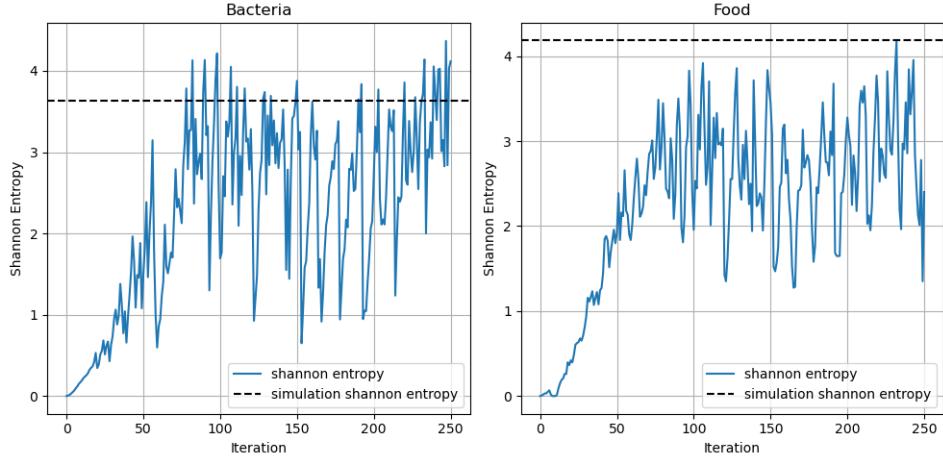


Figure 16. Bacteria and food mean cell populations and Shannon Entropy at each timestep. Shannon Entropy for the entire simulation is also shown as a dashed line. Bacteria growth rate was fixed at 2, food growth rate at approximately 13.465, max food at 100, and bacteria move rate at 0.5. von Neumann neighborhood was used for bacteria movement. System rule order was randomized at each timestep.

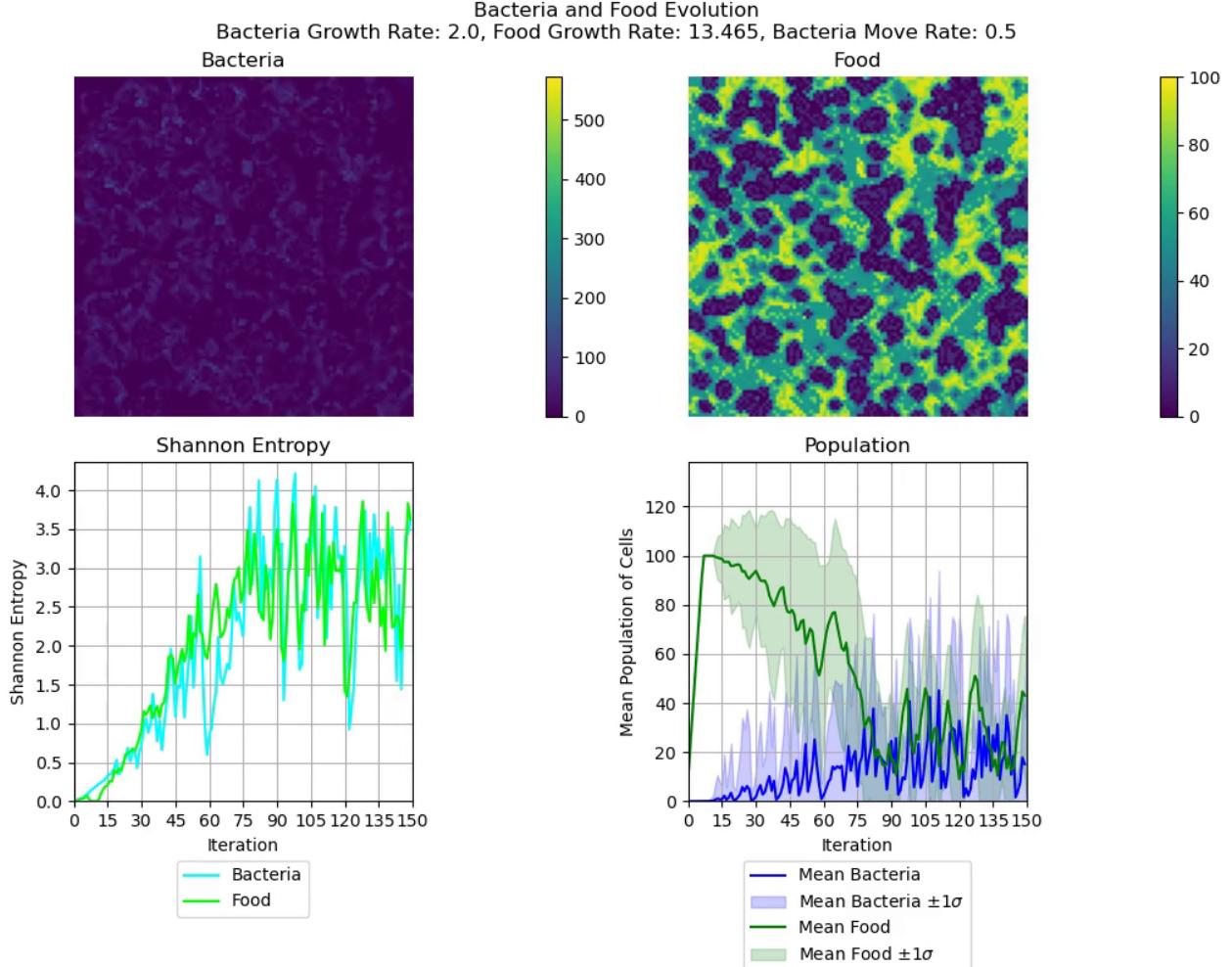


Figure 17. Animation of the population spatial distribution, the accompanying population statistics, and Shannon Entropy at each timestep of the simulation. Bacteria growth rate was fixed at 2, food growth rate

at approximately 13.465, max food at 100, and bacteria move rate at 0.5. von Neumann neighborhood was used for bacteria movement. System rule order was randomized at each timestep.

With the introduction of randomness into the system, the deterministic collapse seen in the original simulation no longer happens. Random rule order allows the populations to survive, even at the lower food growth rate.

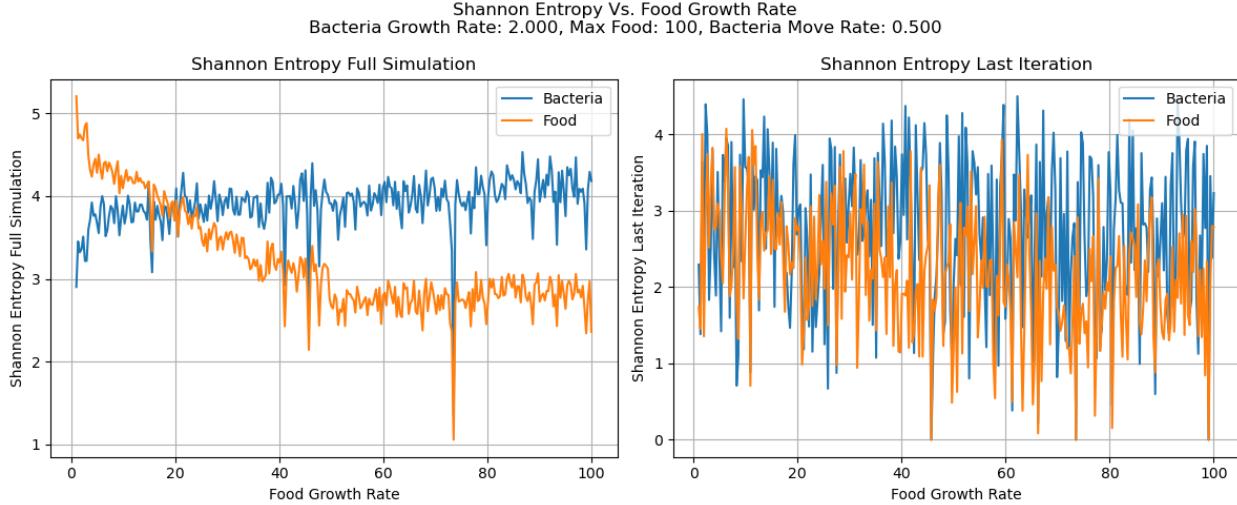


Figure 18. Shannon Entropy results for full simulation (left) and last timestep only (right) of 300 simulations. Food growth factor is varied linearly between each simulation from 0 to 100. Bacteria growth rate was fixed at 2, food growth rate at 2, max food at 100, and bacteria move rate at 0.5. von Neumann neighborhood was used for bacteria movement. System rule order was randomized at each timestep.

Running the changing food growth rate experiment again with random rule order gives a better picture of what's going on. Populations are able to survive even at low food growth rates. This supports the hypothesis that this is a property of the deterministic structure of the system, and suggests that introduction of randomness may be valuable to modeling a bacteria growth system.

6 Conclusion

Though I was able to answer all of my original research questions, I found myself repeatedly asking more questions as to the behavior I was observing in my system than I had started with! Though I now have a much broader understanding of how this system behaves (and how it can be further utilized to model cell behavior), there is plenty more exploration that can be done. I looked at the von Neumann and Moore neighborhoods, but the parameters used were the same. I did not explore changing of the movement rule. I observed that the Moore neighborhood system never died off, but I can only say this is true for the parameters used in the simulations; changing hyperparameters could alter this behavior.

A larger takeaway from this project, I feel, is that a deterministic system may be less effective and representing a competitive cell (in this case, bacteria) system. Though my system rules attempt to represent the ways in which bacteria move towards food in reality, their movement is emergent and perhaps the deterministic model being used here is unable to capture behavior more akin to the real thing.

As a final thought as to measures of complexity, using Shannon Entropy as a measure of complexity results in zero complexity with a static population and a maximum complexity with a completely random cell population. However, one can argue that total randomness should actually have zero complexity in the sense that it is also static from a statistical perspective. Looking at the animations, some of the most interesting behavior seems to occur when there are rapid changes in the Shannon Entropy rather than the absolute value of the Shannon Entropy being high. Hence, perhaps a better measure of complexity for this project would have been the absolute value of the *rate of change* of the Shannon Entropy, or perhaps the square

of the rate of change of the Shannon Entropy. This measure requires an evolution in time, however, one could also come up with similar measures to characterize spatial complexity. Perhaps an interesting topic for another project.

7 References

- [1] Lloyd, Seth, Measures of Complexity: a non-exhaustive list, IEEE Control Systems Magazine (Volume: 21, Issue: 4, August 2001), <https://web.mit.edu/esd.83/www/notebook/Complexity.PDF>
- [2] The Bell System Technical Journal, Vol. 27, pp. 379–423, 623–656, July, October, 1948. <https://web.archive.org/web/19980715013250/http://cm.bell-labs.com/cm/ms/what/shannonday/shannon1948.pdf>
- [3] Scott, David W. (1979). "On optimal and data-based histograms". Biometrika. 66 (3): 605–610. <https://doi.org/10.1093>
- [4] Krawczyk K., Dzwinek W., & Yuen D. A. (2003). Nonlinear Development of Bacterial Colony Modeled With Cellular Automata and Agent Objects. International Journal of Modern Physics C, 14(10), 1385-1404. <https://doi.org/10.1142/s0129183103006199>
- [5] Lindgren K., & Nordahl M. G. (1988). Complexity Measures and Cellular Automata. Complex Systems, 2, 409-440.
- [6] Muci A. L., Jorquera M. A., Ávila Á. I., Rengel Z., Crowley D. E., & De La Luz Mora M. (2012). A combination of cellular automata and agent-based models for simulating the root surface colonization by bacteria. Ecological Modelling, 247, 1-10. <https://doi.org/10.1016/j.ecolmodel.2012.07.035>