Modeling and Analysis of Complex Systems

Predator-Prey Ecosystem Analysis
Minerva University
Fall 2025
Tambasco, TTh@3PM UTC

October 26, 2025

Contents

1 Introduction

This assignment will implement a "reaction-diffusion" model between predators (foxes) and prey (bunnies). From a small set of rules that define the interactions between foxes and bunnies across the cells, a surprisingly large variety of patterns emerged. Before the empirical and theoretical analysis of this system, I explain a few assumptions and simplifications of the model that allow me to produce useful and computable data as well as visualizations. Several test cases and an explanation of how convergence was asserted precedes the empirical implementation. Then, the effect of varying several of the model parameters was plotted and explained. Finally, I conclude with a theoretical analysis of the mean-field approximation and discuss the differences between the empirical and theoretical results.

2 Model Description and Python Implementation

After some initial experimentation with the given parameters, I decided to increase the `f_consump` (which represents the number of bunnies a fox needs to survive) from 0.1 to 0.3, as this seemed like a more realistic assumption and led to visible oscillating patterns that piqued my curiosity to investigate further. Additionally, I also increased the probability of the fox population in a cell randomly dying out. This could model a type of epidemic or human hunting behavior, but was chosen to (potentially) increase the effect of randomness on the model's behavior.

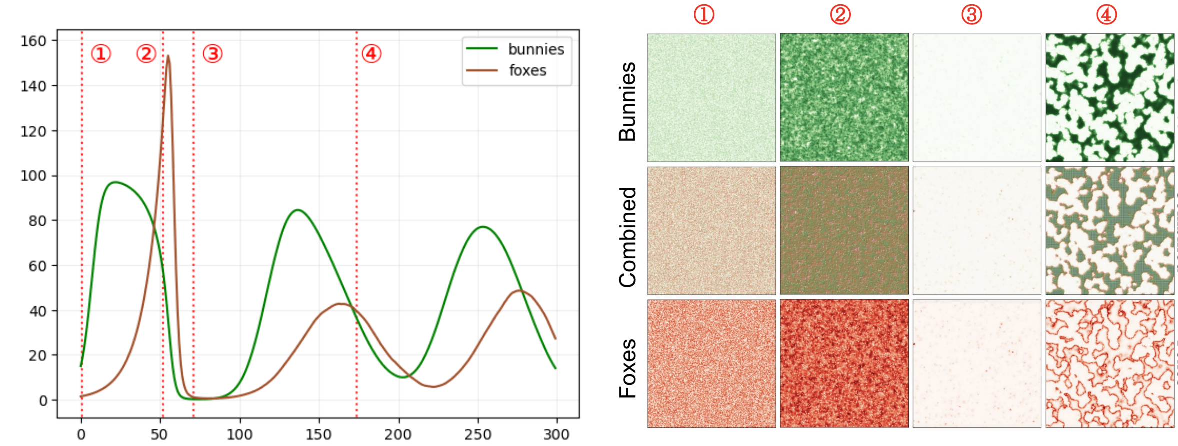
To initialize the grid, I also added `p_bunny=0.3` and `p_fox=0.1`. When creating the grid, the initial populations are randomly drawn from a uniform distribution $B \sim U(0, p_B \cdot k_b) = U(0, 30)$ for the bunnies and $F \sim U(0, p_F \cdot p_B \cdot k_b) = U(0, 10)$. This way, we get small initial populations with a higher population of bunnies than of foxes. This way, the simulation is started with a setup that allows both bunnies and foxes to reproduce. As it is assumed that the bunnies and foxes live in a closed ecosystem, periodic boundary conditions are used. While this limits the realisticness (as most systems like a forest have non-periodic boundaries), it will likely simplify the analysis as we do not have any hard borders at which the behavior would deviate from the rest of the grid (e.g. as the bunny and fox diffusion happen in only three or two rather than four neighboring cells at the border or corner of the grid). Hence, the base parameter that will be used throughout this assignment are:

1. n: Grid size (200)
2. g_b: Bunny Growth Rate (0.3)
3. k_b: Bunny Carrying Capacity (100)
4. g_f: Fox Growth Rate (0.1)
5. c_f: Fox Consumption Rate (0.3)
6. m_f: Fox Mortality Rate (0.03)
7. d_f: Fox Diffusion Rate (0.1)
8. d_b: Bunny Diffusion Rate (0.1)
9. p_b: Bunny Initialization Constant (0.3)
10. p_f: Fox Initialization Constant (0.1)

2.1 Test cases

2.1.1 Sample plot and animation

To validate the simulation's behavior, I compared the temporal evolution of average population densities with their corresponding animation frames. This verification confirms that the averaged data accurately represents the spatial dynamics visible in the animation. For instance, at timestamp (2), the spike in fox density visible in the average plot corresponds to hunting in the animation, while the low densities at (3) align with post-predation population crashes. The animation provides additional spatial information not visible in the average plots, and shows how fox populations concentrate at the boundaries of bunny-dense regions.



(a) Plot of the average fox and bunny density over a limited interval of time.

(b) Snapshots of the simulation showing different stages of the system.

Figure 1: Comparing average plot with animation snapshots at $t = 1$ (1), $t = 52$ (2), $t = 71$ (3), and $t = 174$ (4).

2.1.2 Other test cases

Firstly, the bunny growth rate g_b was tested by setting the fox population and all diffusion to 0 and observing whether the population growth aligns with the defined logistic growth.

Secondly, test cases were implemented to check whether the fox population correctly dies out amid the absence of bunnies. For this, the grid was initialized without any bunnies to assert whether the amount of starved foxes corresponded exactly to the number of foxes it was initialized with.

Thirdly, the bunny diffusion d_b was tested by initializing the grid without foxes and setting all growth parameters to 0. Hence, we can assert that the initial bunny population placed in a known cell would diffuse to its neighborhood in equal parts $(0.25 * d_f)$, leaving $1 - d_f$ bunnies in the cell itself.

All three test cases were repeated for different sets of parameters (e.g. high and low growth rates), and passed in all cases.

```
Running simulation tests...
[SUCCESS] test_bunny_growth (standard)
[SUCCESS] test_bunny_growth (low pop)
[SUCCESS] test_bunny_growth (high pop)
[SUCCESS] test_fox_starvation (standard)
[SUCCESS] test_fox_starvation (low pop)
[SUCCESS] test_bunny_diffusion (standard)
[SUCCESS] test_bunny_diffusion (high diffusion)
----------------------------------------
All 7 tests passed.
----------------------------------------
```

Figure 2: Test cases passed (output of code cell 2).

2.2 Stable State Analysis

To conduct an analysis for this system where we compare the effect of varying one metric on the population densities, we need to ensure it has reached a stable state. This means that the *average* density values (foxes or bunnies) across the system converge to a single value. While this means that the state of individual cells is not fixed, the average density converges to a constant number. While I initially started by plotting the different graphs to observe if the populations converged to a single horizontal line, this visual approach can be challenging for multiple reasons: for instance, the y-axis scaling might limit the precision and make it seem like the density has become horizontal despite remaining fluctuations, and plotting all these graphs would also complicate this analysis. Hence, while convergence theoretically only works as we increase the number of steps $n \to \infty$, my computational limitations led me to make a simplifying assumption: I assume that a stable state is reached if the last 10 equally spaced x-intervals (with a width of $x_{max} - x_{min}/200$) have an average difference in their y-mean (density) that is less than 0.1%, which is a very low value. This function was called with an upper threshold (15,000 - 25,000) for every simulation and plot that was run to assert that a stable state is reached. While this might not allow some functions to converge beyond this upper limit, the vast majority of the simulations plotted for this assignment stayed below these bounds.

2.2.1 Justifying convergence

As both the grid initialization as well as the fox mortality evens are random processes, the time taken until a system converges varies in ever run. While repeating simulations for each set of parameter values for hundreds of trials mitigates this, it is computationally extremely expensive. However, showing that the distribution convergence times has a clear peak around a reasonably small interval would help justify using only a limited number of repeated trials. Running the convergence check on a smaller grid with sizes 150×150 for 300 times confirms that the distribution of convergence times for a set of parameters does indeed have most of its probability mass clustered around a central interval (5000-8000). Note that there is a slight right skew with some very large convergence times. This confirms that running a single simulation might be insufficient as the randomness contained in the simulation introduces variability to the convergence times. For the computationally more expensive 200×200 grid used for the simulation, results will be averaged over 5-10 simulated trials to smooth out the random noise.

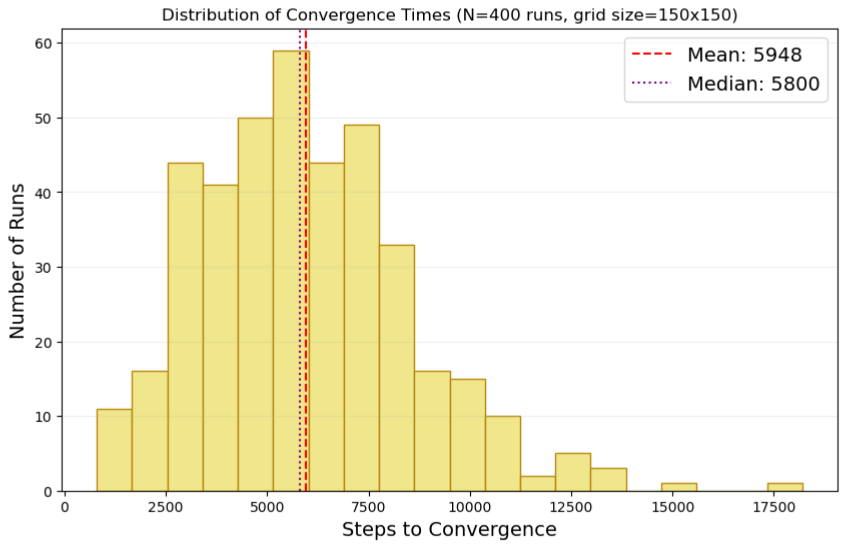


Figure 3: Distribution over the time it took for the simulation with the base parameters to converge. The upper bound was set at 20,000 which prevents possible convergence times beyond that from being shown.

3 Empirical Analysis

3.1 System Viability: Fox Growth Rates

The fox growth rate g_f represents the percentage by which the surviving fox population grows after each time step. While very low values likely lead the fox population to die out as they cannot reproduce enough to survive the random mortality events, a very high growth rate likely also prevents the foxes from forming a large population as too much offspring puts a strain on the bunny population available for hunting. In other words, foxes produce a lot of offspring they cannot feed.

As we run each simulation 10 times and want to find the runs that are always dying out, we set the threshold for the percentage change in the convergence check function to 0.0. This is because the only case in which we would expect an average change of 0.0% for foxes is when they are dead and the average population from one time step to the next has virtually no change at all (to avoid division by zero, the change between consecutive values of 0 was defined as 0.0). While this lets simulations with higher values of g_f run until the convergence time upper bound (12,000 here), it accurately captures the cases where entire fox population is dead.

Let us compare the converged fox population as we vary g_f over its entire domain $[0, 1]$, with each density value averaged over 10 trials (Figure 4, left). This shows a maximum point at around $g_f = 0.05$, with all fox populations dying out at g_f values lower than that. While low density values are also reached as g_f approaches 1, the average density value is always positive, meaning that foxes survived in at least 1 of the 10 trials. To increase the precision of this observation, let us reduce the interval to the range of interest $[0.02, 0.04]$ to find the exact point for the phase change. Figure 4 (right) shows that the fox population always dies out for values lower than $\mathbf{g_f \approx 0.306}$ while stabilizing at drastically larger density values for growth rates larger than that.

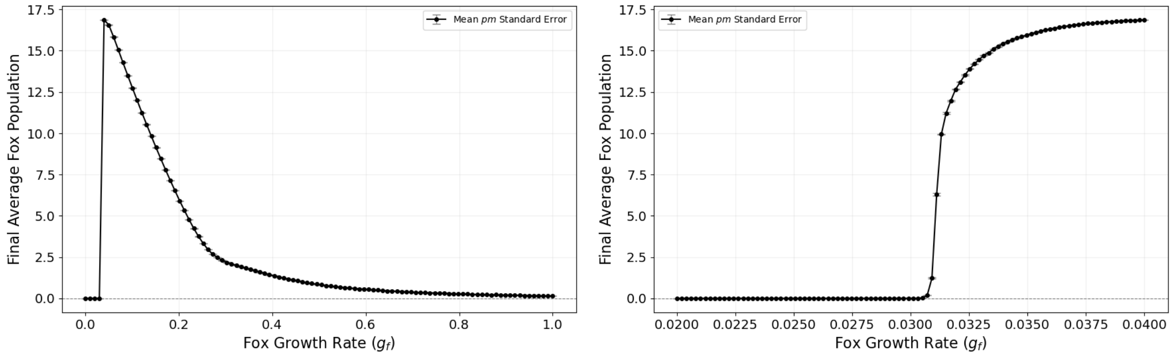


Figure 4: Fox Growth rates vs final converged population over 10 trials. The left plot is over the entire interval of g_b and the right plot only over the interval where the fox population stops dying out. The threshold for convergence was set to 0.0 to determine g_f values where the population dies out entirely.

This aligns with the initial expectation that the growth rate g_f is related to the mortality probability m_f, which randomly causes the fox population in a cell to die out. The plots suggest that whenever $g_f < m_f$, the population cannot sustain itself as more foxes die than are reborn.

3.2 Pattern Formation

This section investigates the pattern formation given different values of the diffusion rate ratio $\frac{d_b}{d_f}$. For this, the fox rate was kept constant while the bunny diffusion rate was altered. The diffusion rate affects the spread of bunnies and foxes through the system, and, in the model, calculates the share of bunnies or foxes that leave to each neighboring cell after each time step. A higher diffusion rate ($\frac{d_b}{d_f} > 1$) will likely give the bunnies an advantage as they might be able to 'outrun' the foxes before they can catch up and reproduce. These figures show different pattern formations that can roughly be divided into three categories:

1. **Very scattered small populations** ($d_b \leq 1$): for a diffusion ratio of 0, the bunny population stays static while the fox population diffuses. This means that foxes will move to places with enough rabbits right in the beginning of the simulation, almost eliminating the bunny population which cannot 'retreat' to areas with lower fox densities to repopulate. At the same time, foxes might also move away from highly populated bunny clusters as a significant fraction moves to neighboring cells. This 'inefficient' diffusion might cause foxes to spend too much time in empty cells which makes them prone to die of starvation or random mortality events.

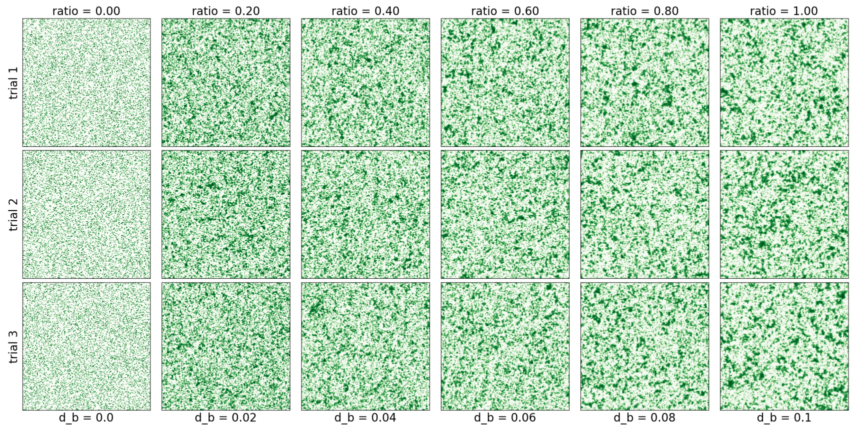


Figure 5: Bunny density patterns for small diffusion ratios $0 < d_b \leq 1$

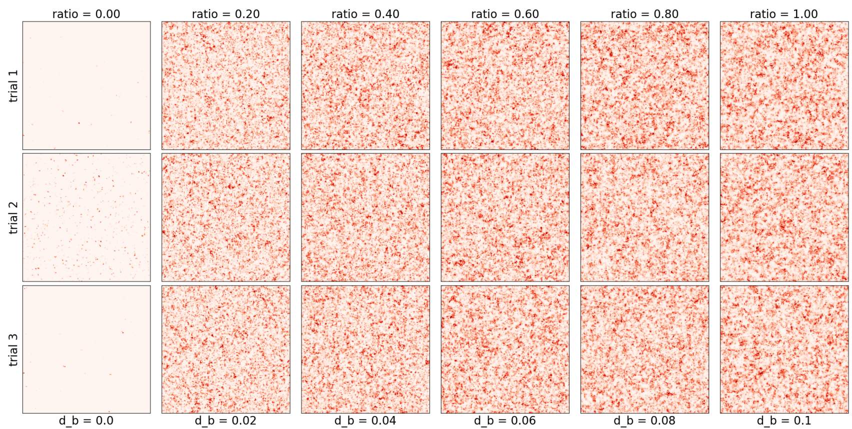


Figure 6: Fox density patterns for small diffusion ratios $0 < d_b \leq 1$

2. **Clustering** ($1 < d_b \leq 8$): For ratios slightly greater than 1, bunnies diffuse faster than foxes. This means that a greater share of the bunnies moves to the surrounding cells after each step than the share of foxes that follows them. This means that bunnies can 'escape' from the foxes, which allows them to form short-lived, unstable clusters in places where bunnies can quickly reproduce before the foxes can catch up with them. For larger ratios like $d_b = 4$ or $d_b = 7$, these clusters get thicker and more attached to each other. Simultaneously, there are large areas with hardly any bunnies because foxes are hunting in them. More specifically, the foxes now form a 'surrounding' pattern whereby they encircle the bunny clusters and slowly move inwards. While the fox density increases

as they penetrate into the dense bunny clusters, foxes die off in the sparsely bunny-populated areas they just hunted in. This allows for bunnies to reproduce in these areas again after the foxes have moved through them.

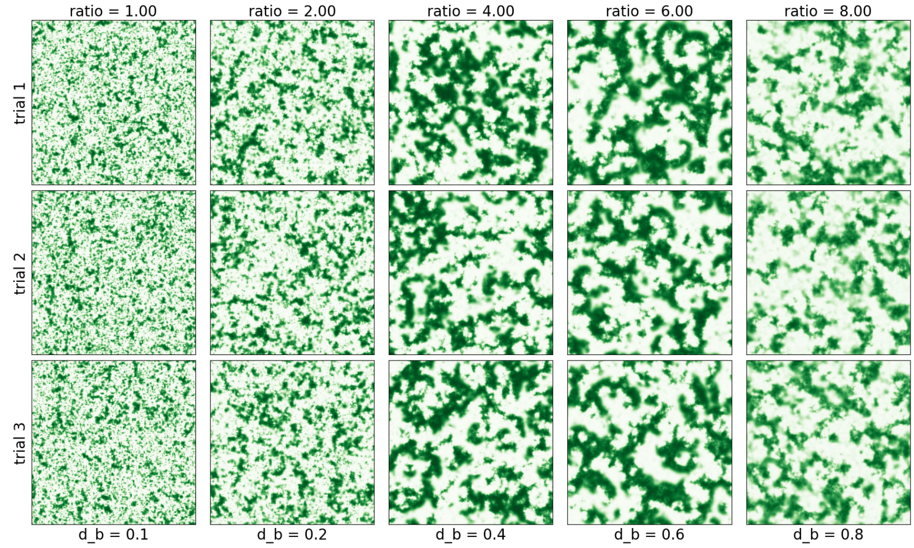


Figure 7: Bunny density patterns for medium diffusion ratios $1 < d_b \leq 8$

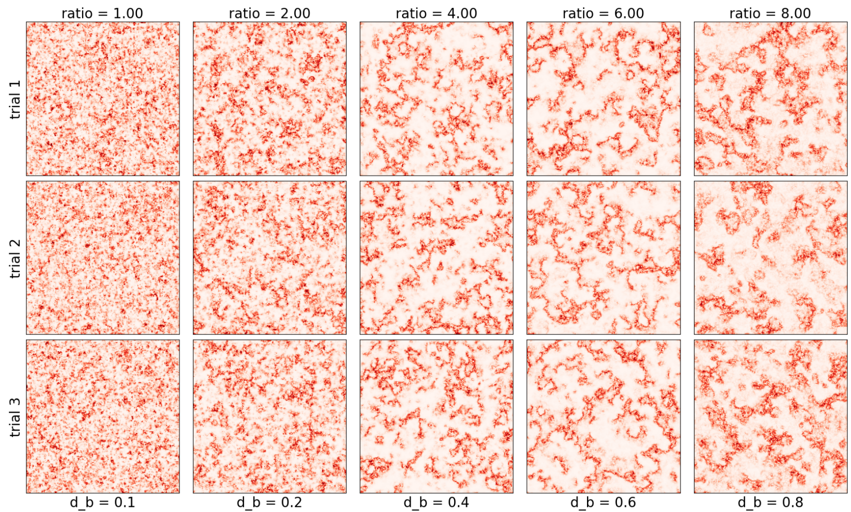


Figure 8: Fox density patterns for medium diffusion ratios $1 < d_b \leq 8$

3. **More uniform distribution** ($d_b > 8$): When the bunny diffusion rate is very high, the bunny density becomes a lot more uniformly distributed overall. This is plausible as bunnies move very quickly after reproducing, with the majority of the bunnies moving to neighboring cells after each time step and hence diffusing quickly through the entire system. This creates a 'smoothing' effect whereby clusters do not stand out as much as for lower values of d_b. The omniscience of bunnies also makes the fox density more uniformly distributed as foxes never need to diffuse far to find bunnies to eat. While the 'surrounding' pattern remains, the entire system seems to host at least *some* foxes and bunnies as foxes do not need to move fast to find prey: bunnies are virtually diffusing into the foxes cells, making it unlikely for them to die of starvation.

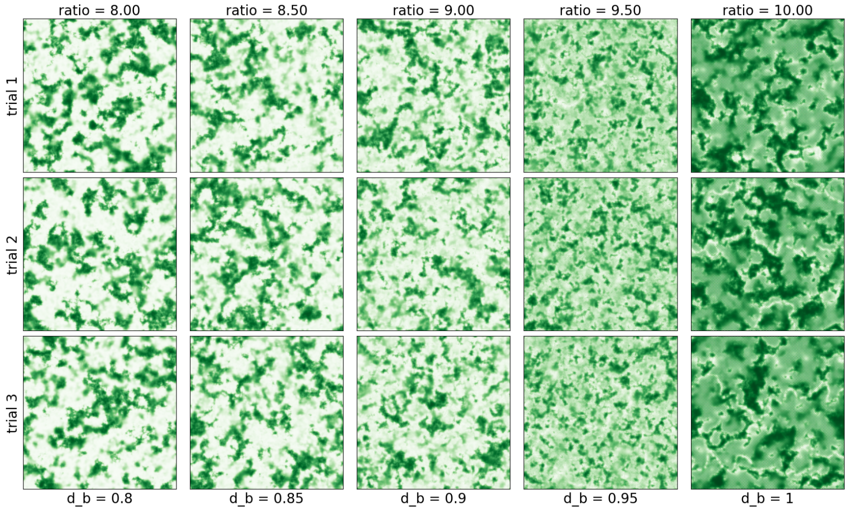


Figure 9: Bunny density patterns for medium diffusion ratios $d_b \geq 8$

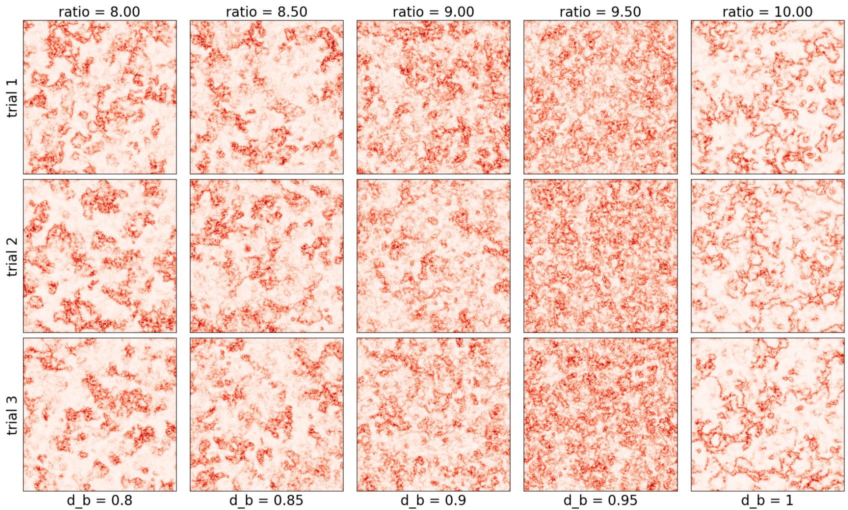


Figure 10: Fox density patterns for medium diffusion ratios $d_b \geq 8$

3.3 Stability: converged populations vs bunny growth rate

In order to identify specific tipping points where the system begins to change or oscillate, a variety of parameters for b_g were tested over $n = 3$ repeated trials and a fixed interval of $5,000$ because oscillations, if they occurred, would always do so within the first 500 steps of the interval.

Plotting some initial trajectories of the mean fox and bunny density as the initial parameters are changed (Figure 11 left) allows for the initial insight that the range of interest is somewhere between 0 and 0.1. For $g_b = 0$, the bunny population (and consequently the fox population) die out, which makes sense as bunnies cannot reproduce. Despite somewhat unstable and 'bumpy' as the simulation progresses, a clear oscillating pattern is observed after $g_b > 0.1$. For these larger growth rates, the oscillations decrease in amplitude within the first few thousand steps and then stabilize at a much lower amplitude. The steps taken to stabilize decreases as we increase g_b, e.g. it takes about 2,500 steps for stable oscillations with $g_b = 0.1$ but only about 100 with $g_b = 0.9$.

Figure 11 (right) shows the mean densities for g_b values around 0.1 which the left plot suggested would be the interval for which oscillations begin.

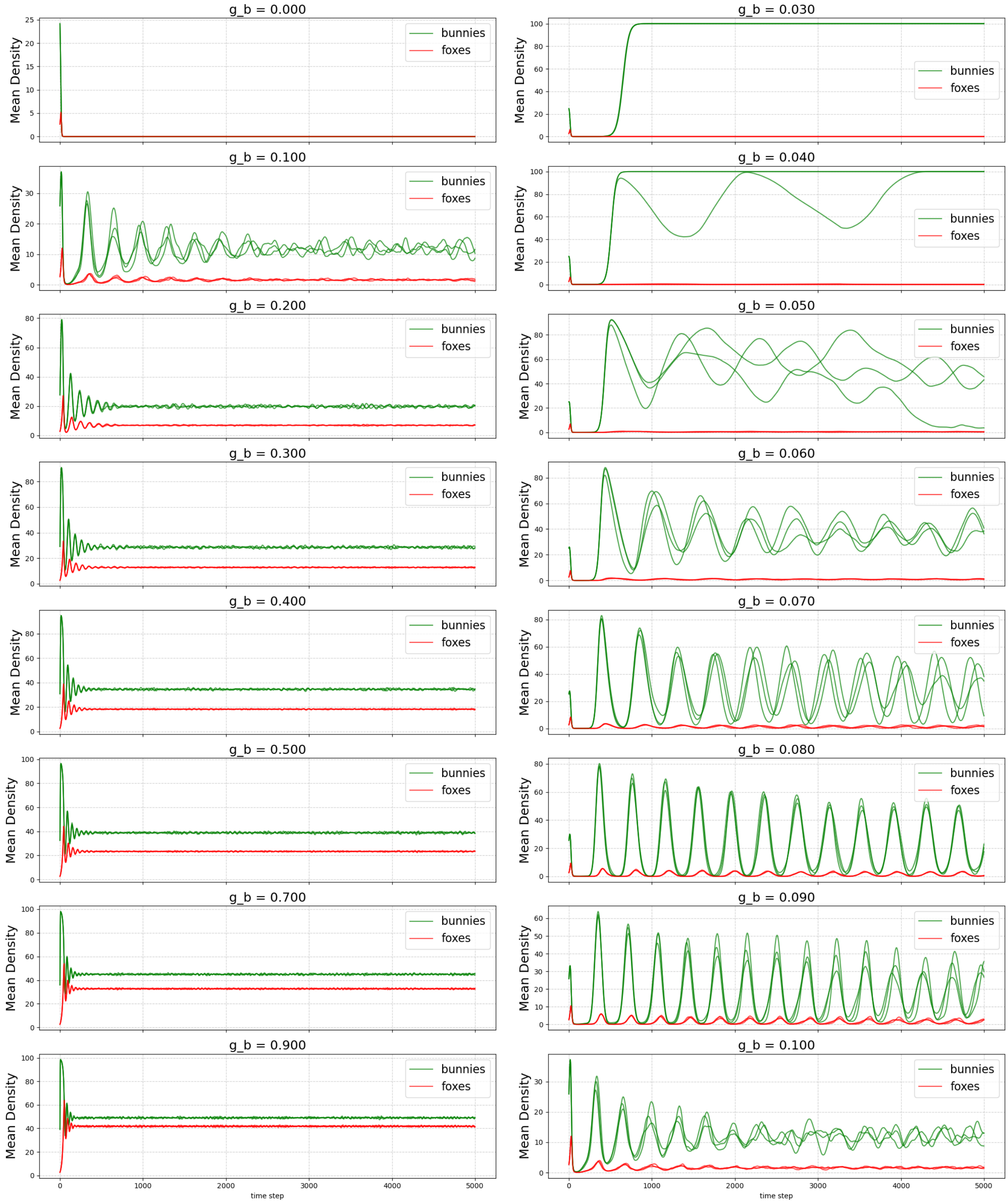


Figure 11: Average bunny and fox density over varying values of g_b over 5,000 steps. Each set of parameters was run and plotted three times to get a more complete picture. The transition to oscillatory behavior occurs between 0.03 and 0.04. Higher growth rates lead to faster stabilization of these oscillations.

To better understand this threshold, Figures 12-14 below were plotted to complement this analysis by explaining the spatial dynamics of the identified threshold. The videos the snapshots were taken from demonstrate the dynamics even more and should be watched while reading this section.

1. For values of $g_b \leq 0.03$, the system is 'damped'. This means that after an initial decrease in the bunny population - as foxes immediately begin hunting - the bunnies do not reproduce fast enough to sustain the fox population with their mortality and consumption rate. Bunnies reproduce too slowly to be populated enough allowing the bunnies to reproduce to their maximum capacity.

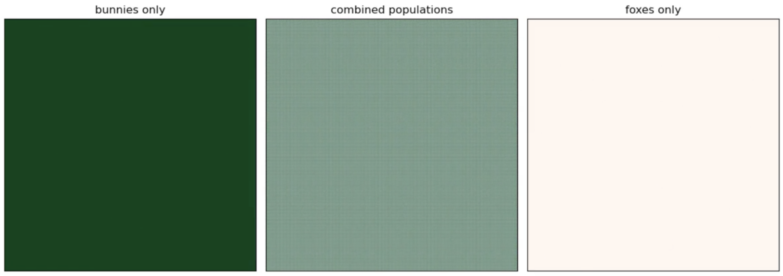


Figure 12: Screenshotted step 2265 of `growth_0.03.mp4` showing that fox populations cannot sustain themselves with a bunny growth rate of $g_b = 0.03$.

2. First oscillations begin at $g_b = 0.04$. This means that after an initial hunting period, the bunny population has decreased, leading a lot of the newly produced foxes to starve as a consequence. However, this time, bunnies reproduce at a slightly higher rate than before, allowing foxes to capture some new bunnies before they entirely die off. The very small increase of bunny growth rates from 0.03 to 0.04 seems to be just enough to sustain some of the foxes, leading to a repeated cycle of recovery, hunting, starvation, and recovery again. The screenshot of $s = 1080$ from the visual simulations shows the survival of a single tiny fox population in the centre of the donut-shape: from there, the foxes diffuse outwards into the highly populated areas. This leaves very small bunny populations in the hunted areas that do not allow foxes to survive, but bunnies to reproduce.

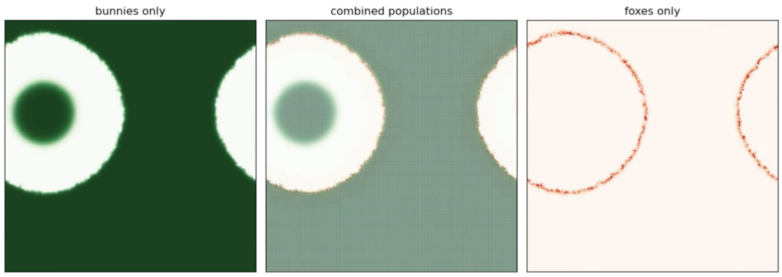


Figure 13: Screenshotted step 1080 of `growth_0.04.mp4` showing the beginning of oscillatory densities in a 'wave' pattern.

3. However, 'regular' oscillations only begin for values $g_b \geq 0.06$, with the most stable and regular oscillations recorded take place at $g_b = 0.08$, with growth rates beyond $g_b > 0.1$ leading to more 'bumpy' and uneven oscillations. This is plausible as after phases of hunting and starvation, bunnies can populate quickly enough to sustain some of the foxes. The snapshot below shows how many small fox populations are able to survive the initial collapse of the bunny population and penetrate into the densely populated areas around them.

To quantitatively measure whether the system has started to oscillate, simply plotting these graphs might not be sufficient. Some research into 'bifurcation theory', which studies the parameter values for which the properties of the larger system suddenly changes, led me to identify the Fast Fourier Transformation as a possible metric to determine whether the system has taken on a regular frequency of repeated oscillations (Zandi-Mehran et al., 2022). The FFT separates the plot into different frequencies and determines whether there are any dominant repeating patterns. According to this, flat, non-oscillating lines (like the ones observed for $b_g \leq 0.03$ would result in a low 'power' value in the FFT plot (as they

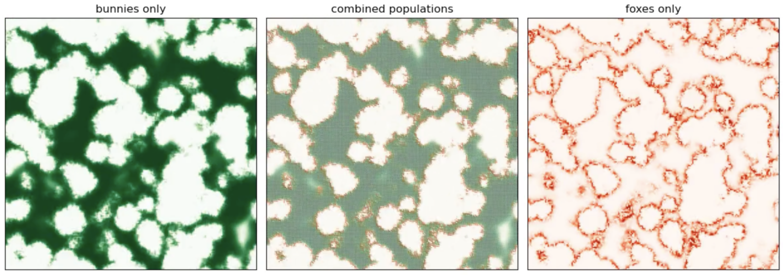


Figure 14: Screenshotted step 1095 of `growth_0.08.mp4` showing the periodic interplay between population increase and decrease.

contain no periodic movement), while more stable and periodic oscillations will be assigned a higher power score (Julia Programming, 2021). The FFT plot (Figure 15) confirms the observation that a growth rate of 0.04 acts as a threshold for increasingly constant oscillations for the fox and the bunny population. This creates a delayed feedback loop: bunny abundance allows fox growth, which then suppresses bunnies, which then causes fox decline, allowing bunnies to recover.

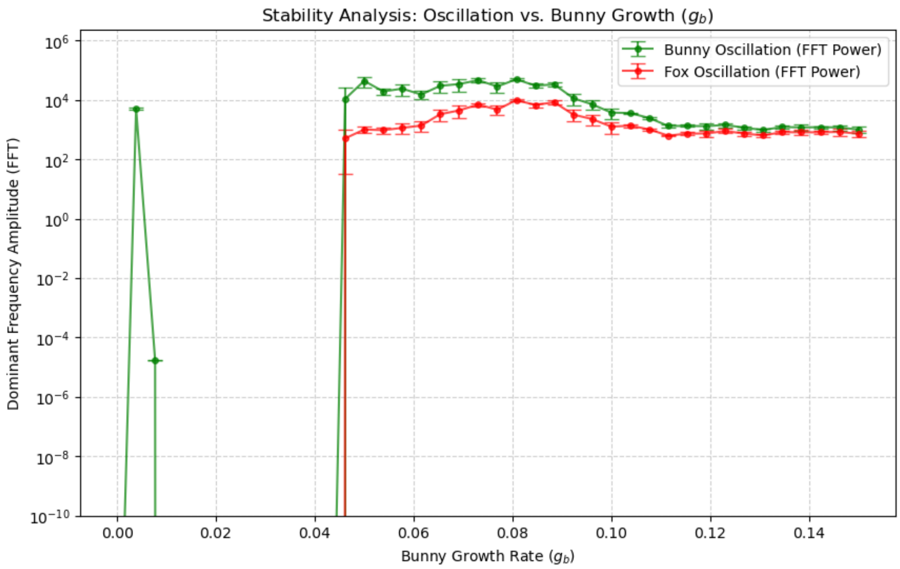


Figure 15: FFT Plot showing the sudden increase in FFT Power values around 0.04 threshold.

4 Theoretical Analysis

Rather than looking at individual cell states, we can reduce the dimension of the grid to a single average value, using the mathematical relationships we are given about growth, diffusion, and death. Different to previously discussed model, the states in this model are continuous, meaning we cannot simply enumerate the probabilities of moving from a discrete state "0" to another "1". Instead, we connect our knowledge of a stable state where the mean field value of the next state is equal to the current value to the 'average', mean-field interactions between foxes and bunnies. As we reduce the entire grid to a single mean-field, we make the assumption that the bunny and fox diffusion does not matter as the average population densities are constant regardless of the distribution of bunnies and foxes across the grid. Leaving out the diffusion rates, we still need to consider the following parameters:

1. g_b: Bunny Growth Rate (0.3)
2. k_b: Bunny Carrying Capacity (100)
3. g_f: Fox Growth Rate (0.1)
4. c_f: Fox Consumption Rate (0.3)
5. m_f: Fox Mortality Rate (0.03)

The change in bunny population from one time step to the next, $\Delta b = b_{t+1} - b_t$, can be expressed as the difference between the bunnies gained (by reproduction) and the bunnies lost (by fox hunting).

$$b_{t+1} = b_t + \Delta b \tag{1}$$
$$\Delta b = b_g - b_d \tag{2}$$
$$\tag{3}$$

The bunnies gained by reproduction are

$$b_g = b_t \cdot g_b \cdot (1 - \frac{b_t}{k_b}), \tag{4}$$

which is the same formula as in the instructions, only without the addition of b_g (as we are interested in the *change* of bunny density between two steps). The bunnies lost can be calculated as a product of the current number of foxes, their individual consumption rate, and the number of bunnies present in the cell. This is a deviation from the model because in the model, $min(b, c_f \cdot F)$ bunnies are hunted. In the model, c_f is a fixed limit of how much a fox eats, while in this case, foxes consume the bunny in a fraction of c_f times they encounter each other. As foxes and bunnies are perfectly mixed in this MFA, we can broadly assume that the consumption of bunnies scales with both the number of foxes and the bunny population. This allows for an analytical solution of the stable state system. Hence, we assume:

$$b_d = f_t \cdot c_f \cdot b_t. \tag{5}$$

Hence, the change in the bunny population can be expressed as:

$$\Delta b = b_t \cdot g_b \cdot (1 - \frac{b_t}{k_b}) - f_t \cdot c_f \cdot b_t = b_t(g_b \cdot (1 - \frac{b_t}{k_b}) - f_t \cdot c_f). \tag{6}$$

We need to repeat this process for the fox population as both depend on *each other*. Just as for the bunnies, the change in fox density from one time step to the next is the difference between the number of foxes gained (by reproduction) and the foxes lost (by starvation or a random mortality event).

$$\Delta f = f_g - f_d \tag{7}$$
$$f_{t+1} = f_t + \Delta f \tag{8}$$

The foxes gained are determined by the number of surviving foxes who then reproduce. As the number of hunted bunnies is $f_t \cdot c_f \cdot b_t$, and each fox consumes c_f bunnies, we know that the number of foxes who survived is $b_{hunted}/c_f = f_t \cdot b_t$.

$$f_g = (f_t \cdot b_t) \cdot g_f \tag{9}$$

At the same time, the number of foxes lost can be modeled by the expected loss due to the mortality event at each time step. Note that this approximation now contains no randomness from the mortality event but instead deducts a constant average fraction from the fox population (the resulting limitations are described further below).

$$f_d = f_t \cdot m_f. \tag{10}$$

So the overall change in the fox population from one time step to the next is

$$\Delta f = f_t \cdot b_t \cdot g_f - f_t \cdot m_f = f_t(b_t \cdot g_f - d_f) \tag{11}$$

We are now trying to find the points where $b_t = b_{t+1}$ and $f_t = f_{t+1}$ which only happens for $\Delta b = \Delta f = 0$. In class, we were able to model this as a cobweb plot as the result depended on a single variable only, but now we have two variables (the fox and the bunny density), hence we cannot use this one-dimensional plot. Instead, we can solve a system of two equations which is the equivalent of looking for intersection points between $y = x$ and the state transition function for a 1D mean-field approximation. Let us start with the equation for the fox population.

$$0 = f_t(b_t \cdot g_f - m_f) \tag{12}$$

This is either 0 if $f_t = 0$ or if $b_t \cdot g_f - m_f = 0$. If $f_t = 0$, the fox population has died out, leading to uninhibited bunny growth and a stable point where $f_t = 0$ and $b_t = k_b$. We want to investigate the second case because this allows bunnies and foxes to coexist. We can rearrange this to express the bunny-related parameter b_{t*} (the asterisk implies that this is the special constant average value for which the average fox density remains constant) in terms of the fox-related ones:

$$b_{t*} \cdot g_f = m_f \tag{13}$$

$$b_{t*} = \frac{m_f}{g_f} \tag{14}$$

So in order for the fox population to stay constant, the ratio of the fox death rate to the growth rate needs to be equal. This sounds very plausible as any deviation would either lead to increased growth (if $g_f > m_f$) or death (if $g_f < m_f$). Let us use the second equation:

$$0 = b_t(g_b \cdot (1 - \frac{b_t}{k_b}) - f_t \cdot c_f) \tag{15}$$

Again, this would be zero if $b_t = 0$ or when $g_b \cdot (1 - \frac{b_t}{k_b}) - f_t \cdot c_f = 0$. For the former, which implies that the bunny population has completely died out, the fox population would also die out amid the lack of food, so $d_b = m_f = 0$ is another stable state here. Investigating the co-existing case further:

$$g_b \cdot (1 - \frac{b_t}{k_b}) - f_{t*} \cdot c_f = 0 \tag{16}$$

$$g_b \cdot (1 - \frac{b_t}{k_b}) = f_{t*} \cdot c_f \tag{17}$$

$$\tag{18}$$

Substituting b_{t*} from above:

$$g_b \cdot (1 - \frac{\frac{m_f}{g_f}}{k_b}) = f_{t*} \cdot c_f \tag{19}$$

$$f_{t*} = \frac{g_b}{c_f}(1 - \frac{m_f}{k_b \cdot g_f}) \tag{20}$$

So for our given set of parameters:

$$b_{t*} = \frac{m_f}{g_f} = \frac{0.03}{0.1} = 0.3 \tag{21}$$

and

$$f_{t*} = \frac{g_b}{c_f}(1 - \frac{m_f}{k_b \cdot g_f} = \frac{0.3}{0.3}(1 - \frac{0.03}{100 \cdot 0.1})) = 0.997. \tag{22}$$

This means that the average bunny and fox populations reach and maintain a stable co-existing state for the given set of parameters if and only if the bunny density is equal to 0.3, and the fox density is equal to 0.997. The theoretical results of this mean-field approximation are clearly very different from the empirical results. This makes sense as the MFA has some strong simplifying assumptions as it reduces the entire grid to a single cell that only contains a single average value of the fox and the bunny density. On this field, diffusion does not matter as the entire bunny and fox population is contained within it. However, in the simulation, space matters: my empirical analysis for the different diffusion parameters have shown that the densities are spatially correlated, meaning that high fox densities appear close to densely bunny-populated areas. The MFA also assumes a mortality that is directly proportional to the fox population, rather than randomly wiping out certain fox populations (which may vary in density).

This means that we have an unstable steady state at $b_t = 0.3$, $f_t = 0.997$ for which every disruption (e.g. diffusion or random mortality) leads us away from it. While this theoretical equilibrium of suggests a counter-intuitive scenario with more foxes than bunnies, it is consistent with the mean-field equations. In this model, the very low bunny density severely limits *both* the predation rate and the fox reproduction rate. The foxes exist at a higher density but in a state of severely limited resources, where their growth is perfectly balanced by mortality. With these parameters, foxes are always at the brink of starvation, just so being able to maintain their population. Therefore, this MFA does not predict the actual population densities observed in the spatial simulation. The simulation shows that spatial effects allow convergence to happen at much higher population levels.

4.0.1 Theoretical Effect of g_f on Converged Densities

To compare the results of the theoretical and the empirical solutions, we can plot the effect of increasing the fox and the bunny growth rate on the converged populations. As

$$b_{t*} = \frac{m_f}{g_f}, \tag{23}$$

the fox growth rate is inversely proportional to the converged bunny population. This makes sense as a higher growth rate can be interpreted as a higher 'efficiency' (the ratio of the number of offspring to the number of bunnies consumed), which implies that the fox population can sustain itself on a lower number of bunnies. Additionally,

$$f_{c*} = \frac{g_b}{c_f}(1 - \frac{m_f}{k_b \cdot g_f}) \tag{24}$$

tells us that as the fox growth rate increases, so does the converged fox population (as the denominator of the term we subtract increases). This seems very counterintuitive but a higher fox growth rate means that more offspring are generated with the same amount of bunnies consumed, allowing a larger population of foxes to be sustained.

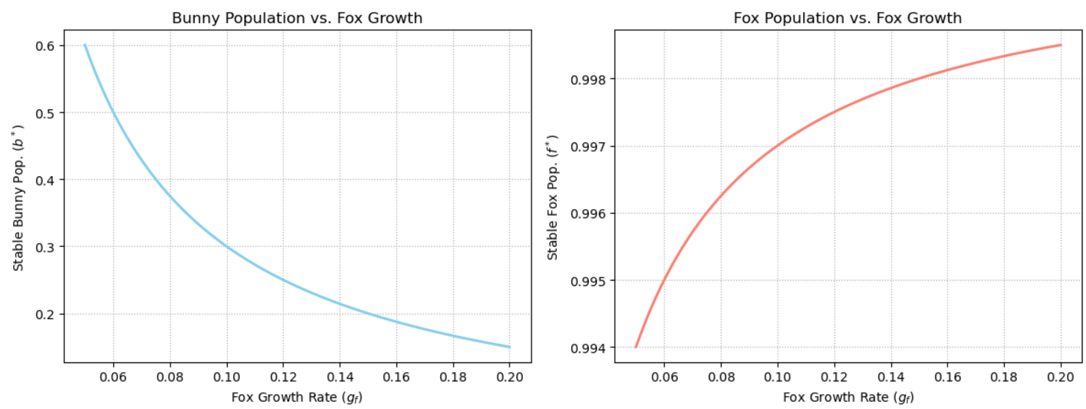


Figure 16: Theoretical Relationships between the fox growth rate and the stable population sizes.

5 Conclusion

This empirical and theoretical investigation of the predator-prey reaction-diffusion model revealed several new insights into the model's prey-predator dynamics. The spatial simulation demonstrated that both species can persist through emergent oscillatory behavior, with the transition to stable oscillations occurring at a critical bunny growth rate threshold of $g_b \approx 0.04$. This point, which was verified through Fast Fourier Transform analysis, represents the emergence of a cyclic behavior where bunny abundance enables fox growth, which subsequently suppresses bunnies, allowing the cycle to repeat.

The pattern formation analysis revealed that the diffusion rate ratio $\frac{d_b}{d_f}$ critically determines spatial organization. When $d_b \leq d_f$, populations remain scattered and unstable, while moderate ratios ($1 < \frac{d_b}{d_f} \leq 8$) enable the formation of persistent clusters where foxes surround and penetrate bunny-dense regions. Higher ratios lead to uniform distributions as rapid bunny diffusion lead to a more consistent spread across the entire system.

The mean-field theoretical analysis yielded equilibrium values of $b^* = 0.3$ and $f^* = 0.997$ for the given parameters, fundamentally different from empirical results. This discrepancy was to expect as we compare the 'correct' empirical model with a strongly simplified, non-stochastic mean-field approximation. The theoretical results predict that, within the bounds of this simplified model, the stable bunny population is determined by the fox mortality-to-growth ratio ($b^* = m_f/g_f$), but fails to capture how spatial diffusion allows both populations to maintain higher densities than the well-mixed theory predicts.

This shows that spatial structure is a fascinating mechanism enabling species coexistence. Through increased runtimes, more precise analyses of variations in the parameters, and primarily a more advanced theoretical analysis using differential equations to obtain an improved mean field approximation that follows the model behavior more closely, this work could be further extended.

6 References

Programming, J. (2021, April 23). How to determine whenever there's an oscillation. Julia Programming Language. https://discourse.julialang.org/t/how-to-determine-whenever-theres-an-oscillation/48936/9

Zandi-Mehran, N., Nazarimehr, F., Rajagopal, K., Ghosh, D., Jafari, S., & Chen, G. (2022). FFT bifurcation: A tool for spectrum analyzing of dynamical systems. Applied Mathematics and Computation, 422, 126986. https://doi.org/10.1016/j.amc.2022.126986

7 Acknowledgments

Thank you to our TA Daria with who I discussed showing how my graphs reached a stable state, and who encouraged me to do some more experimentation with the parameters.

Thank you to my friend and classmate Meri Avetisyan for productive discussions about this assignment.

Thanks to Prof Tambasco for his directions in OH and the encouragement to have fun experimenting with this system!

AI Statement: I used AI alongside the library documentations to debug parts of my code, primarily when I used libraries I had not used before, e.g. the scipy.fft or the np.krog functions for my animations.