MXB261 – Assignment 2

A SIMULATION PROJECT

Group 17 | 25/10/2021

Contents

Introduction

The study of population dynamics investigates the change in the size and structure of the population over time. It provides significant insight into important parameters in population dynamics such as death rate, birth rate, and migration rate. The Lotka-Volterra is one of the most well-known models that describe the relationship between species in the biological systems through dependent first-order nonlinear differential equations. The project considers the following differential equation system:

$$\frac{dX_1}{dt} = k_1 X_1 X_2 - k_2 X_1$$

$$\frac{dX_2}{dt} = k_3 - k_4 X_2 - k_5 X_1$$

In the system, X_1 is the population of parasites and X_2 represents the host population. Parameters k1, k2, k3, k4, and k5 represent the birth rate of X_1, the death rate of X_1, the growth rate of food source, the decay rate of food source, and food consumption by the parasites respectively. The project studies how the parameter values would affect the temporal and spatial system dynamics of the model. Task 1 and Task 2 implement parameter sweeps of multiple variables and Latin Hypercube Sampling in 3D to investigate the deterministic system. Task 4 employs a spatial agent-based model to analyze the interactions between species and explore the system dynamics.

Task 1

METHOD

The system is analyzed over the time of [0, 20], and the following parameters are fixed: $k_1 = 1, k_2 = 2$. Setting the differential equations equal to 0, we can solve the equilibrium points as followed:

$$\frac{dX_1}{dt} = X_1(X_2 - 2) = 0 => \begin{cases} X_1 = 0 \\ X_2 = 2 \end{cases}$$

If $X_1 = 0$ then:

$$\frac{dX_2}{dt} = k_3 - k_4 X_2 = 0 => X_2 = \frac{k_3}{k_4}$$

If $X_2 = 2$:

$$\frac{dX_2}{dt} = k_3 - 2k_4 - k_5 X_1 = 0 \Rightarrow X_1 = \frac{k_3 - 2k_4}{k_5}$$

We have, the system has two stable state solutions: $(\overline{X_1}, \overline{X_2}) = (0, \frac{k_3}{k_4})$ and $(\overline{X_1}, \overline{X_2}) = (\frac{k_3 - 2k_4}{k_5}, 2)$.

For part (a) of this task, the dynamics of the numerical solution for different values of delay were analyzed by running the difference scheme equation: $X_{n+1} = X_n \left(1 + hr - \frac{hrX_{n-s}}{K}\right)$ for 2000 steps. The initial value X_0 is defined on $[-sh, 0]$, meaning all X_r with $r \leq 0$ were set to X_0. The *PopGrowDelay.m* function runs the delay growth model over 2000 steps. It takes the following parameters: Initial value X_0, Growth rate r, Step size h, Carrying capacity K, delay s, and Number of steps. The results were then visualized and interpreted regarding the population size, the relationship between oscillatory peaks, period, and the delay term.

The next three parts analyzed the differential equations of the 2D representation of the parasite/food system. The system is defined on the time $[0, 20]$ with the fixed birth rate $k_1 = 1$, and death rate $k_2 = 2$. We performed multiple parameter sweep processes to analyze the effect of the rate of food growth k_3, food decay k_4, and food consumption k_5:

Parameters that do not satisfy the following conditions will be ignored:

- The population is non-negative at all times.
- The system tends to enter a stable state of either $X_1 \rightarrow 0 + Tol$ or $X_2 \rightarrow 2 \pm Tol$ with $Tol = 10^{-1}$ and $Tol = 10^{-2}$.

MATLAB ODE45 built-in function was used with the *ParasiteGrowthModel.m* function to solve the differential equations of X_1 and X_2. The results then are then applied through the above conditions to determine successful parameters. Three parameter sweeps function *SweepK3.m, SweepK3K4.m,* and *SweepK4K5.m* are used for tasks b, c, and d respectively. We used the increment of 0.5 for all sweeping functions.

RESULT
a) Delay Growth Model
After introducing the delay term into the equation, we plotted the population size against steps to see the dynamics of the system [**Figure 1**].
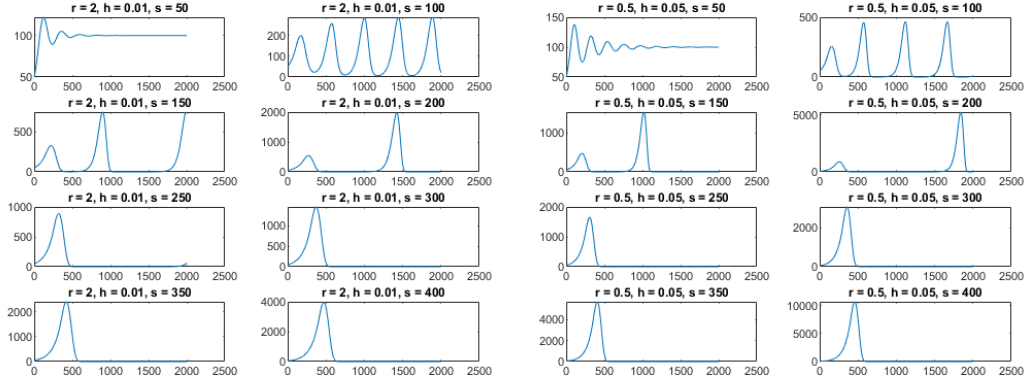
Figure 1: Population growth with delay term (s)

For both cases of growth rate r and step size h, with a small delay term ($s = 50$), the population temporarily oscillates and approaches the stable point of carrying capacity $K = 100$. As s increases to 100, we observed permanent oscillations around the carrying capacity with a larger period. However, after reaching some delay threshold, the population goes extinct because the degree of the delay is so violent. Even though we see the oscillation peaks are higher, the population is extinct after it went down to 0.

b) Parameter Sweep: Rate of Food Growth (k_3)

Setting $k_4 = 4$ and $k_5 = 3$, we perform parameter sweep for k_3. **Figure 2** shows the results of successful k_3 that satisfied following conditions:

- The population is non-negative at all times.
- The system tends to enter a stable state of either $X_1 \rightarrow 0 + Tol$ or $X_2 \rightarrow 2 \pm Tol$ with $Tol = 10^{-1}$ and $Tol = 10^{-2}$.
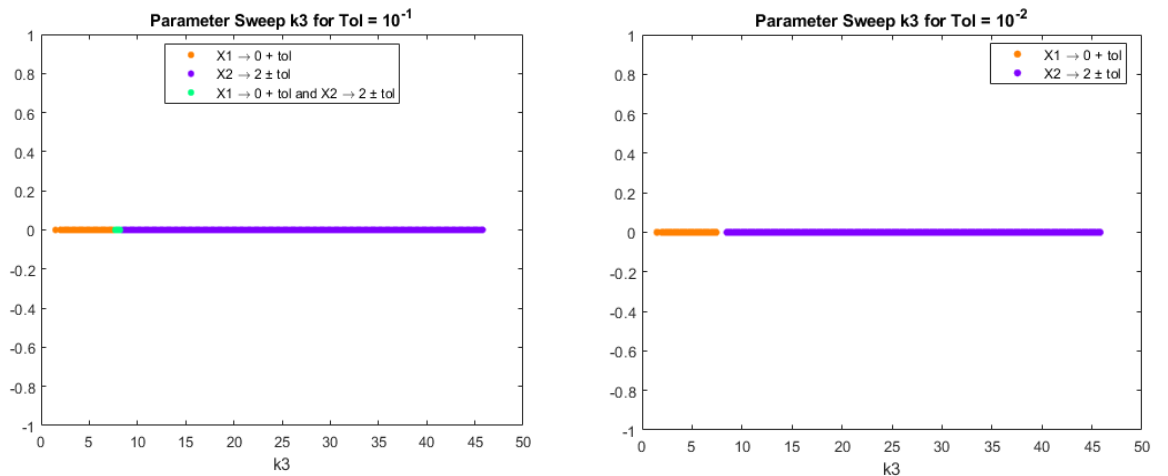

Figure 2: Parameter Sweep for k3

For $Tol = 10^{-1}$, k_3 is in the range of [1.5, 45.5]. To satisfy both conditions $X_1 \rightarrow 0 + Tol$ and $X_2 \rightarrow 2 \pm Tol$, k_3 is found to be $7.5 \leq k_3 \leq 8$. For smaller Tol ($Tol = 10^{-2}$), k_3 is still successful in the same but instead, there is no value of k_3 satisfied both conditions.

c) Parameter Sweep: Rate of Food Growth (k_3) and Food Decay (k_4)

Every parameter is fixed except k_3 and k_4, we perform sweeping for both big and small values of Tol.
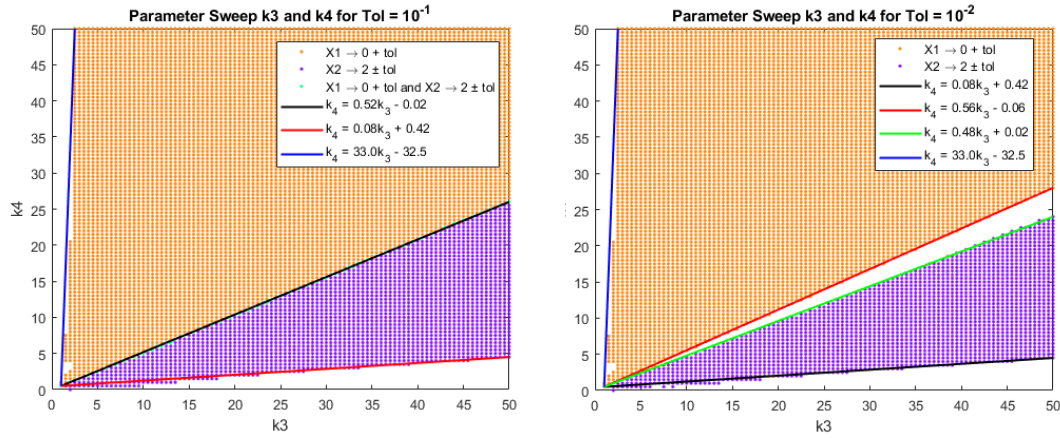

Figure 3: Parameter Sweep for k3 and k4

For $Tol = 10^{-1}$, successful parameters must be in the following range:

- $0.08k_3 + 0.42 \leq k_4 \leq 33k_3 - 32.5$ for k_3 in the range of [2.5, 50].
- If $k_4 = 0.52k_3 - 0.02$, then the result satisfies both conditions $X_1 \rightarrow 0 + Tol$ and $X_2 \rightarrow 2 \pm Tol$.

For $Tol = 10^{-2}$, k_3 and k_4 are found to be:

- $0.56k_3 - 0.06 \leq k_4 \leq 33k_3 - 32.5$ to satisfy $X_1 \rightarrow 0 + Tol$.
- $0.08k_3 + 0.42 \leq k_4 \leq 0.48k_3 + 0.02$ to satisfy $X_2 \rightarrow 2 \pm Tol$.

d) Parameter Sweep: Food Decay (k_4) and Food Consumption (k_5)

As shown in **Figure 4**, for all cases, successful k_5 must be smaller than $0.003k_4^2 - 0.25k_4 + 16$. For $0 \leq k_4 \leq 5$, k_5 needs to be bigger than $-0.78k_4 + 3.89$. For $5 \leq k_4 \leq 50$, all k_5 lying

under the black parabola satisfies the conditions. The result is the same for $Tol = 10^{-2}$, except for $k_4 = 5$, k_5 can only be equal to 0.
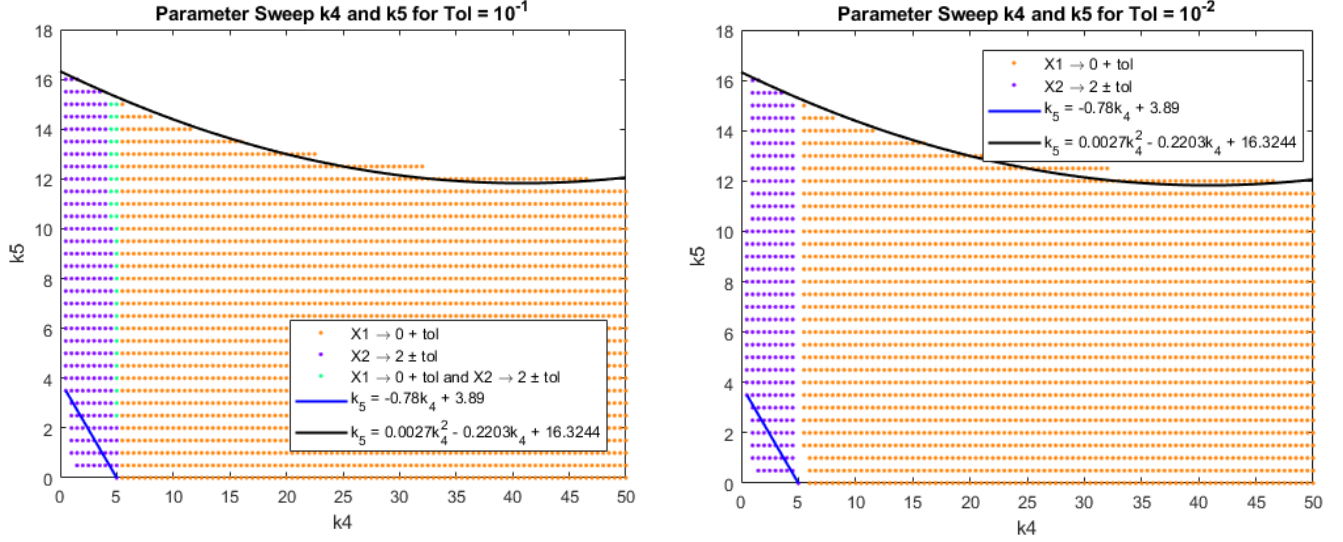

Figure 4: Parameter Sweep for k4 and k5

Part (d) set $k_3 = 10$, which is in the successful range in the result of part (b). With $k_4 = 4$, and $k_5 = 3$ (from part b), the parameters lie in the first case of part (d), which $0 \leq k_4 \leq 5$ and k_5 needs to be bigger than $0.635k_4^2 - 4.27k_4 + 5.48$. Substituted $k_4 = 4$ into the quadratic function, we have the result of -1.44, which is smaller than $k_5 = 3$. Therefore, this result is consistent with the result from part (b).

Task 2

This task uses Latin hypercube sampling in order to model the given system of equations in three-dimensional space. For this task, the parameters k_1 (birth rate) and k_2 (death rate) are set to fixed values. The following parameters are also fixed. The values of the parameters are: $k_1 = 1$ and $k_2 = 2$. Additionally, the tolerance selected for this task was chosen as 10^{-2}. After sampling. The values generated must fit the characteristics specified. The characteristics are defined as $X_1 \rightarrow 0 + \text{Tol}$ and $X_2 \rightarrow 2 \pm \text{Tol}$. The populations must also be greater than zero ($X\_1 \geq 0$ & $X\_2 \geq 0$).

Latin hypercube sampling was done in three dimensions to generate the samples for the values of $k_3, k_4 \text{and} k_5$. These parameters correspond to food growth, food decay and food consumption. The values for each of the parameters were generated in the range of [0 50], as specified in the task description.

Samples are generated using the Latin hypercube function defined in the script "lhs_impl.m". The implementation takes three arguments: the number of samples, the number of dimensions and the range of the samples being generated. The function generates the stratifications before generating the samples. The stratifications are generated by taking the number of samples to be generated and the upper bound of the range of values and dividing them. This gives the minimum and maximum value of each cell.

Each sample was then generated by selecting a random stratum and then generating a uniform random value between the lower and upper bound of the strata boundaries. This was done d times as specified in the number of dimensions for the function. These were then stored in the output matrix which has size of (n, d).

To use the samples to compute the behaviour of the parasite model, it needed to fulfill a set of defined characteristics. The inputs generated using the Latin hypercube sampling method had to be used as inputs to the ordinary differential equation function to compute the values for X_1 and X_2. This is done in the function "lhs_system" which has been defined in the script "lhs_system.m". This function takes the tolerance, time, a row vector containing the values for $X_1(0), X_2(0), k_1, k_2$ and the matrix containing the samples generated from the Latin hypercube function.

This function then iterates over the number of samples located in the LHS matrix. It calls ode45 using a function containing the equations for the parasite model. The function used is the "model" function defined in the script "lhs_system.m". This function takes the arguments tspan, $\{[X]_1(0), X_2(0)\}$, and k_1, k_2, k_3, k_4 and k_5. This will then generate a column vector with the samples for X_1 and X_2.

The values generated for X_1 and X_2 then need to be verified to be determined if the samples are successful or unsuccessful. First all the samples generated need to be checked to ensure that the values were greater than zero. Then the values are checked to sure that they fulfill the characteristics defined. If the values fit the characteristics, they are deemed successful. Success is represented in MATLAB as being a value of '1' or '2' whereas an unsuccessful sample is '0'. '1' represents a sample that satisfies the characteristic 'x1' and '2' represents a sample that satisfies the characteristic 'x2'. The success of a sample is stored in a column vector as output.

RESULTS AND DISCUSSION

To plot the data generated, 100 samples were generated and concatenated with the column vector from the "lhs_system" function. This was then used with the scatter3 function to generate visualisations of the data. In the first figure, the behaviour of k_4 shows a high concentration of points between the groups of x_1 and x_2 samples. Removing the unsuccessful populations, shows a distinct boundary between x_1 and x_2. The number of successful populations is also skewed to the right. This is consistent with expectations as food growth increases, the number of successful populations also increases.
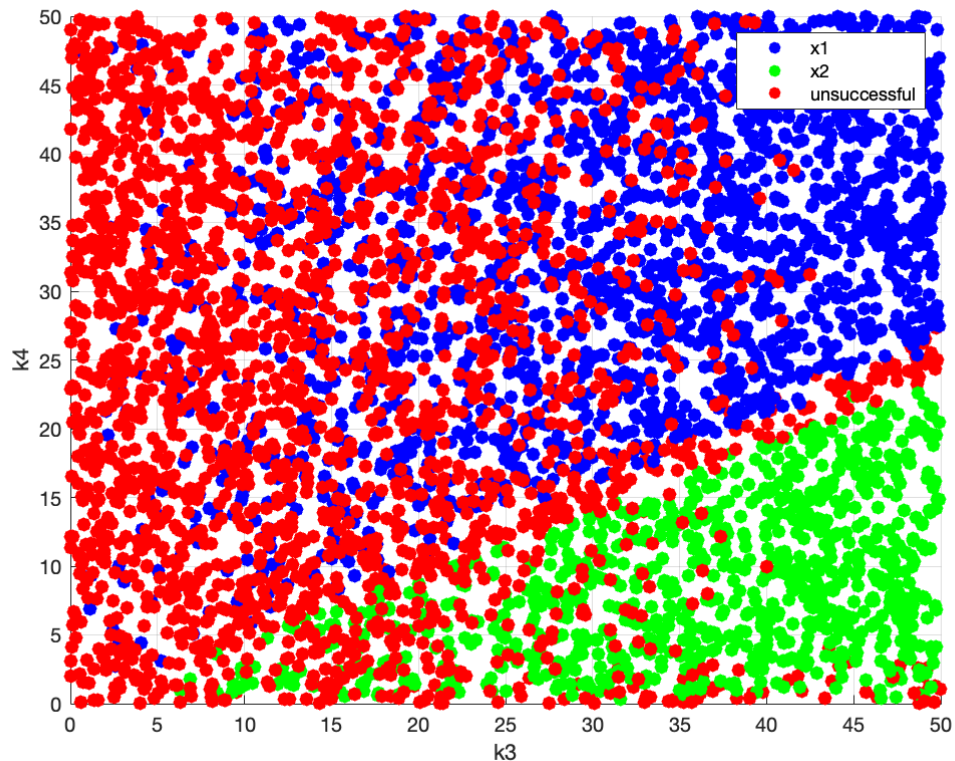

Figure 5. k3 vs k4
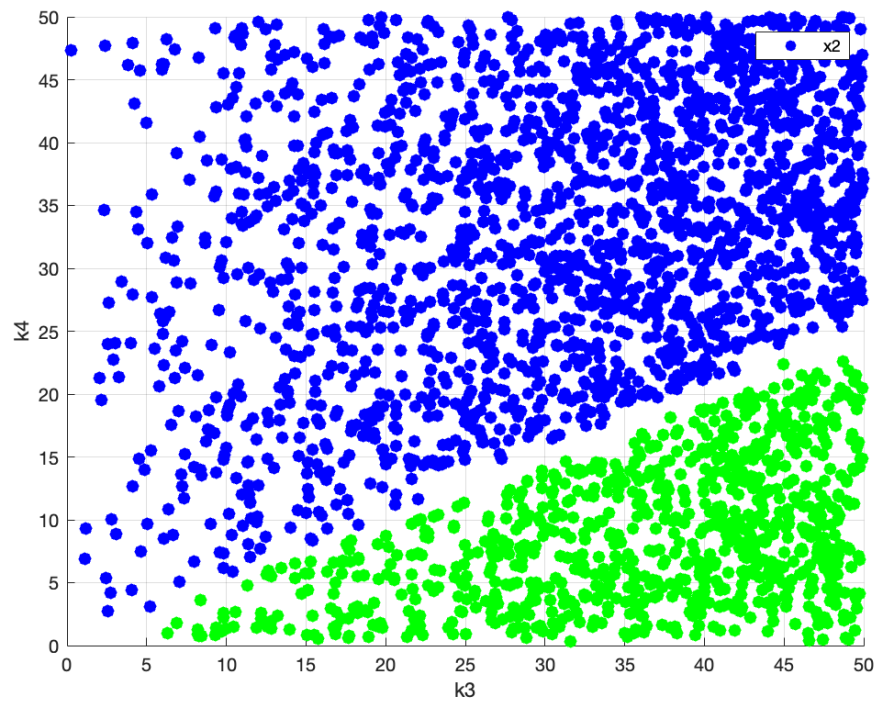
Figure 6. k₃ vs k₄, X₁, X₂ only

In the third figure which is the same visualisation with a side on view, the relationship between the variables k_3 and k_5 is made obvious. As k_3 increases, there are more successful populations. In contrast, when k_5 increases there is an increase in the number of unsuccessful populations. A clear divide can be seen between the two areas of successful and unsuccessful populations. This divide represents the equilibrium between food growth rate and food consumption.
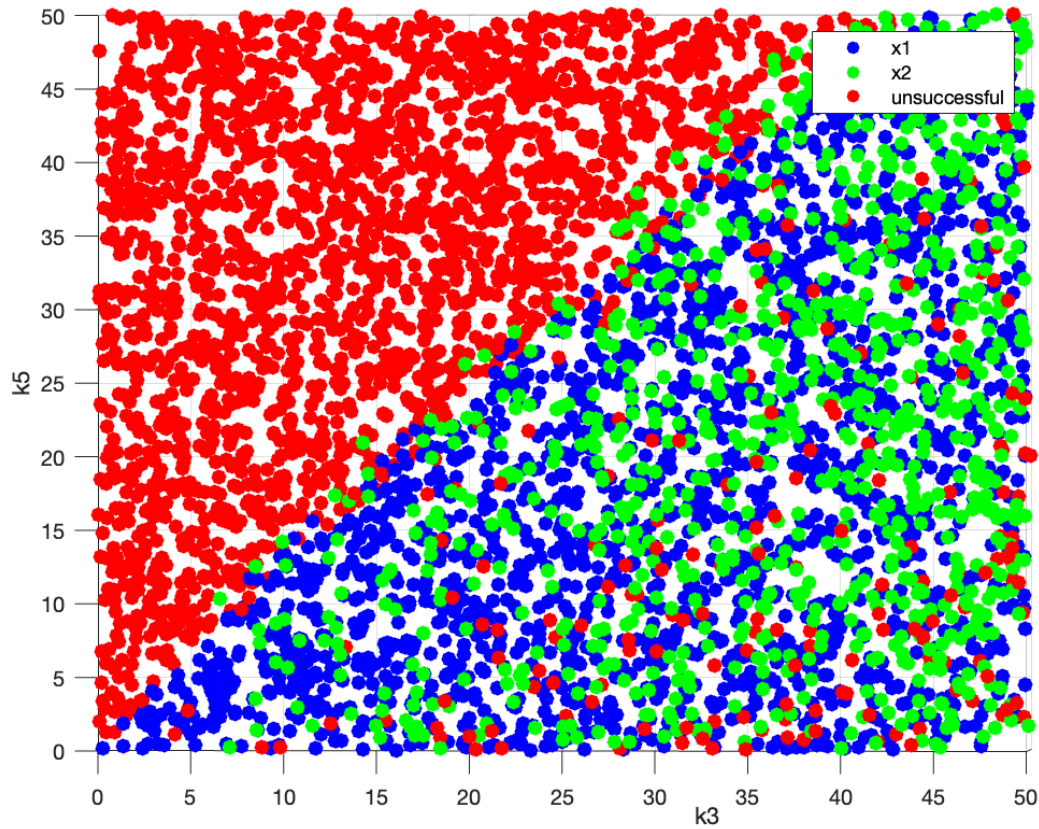
Figure 7. k5 vs k3, X1, X2, Unsuccessful

In the orthogonal perspective, it can be concluded that the primary trends that exist in the given system are between the food growth rate and the food consumption rate. As expected, when food decay rate increases, the number of unsuccessful populations also increases. This is also true for the rate of food decay.
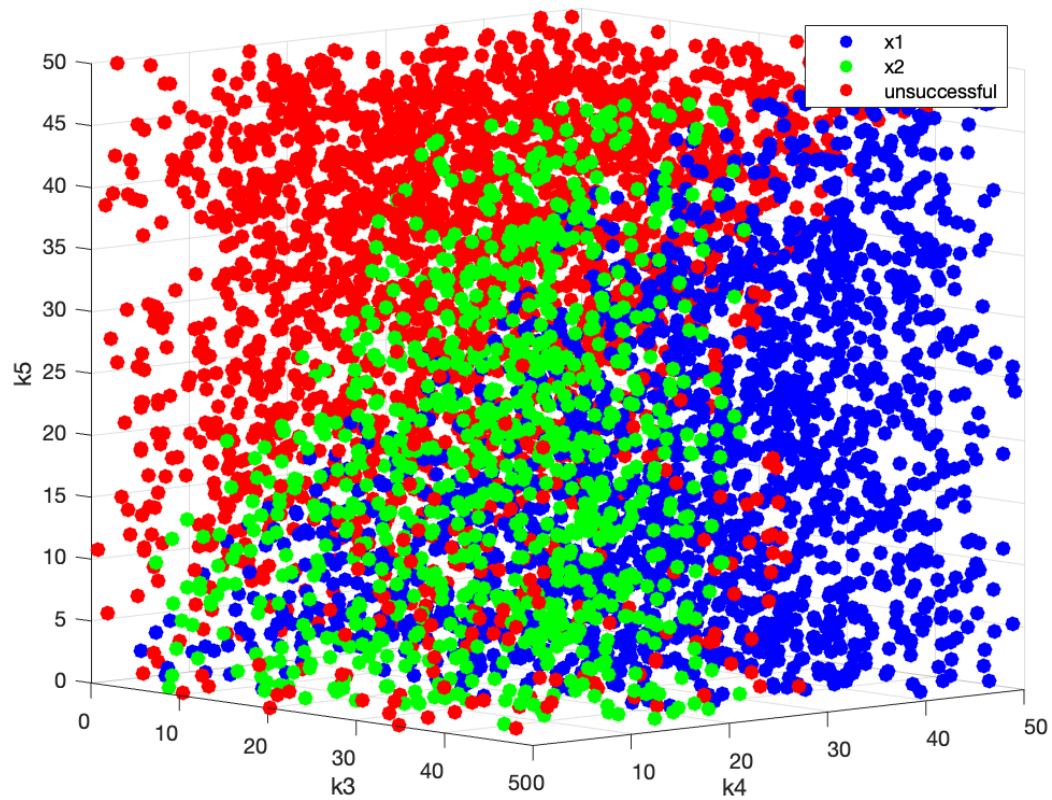
Figure 8. k3, k4, k5

In comparison to task 1, the results of the Latin hypercube modelling correspond with those of task 1. The relationships between food growth, food decay and food consumption. In task 1, the relationship between food decay and food growth is the same as the results in task 2. When food decay increases, the number of unsuccessful populations decreases.

This is also true for increases in food consumption. This can be observed in the k3, k5 Latin hypercube figure where a defined boundary is shown between the successful and unsuccessful populations. As such the results of task 2 confirm the results and the expected behaviour of the system in task 1.

Task 4

METHOD

The spatial agent-based implementation of the parasite model illustrates the interactions between species at an individual level. A 200x200 domain is created to store parasites and food agents at each iteration step. In the domain, the food agents are represented by -1's, the parasites agents are represented by 1's and the empty cell is 0. The attributes of each parasite are row, col, and age where [row, col] is the position of the parasite in the domain, and age is the life span. The attributes of each food agent are row and col. When the food or parasites are removed from the domain, their row, col values are 0. The *initial_random_position.m* function positions the food and parasites randomly. Then the simulation iterates over each parasite, generate a random walk and check for the collision. If the new neighboring cell is empty, it moves the parasite there. If the new neighboring cell is vacated by a food agent, the food is replaced by the parasite, and a new parasite is placed in the original cell (reproduction process). If there exists a parasite in the neighboring cell, the movement does not take place. After each iteration, the age of the parasite increases by 1 and row, col values are updated. A parasite would be removed if its age value is greater than the f1 value. Then the simulation iterates over each food agent. For each food, a uniform random sample u is created. If u is smaller than the f2 value, the food is removed from the domain. The domain and the arrays that contain data of food and parasites are updated at each iteration. The total number of parasites and food agents are stored after each simulation step.

RESULT

The graph shows the population of parasites and food agents over 1000 steps (red is a parasite, blue is food). The system is simulated with the selection of different parameter values (population densities and food respawn rate f3). For all the cases, the number of food agents is varied during the first 100 steps, then it maintains the stable trend of around 5000 agents. Due to the constant number of food agents respawning after each step, the food never goes extinct. With a small f3 value (f3 = 100), the parasites are extinct. As the value of food birth rate increases, there are more cases that parasites would not go extinct. However, with f3 = 200, we can see the huge gap between the population of food and parasites. The gap is narrow as the value of f3 increases. From the graph, we can conclude that food-placement strategies would exert a more significant effect on the characteristics of the system. It is easier for the system to reach an equilibrium state with high food birth rate.
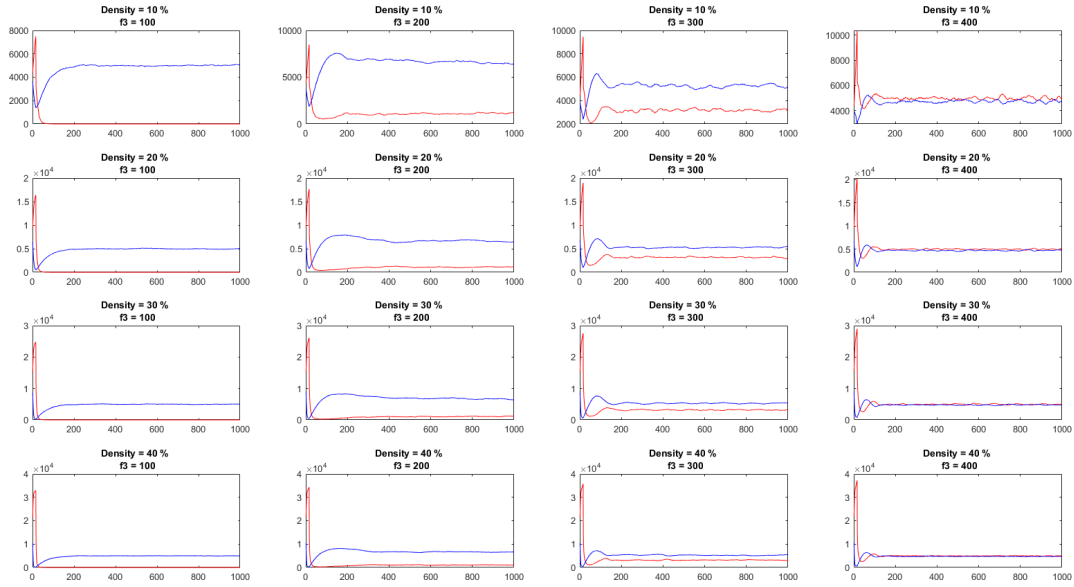
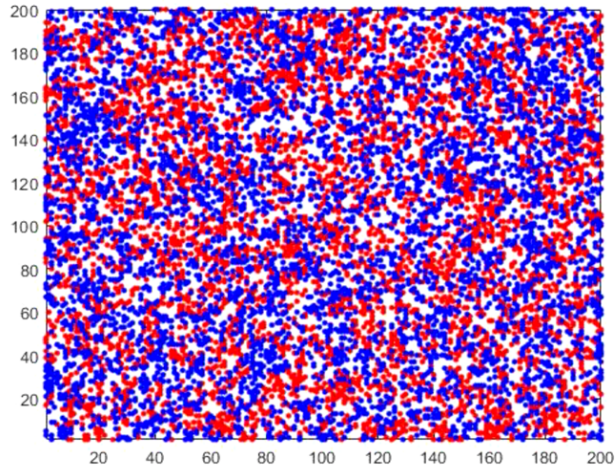**Figure 8**: Parasite (red) and food (population) vs step

Figure 9: Parasite (red) and food (population) at a regular time point in the equilibrium scenario

Comparison

DISCUSSION

As mentioned above, the system equilibrium solutions are $(\overline{X_1}, \overline{X_2}) = (0, \frac{k_3}{k_4})$ and $(\overline{X_1}, \overline{X_2}) = (\frac{k_3 - 2k_4}{k_5}, 2)$. In the case of $k_3 = 2k_4$, two stages are equal with $(\overline{X_1}, \overline{X_2}) = (0, 2)$. This result

remains consistent in all parameter sweeps performed in Task 1. In the first sweep with one parameter k_3, we observed the merged point of the two stages is in the range of $7.5 \le k_3 \le 8$, which is approximately equal to $2k_4$. The second sweep showed a linear line $k_4 = 0.52k_3 - 0.02$ separating two states. And the final instance has the merged point at $k_4 = 5 = \frac{1}{2}k_3$. It can be inferred from the results that:

- If $k_3 = 2k_4$, the rate of growth of the host has doubled the rate of its decay, the population tends to get stable at $(0, 2)$.
- If $k_3 < 2k_4$, the rate of growth of the host is less than double of the rate of its decay, the population will approach $\left(0, \frac{k_3}{k_4}\right)$, meaning the parasite will go extinct while the host gets to $\frac{k_3}{k_4}$.
- If $k_3 > 2k_4$, the rate of growth of the host is more than double of the rate of its decay, the population will approach $\left(\frac{k_3 - 2k_4}{k_5}, 2\right)$.

Therefore, we can observe that the parasite population will go extinct if the food growth rate tends to 0, food decay rate tends to infinity or the food consumption rate goes to infinity.

This result is consistent with the findings in Task 2 where we implemented the Latin Hypercube Sampling techquique. We can see in Figure 6 and Figure 7 that as the rates of successful simulations is directly proportional with k_3 (food growth rate). And as k_5 (food consumption) increases, the number of successful cases reduce accordingly.

Conclusion

Tasks 1 and 2 explore the system dynamic with the implementation of differential equations and produce deterministic solutions. The consistent result is obtained with fixed parameters and interaction rules. The relationship between the behavior of the system and its parameters were investigated. Task 4 studies the behavior of the system dynamics when interactions occur at an individual level. The spatial agent-based implementation reveals similar properties as the deterministic method. Overall, these two techniques give a better understanding of the parasite model and showcase the impact of different parameters on the species population.