

Research on Evolutionary Dynamics of Artificially  
Evolving Virtual Organisms in Perlin Noise  
Generated Environments

Ricardo Alberto Carrillo  
(Master's Program in Computer Science)

Advised by Claus Aranhna de Castro

Submitted to the Graduate School  
of Science and Technology  
Degree Programs in Systems and  
Information Engineering  
in Partial Fulfillment of the Requirements  
for the Degree of Master of Engineering  
at the  
University of Tsukuba

September 2024

## Abstract

This thesis presents the development and exploration of a software simulation (Evo Island) designed to study evolutionary processes within virtual environments. Taking inspiration from the *E. coli* Long-Term Evolution Experiment (LTEE), this project aims to replicate traditional evolutionary studies in silico using computational techniques, allowing for very fast observational capabilities. By creating a digital ecosystem where agents evolve over time, the simulation provides a platform for examining mechanisms such as adaptation, mutation, competition, speciation, and natural selection.

The primary research objective is to evaluate the speciation of evolving AI agents when embedded inside a Perlin noise environments when compared to simpler terrains. Additionally, the study examines the effects of adding various complex agent behaviors and attributes to the simulation: inter-agent conflict, environmental resilience, aging, and food metabolism. These behaviors and attributes affect survival and reproductive success, creating interesting trade-offs that agents must navigate in order to thrive.

Results from multiple simulation runs indicate that embedding these agents in realistic Perlin noise environments significantly increases genetic diversity and encourages the emergence of unique adaptive strategies. The activation of new more realistic attributes further amplifies these effects, demonstrating how the interaction of environmental complexity and organism dynamics drive evolutionary divergence. Each effect also created interesting collective and emergent effects visible in simulation visualizations.

This research presents a different approach to studying evolutionary biology by integrating terrain generation with organism modeling on a computational model, allowing control of the parameters associated to ecosystems that are not possible to do in reality. Finally, this thesis offers detailed documentation of the simulation framework, serving as a resource for researchers interested in utilizing this simulation for future studies.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Background . . . . .	1
1.2	Problem Statement . . . . .	1
1.3	Research Objectives . . . . .	2
1.4	Thesis Structure . . . . .	2
<b>2</b>	<b>Literature Review</b>	<b>3</b>
2.1	The LTEE Experiment . . . . .	3
2.2	Alife . . . . .	4
2.3	Evolutionary Algorithms and Genetic Representations . . . . .	5
2.4	Speciation in Asexually Reproducing Organisms and Alife Agents . . . . .	8
2.5	Possible Solution to Simulated Asexual Speciation Inspired by NEAT Algorithm . . . . .	8
2.6	Procedural Environment Generation Using Perlin Noise . . . . .	9
<b>3</b>	<b>Method, Algorithms, and Technical Manual</b>	<b>12</b>
3.0.1	Git Repository . . . . .	12
3.1	Main Algorithms . . . . .	12
3.2	Setting Up the Simulation . . . . .	17
3.2.1	Config File . . . . .	17
	Example Config File . . . . .	17
3.3	Running the Simulation . . . . .	18
3.3.1	Simulation Result Artifacts . . . . .	18
3.3.2	CSV Metrics Collection . . . . .	18
3.3.3	Analysis Notebooks . . . . .	19
3.3.4	Visualizations . . . . .	19
<b>4</b>	<b>Experimental Results</b>	<b>20</b>
4.1	Common Experimental Setup . . . . .	20
4.2	Experiment 1: Replicating the LTEE Petri Dish with simulation of Hardiness and Strength . . . . .	20
4.2.1	Goal: . . . . .	20
4.2.2	Hypothesis: . . . . .	21
4.2.3	Experiment Configuration . . . . .	21
4.2.4	Observation of Fan Shaped Pattern . . . . .	30
4.2.5	Summary of Findings: . . . . .	30
4.2.6	Interpretation: . . . . .	30
4.2.7	Conclusion: . . . . .	31
4.3	Experiment 2: Effect of Agent Aging on Speciation and Elitism . . . . .	32

4.3.1	Goal:	32
4.3.2	Hypothesis	32
4.3.3	Configuration	32
4.3.4	Summary of Findings:	42
4.3.5	Interpretation	42
4.3.6	Conclusion:	42
4.4	Experiment 3: Effect of Map Complexity on Agent Diversity	43
4.4.1	Goal:	43
4.4.2	Hypothesis	43
4.4.3	Configuration	44
4.4.4	Experiment 3 Maps	44
4.4.5	Summary of Findings:	55
4.4.6	Interpretation:	55
4.4.7	Conclusion:	55
4.5	Experiment 4: Effect of Food Simulation on Speciation	56
4.5.1	Goal:	56
4.5.2	Hypothesis:	56
4.5.3	Configuration	56
4.5.4	Summary of Findings:	67
4.5.5	Interpretation:	67
4.5.6	Conclusion:	67
<b>5</b>	<b>Conclusion</b>	<b>68</b>
5.1	Restatement of Research Statement and Objectives	68
5.2	Summary of Experiments	68
5.2.1	Experiment 1: Replicating the LTEE Petri Dish with Evolution of "Hardiness" and "Strength"	68
5.2.2	Experiment 2: Effect of Agent Aging on Speciation and Elitism	68
5.2.3	Experiment 3: Effect of Map Complexity on Agent Diversity	69
5.2.4	Experiment 4: Effect of Food Simulation on Speciation	69
5.3	Limitations and Future Work	69
<b>Acknowledgements</b>	<b>70</b>	
<b>Bibliography</b>	<b>70</b>	

# List of Figures

2.3	Basic Process of Evolutionary Computation[1] . . . . .	6
2.4	Vector Representation of Agent Genome and Mutation Method . . . . .	7
2.5	Speciation Method using the L2 Distance and clustering around species representatives . . . . .	9
2.6	Examples of Patterns created by Perlin Noise[2] . . . . .	10
2.7	Example Simulation Map Created by Perlin Noise . . . . .	11
4.1	This is the map used for experiment 1. It is designed to simulate the experimental setup of the LTEE and has graduated bands of increased survival difficulty, analogous to the antibiotic concentration. . . . .	22
4.2	Experiment 1 - Agent Strength over 1000 Time Steps. Note that at about 300/1000 mark of the simulation, the strength attribute was maximized for all agents. This capping of the strength value greatly reduced diversity, and eventually gridlocked the simulation when no new agents could occupy new cells from existing agents. . . . .	23
4.3	Experiment 1 - Agent hardiness over 1000 time steps. Notice that the agents stratify themselves by the required hardiness needed to occupy areas of higher difficulty. Agent hardiness steadily increased from left to right. This suggests adaptation, and geographic isolation occurred due to geographic barriers. . . . .	24
4.4	Experiment 1 - Agent Age over 1000 Time Steps. This result heavily suggests that the grid locking of the simulation was caused by elitism. Once agents maxized their strength attribute, they could not be displaced by any newer agent. With nothing to check them, the same agents continued living until the end of the simulation. . . . .	25
4.5	Experiment 1 - Agent Species distribution over 1000 Time Steps. The distribution of species over time showed how after the strength gene was maximized, diversity fell dramatically. . . . .	26
4.6	Experiment 1 - Genetic Distance from common ancestor over 1000 Time Steps. Note how after the 3rd image, the genetic distance from the common ancestor for all agents stopped changing. This implies that the grid locking of the simulation due to elitism, hampered the evolutionary process. . . . .	27
4.7	Experiment 1 - Average gene values over 30 trials. While both attributes increased over time, this shows that the strength attribute was under the most significant evolutionary pressure. Once it was maximized, and due to the elitism problem, the hardiness gene stopped changing as well. . . . .	28
4.8	Experiment 1 - Average cause of Agent Death over 30 Trials. This again shows that the strength attribute was under the most significant evolutionary pressure. . . . .	28

4.9 Experiment 1 - Average number of species per time step over all trials. This graph illustrates how the saturation of the strength gene reduced diversity until it eventually grid locked the simulation. Across all experimental trials, this saturation occurred roughly at time step 300. At that point, the number of species decreased until it stabilized at an equilibrium. This suggests that elitism is capping the diversity of the simulation. . . . .	29
4.11 Experiment 2 - Agent death probability is calculated using a sigmoid function. The chance of an agent dying per turn is a function of its current age with respect to its maximum genetic lifespan. . . . .	32
4.12 Experiment 2 - Agent Strength over 1000 Time Steps. Agent Strength still reached a cap, but we notice a noisy static effect on the images, these are dying agents, dying due to the probabilistic aging mechanism. . . . .	33
4.13 Experiment 2 - Agent hardiness over 1000 time steps. As before we note that agents stratify themselves according to the required hardiness of each difficulty region. Additionally we note that in several simulations, the agents were able to reach even the most difficult regions of the map when they could not before. This would suggest that the elimination of elitism allowed the hardiness gene to continue developing, allowing the agents to continue the adaptation process and explore farther. . . . .	34
4.14 Experiment 2 - Agent Maximum Lifespan over 1000 Time Steps. Here we see there was a clear positive selection pressure for longer lifespans. A positive trend towards increasing maximum lifespan was observed as the simulation progressed. . . . .	35
4.15 Experiment 2 - Agent Age over 1000 Time Steps. This result heavily suggests that the grid locking of the simulation was caused by elitism. With the addition of agent aging and spontaneous death, the average age for each agent is vastly reduced, but still allows for continued propagation of the agent population as a whole. . . . .	36
4.16 Experiment 2 - Species Distribution. We note that the addition of agent aging, while not reducing the amount of species itself, caused a reduction in the variation of species distribution especially as the simulation reached later stages. Unlike before where several species were visible, we see the green species take over the majority of the map. We suggest that this is because without the ability of elites to lay claim to an area, it was much more difficult for any particular species to hold on to a particular region. . . . .	37
4.17 Experiment 2 - Genetic Distance from common ancestor over 1000 Time Steps. We note with the addition of aging, this visualization did not stop changing at any point of the simulation. This suggest that the elimination of elitism via aging, allowed the simulation to avoid the gridlock seen before. . . . .	38
4.18 Experiment 2 - Average gene values over 30 trials. We note that unlike before, the maximization of the strength gene did not cause a plateauing in the other genes. Both hardiness and maximum lifespan both continued slowly increasing even when strength is maximized. . . . .	39
4.19 Experiment 2 - Average cause of Agent Death over 30 Trials. Here we see that even with the ability to die from aging, the leading cause of death was still conflict with other agents over territory. This would explain why the largest genetic pressure was still on the strength gene. . . . .	40

4.20 Experiment 2 - Average number of species per time step over all trials. Here we note an interesting result. We still see the same fall in overall species numbers around step 300 once the strength gene is maximized, but the number of species does not stabilize afterward as before. We now see that the number of distinct species was still increasing at program termination, and suggests that aging successfully uncapped the simulation's diversity. . . . .	41
4.22 Experiment 3 Petri Map - Agent Species Distribution over 1000 Time Steps . . . . .	47
4.23 Experiment 3 Petri Map - Average number of species per time step over all trials. . . . .	48
4.24 Experiment 3 Perlin Map - Agent Species Distribution over 1000 Time Steps . . . . .	49
4.25 Experiment 3 Map - Average number of species per time step over all trials. . . . .	50
4.26 Experiment 3 Random Map - Agent Species Distribution over 1000 Time Steps . . . . .	51
4.27 Experiment 3 Random Map - Average number of species per time step over all trials. . . . .	52
4.28 Experiment 3: Comparison of the mean number of species per map, over all trials. . . . .	53
4.29 Experiment 3: Comparison of the max number of species per map, over all trials. . . . .	54
4.30 Experiment 4 - Agent Strength over 1000 Time Steps. . . . .	57
4.31 Experiment 4 - Agent hardiness over 1000 time steps. . . . .	58
4.32 Experiment 4 - Agent Maximum Lifespan over 1000 Time Steps. . . . .	59
4.33 Experiment 4 - Agent Age over 1000 Time Steps. . . . .	60
4.34 Experiment 4 - Agent Average Metabolism over 1000 Time Steps. . . . .	61
4.35 Experiment 4 - Agent Average Reproduction Threshold over 1000 Time Steps. . . . .	62
4.36 Experiment 4 - Agent Species Distribution. . . . .	63
4.37 Experiment 4 - Genetic Distance from common ancestor over 1000 Time Steps. . . . .	64
4.38 Experiment 4 - Average gene values over 30 trials. . . . .	65
4.39 Experiment 4 - Average cause of Agent Death over 30 Trials. . . . .	65
4.40 Experiment 4 - Average number of species per time step over all trials. . . . .	66

# Chapter 1

## Introduction

### 1.1 Background

The study of evolutionary biology provides our understanding of the origin, diversity, and complexity of life on Earth. But historically, studying evolutionary processes has required long periods of data collection, often spanning decades, to observe changes in natural environments. But relatively recent advances in computational technology, enable more efficient and detailed studies of these processes in simulation.

This thesis introduces a computational model, *Evo Island*, inspired by the ongoing *E. coli Long-Term Evolution Experiment* (LTEE). The LTEE is an ongoing study on evolutionary processes now entering its 36th year, and has yielded insights into mutation rates, adaptation, and genetic drift of *E. Coli* bacteria over very long periods of time. Building on the LTEE methodologies, this project uses digital simulations of asexually reproducing organisms, similar to simple bacteria, to replicate long-term evolutionary processes in a controlled virtual environment. This approach enables the examination of emergent evolutionary dynamics across various conditions in significantly less time than traditional methods.

### 1.2 Problem Statement

Traditional evolutionary studies, including the LTEE, are valuable but have limitations due to the slow pace of genetic changes and logistical difficulties in long-term experiments. Computational simulations and the study of artificial life (ALife) offer a solution, allowing the rapid simulation of many generations with the ability to manipulate variables and conditions not feasible in physical biological experiments.

However, creating a digital simulation that faithfully replicates the complexity of natural environments and biological interactions is very challenging. The simulation must balance the complexity of ecological and genetic interactions with computational feasibility in order to provide meaningful insights into evolutionary processes, and the interaction between the environment and evolutionary outcomes.

## 1.3 Research Objectives

The main goals of this thesis are:

- **Develop an Evolutionary Simulation Framework:** Create a flexible computational model that simulates evolutionary processes within dynamically generated terrains. This framework will provide a variety of environmental conditions, enabling the study of digital organisms and their evolutionary trajectories.
- **Investigate the Impact of Perlin Noise Environments:** Examine how the complexity of virtual terrains generated using Perlin Noise procedural generation influences evolutionary outcomes for these virtual creatures. Specifically, this objective focuses on understanding how such environments affect the adaptation strategies, survival rates, and overall genetic diversity of digital organisms.
- **Assess Genome Diversity and Facilitation of Allopatric Speciation in Simulation:** This includes investigating whether these Perlin Noise environments are sufficiently realistic to allow for the emergence of allopatric speciation and geographic isolation of species in simulation.

## 1.4 Thesis Structure

This thesis is divided into six chapters:

1. **Chapter 1: Introduction** — Background, problem statement, and research objectives.
2. **Chapter 2: Literature Review** — Review of existing research on evolutionary algorithms, agent-based models, and procedural terrain generation.
3. **Chapter 3: Method and Technical Manual** — Implementation details and guide to the simulation software developed for this project.
4. **Chapter 4: Experimental Results and Discussion** — Results from various simulation scenarios, analysis and interpretation of results for each experiment
5. **Chapter 5: Conclusion and Future Work** — Summary of findings and potential areas for further research and development.

# Chapter 2

## Literature Review

This chapter explains the main concepts which give context to this experiment, and the main technical methods used to implement the simulation.

### 2.1 The LTEE Experiment

This simulation drew inspiration from a wide collection of existing studies, but one of the most influential was the LTEE. The Long-Term Evolution Experiment (LTEE) initiated by Richard Lenski in 1988 aims to observe evolutionary processes over an extended period. The experiment involves twelve populations of *Escherichia coli* bacteria, which have been propagated for over 70,000 generations under controlled conditions[3]. The primary objectives of the LTEE are to investigate the dynamics of adaptation, the repeatability of evolutionary processes, and the genetic basis of evolutionary changes. One of the key findings of the LTEE is the evolution of citrate utilization and metabolism in one of the populations, which serves as an example of adaptive evolution.

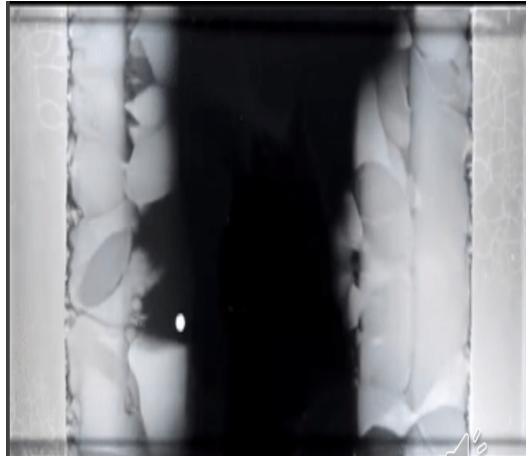
Cooper and Lenski (2000) explored the population genetics of ecological specialization in evolving *Escherichia coli* populations within the LTEE. Their study demonstrated how different *E. coli* populations adapted to the same environment through divergent evolutionary paths, leading to ecological specialization. By mapping these paths, they showed how this specialization was driven by mutations that conferred specific advantages in the given environmental conditions. [5].

Then, Elizabeth Pennisi's article "The Man Who Bottled Evolution" provides another detailed overview of the LTEE and its significance in understanding evolutionary biology. The LTEE's continuous observation of *E. coli* over thousands of generations revealed how new traits can emerge and how populations can adapt to changing environments. She detailed the discovery of citrate metabolism in one of the *E. coli* populations, a trait not typically exhibited by *E. coli* in the wild, and shows the power of natural selection and mutation over long periods, [3].

And more recently, Lamrabet et al. (2019) focused on the changes in intrinsic antibiotic susceptibility (antibiotic resistance) exhibited by *E. coli* during the LTEE. Their study revealed that *E. coli* populations evolved varying levels of resistance to different antibiotics over the course of the experiment, and these resistance traits were passed to their offspring. A key observation from their study was the formation of characteristic fan-shaped growth patterns visible in the growth medium, starting from single mutants [4] when the bacteria were exposed to graduated levels of antibiotic in their growth substrate. These patterns indicated that beneficial traits, such as antibiotic resistance, were spreading throughout the population from initial extreme mutations of single individuals and could be clearly visualized by the experimental setup. This phenomenon was particularly significant for the simulation explored in this thesis, as it provided a clear visual confirmation that could



(a) LTEE Experimental Setup: The color banding indicate antibiotic concentration.[4]



(b) LTEE Growth Pattern: Growth of *E. coli* was limited until resistant mutants appeared.[4]

be searched for to determine if the phenomenon of a single beneficial mutation being propagated through a population was successfully captured by the model.

By enabling the observation of genetic changes over multiple generations in a fast computational environment, and under varying environmental conditions, the simulation framework aimed to emulate the LTEE and facilitate the creation of dynamic and realistic evolutionary scenarios.

## 2.2 Alife

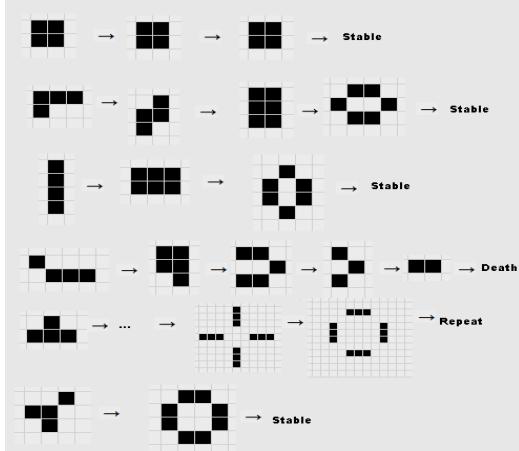
To provide further context, we wish to present the field of Artificial life. Artificial Life (Alife) is a field of study that examines life through the lens of artificial systems, aiming to understand the properties of living systems by recreating biological phenomena in computational environments. The simulation explored in this thesis would therefore be one such example application of an Artificial Life simulation. Alife encompasses various sub-disciplines, including cellular automata, artificial evolution, and digital organisms. This study belongs to the sub-discipline on artificial evolution, where digital organisms evolve in a simulated environment, and are subject to natural selection and mutation, much like their biological counterparts.

Martin Gardner's article "The fantastic combinations of John Conway's new solitaire game 'life'" summarized the famous Conway's Game of Life, a cellular automaton that demonstrated how simple rules could lead to complex behaviors and patterns. Conway's Game of Life consists of a grid of cells that can be alive or dead, and the state of each cell in the next generation is determined by a set of simple rules based on the states of its neighbors. This game became a landmark in the study of Alife and complexity, showing that unpredictable patterns could emerge from straight forward initial conditions and rules. While not a cellular automata, the simulation explored in this thesis can be thought of as a similar type of two dimensional game, where initial conditions can have a large affect on final outcomes. [6].

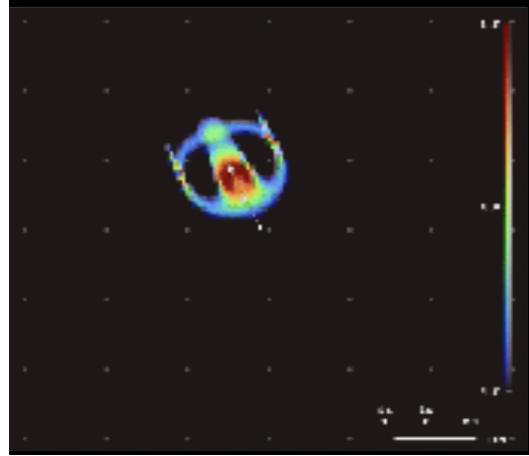
Building on these foundational concepts from Alife more recently, Bert Wang-Chak Chan's "Lenia: Biology of Artificial Life" explores a continuous extension of the cellular automaton from the game of life, called Lenia. Lenia extends the principles of Conway's Game of Life by allowing for a broader range of states and more complex interactions between cells. Lenia demonstrates how life-like behaviors and structures can emerge from simple rules applied in a continuous, rather than

discrete, space. This approach has led to the discovery of new, dynamic patterns and behaviors that resemble biological organisms more closely than previous models. Lenia provided further evidence of emergent phenomena inside of an ALife simulation, which this study also looked to reproduce.[7].

In the context of the simulation, Alife principles were applied to create digital agents that evolve over time. The simulation uses digital organisms with genomes that mutate and adapt to their environment, creating an ecology simulation able to study evolutionary dynamics and the emergence of complex traits. The use of simple rules to govern the interactions and behaviors of these digital agents draws directly from the concepts explored by Conway’s Game of Life and Lenia.



(a) Conway’s Game of Life[6]



(b) Lenia Artificial Life[7]

### 2.3 Evolutionary Algorithms and Genetic Representations

Evolutionary algorithms (EAs) are computational methods that mimic natural evolutionary processes to solve optimization and search problems. Genetic algorithms (GAs), a subset of EAs, utilize mechanisms inspired by biological genetic evolution such as genome representation, selection, mutation, and crossover.[8, 1]. In constructing the agents within this thesis’ simulation, these concepts from the study of GAs and EAs were utilized extensively.

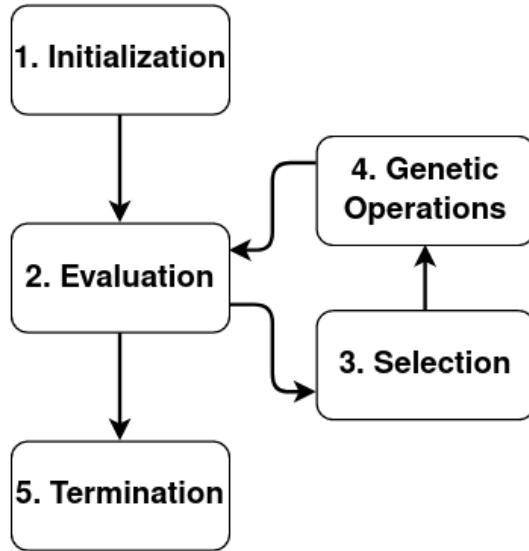


Figure 2.3: Basic Process of Evolutionary Computation[1]

Genetic algorithms typically involve several key components. Genome representation is often encoded as a string of bits, integers, or real numbers, representing the solution to a problem. Selection processes select the fittest individuals from the population to pass their genes to the next generation. Various selection strategies can be used, such as roulette wheel selection, tournament selection, or rank-based selection. Mutation introduces random changes to an individual's genome to maintain genetic diversity within the population and explore new solutions. This could involve flipping bits in a bit string, altering numeric values, or modifying genes in other ways. Crossover, or recombination, combines the genetic information of two parent individuals to produce one or more offspring, promoting the mixing of genes and the discovery of new gene combinations.[8, 1].

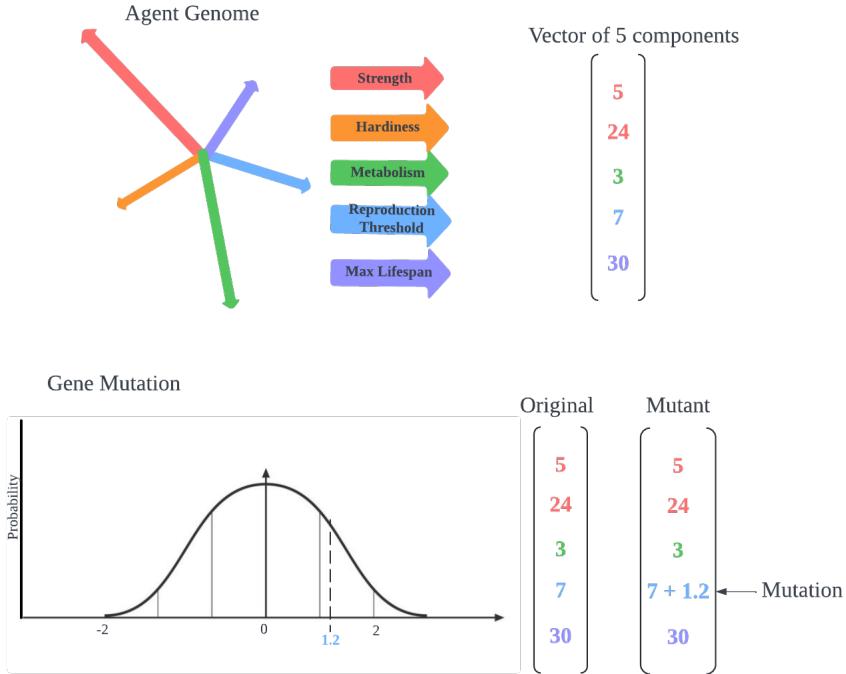


Figure 2.4: Vector Representation of Agent Genome and Mutation Method

In this simulation, a vector encoding of the genome was implemented. The individual traits of these organisms are encoded as a matrix, with each attribute represented as a vector component. This encoding allows for the representation of various traits in a straightforward manner that can be quickly manipulated by a CPU or GPU. Mutation, a critical mechanism in evolutionary algorithms, is applied by introducing random changes to these vector-encoded attributes by modifying the vector components of each matrix genome. This method enables the exploration of a wide range of potential adaptations and evolutionary paths, similar to the mutation processes used in traditional GAs.

Mutation in the simulation introduces variability by randomly altering the values of the vector components. For example, if an agent's genome is represented as a vector [20, 10, 5, 5, 3], a mutation might alter it to [20, 11, 5, 4, 3]. This randomness allows the simulation to explore a broad range of genetic variations, facilitating adaptation to changing environments.

Selection in the simulation mimics natural selection by favoring agents that exhibit traits conferring higher fitness. Fitness in this context could be based on the agent's ability to survive, reproduce, or acquire resources. Agents with higher fitness are more likely to pass their genes to the next generation, ensuring that advantageous traits become more common over time.

While the current implementation primarily relies on mutation to introduce genetic diversity, crossover mechanisms could also be integrated. Crossover would involve combining the genomes of two parent agents to produce offspring with mixed traits, promoting genetic diversity and potentially leading to the emergence of beneficial gene combinations.

## 2.4 Speciation in Asexually Reproducing Organisms and Alife Agents

Since a large part of the simulation involved measurement of how several factors affect the process of speciation, it became necessary to find a rigid definition of it that could be applied to the software. This was easier said than done. Speciation, the evolutionary process by which populations evolve to become distinct species, is well-studied in sexual organisms but less understood in asexual organisms as the Alife agents of this simulation would approximate. Asexual reproduction, where genetic material is not mixed as in sexual reproduction, poses unique challenges to emulate in simulation. Theoretical and empirical studies suggest that real life asexual organisms might experience speciation through mechanisms like geographic isolation, population clustering, and ecological specialization, albeit at different rates and under different conditions compared to sexual organisms [9]. However, implementing these concepts in simulation proved challenging due to computational complexity. So we opted instead to find a different quantitative measure to represent a speciation event.

## 2.5 Possible Solution to Simulated Asexual Speciation Inspired by NEAT Algorithm

The NEAT (NeuroEvolution of Augmenting Topologies) algorithm, introduced by Stanley and Miikkulainen in 2002, offered a solution. NEAT is an algorithm that evolves both neural network topologies and weights with evolutionary methods. NEAT starts with simple neural networks and incrementally adds complexity through mutations, which introduce new nodes and connections. To manage genetic diversity and speciation, NEAT uses a method that groups similar genomes into species based on a compatibility distance measure. This measure considers the number of excess and disjoint genes and the differences in the weights of matching genes. Speciation prevents premature convergence by protecting innovative structures within the population, fostering a variety of evolutionary paths [10].

Although not directly relevant as NEAT produces neural network instead of evolved Alife agents, they are both based on an artificial genome. Therefore, NEAT's speciation method could be utilized to model speciation in the current simulation. Agents are grouped into species using Euclidean distance between vector genomes as the measure of genetic distance. First, an initial species representative is found. Then, all agents that are within a certain threshold of genetic similarity are considered that species. When a new agent that is outside of this threshold appears, it is classified as a new species representative, and all further individuals are classified into those two representation, until the process repeats. The adaptation of NEAT's technique to classify species based on genetic distance allows the simulation to maintain a diverse population of agents and study speciation in a controlled discrete manner, while still maintaining high performance. This approach tracks the emergence of distinct species over time, even without needing sexual reproduction, and gives us our quantitative measure from which to measure speciation events in the Evo Island simulated organisms.

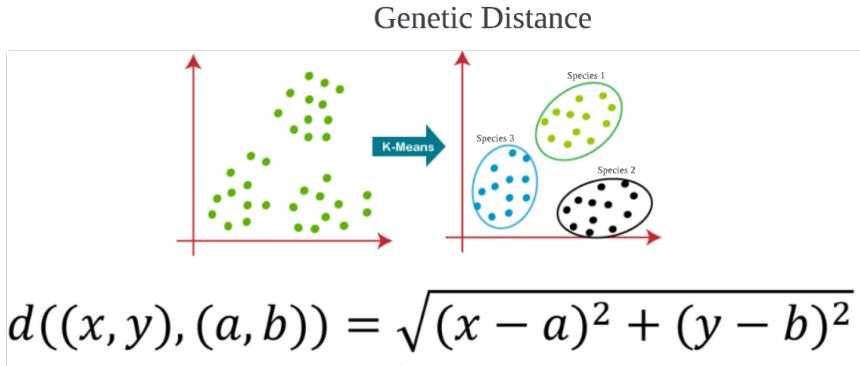


Figure 2.5: Speciation Method using the L2 Distance and clustering around species representatives

## 2.6 Procedural Environment Generation Using Perlin Noise

Finally, we searched for a method to produce a realistic environment for the agents to explore and inhabit. For this purpose we looked towards procedural generation techniques from video games and movies. Procedural generation of virtual environments, particularly using Perlin noise, is a method for creating complex, lifelike textures and terrains algorithmically. Perlin noise, a type of gradient noise developed by Ken Perlin, is widely used in graphics and game development to generate natural-looking textures and terrains without the need for manual design. It is particularly effective in simulating the randomness and diversity found in natural environments, making it suitable for studies involving ecological simulations [2].

Ken Perlin's paper "An Image Synthesizer" introduced the concept of Perlin noise, which has had a significant impact on various fields, including computer graphics and procedural generation. The noise function he developed creates continuous, smooth variations in textures, which can mimic the complexity of natural landscapes. This technique involves generating noise values that are interpolated to create coherent structures, avoiding the repetitive patterns common in simple random noise [2].

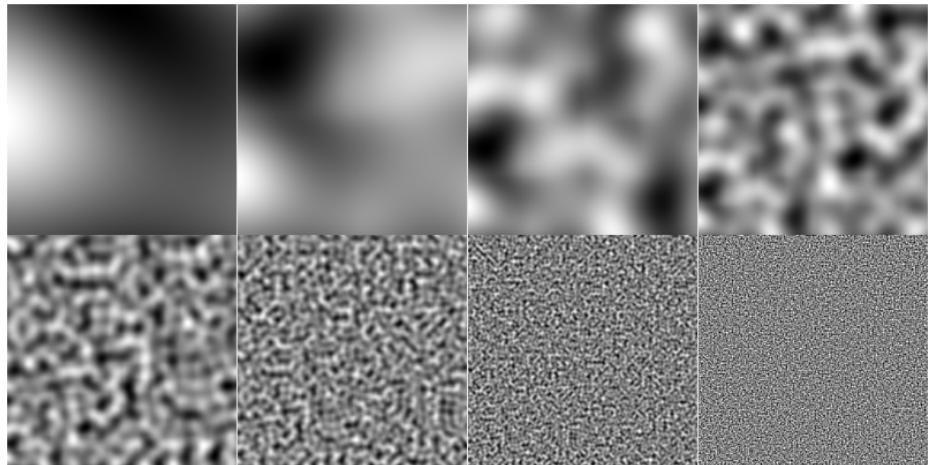


Figure 2.6: Examples of Patterns created by Perlin Noise[2]

In this simulation, Perlin noise is used to generate the environment in which the agents evolve. This procedural generation technique creates diverse and realistic terrains, providing a dynamic and challenging environment for the agents to live and explore. By adjusting input parameters of the perlin noise generator such as frequency, amplitude, and persistence, researchers can control the characteristics of the generated environments. [2, 11].

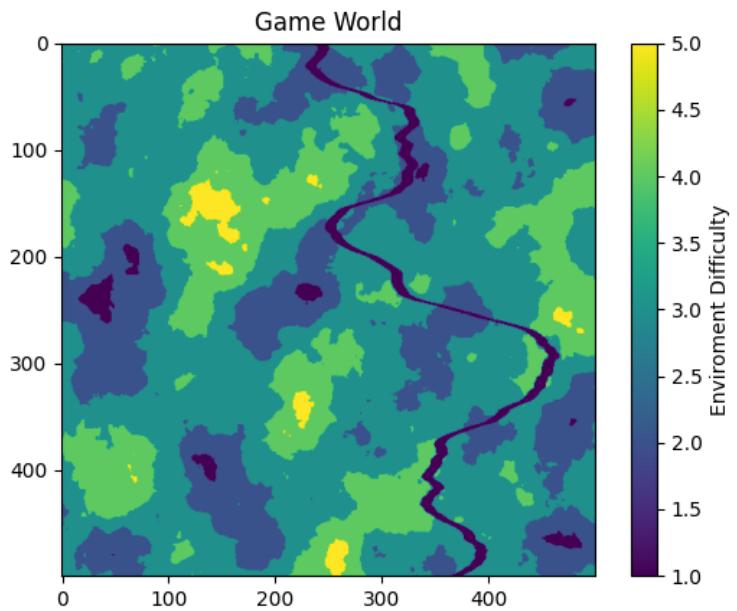


Figure 2.7: Example Simulation Map Created by Perlin Noise

This application of Perlin noise allow us to change a parameter that is wholly infeasible for traditional biologists: the replacement and modification of entire environments instantaneously.

These concepts came together to form our simulation.

# Chapter 3

## Method, Algorithms, and Technical Manual

This chapter details the methods used to implement the Evo Island evolutionary simulation, providing an overview of the codebase structure, pseudocode for the main algorithms, and a technical manual for setting up and running the simulation. The goal of this chapter is to offer a comprehensive understanding of how the simulation is constructed and operated.

### 3.0.1 Git Repository

The project source code to produce the experiments and data in this thesis is available for download at: <https://github.com/ract93/Morphanimals>

### 3.1 Main Algorithms

Evo Island simulates the evolution of digital organisms in a dynamically changing environment. The primary goal is to study how different genetic attributes and environmental pressures influence the evolutionary process. The system achieves this by allowing agents to mutate, reproduce, and compete for resources, mimicking natural selection. The core functionality of Evo Island is built around the interaction between agents and their environment. Below is an explanation of the main algorithms used in the simulation, detailing their roles and justifying their inclusion.

### Agent Module

The Agent Python module represents individual organisms within the simulation. Each agent has genetic attributes that determine its behavior, survival capabilities, and reproductive success. The primary operations agents can perform include mutation, reproduction, aging, consuming food, fighting with other agents, and interacting with their environment.

#### Algorithm: Agent Vector Genome Initialization

The initialization of an agent's genome is needed for setting the starting conditions of the simulation. Each agent begins with a set of default traits, which can evolve over time.

---

**Algorithm 1** Agent Vector Genome Initialization

---

```
1: Initialize common ancestor genome: [20, 10, 5, 5, 3]    ▷ Lifespan, Hardiness, Strength,  
   Metabolism, Reproduction Threshold  
2: function DECODEGENOME(genome)  
3:   if genome is not None then  
4:     lifespan ← genome[0]  
5:     hardiness ← genome[1]  
6:     strength ← genome[2]  
7:     metabolism ← genome[3]  
8:     reproduction_threshold ← genome[4]
```

---

Mutation introduces genetic variation, which is a fundamental mechanism for evolution. This algorithm simulates the process by randomly altering genes within an agent's genome.

---

**Algorithm 2** Agent Mutation

---

```
1: function MUTATEGENOME(genome, mutation_rate)  
2:   new_genome ← genome.copy()  
3:   for i = 0 to length(new_genome) - 1 do  
4:     if random_chance()  $\geq$  mutation_rate then  
5:       new_genome[i] += random_normal_variation()  
6:       new_genome[i] ← clamp(new_genome[i], 0, 100)  
7:   return new_genome
```

---

Reproduction is vital for the propagation of genetic material. This algorithm handles the creation of offspring through asexual reproduction, incorporating mutations to introduce genetic diversity.

---

**Algorithm 3** Agent Reproduction

---

```
1: function REPRODUCEASEXUALLY(parent)  
2:   child_genome ← MutateGenome(parent.genome, mutation_rate)  
3:   child ← CreateAgent()  
4:   child.genome ← child_genome  
5:   child.decode_genome()  
6:   return child
```

---

Hardiness determines an agent's ability to survive in different environmental conditions. This algorithm assesses whether an agent can thrive in its current location.

---

**Algorithm 4** Agent Hardiness

---

```
1: function CANSURVIVE(agent, environment)  
2:   cell_difficulty ← environment.get_difficulty(agent.position)  
3:   return agent.hardiness  $\geq$  cell_difficulty
```

---

Strength determines an agent's ability to compete for resources. This algorithm simulates con-

flicts between agents over territory.

---

**Algorithm 5** Agent Strength

---

```

1: function COMPETEFORCELL(new_agent, existing_agent)
2:   if new_agent.strength < existing_agent.strength then
3:     existing_agent.kill()
4:     return True
5:   else
6:     new_agent.kill()
7:   return False

```

---

Aging introduces a lifecycle to agents, affecting their longevity and mortality. This algorithm models the aging process and its impact on survival.

---

**Algorithm 6** Agent Aging

---

```

1: function AGEAGENT(agent)
2:   agent.age += 1
3:   if ShouldDieFromAging(agent.age, agent.lifespan) then
4:     agent.kill()
5: function SHOULDDEFROMAGING(age, lifespan)
6:   midpoint ← lifespan / 2
7:   steepness ← 10   death_probability ←  $\frac{1}{1 + \exp(-steepness \times \frac{(age - midpoint)}{midpoint})}$ 
8:   return random_chance() | death_probability

```

---

Metabolism affects an agent's energy balance, needed for survival. This algorithm manages the energy intake and expenditure of agents.

---

**Algorithm 7** Agent Metabolism

---

```

1: function METABOLIZE(agent, food_available)
2:   agent.energy_reserves += food_available
3:   agent.energy_reserves -= agent.metabolism
4:   if agent.energy_reserves ≤ 0 then
5:     agent.kill()
6: function CONSUMEFOOD(agent, environment, current_step)
7:   food_available ← environment.calculate_food_available(agent.position, current_step)
8:   Metabolize(agent, food_available)

```

---

## Environment Module

The `Environment` class simulates the world in which agents operate, providing the terrain and resources they interact with. It is responsible for generating different types of maps and managing food distribution.

### Map Generation Methods

#### Petri Dish Map

The Petri dish map provides a simplified environment where the difficulty increases linearly across the grid. This map type is useful for testing basic evolutionary principles. Its function is to reproduce the experimental conditions of the LTEE Experiment.

---

#### Algorithm 8 Petri Dish Map Generation

---

```
1: function GENERATE_PETRI_DISH(n)
2:   if n ≤ 0 then
3:     return empty array
4:   Initialize array of size  $n \times n$  with zeros
5:   Set increment  $\leftarrow 4 / (n - 1)$ 
6:   for each row  $i$  in array do
7:     for each column  $j$  in row do
8:       array[i][j]  $\leftarrow$  round(( $j * \text{increment}$ ) + 1)
9:   return array
```

---

#### Random Map

The random map generates a completely unpredictable environment with difficulty levels ranging from 1 to 3. This map type is useful for testing how agents adapt to unpredictable environments.

---

#### Algorithm 9 Random Map Generation

---

```
1: function GENERATE_RANDOM_MAP(n)
2:   Initialize array of size  $n \times n$ 
3:   for each row  $i$  in array do
4:     for each column  $j$  in row do
5:       array[i][j]  $\leftarrow$  random integer between 1 and 3
6:   return array
```

---

#### Perlin Noise Terrain

Perlin noise is a gradient noise function used to generate natural-looking textures. In the `Environment` class, Perlin noise is used to create varied and realistic terrain.

---

**Algorithm 10** Perlin Noise Terrain Generation

---

```
1: function GENERATE_PERLIN_NOISE_TERRAIN(n)
2:   if use_random_perlin_params is True then
3:     Set scale, octaves, persistence, lacunarity to random values
4:   else
5:     Set scale ← 100.0
6:     Set octaves ← 6
7:     Set persistence ← 0.5
8:     Set lacunarity ← 2.0
9:   Initialize world array of size  $n \times n$  with zeros
10:  for each row  $i$  in world do
11:    for each column  $j$  in row do
12:      world[ $i$ ][ $j$ ] ← Perlin noise value at  $(i/scale, j/scale)$ 
13:  Normalize world values to range from 1 to 5
14:  return rounded world array
```

---

## River Generation

The river generation method modifies the Perlin noise terrain to add a meandering river path. Agents can use the river as a migrational path to reach regions of the map that would otherwise be unaccesible to them given their current attributes.

---

**Algorithm 11** River Generation

---

```
1: function GENERATE_RIVER(world_matrix)
2:   Set  $n \leftarrow$  map size
3:   Set scale ← 200.0, octaves ← 6, persistence ← 0.4, lacunarity ← 2.0
4:   Initialize grid array of size  $n \times n$  with zeros
5:   for each row  $i$  in grid do
6:     for each column  $j$  in row do
7:       grid[ $i$ ][ $j$ ] ← Perlin noise value at  $(i/scale, j/scale)$ 
8:   Normalize grid values
9:   Set river_width ←  $n // 50$ 
10:  Initialize river_path as list of coordinates based on normalized grid values
11:  for each row  $i$  in world_matrix do
12:    for each column  $j$  in row do
13:      if distance from river path  $\leq$  river_width // 2 then
14:        world_matrix[ $i$ ][ $j$ ] ← 1
15:  return modified world_matrix
```

---

## 3.2 Setting Up the Simulation

### 3.2.1 Config File

The simulation is configured using a JSON file (`config.json`). This file contains parameters that influence various aspects of the simulation, such as the number of trials, map size, simulation steps, and toggles for different features.

#### Example Config File

```
{  
    "experimental_trials": 30,  
    "map_size": 300,  
    "simulation_steps": 1000,  
    "map_type": "petri_dish",  
    "use_random_perlin_params": false,  
    "use_rivers": false,  
    "enable_violence": false,  
    "enable_food": false,  
    "enable_aging": false,  
    "enable_reproduction_threshold": false,  
    "food_generation_rate": 4,  
    "initial_food": 50,  
    "max_food_capacity": 50,  
    "mutation_rate": 0.034,  
    "frame_save_interval": 3,  
    "frame_rate": 30  
}
```

#### Configuration Parameter Descriptions

- `experimental_trials`: (Integer) The number of times the experiment is repeated. Default is 30 trials.
- `map_size`: (Integer) The size of the 2D grid representing the environment. A value of 300 means the grid is 300x300 cells.
- `simulation_steps`: (Integer) The total number of steps the simulation will run. For example, 1000 steps per trial.
- `map_type`: (String) Defines the type of map used in the simulation. Options include "petri\_dish", "perlin", or "randomized". The default is "petri\_dish".
- `use_random_perlin_params`: (Boolean) If set to true, the Perlin noise parameters will be randomized to create different terrain each time. Default is false. Is only activated if used with the perlin map
- `use_rivers`: (Boolean) Enables or disables the presence of rivers in the environment. Rivers can influence agent movement and resource distribution. Default is true. Is only active with the perlin map.
- `enable_violence`: (Boolean) If true, agents can interact violently, by using strength attribute to compete for a cell, potentially impacting survival and reproduction. Default is true.

- `enable_aging`: (Boolean) If true, agents will age and eventually die over time, which can affect their survival and reproduction. Default is false.
- `enable_food`: (Boolean) If true, food resources are included in the simulation, and agents must find and consume food to survive. Default is false.
- `enable_reproduction_threshold`: (Boolean) If true, agents must meet certain threshold of food resource in order to reproduce. Default is false.
- `food_generation_rate`: (Integer) The rate at which new food is generated in the environment per step. A value of 4 means 4 food is generated in each cell every step. Only applicable if `enable_food` is true.
- `initial_food`: (Integer) The initial amount of food available in each cell at the start of the simulation. Default is 50 units.
- `max_food_capacity`: (Integer) The maximum amount of food a cell can hold. Default is 50 units.
- `speciation_threshold`: (Integer) The genetic distance threshold at which agents are considered different species. A lower value means more speciation events, but a slower simulation.
- `mutation_rate`: (Float) The probability of a mutation occurring in an agent's genome during reproduction. Default is 0.034.
- `frame_save_interval`: (Integer) The interval at which frames (snapshots of the simulation state) are saved to create the visualization gifs. A value of 3 means a frame is saved every 3 steps. Greatly impacts simulation performance.
- `frame_rate`: (Integer) The rate at which the simulation updates are visualized in the gifs. Default is 30 frames per second.

### 3.3 Running the Simulation

1. To execute the simulation, check that the configuration file (`config.json`) is set up correctly with the desired parameters.
2. Run the `evo_island.py` script.
3. Enter a name for the experiment.
4. The simulation results, including logs, images, GIFs, and analysis notebooks will be created and saved in the `Experimental-Results/` directory.

#### 3.3.1 Simulation Result Artifacts

The simulation generates various metrics and visualizations to analyze evolutionary dynamics, including population counts, death causes, and genetic diversity. Visualizations encompass time-series plots of genetic traits and GIFs showing agent attributes over time.

#### 3.3.2 CSV Metrics Collection

Metrics are collected and logged at each simulation time step, saved in `simulation-metrics.csv` per trial. These files are aggregated into `aggregated-metrics.csv` for simulations with multiple trials.

### **3.3.3 Analysis Notebooks**

The Jupyter analysis notebook records and preliminary analyzes simulation data, including plots and statistical summaries. Notebooks are produced for each trial and aggregated at the conclusion of all trials.

### **3.3.4 Visualizations**

Visualization techniques, including GIFs and timelapse images, represent time series data of agents' evolving attributes and their interactions with the environment. These visualizations, saved in the Gifs and Images folders after each simulation run, help analyze the simulation's outcomes.

# Chapter 4

# Experimental Results

This chapter presents the results of experiments designed to test our evolutionary model. Each experiment explores different aspects of the model by varying specific parameters or conditions. The findings from these experiments will help validate the model's ability to simulate complex evolutionary processes and provide insights into the mechanisms driving these processes.

## 4.1 Common Experimental Setup

All experiments were conducted within a consistent simulation environment, as described in Chapter 3. To maintain uniformity and ensure comparability across experiments, certain parameters were kept constant unless otherwise specified:

- Simulation Steps: Each experiment runs for 1000 simulation steps.
- Experimental Trials: Each experiment is repeated 30 times for statistical significance.
- Attributes: All simulations will activate hardness and strength attributes in agents.
- Speciation Threshold: A genetic distance threshold of 10 is used to determine speciation.
- Frame Save Interval: Snapshots of the simulation state are saved every 3 steps.
- Frame Rate: The simulation GIFs are visualized at a rate of 30 frames per second.
- Mutation Rate: All experiments will use a mutation rate of 0.034 per agent replication.

## 4.2 Experiment 1: Replicating the LTEE Petri Dish with simulation of Hardiness and Strength

These consistent settings allowed us to isolate the effects of the specific variables being tested in each experiment.

### 4.2.1 Goal:

The aim of this experiment was to replicate the environment of a petri dish with abundant nutritional substrate and areas of gradually increasing antibiotic concentration, similar to the original LTEE setup. In the LTEE, regularly spaced bands of increasing antibiotic concentration were placed in a

Petri Dish to study the emergence of antibiotic resistance in E. Coli. By simulating this setup, we sought to determine if our model could capture the fan-shaped growth patterns observed in LTEE, thereby validating its ability to simulate adaptive changes such as antibiotic resistance.

For the initial configuration, we focused on two primary genes: "hardiness," which controlled the difficulty of the cells agents could occupy, and "strength," which mediated inter-agent conflict and competition over cells. Other attributes were deactivated to simplify interactions and mirror real microbial growth.

A single agent was placed in the top left of the Petri Dish map, and its growth patterns were observed over 1000 simulation steps. We collected heat map visualizations of how the attributes of the agents changed over time, the causes of death for the agents, and the number of speciation events.

#### 4.2.2 Hypothesis:

**Null Hypothesis (H0):** The agents will not exhibit an increase in 'hardiness' over time, and no distinct fan-shaped growth patterns will emerge, indicating that the evolutionary model does not accurately replicate the adaptive changes observed in the LTEE.

**Alternative Hypothesis (H1):** The agents will exhibit an increase in 'hardiness' over time and form distinct fan-shaped growth patterns at the intersection of difficulty levels, indicating that the evolutionary model successfully replicates the adaptive changes and the spread of antibiotic resistance via the genome, as observed in the LTEE.

#### 4.2.3 Experiment Configuration

: Experiment 1 used the following configuration:

```
{
    "map_type": "petri_dish",
    "use_random_perlin_params": false,
    "use_rivers": false,
    "enable_violence": true,
    "enable_food": false,
    "enable_aging": false,
    "enable_reproduction_threshold": false,
    "mutation_rate": 0.034,
    "frame_save_interval": 3,
    "frame_rate": 30
}
```

Experiment 1 will be conducted in the "Petri Dish" map (Figure 4.1) which simulates the experimental setup of the LTEE.

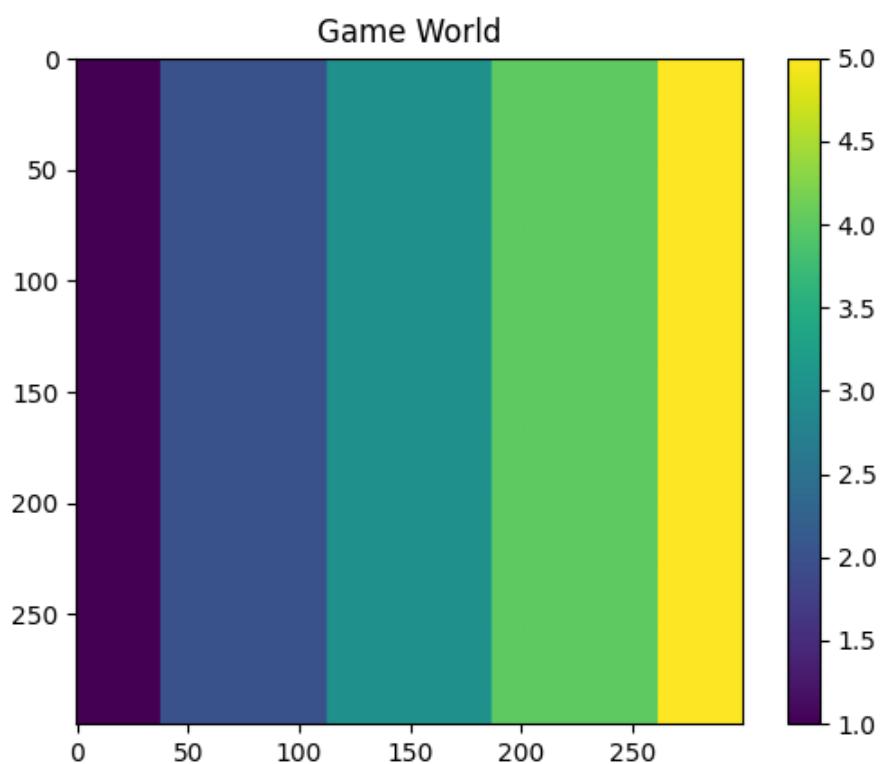


Figure 4.1: This is the map used for experiment 1. It is designed to simulate the experimental setup of the LTEE and has graduated bands of increased survival difficulty, analogous to the antibiotic concentration.

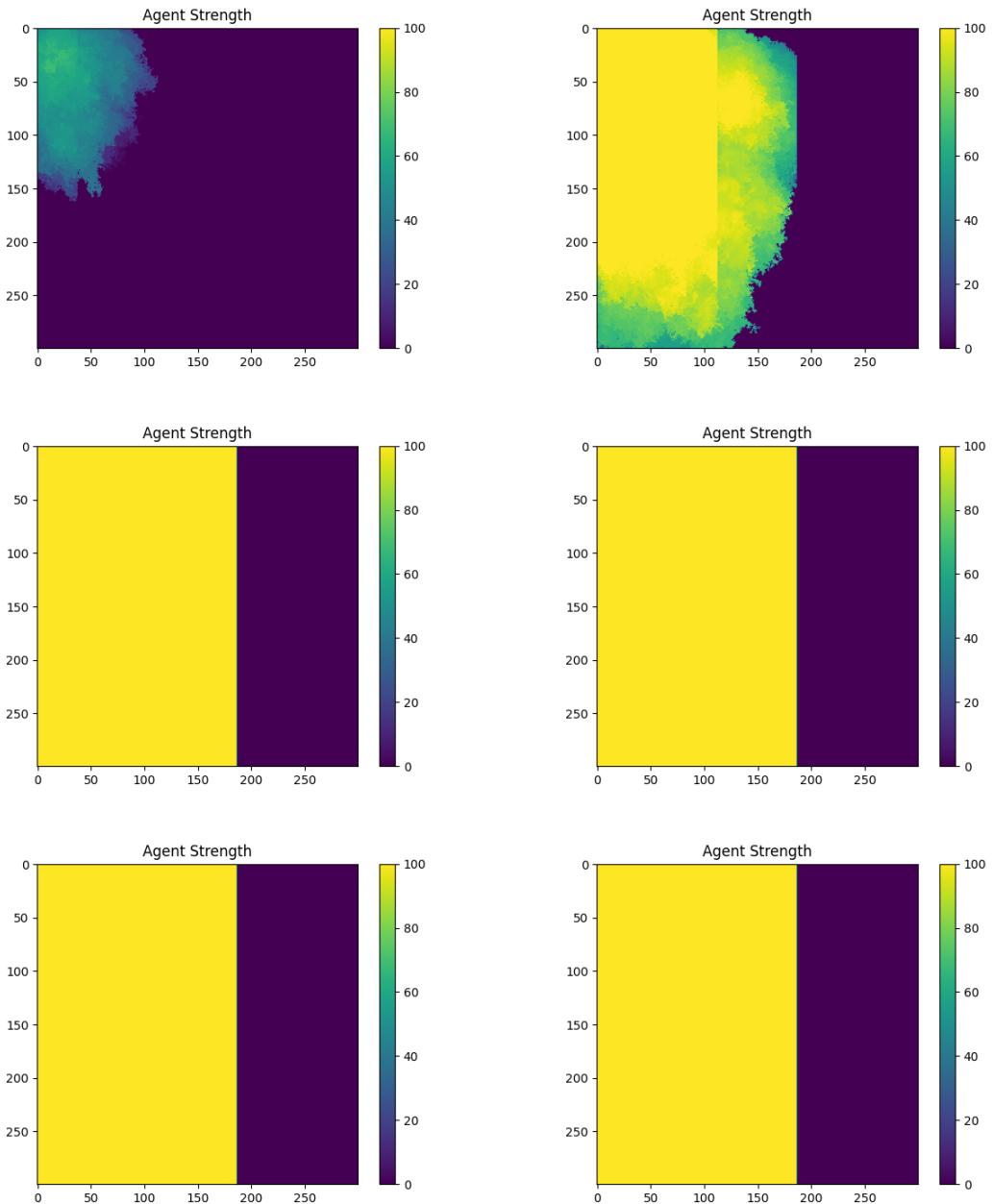


Figure 4.2: Experiment 1 - Agent Strength over 1000 Time Steps. Note that at about 300/1000 mark of the simulation, the strength attribute was maximized for all agents. This capping of the strength value greatly reduced diversity, and eventually gridlocked the simulation when no new agents could occupy new cells from existing agents.

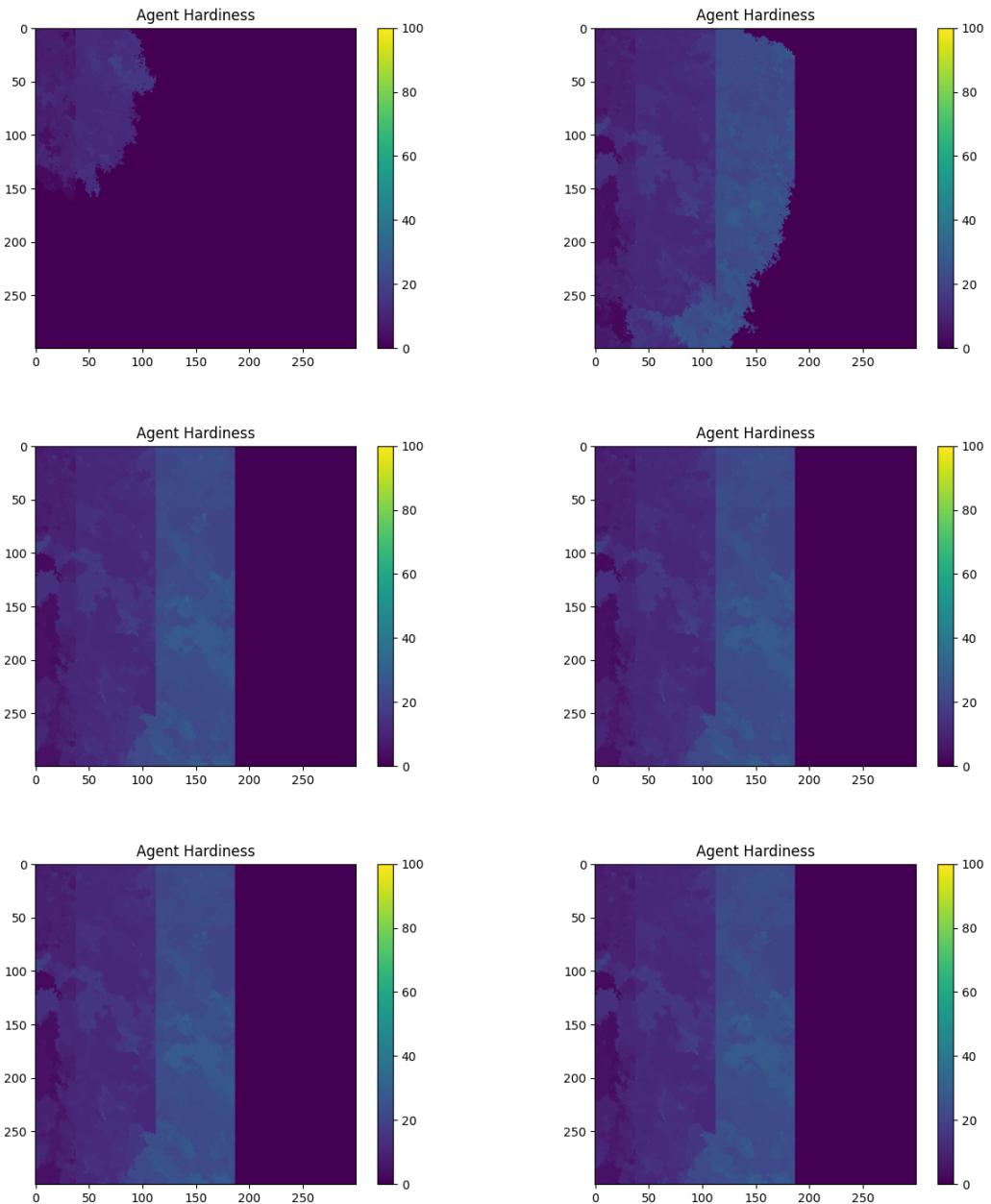


Figure 4.3: Experiment 1 - Agent hardness over 1000 time steps. Notice that the agents stratify themselves by the required hardness needed to occupy areas of higher difficulty. Agent hardness steadily increased from left to right. This suggests adaptation, and geographic isolation occurred due to geographic barriers.

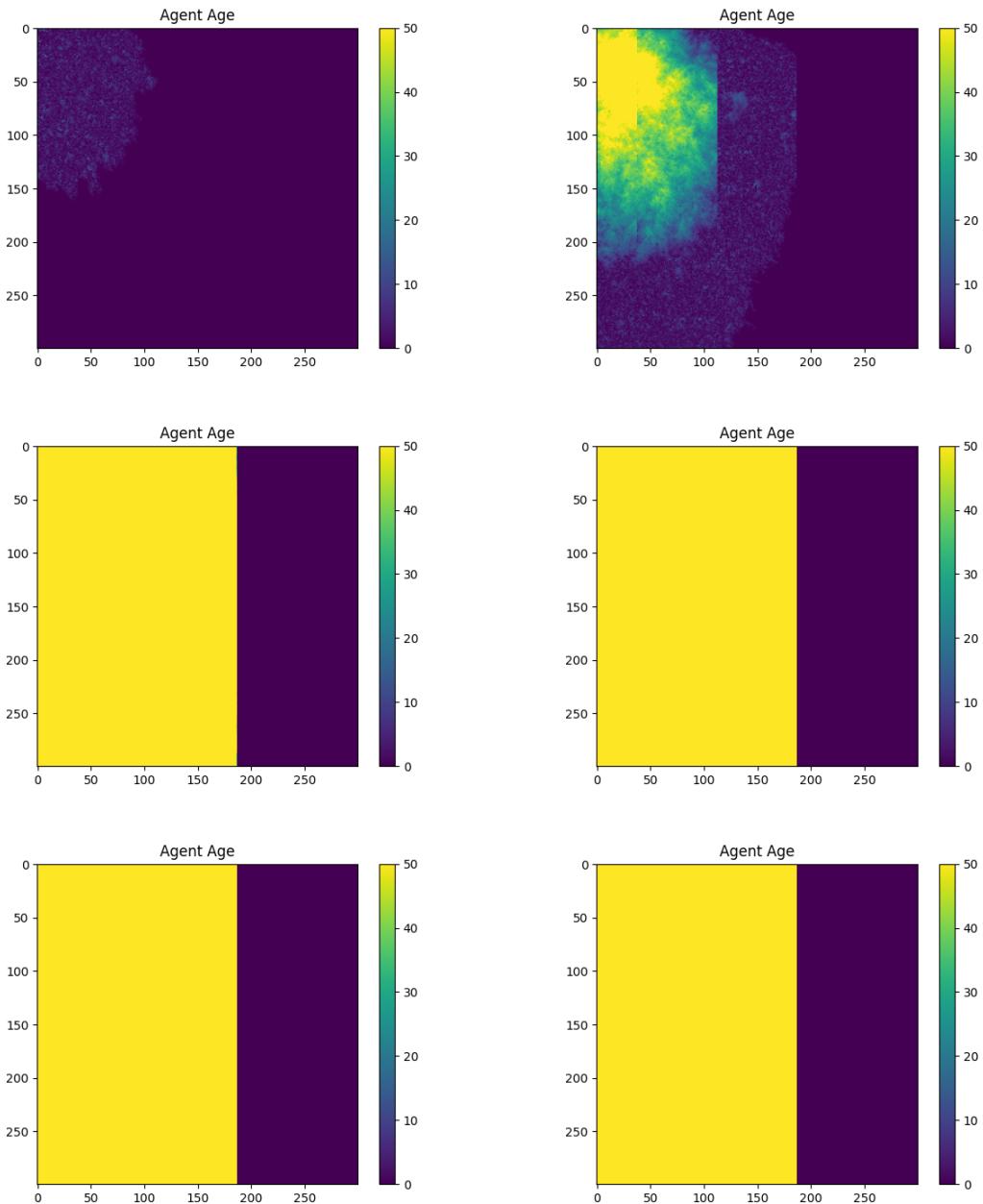


Figure 4.4: Experiment 1 - Agent Age over 1000 Time Steps. This result heavily suggests that the grid locking of the simulation was caused by elitism. Once agents maximized their strength attribute, they could not be displaced by any newer agent. With nothing to check them, the same agents continued living until the end of the simulation.

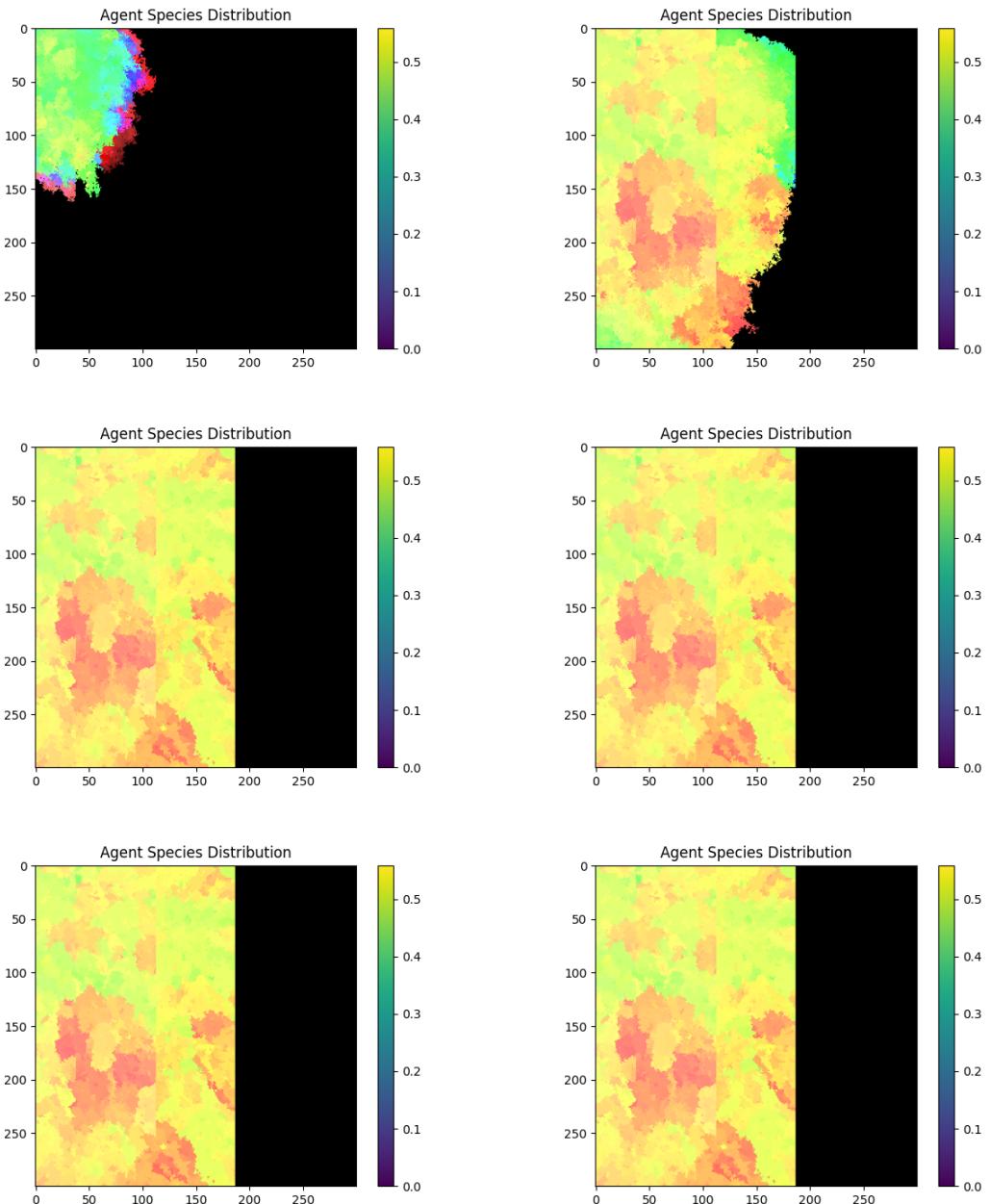


Figure 4.5: Experiment 1 - Agent Species distribution over 1000 Time Steps. The distribution of species over time showed how after the strength gene was maximized, diversity fell dramatically.

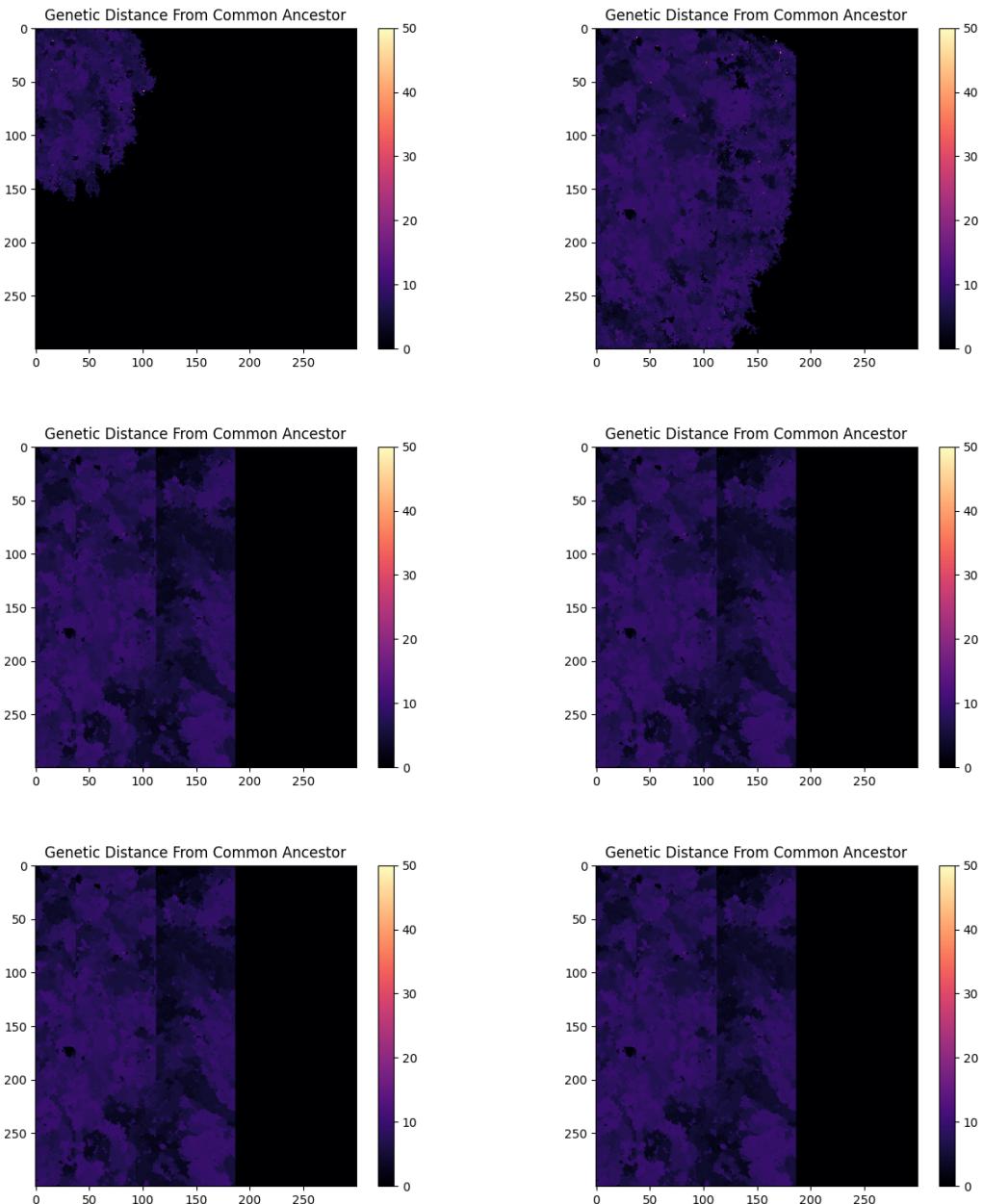


Figure 4.6: Experiment 1 - Genetic Distance from common ancestor over 1000 Time Steps. Note how after the 3rd image, the genetic distance from the common ancestor for all agents stopped changing. This implies that the grid locking of the simulation due to elitism, hampered the evolutionary process.

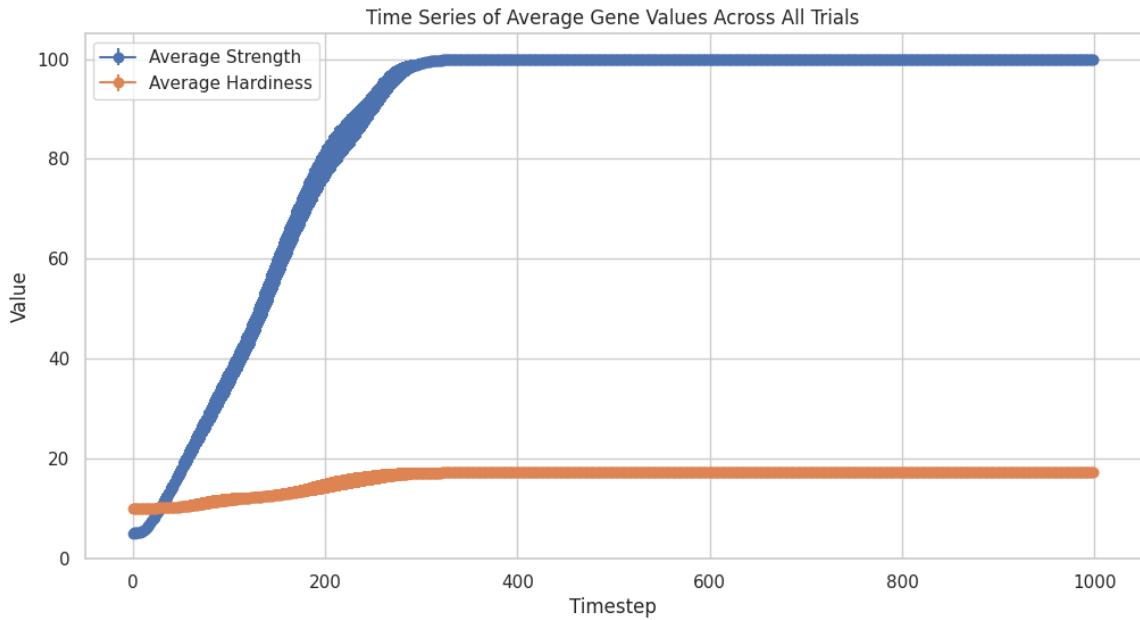


Figure 4.7: Experiment 1 - Average gene values over 30 trials. While both attributes increased over time, this shows that the strength attribute was under the most significant evolutionary pressure. Once it was maximized, and due to the elitism problem, the hardiness gene stopped changing as well.

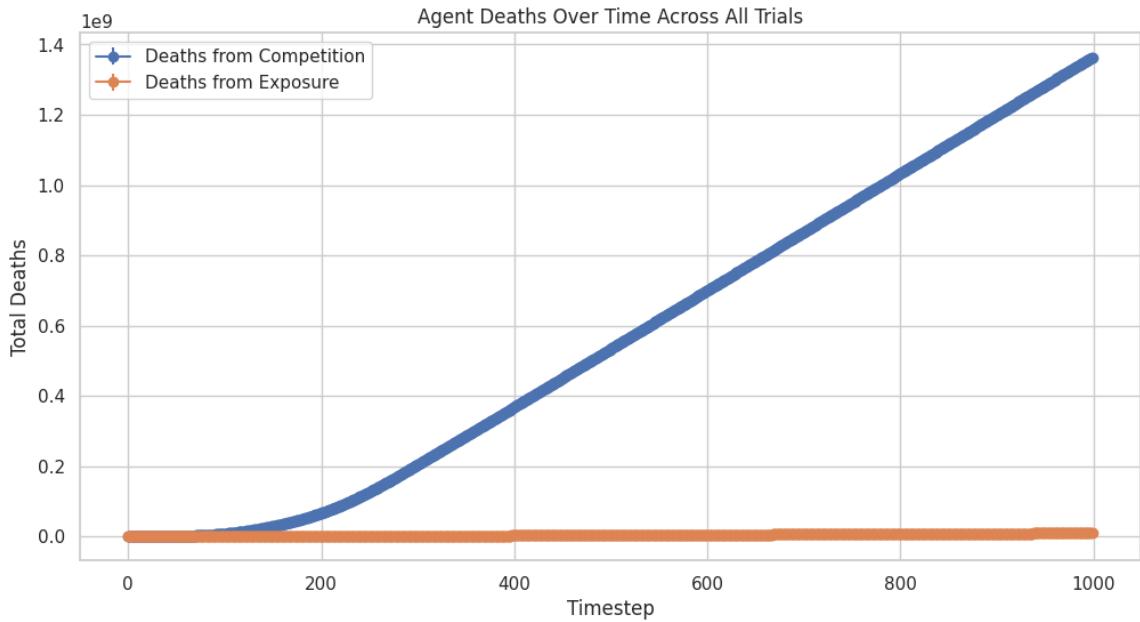


Figure 4.8: Experiment 1 - Average cause of Agent Death over 30 Trials. This again shows that the strength attribute was under the most significant evolutionary pressure.

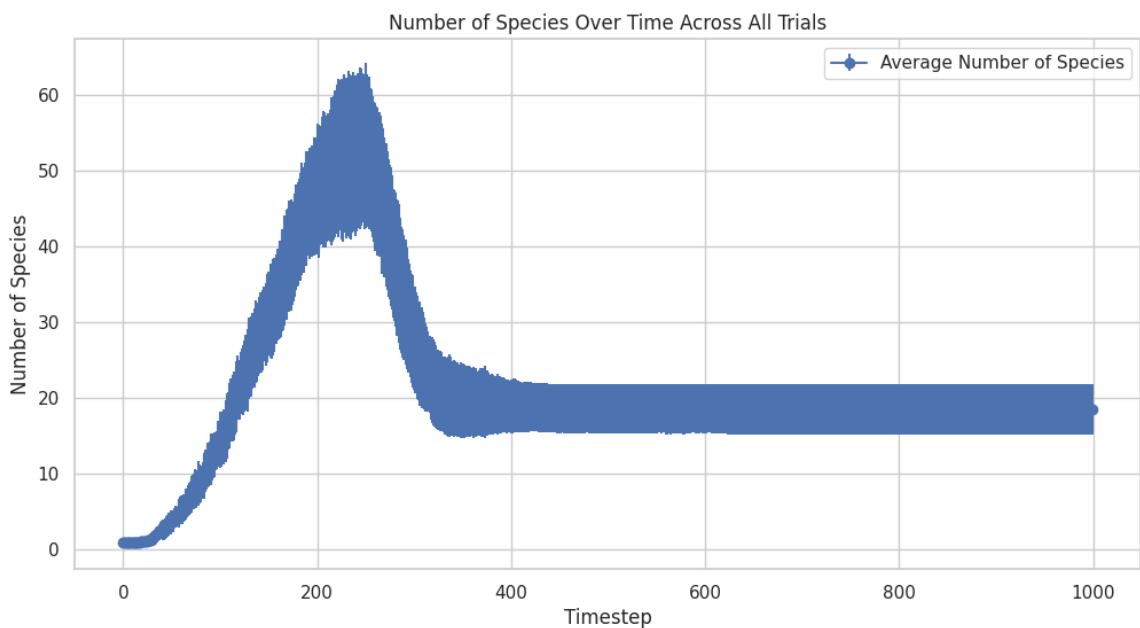
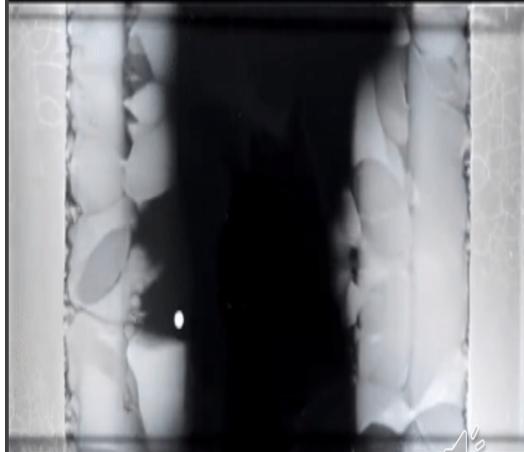
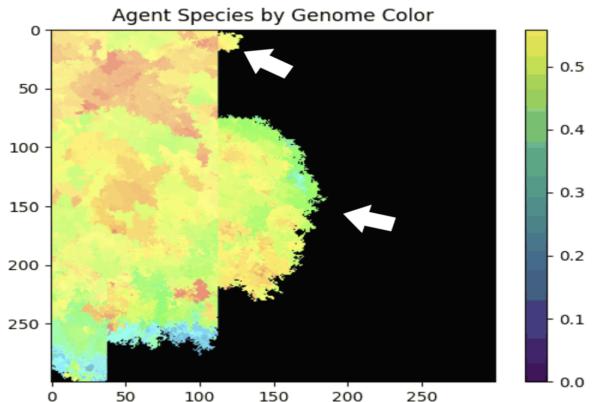


Figure 4.9: Experiment 1 - Average number of species per time step over all trials. This graph illustrates how the saturation of the strength gene reduced diversity until it eventually grid locked the simulation. Across all experimental trials, this saturation occurred roughly at time step 300. At that point, the number of species decreased until it stabilized at an equilibrium. This suggests that elitism is capping the diversity of the simulation.

#### 4.2.4 Observation of Fan Shaped Pattern



(a) LTEE Experiment Growth Pattern. [4]



(b) Growth Pattern of Evo Island Agents. Note the similar fan shape.

#### 4.2.5 Summary of Findings:

Figures 4.1 - 4.9 contain the results of Experiment 1. Results showed a clear increase in hardness over multiple generations, forming fan-shaped descent patterns similar to those observed in the LTEE. Agents appearing later in the simulation exhibited genomes increasingly distant from the common ancestor, with the greatest concentrations of these genomes located in higher difficulty biomes. This suggests that allopatric speciation may have occurred due to geographical boundaries created by increasing difficulty levels. However, evolutionary dynamics slowed significantly around step 375, as most agents achieved a maximum strength value of 100, causing the number of species to stabilize.

#### 4.2.6 Interpretation:

The evolution of hardness within the digital Petri dish and the fan-shaped patterns of descent observed in all simulation runs mirror microbial adaptation to varying antibiotic levels seen in the LTEE. The visual similarity between the LTEE and the simulation indicates that the digital model can effectively capture key evolutionary processes such as adaptation, migration, and genetic drift.

However, the cessation of evolutionary dynamics appears to be due to a "gridlock" caused by elitism. Individuals with a maximized strength attribute faced no mechanisms forcing them to relinquish their cells to younger generations. This gridlock occurred because the only death mechanisms implemented were competition from other agents and exposure to the environment, neither of which affected agents with maximized strength and sufficient hardness to occupy any locally available cell. Consequently, these elites prevented further evolutionary progress and diversification, indicating the need to introduce additional mechanisms to prevent such gridlock and facilitate continuous evolution.

#### **4.2.7 Conclusion:**

We accept the Alternative Hypothesis (H1). The experiment supports that the agents exhibited an increase in 'hardiness' over time and form distinct fan-shaped growth patterns at the intersection of difficulty levels, indicating that the evolutionary model successfully replicates the adaptive changes and the spread of antibiotic resistance via the genome, as observed in the LTEE.

## 4.3 Experiment 2: Effect of Agent Aging on Speciation and Elitism

### 4.3.1 Goal:

In Experiment 1, the evolutionary and speciation processes were greatly limited by elitism, particularly regarding the maximization of the strength gene. Experiment 2 investigates the effect of introducing cellular aging to combat elitism. Agents are now able to age, with a probabilistic death chance each turn, based on the agent's inherited maximum lifespan gene. This experiment aims to determine if aging can prevent the dominance of a few strong agents and promote greater diversity and speciation.

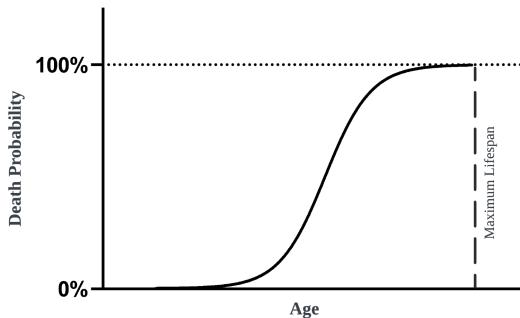


Figure 4.11: Experiment 2 - Agent death probability is calculated using a sigmoid function. The chance of an agent dying per turn is a function of its current age with respect to its maximum genetic lifespan.

### 4.3.2 Hypothesis

**Null Hypothesis (H0):** Introducing aging will not significantly impact speciation patterns or reduce elitism. The evolutionary dynamics and diversity of agents will remain similar to those observed in Experiment 1.

**Alternative Hypothesis (H1):** Introducing aging will result in more distinct speciation patterns and a reduction in elitism. Older agents will provide opportunities for new variants to emerge, leading to increased diversity and more dynamic evolutionary processes.

### 4.3.3 Configuration

: Experiment 2 used the same configuration and Petri Dish map as Experiment 1, except for the following modification:

```
{  
    "enable_aging": true,  
}
```

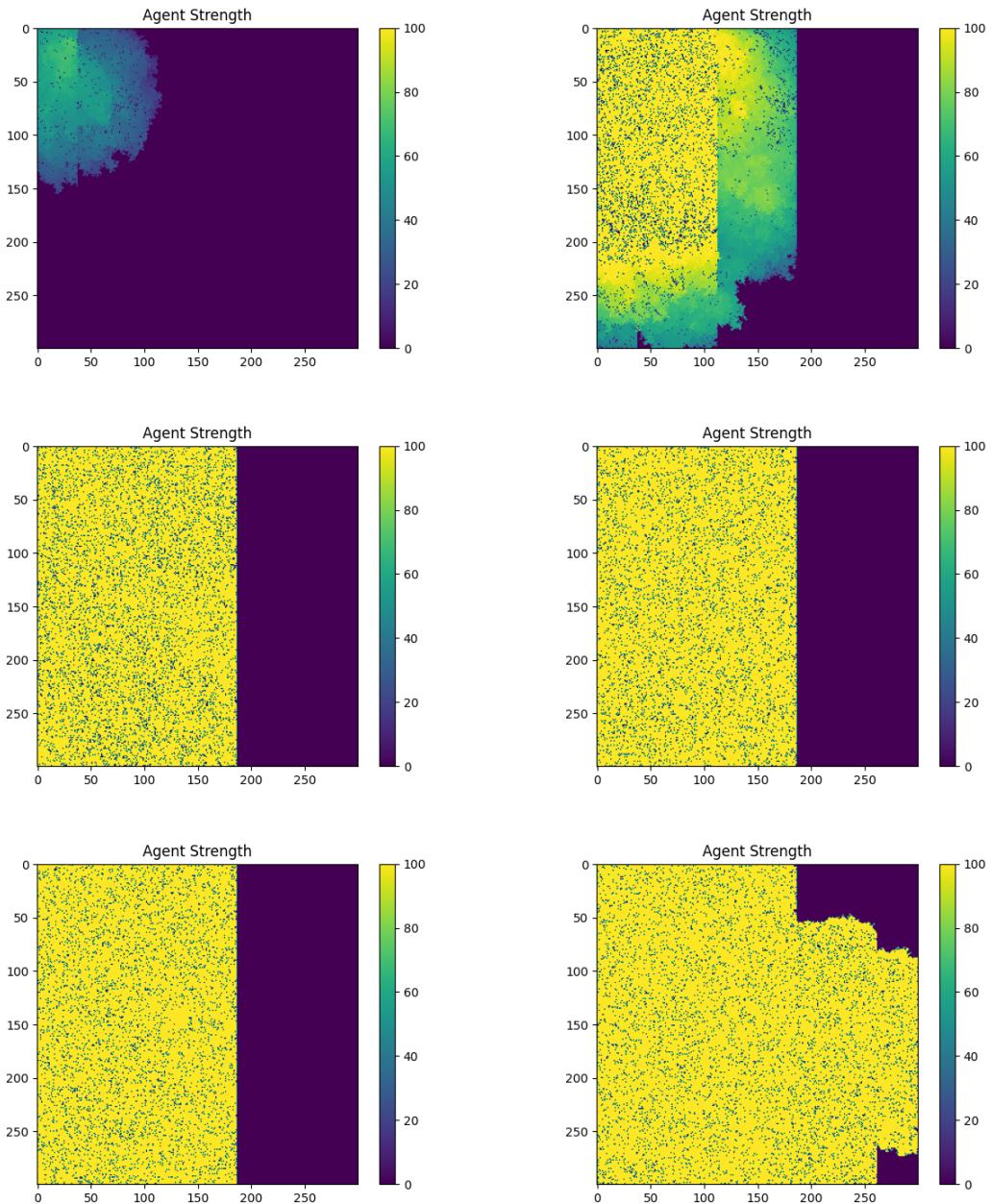


Figure 4.12: Experiment 2 - Agent Strength over 1000 Time Steps. Agent Strength still reached a cap, but we notice a noisy static effect on the images, these are dying agents, dying due to the probabilistic aging mechanism.

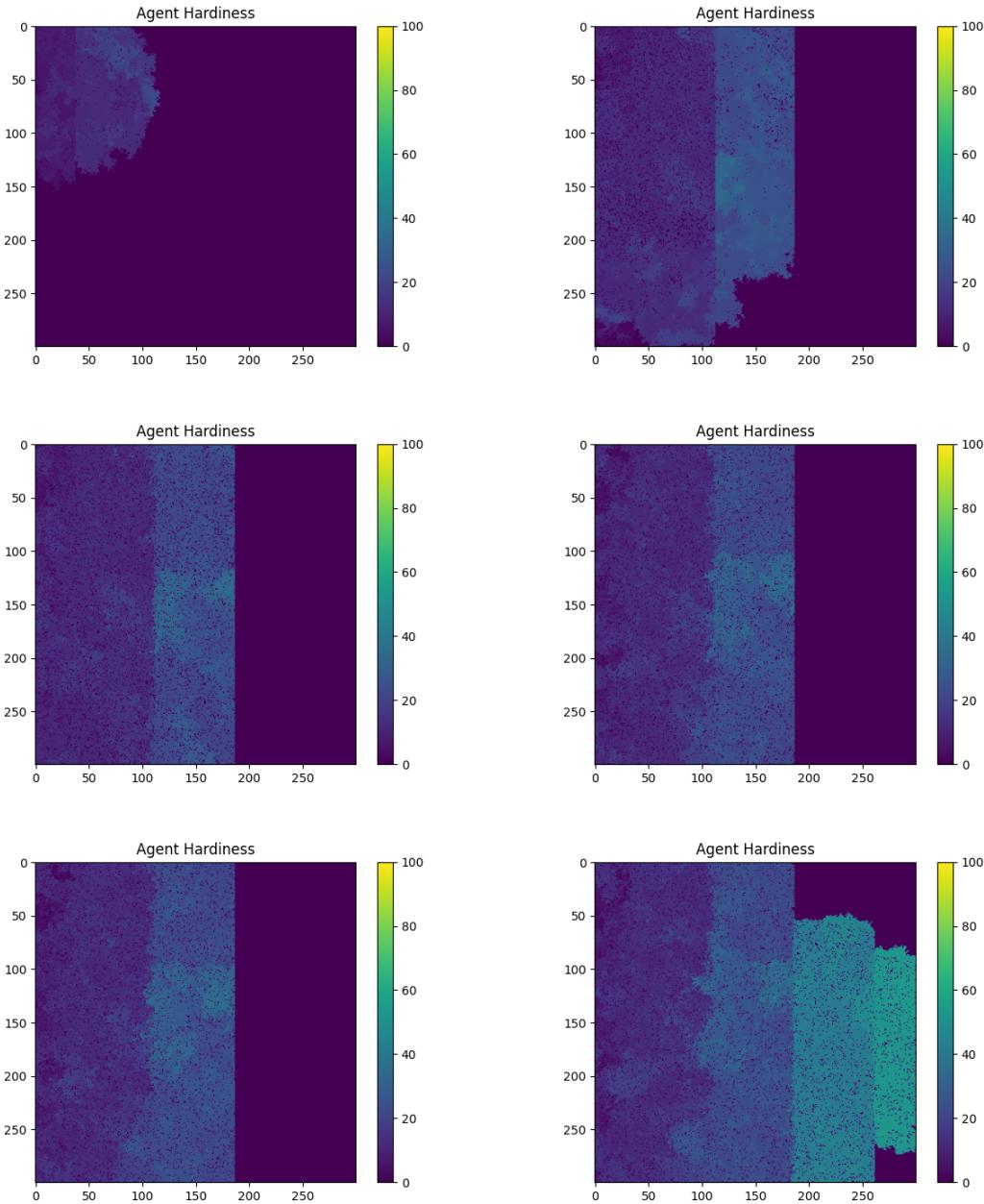


Figure 4.13: Experiment 2 - Agent hardness over 1000 time steps. As before we note that agents stratify themselves according to the required hardness of each difficulty region. Additionally we note that in several simulations, the agents were able to reach even the most difficult regions of the map when they could not before. This would suggest that the elimination of elitism allowed the hardness gene to continue developing, allowing the agents to continue the adaptation process and explore farther.

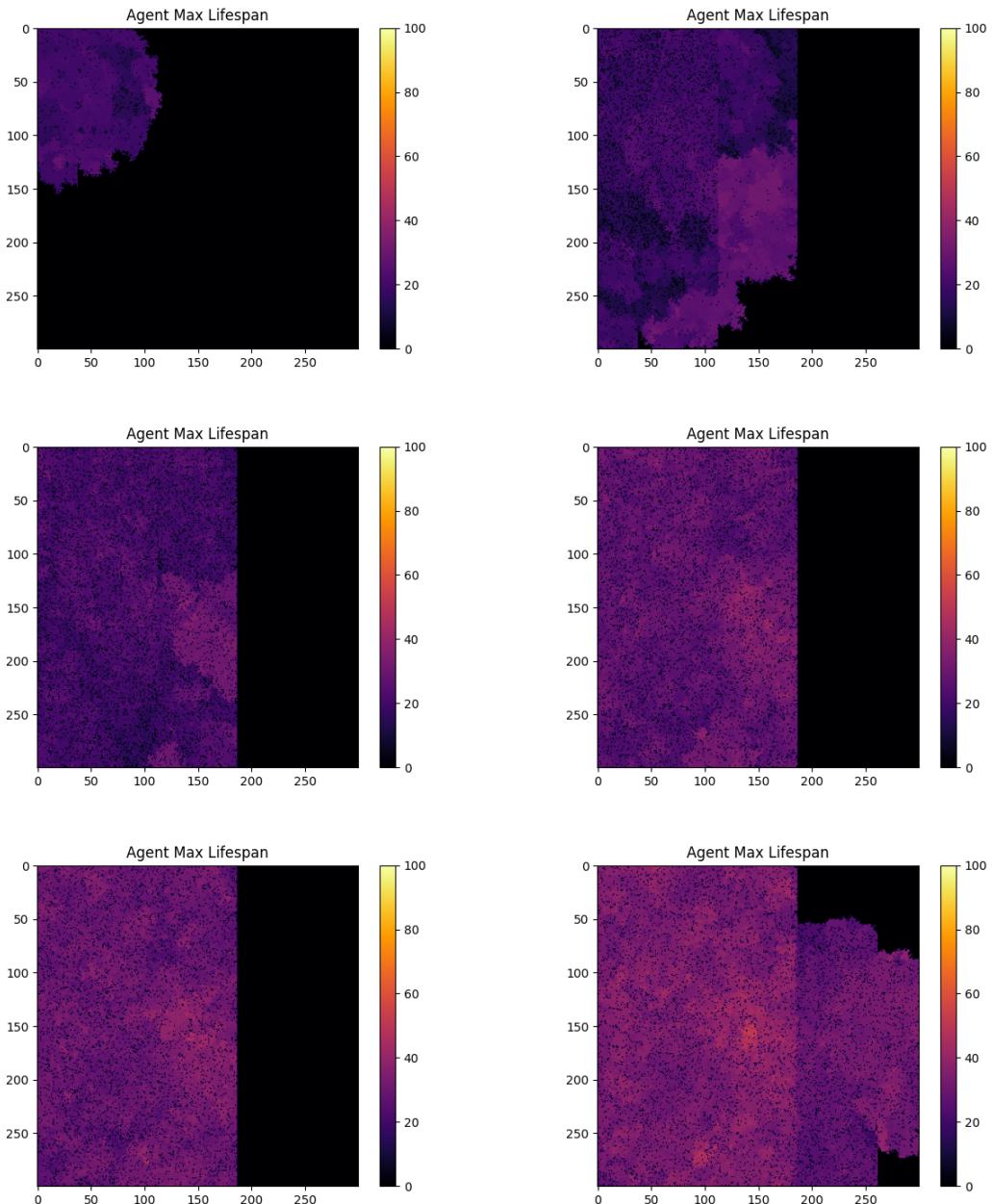


Figure 4.14: Experiment 2 - Agent Maximum Lifespan over 1000 Time Steps. Here we see there was a clear positive selection pressure for longer lifespans. A positive trend towards increasing maximum lifespan was observed as the simulation progressed.

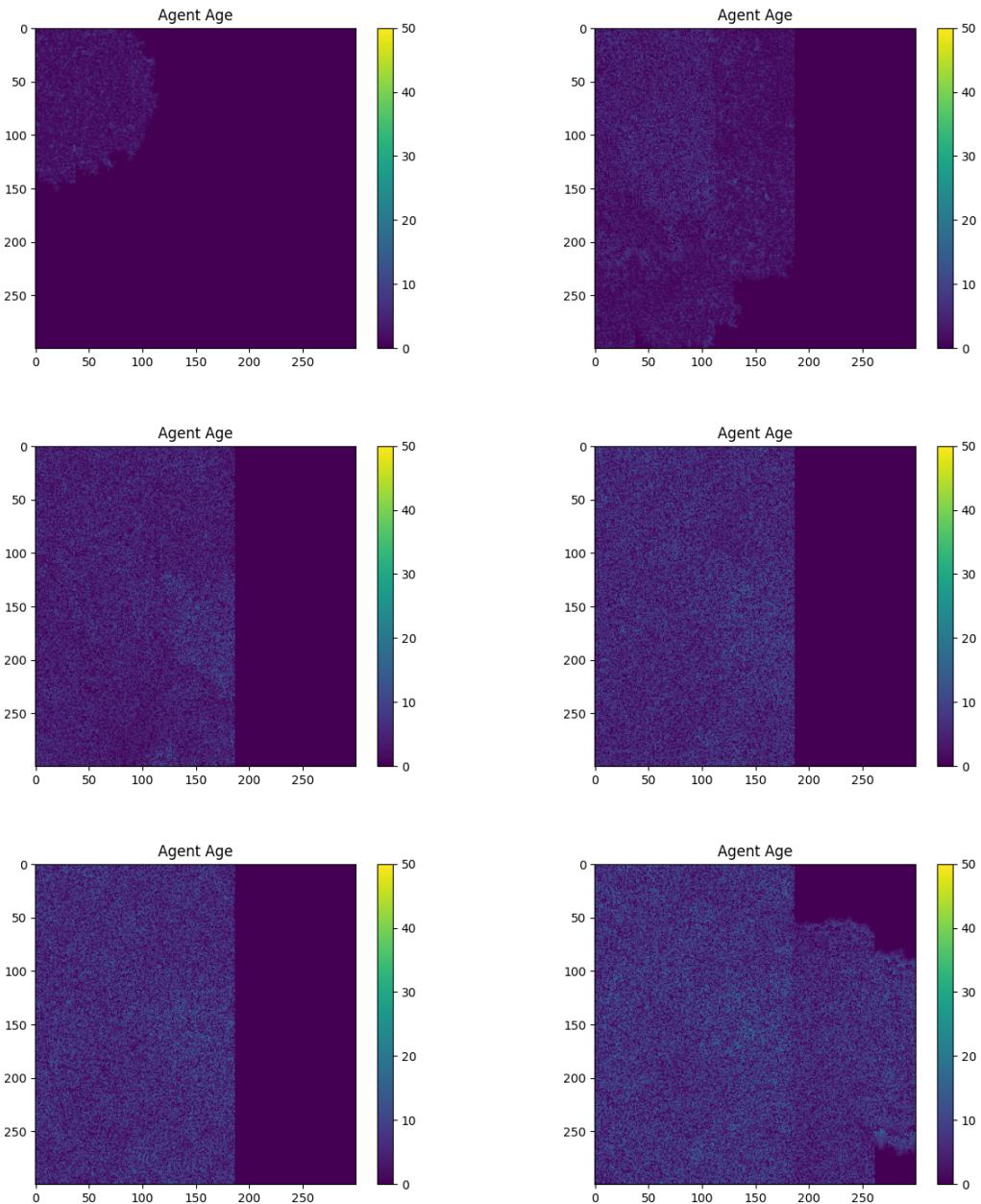


Figure 4.15: Experiment 2 - Agent Age over 1000 Time Steps. This result heavily suggests that the grid locking of the simulation was caused by elitism. With the addition of agent aging and spontaneous death, the average age for each agent is vastly reduced, but still allows for continued propagation of the agent population as a whole.

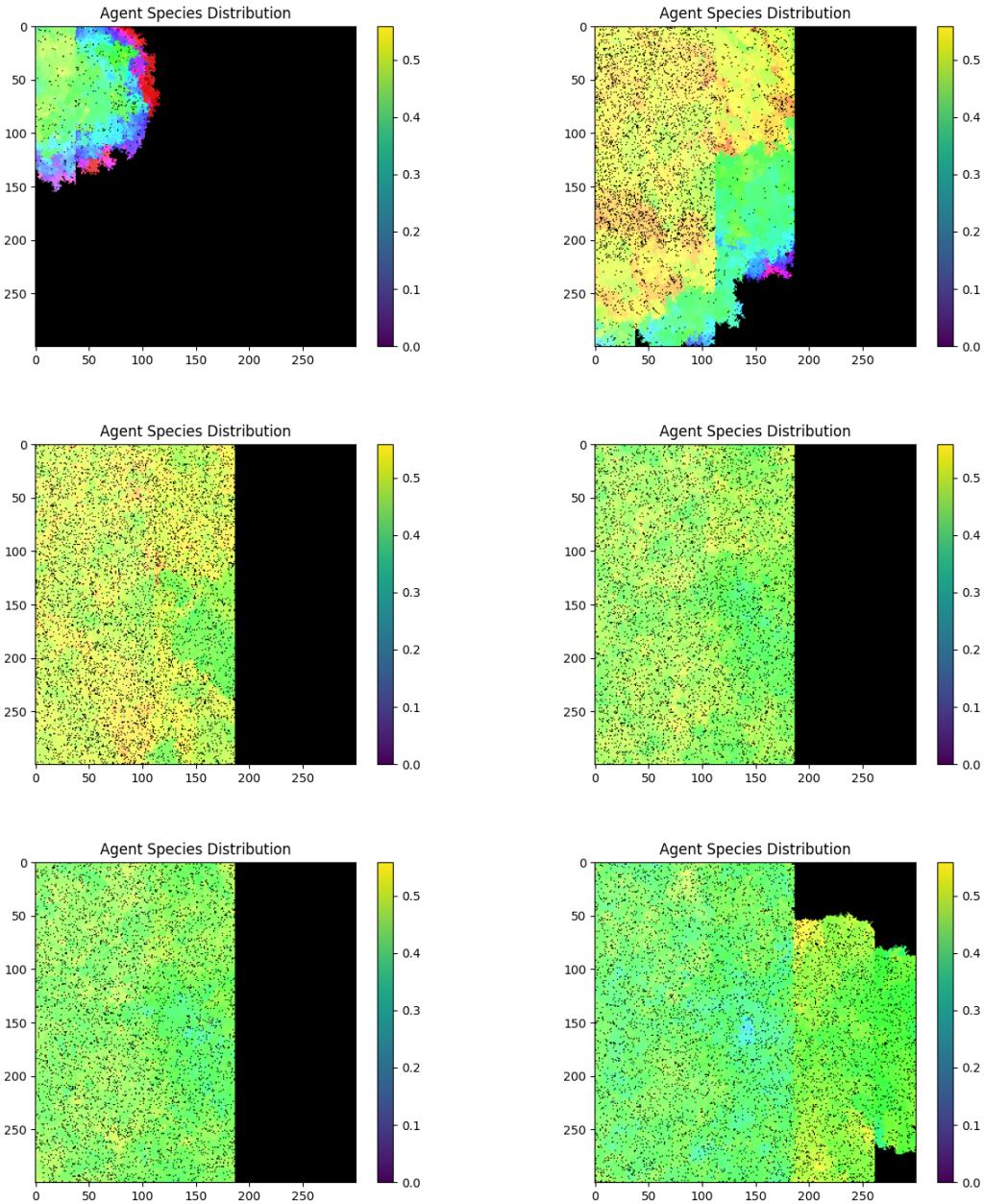


Figure 4.16: Experiment 2 - Species Distribution. We note that the addition of agent aging, while not reducing the amount of species itself, caused a reduction in the variation of species distribution especially as the simulation reached later stages. Unlike before where several species were visible, we see the green species take over the majority of the map. We suggest that this is because without the ability of elites to lay claim to an area, it was much more difficult for any particular species to hold on to a particular region.

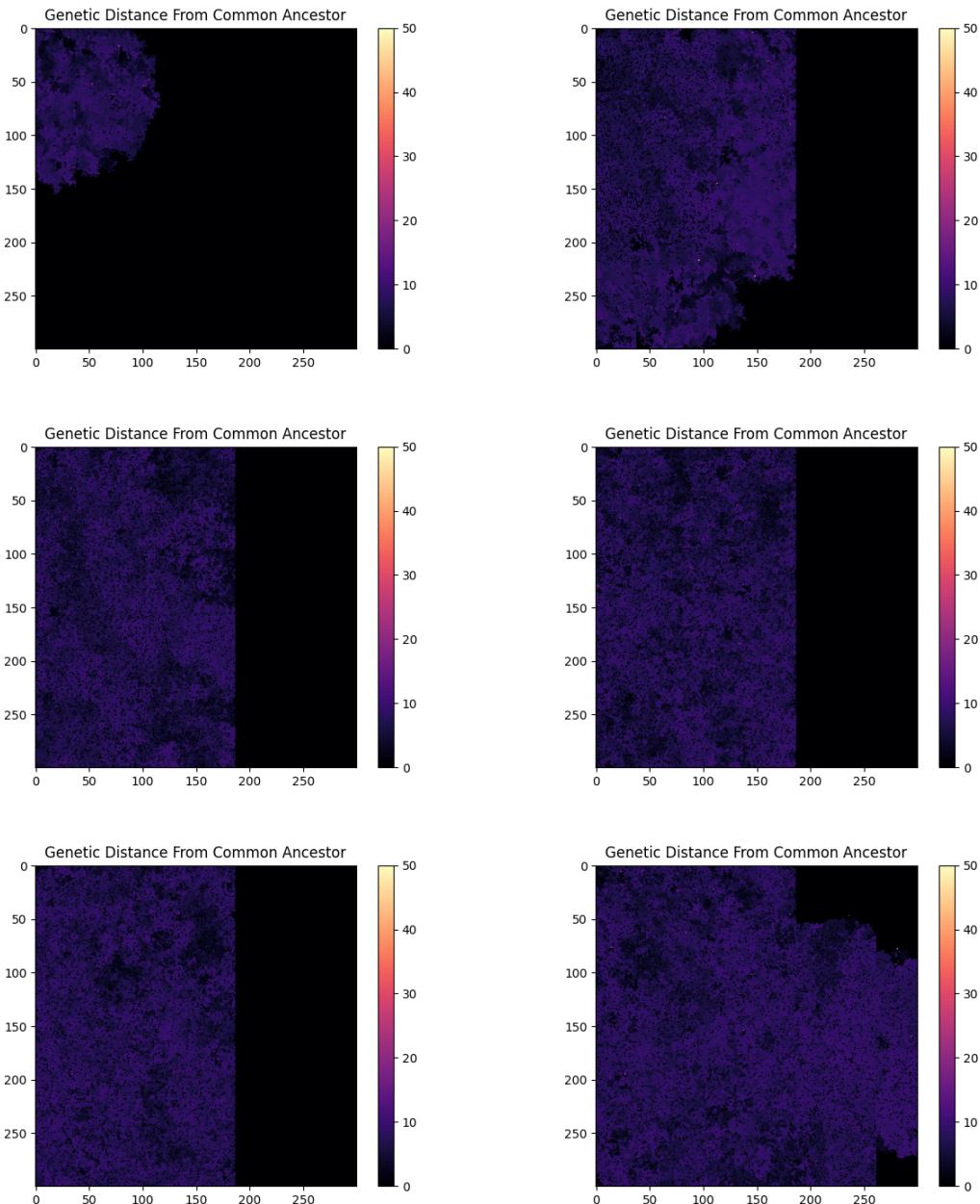


Figure 4.17: Experiment 2 - Genetic Distance from common ancestor over 1000 Time Steps. We note with the addition of aging, this visualization did not stop changing at any point of the simulation. This suggest that the elimination of elitism via aging, allowed the simulation to avoid the gridlock seen before.

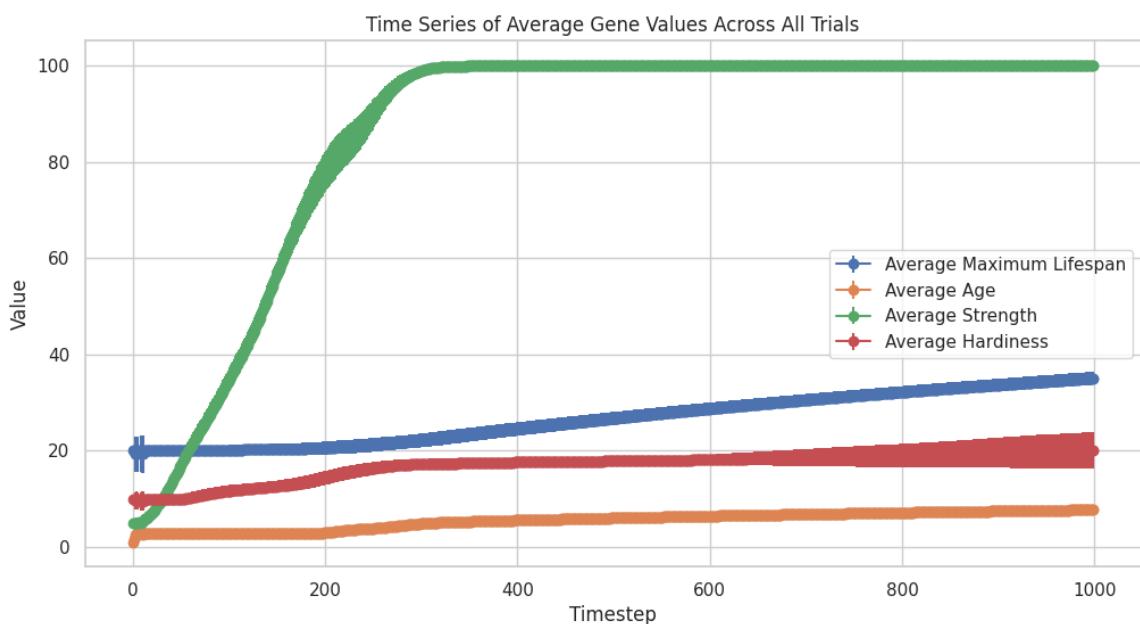


Figure 4.18: Experiment 2 - Average gene values over 30 trials. We note that unlike before, the maximization of the strength gene did not cause a plateauing in the other genes. Both hardiness and maximum lifespan both continued slowly increasing even when strength is maximized.

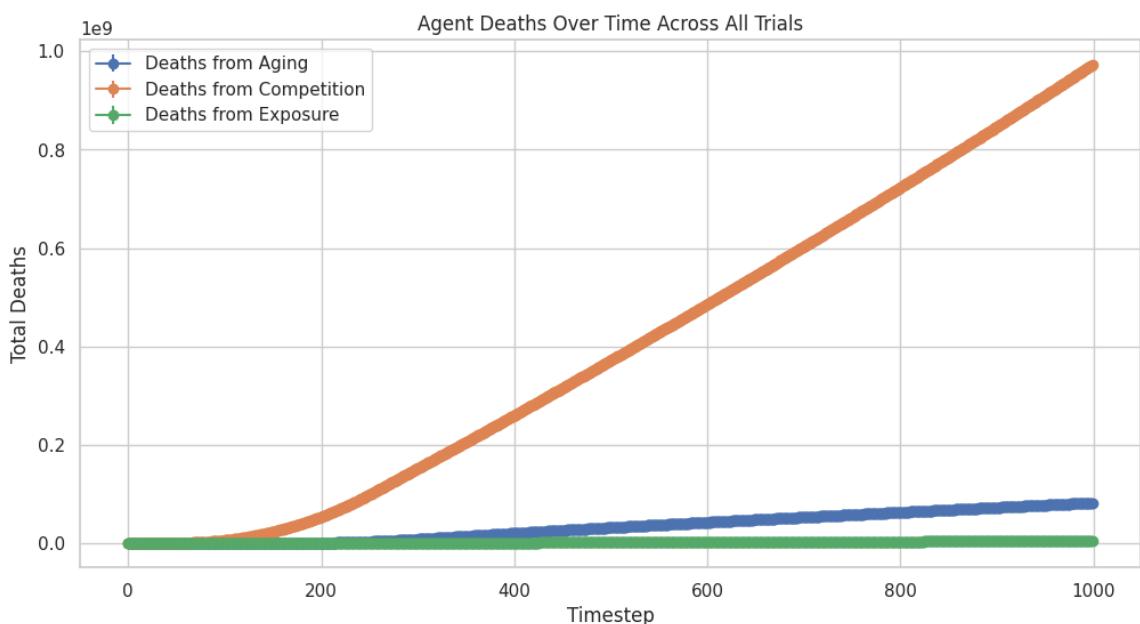


Figure 4.19: Experiment 2 - Average cause of Agent Death over 30 Trials. Here we see that even with the ability to die from aging, the leading cause of death was still conflict with other agents over territory. This would explain why the largest genetic pressure was still on the strength gene.

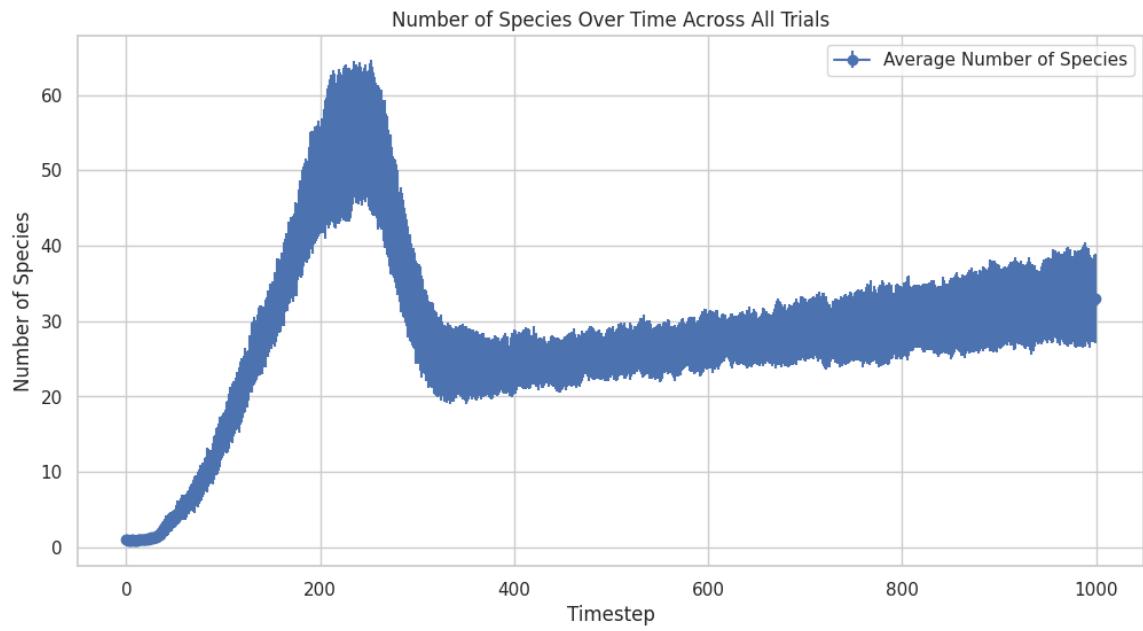


Figure 4.20: Experiment 2 - Average number of species per time step over all trials. Here we note an interesting result. We still see the same fall in overall species numbers around step 300 once the strength gene is maximized, but the number of species does not stabilize afterward as before. We now see that the number of distinct species was still increasing at program termination, and suggests that aging successfully uncapped the simulation's diversity.

#### **4.3.4 Summary of Findings:**

Figures 4.12 - 4.20 contain the results of Experiment 2. In Experiment 2, we introduced cellular aging to mitigate the effects of elitism observed in Experiment 1. The results demonstrated several key observations:

- The strength attribute still reached a maximum value; however, the aging mechanism caused agents to die probabilistically, leading to a "noise" effect that visualizes the impact of aging on the population.
- Agents continued to stratify themselves according to the required hardness of each difficulty region, with many agents now reaching the most difficult areas. This suggests that the elimination of elitism allowed for the continued development of the hardness gene.
- The average age of agents was significantly reduced due to the aging mechanism, but the population continued to propagate effectively.
- Aging did not reduce the maximum number of species but did cause a reduction in species distribution variation. This suggests that without the ability of elites to dominate areas, it was more challenging for any single species to maintain control over a region.
- The genetic distance from the common ancestor continued to change throughout the simulation, indicating ongoing evolution without the gridlock seen previously.
- Both the hardness and maximum lifespan genes continued to increase even when the strength gene was maximized, unlike in Experiment 1 where other genes plateaued.
- Despite the aging mechanism, the leading cause of death remained conflict over territory, maintaining evolutionary pressure on the strength gene.
- The number of species did not stabilize after step 300 but continued to increase, suggesting that aging successfully prevented the simulation's diversity from being capped.

#### **4.3.5 Interpretation**

: The introduction of cellular aging mitigated the elitism problem observed in Experiment 1, preventing the simulation from gridlocking and allowing continuous evolutionary processes. The ongoing changes in genetic distance and the increase in the number of distinct species throughout the simulation indicate that aging allowed for more dynamic speciation patterns.

The continued evolution of the hardness gene and the increase in genetic diversity support the hypothesis that aging promotes more varied survival strategies and prevents stagnation in the evolutionary process. Despite aging, conflict over territory was still the leading cause of death, indicating that the strength gene continued to be under significant evolutionary pressure.

#### **4.3.6 Conclusion:**

We accept the Alternative Hypothesis (H1). The experiment supports that introducing aging will result in more distinct speciation patterns and a reduction in elitism. Older agents will provide opportunities for new variants to emerge, leading to increased diversity and more dynamic evolutionary processes.

## 4.4 Experiment 3: Effect of Map Complexity on Agent Diversity

### 4.4.1 Goal:

Building on the findings from Experiment 2, which addressed the issue of elitism by introducing aging, Experiment 3 aims to investigate the impact of environmental complexity on the evolutionary dynamics of the agents. Specifically, this experiment examines how different map complexities influence agent diversity and adaptation.

To maintain consistency and ensure comparability, the simulation parameters from Experiment 2 will be repeated, including the activation of aging. The independent variable in this experiment is the type of map used, introducing different levels of environmental complexity. The three types of maps used in this experiment are:

- Petri Map: This map replicates the experimental setup of the LTEE. It provides a simple environment with a controlled gradient of difficulty levels. This serves as the baseline environment, representing a simple and predictable setting.
- Perlin Map: This map uses Perlin noise to generate a procedurally created terrain that resembles natural landscapes, including features such as mountains and valleys. The Perlin map introduces a varied and realistic topography, enabling more complex and challenging environment for the agents.
- Random Map: In this map, each cell is assigned a difficulty level at random, creating a highly unpredictable and disordered environment. Unlike the structured gradients of the Petri map or the naturalistic features of the Perlin map, the Random map provides a chaotic setting, testing the agents' adaptability to stochastic variations.

The aim of this experiment is to observe how the evolutionary dynamics change across these different environmental settings. We will measure changes in genetic diversity, speciation rates, and the emergence of adaptive traits in response to the varying levels of environmental complexity.

By comparing the agents' evolutionary trajectories across these three types of maps, we hope to gain insights into how environmental complexity influences the diversity and adaptive strategies of the populations.

### 4.4.2 Hypothesis

**Null Hypothesis (H0):** Increased map complexity will not significantly impact agent diversity. The evolutionary behaviors and speciation patterns of agents will remain similar across environments with varying levels of complexity.

**Alternative Hypothesis 1 (H1):** Increased map complexity will lead to higher agent diversity due to varied environmental pressures and the presence of more ecological niches. Agents will adapt to the diverse challenges presented by complex maps, resulting in greater genetic diversity and more distinct species.

**Alternative Hypothesis 2 (H2):** The change in environmental complexity will affect speciation, but the trend will not be clear. Different types of environmental complexity may lead to varying and unpredictable effects on speciation and diversity, without a consistent pattern across all map types.

#### 4.4.3 Configuration

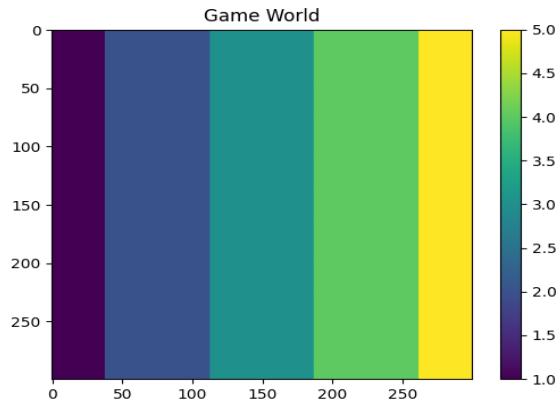
Experiment 3 used the same configuration settings as Experiment 2, except for the selection of map. The 3 maps used are included in the figure displayed on page 45. The configuration file has the following deltas from the previous experiments:

```
Petri Dish Map
{
    "map_type": "petri_dish",
}

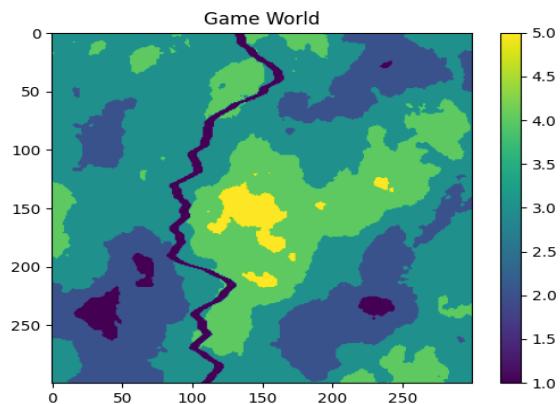
Perlin Noise Map
{
    "map_type": "perlin",
    "use_random_perlin_params": false,
    "use_rivers": true,
}

Randomized Map
{
    "map_type": "random",
}
```

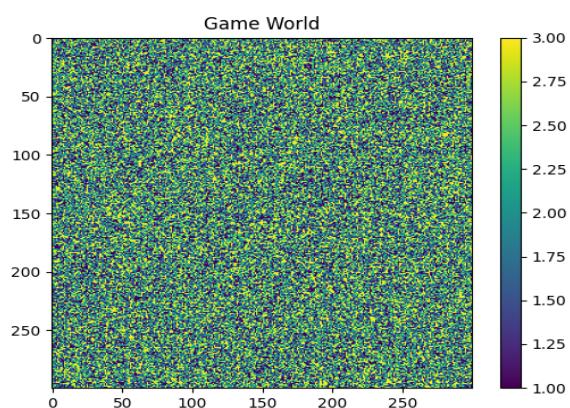
#### 4.4.4 Experiment 3 Maps



(a) Petri Map



(b) Perlin Map



(c) Random Map

Because Experiment 3 focuses on speciation as a function of the environment, only the summary data for species distribution and the average number of species per time step for each of the three maps will be shown.

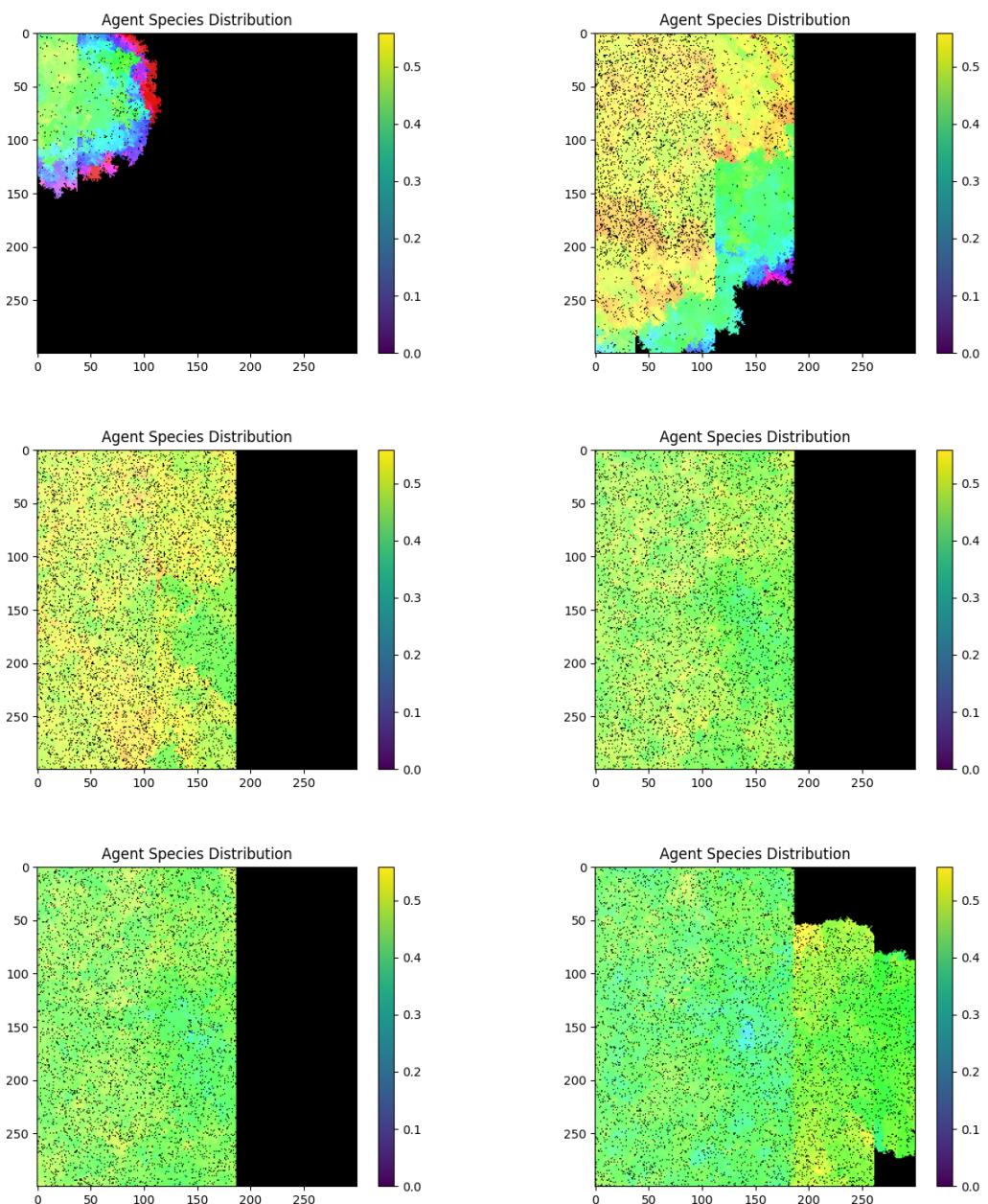


Figure 4.22: Experiment 3 Petri Map - Agent Species Distribution over 1000 Time Steps

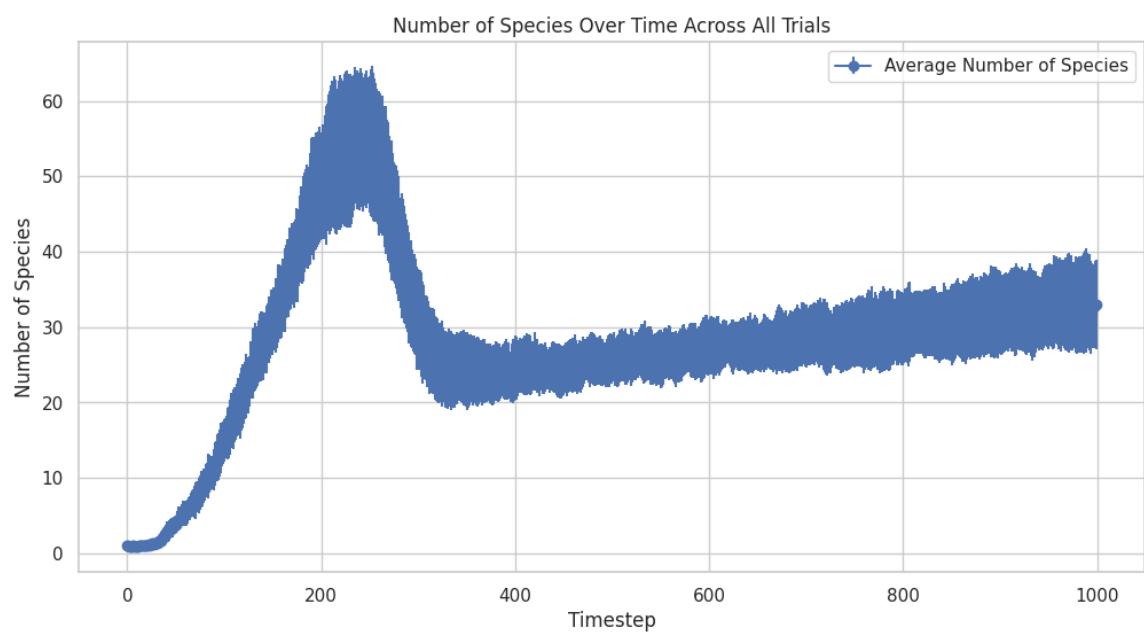


Figure 4.23: Experiment 3 Petri Map - Average number of species per time step over all trials.

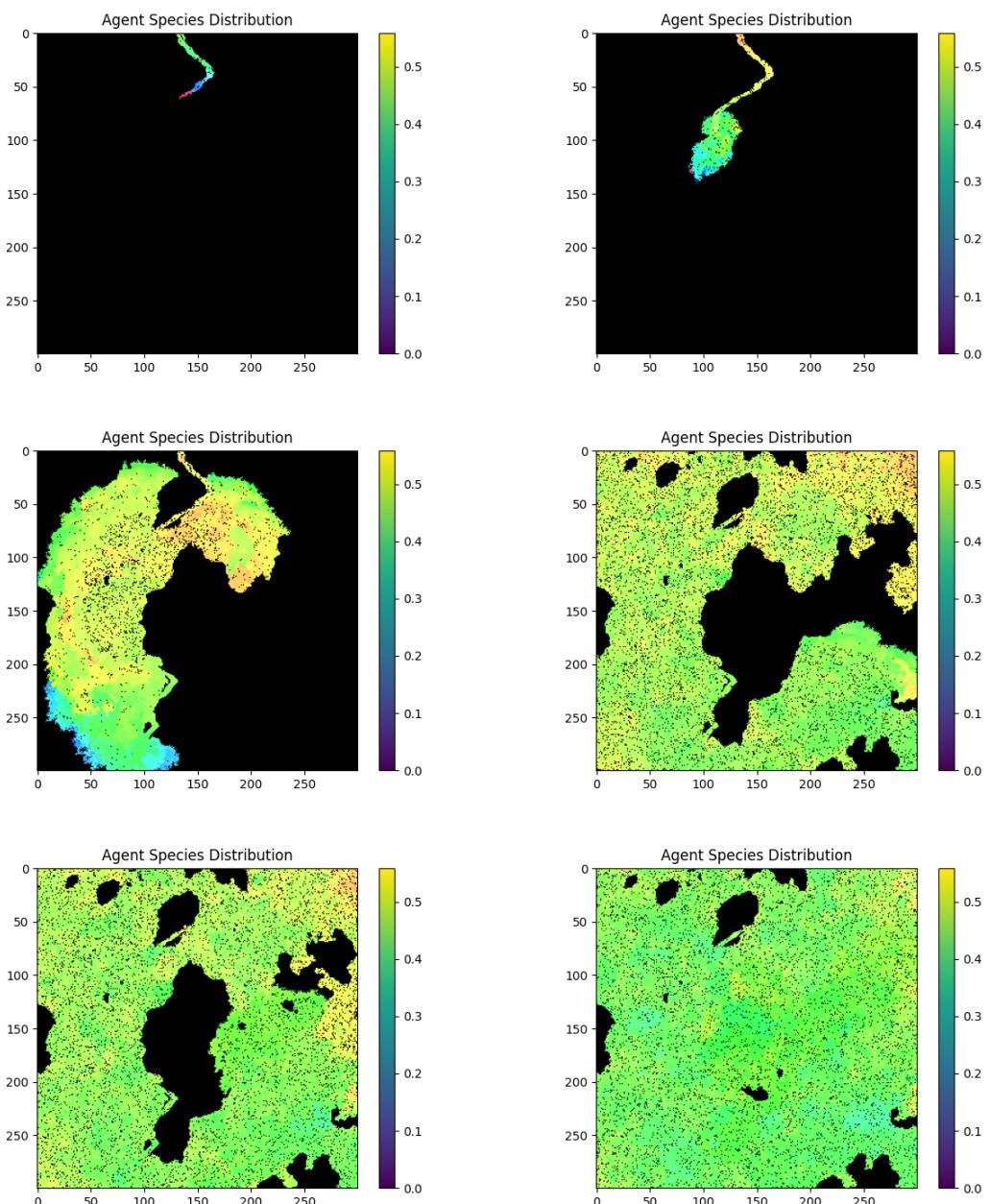


Figure 4.24: Experiment 3 Perlin Map - Agent Species Distribution over 1000 Time Steps

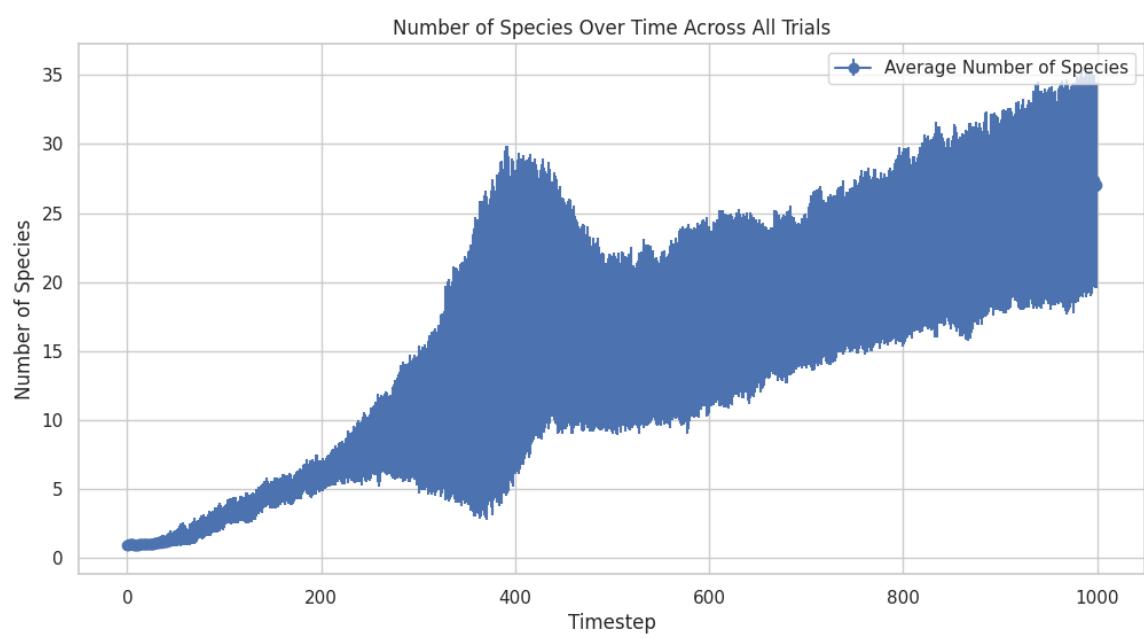


Figure 4.25: Experiment 3 Map - Average number of species per time step over all trials.

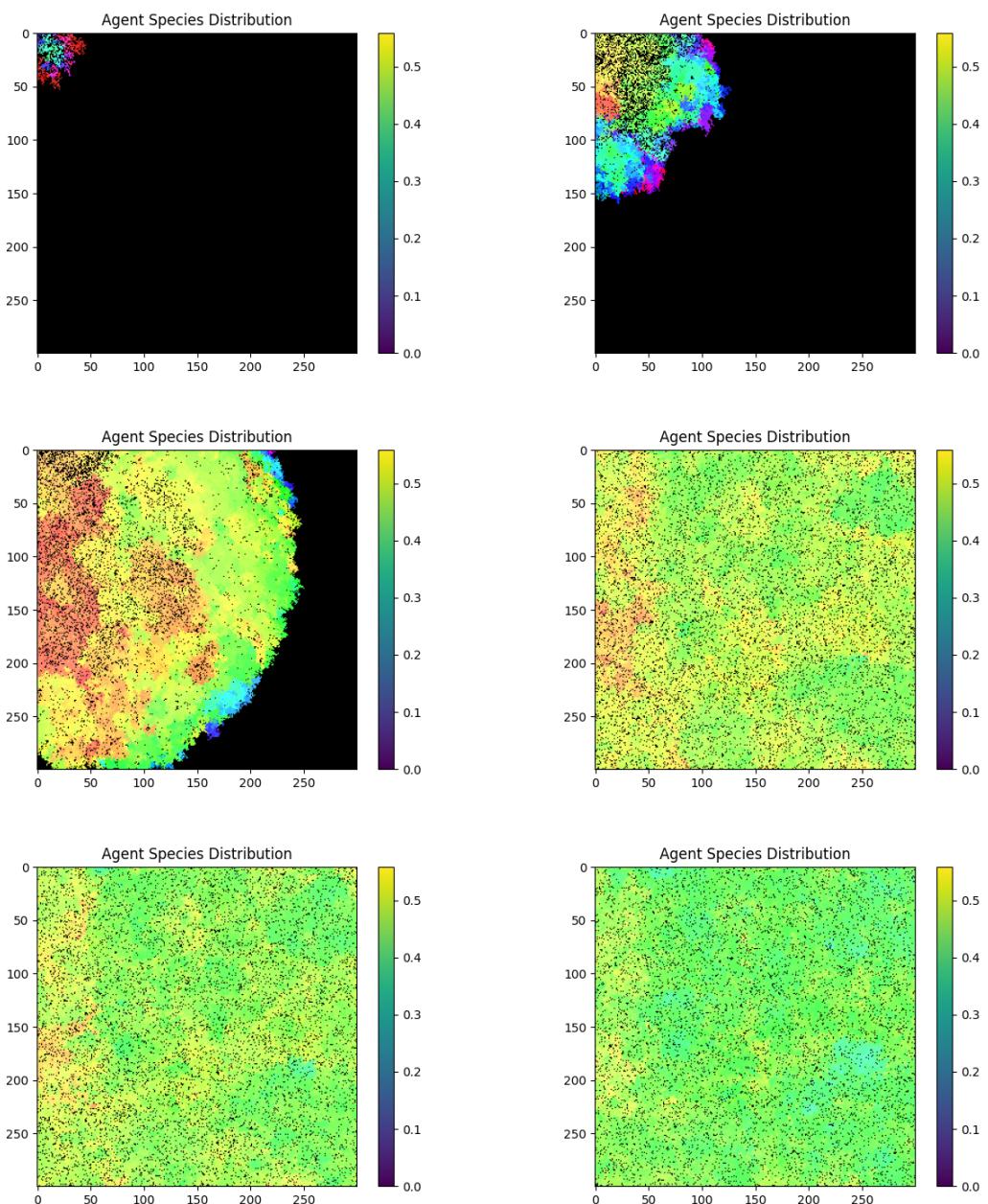


Figure 4.26: Experiment 3 Random Map - Agent Species Distribution over 1000 Time Steps

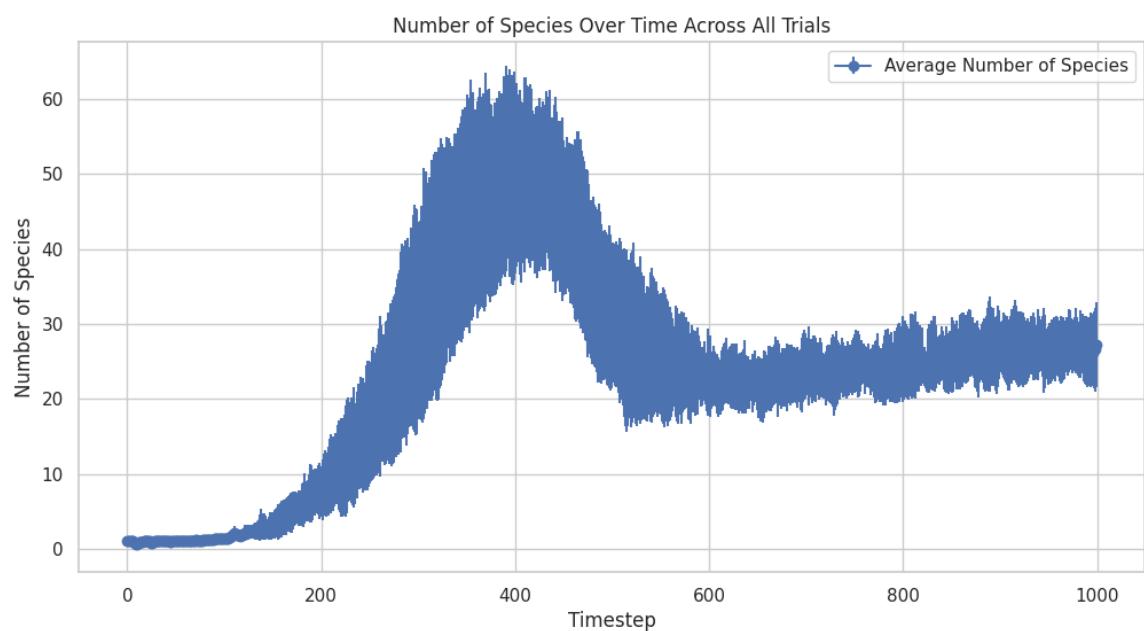


Figure 4.27: Experiment 3 Random Map - Average number of species per time step over all trials.

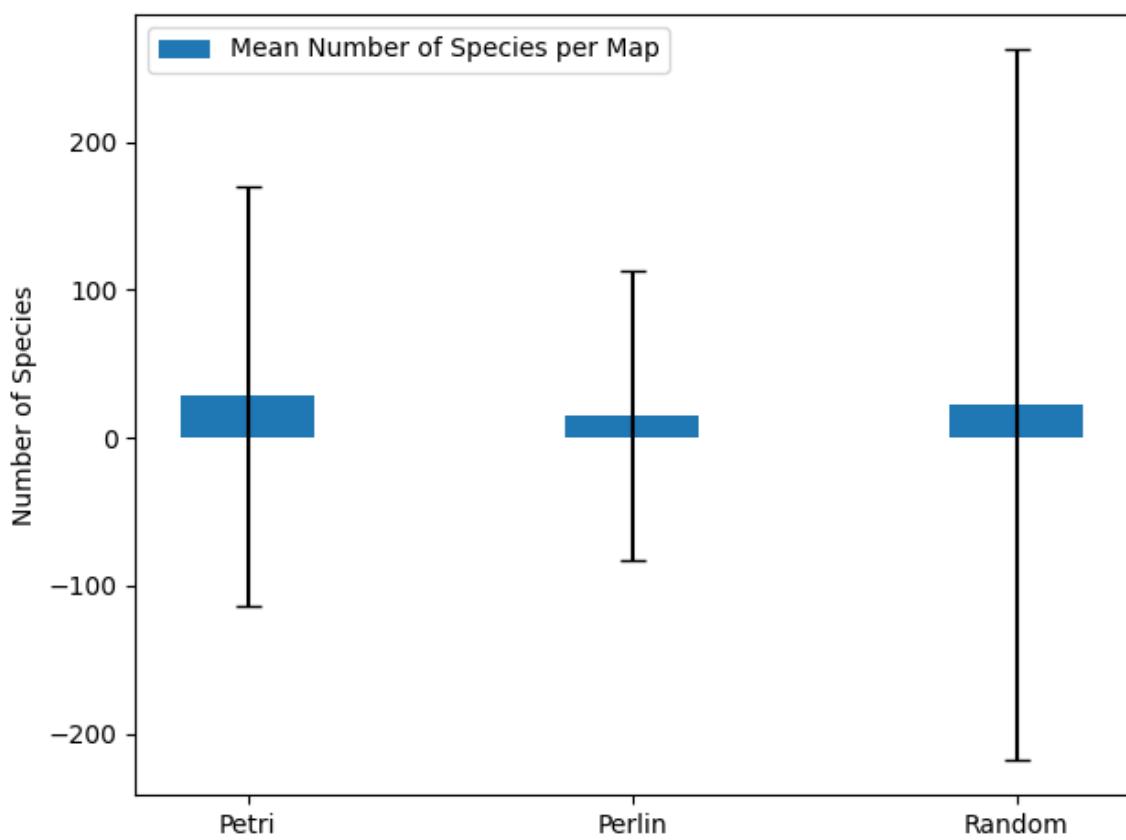


Figure 4.28: Experiment 3: Comparison of the mean number of species per map, over all trials.

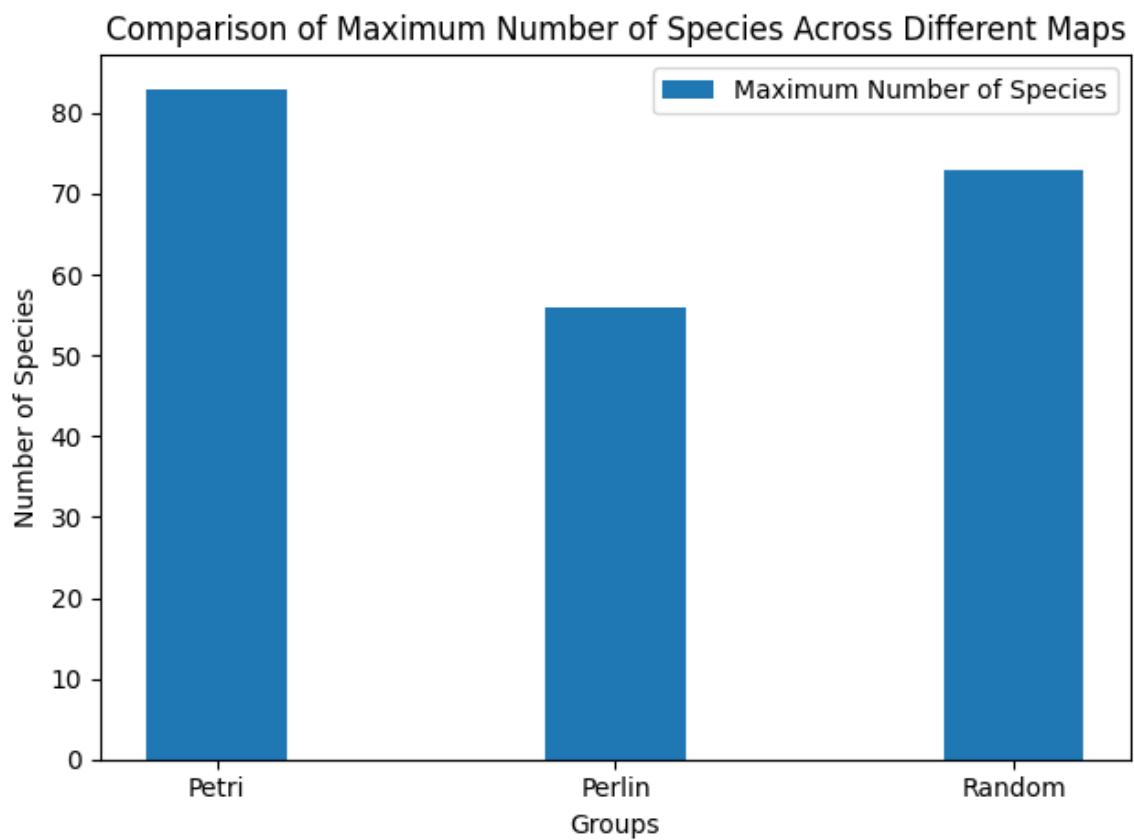


Figure 4.29: Experiment 3: Comparison of the max number of species per map, over all trials.

#### **4.4.5 Summary of Findings:**

Figures 4.22 - 4.29 contain the results of Experiment 3. This experiment demonstrated that map complexity influences species diversity, but the relationship is not straightforward. The different map complexities resulted in varying impacts on speciation and diversity. Specifically:

- The Petri map had the largest mean number of species.
- The Random map had the second largest mean number of species.
- The Perlin map had the smallest mean number of species.
- In terms of variance, the Perlin map had the least variability.
- The Petri map had the second largest variance.
- The Random map had the largest variance by a significant margin.
- For the maximum number of species observed, the Petri map had the most.
- The Random map followed with the second most.
- The Perlin map had the least maximum number of species.

#### **4.4.6 Interpretation:**

The results from Experiment 3 show the role environmental variability plays in shaping evolutionary dynamics and species diversity. The structured gradient of the Petri map created conditions that supported high species diversity and maximum species counts. In contrast, the Random map, despite having the second highest mean species count, showed the greatest variability, reflecting the unpredictable nature of its high entropy environment and the tendency for the agents to die off early and completely in many of the simulations. The Perlin map, with its realistic terrain features, resulted in the lowest mean species count and smallest variance, suggesting that while it provided consistent niches, these were fewer in number compared to the other maps. These findings imply that structured complexities, like those in the Petri map, can enhance diversity, while highly random environments can lead to unpredictable outcomes difficult for the agents to overcome. Realistic but varied terrains might limit the number of adaptive niches available.

#### **4.4.7 Conclusion:**

We reject the Null Hypothesis ( $H_0$ ) because increased map complexity did have an impact on agent diversity. However, we also reject the straightforward version of the Alternative Hypothesis ( $H_1$ ), as the relationship between map complexity and diversity was not linear or straightforward. Thus, the findings indicate that while map complexity affects diversity, the nature of this relationship is complex and requires further investigation to understand fully.

## 4.5 Experiment 4: Effect of Food Simulation on Speciation

### 4.5.1 Goal:

The goal of this experiment is to examine how simulating food availability impacts agent behavior, overall diversity, and speciation. By introducing a dynamic food resource into the environment, we aim to investigate how resource availability influences survival strategies, reproductive success, and the evolutionary pathways of digital organisms. This experiment will help us understand the role of resource distribution and competition in driving evolutionary processes.

### 4.5.2 Hypothesis:

**Null Hypothesis (H0):** The addition of food simulation will not significantly affect the diversity of survival strategies or the rate of speciation. Agents will exhibit similar evolutionary behaviors and speciation patterns as in environments without food simulation.

**Alternative Hypothesis (H1):** The addition of food simulation will promote more diverse survival strategies and increase the rate of speciation. The presence of varying resource availability will drive the evolution of distinct traits and adaptive strategies, leading to greater genetic diversity and the emergence of a greater number of species.

### 4.5.3 Configuration

: Experiment 4 used the same configuration file as experiment 2, with the following deltas:

```
{  
    "enable_food": true,  
    "food_generation_rate": 4,  
    "initial_food": 50,  
    "max_food_capacity": 50,  
}
```

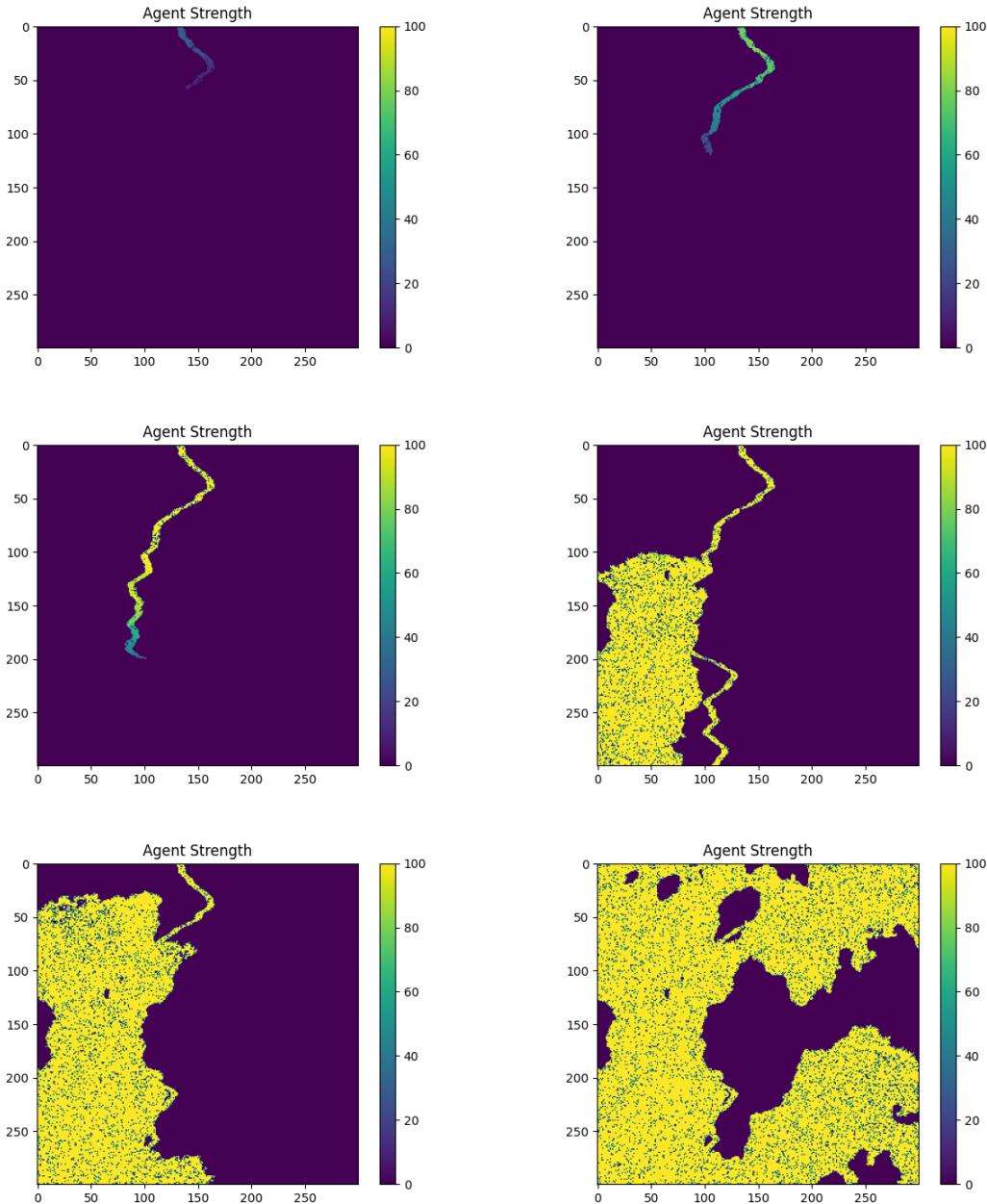


Figure 4.30: Experiment 4 - Agent Strength over 1000 Time Steps.

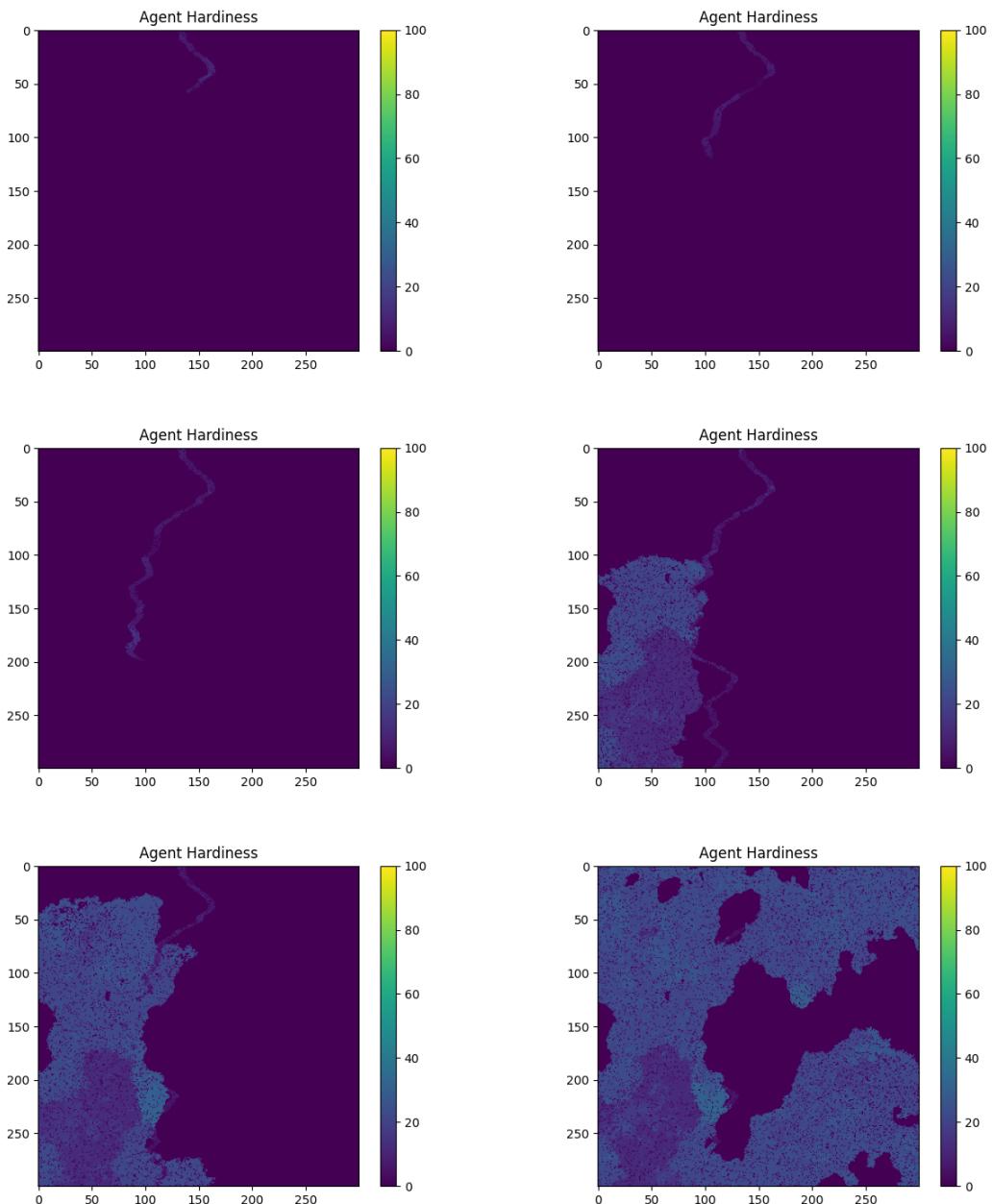


Figure 4.31: Experiment 4 - Agent hardness over 1000 time steps.

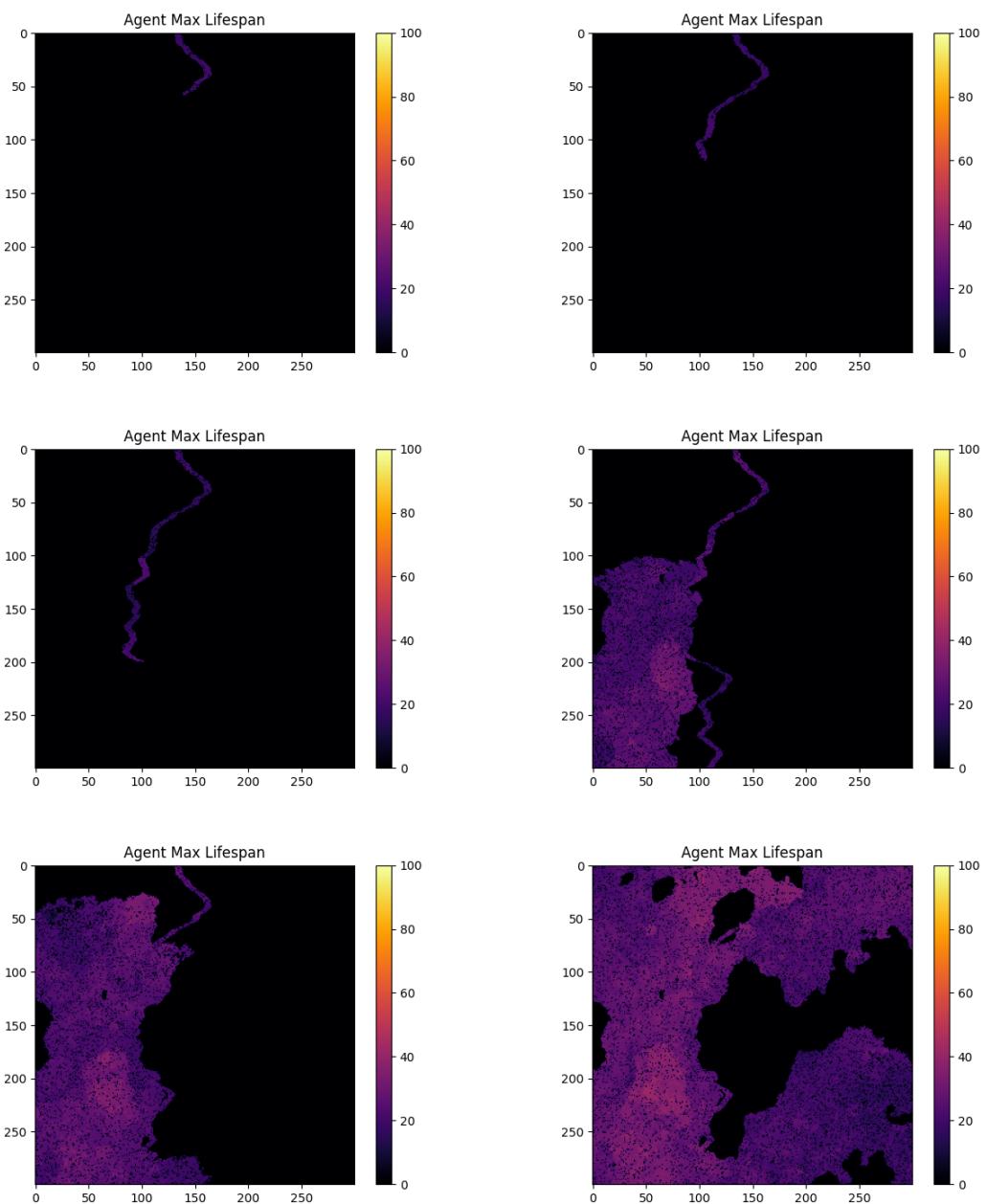


Figure 4.32: Experiment 4 - Agent Maximum Lifespan over 1000 Time Steps.

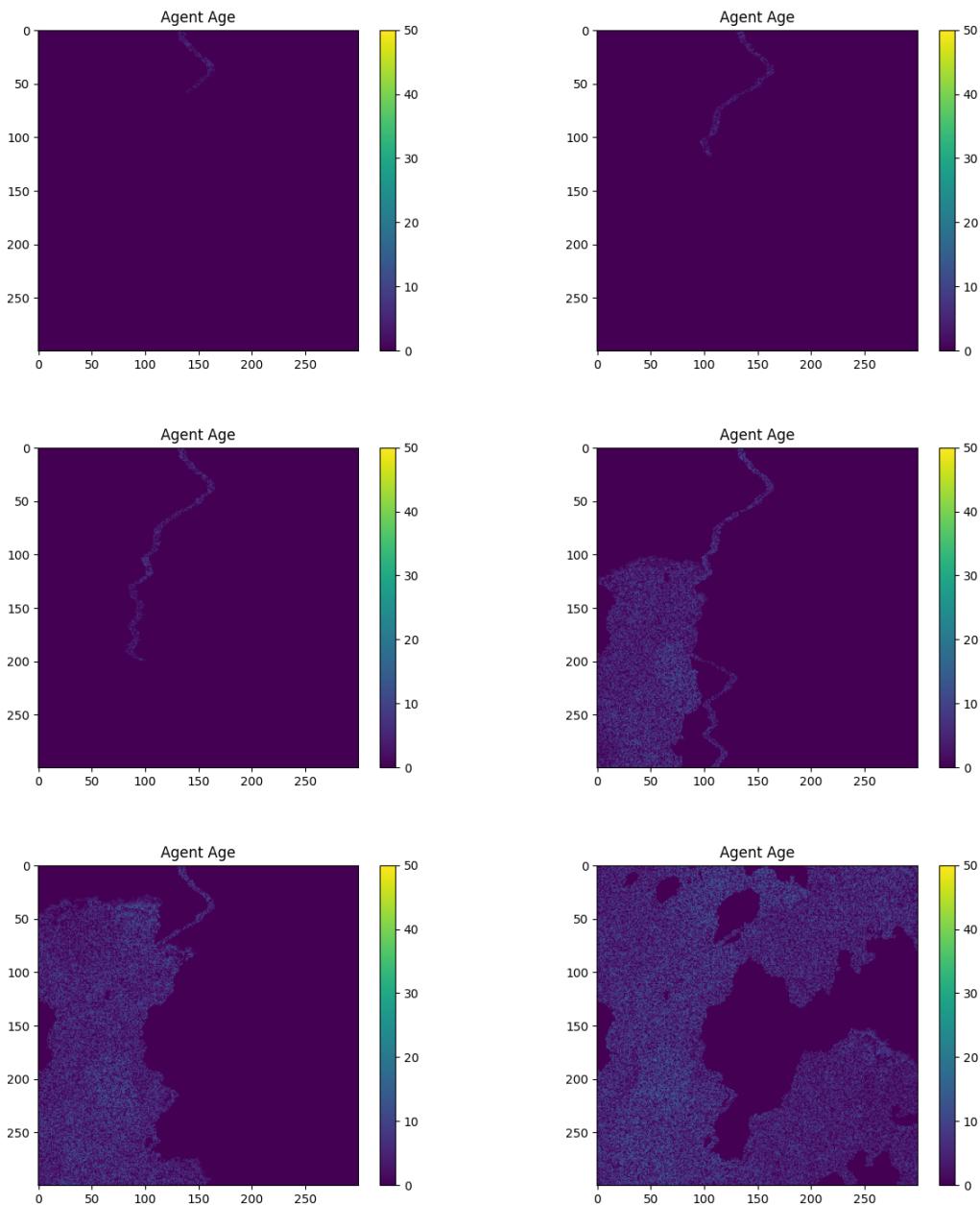


Figure 4.33: Experiment 4 - Agent Age over 1000 Time Steps.

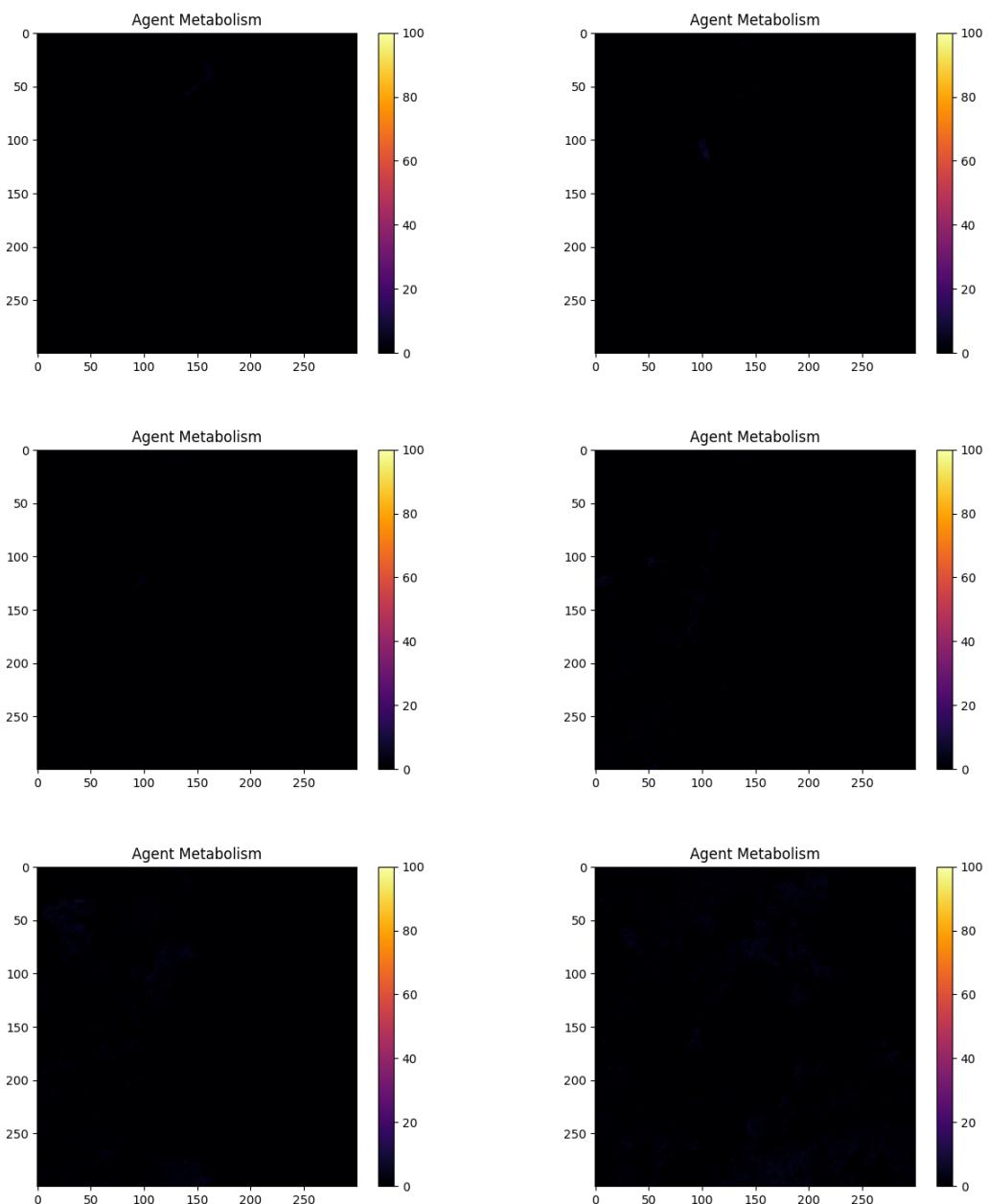


Figure 4.34: Experiment 4 - Agent Average Metabolism over 1000 Time Steps.

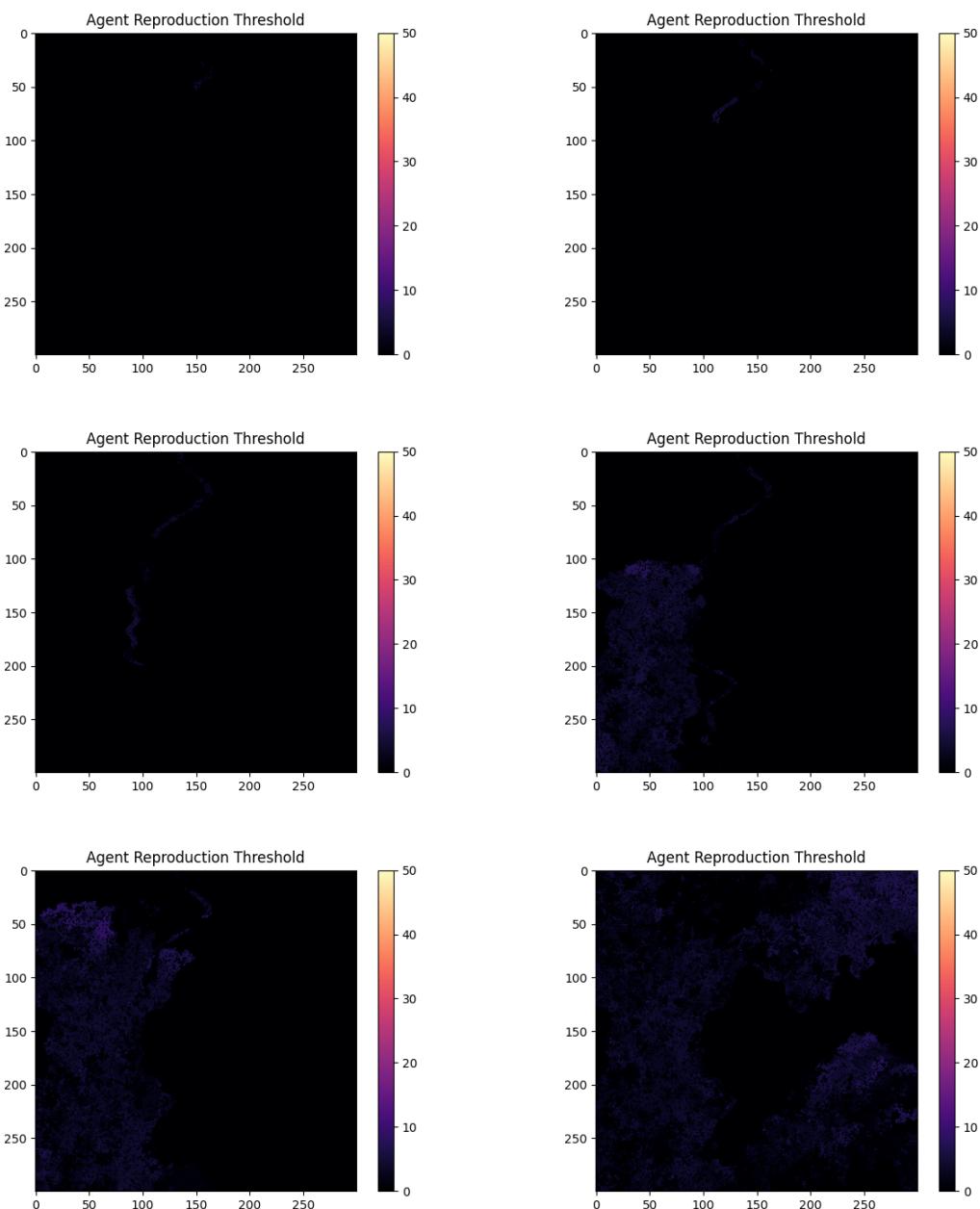


Figure 4.35: Experiment 4 - Agent Average Reproduction Threshold over 1000 Time Steps.

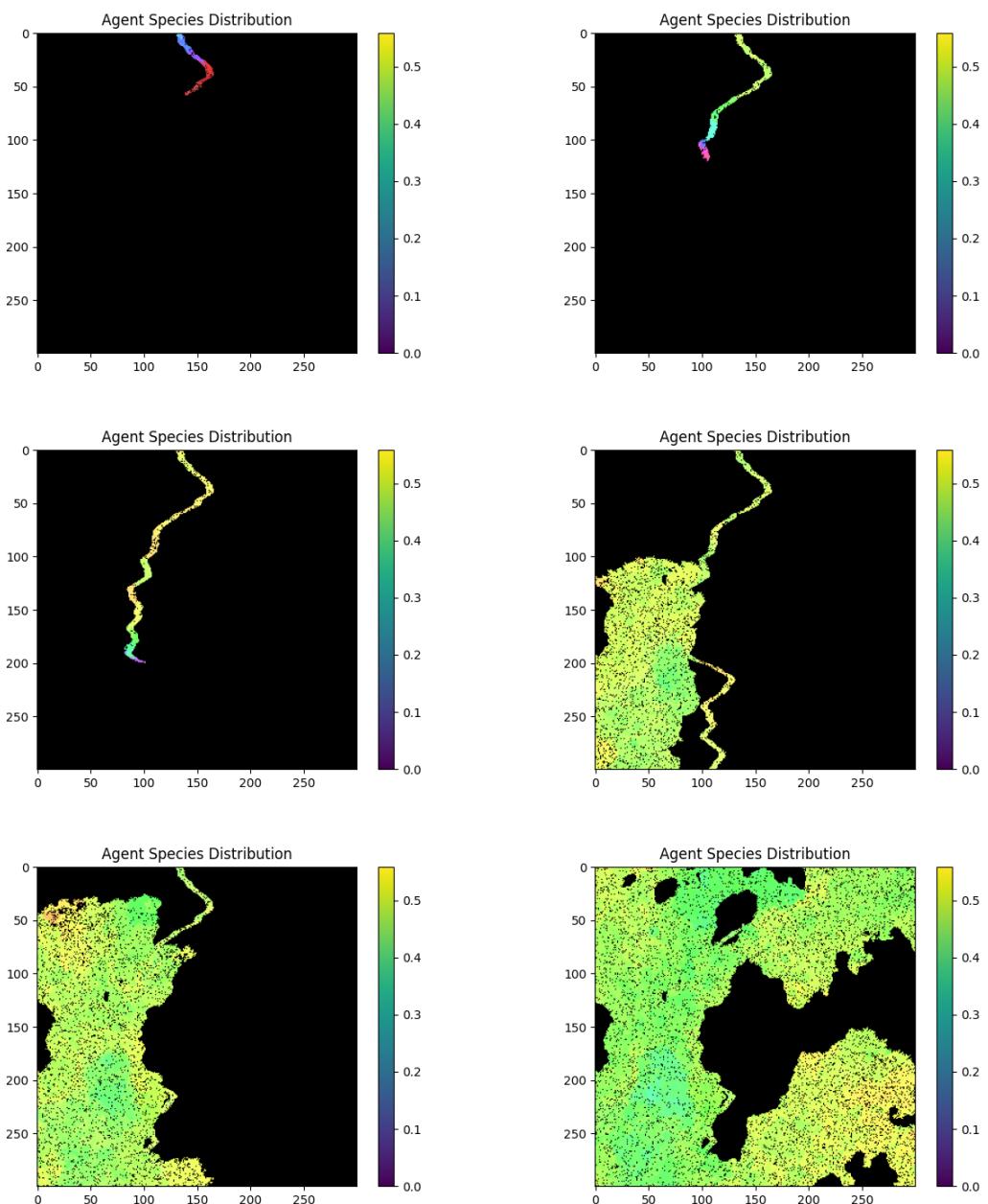


Figure 4.36: Experiment 4 - Agent Species Distribution.

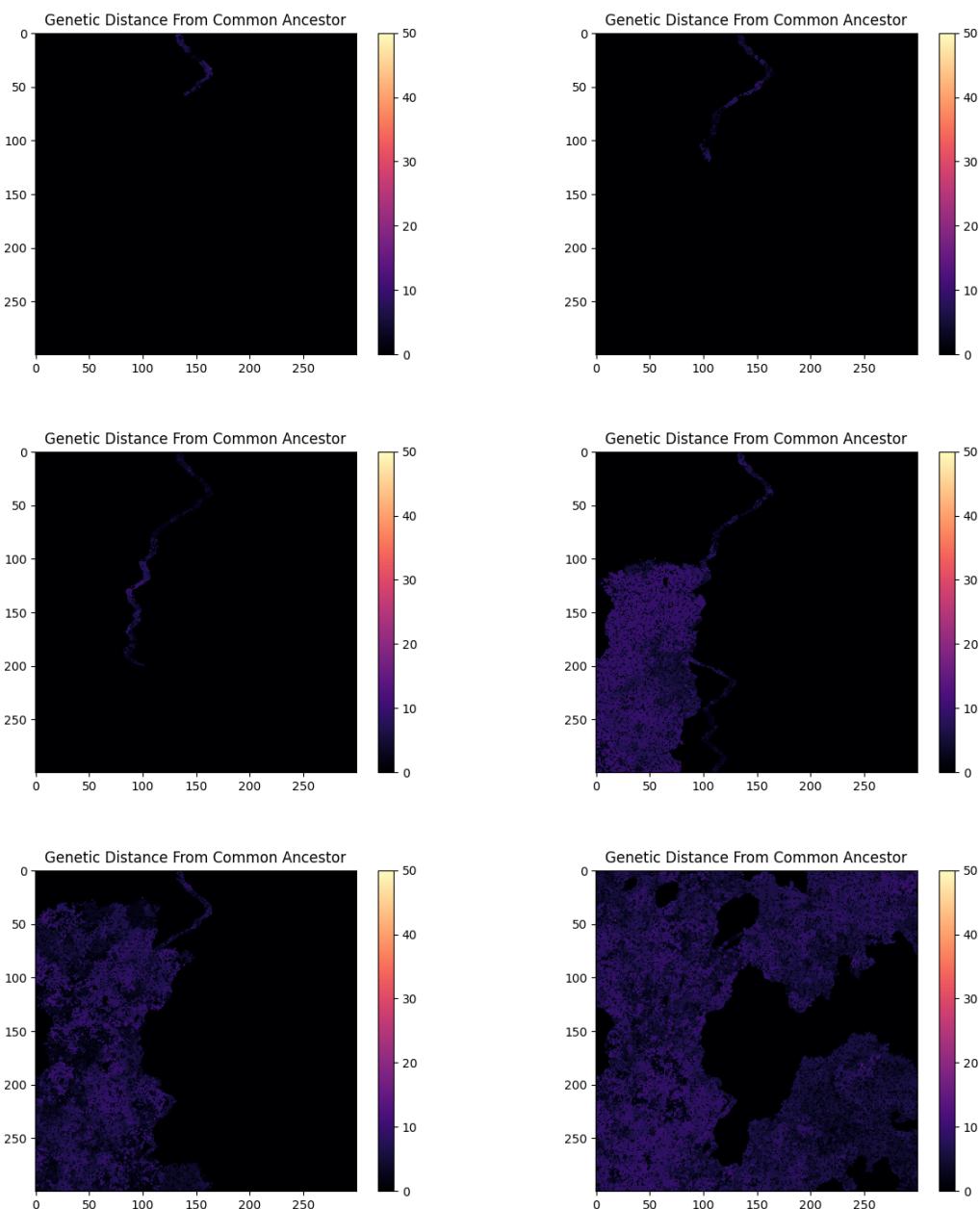


Figure 4.37: Experiment 4 - Genetic Distance from common ancestor over 1000 Time Steps.

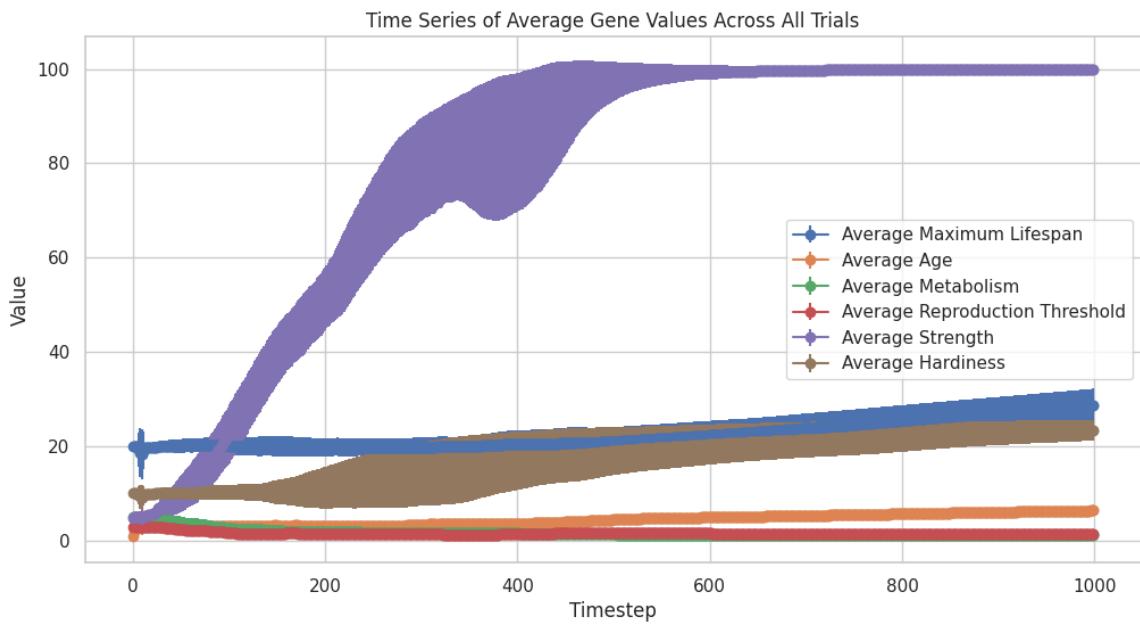


Figure 4.38: Experiment 4 - Average gene values over 30 trials.



Figure 4.39: Experiment 4 - Average cause of Agent Death over 30 Trials.

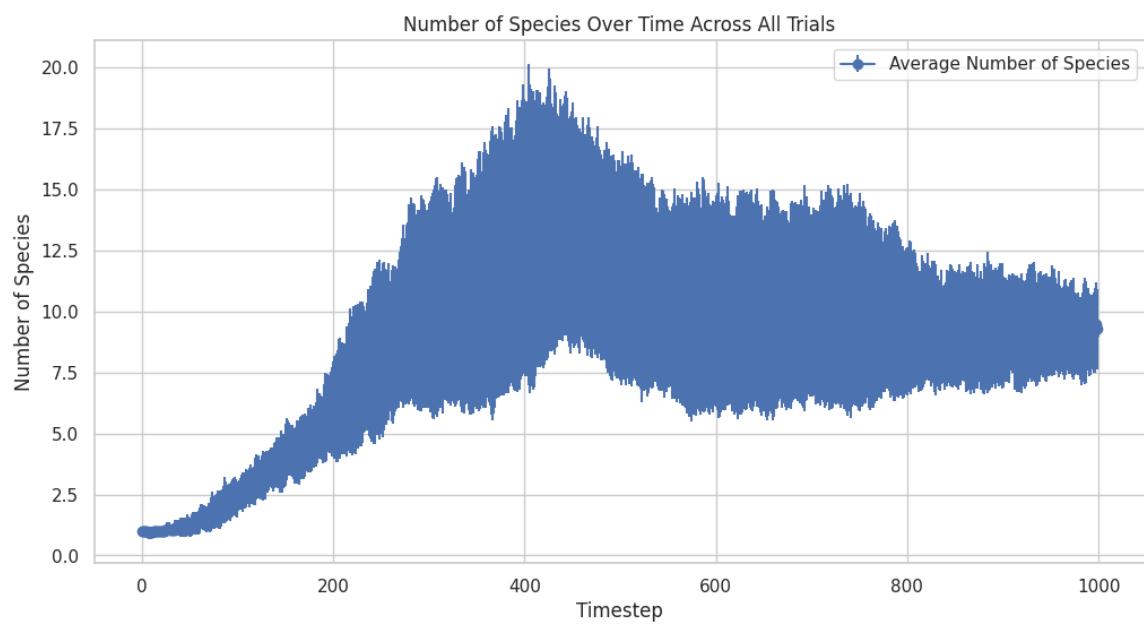


Figure 4.40: Experiment 4 - Average number of species per time step over all trials.

#### 4.5.4 Summary of Findings:

Figures 4.30 - 4.40 contain the results of Experiment 4. The introduction of food simulation had several notable effects on agent behavior and evolutionary dynamics:

- **Minimized Metabolism:** Agents evolved to minimize their metabolism, optimizing their energy usage for survival.
- **Reduced Reproduction Threshold:** Agents also minimized their reproduction threshold, enabling more frequent reproduction with fewer resources.
- **Slower Growth and Change Rates:** The overall growth rate and rates of genetic changes slowed down due to the introduction of food, resulting in fewer speciation events within the allowed time period.
- **Higher Mortality Rate:** Increased agent mortality due to food scarcity slowed down the evolutionary process.

#### 4.5.5 Interpretation:

The findings from Experiment 4 indicate that the introduction of food simulation significantly impacted agent behavior and evolutionary dynamics. Agents evolved to minimize metabolism and reproduction thresholds, focusing on efficient resource use. The higher mortality rate due to food scarcity resulted in a slower overall rate of evolution and fewer speciation events within the given timeframe. However, this does not suggest a limit on the maximum potential diversity. The stable speciation patterns similar to the previous experiments suggest that, despite the reduced rate of evolution, agents continued to adapt to their environment effectively.

#### 4.5.6 Conclusion:

We reject the Alternative Hypothesis (H1) because the findings do not support an increase in the rate of speciation or diversity within the allowed time period after the addition of food simulation. Instead, the data indicates that the higher mortality and slower growth rates slowed down the evolutionary process. However, the data indicates that this investigation is incomplete as a slowing of evolutionary processes does not give information about the maximum diversity supported by an environment. If the simulation was allowed to go on longer, we may find that food simulation supports greater diversity than the other simulations, but this is just speculation and requires further study.

# Chapter 5

## Conclusion

### 5.1 Restatement of Research Statement and Objectives

The primary goal of this thesis was to develop a computational model simulating evolutionary processes within dynamically generated terrains and to study how varying environmental and agent parameters influence evolutionary outcomes. The objectives included: creating an evolutionary simulation framework, investigating the impact of Perlin Noise environments, and assessing genome diversity and allopatric speciation in digital organisms. This research aimed to provide insights into the adaptive strategies and genetic dynamics that emerge in response to environmental pressures and agent attributes.

### 5.2 Summary of Experiments

#### 5.2.1 Experiment 1: Replicating the LTEE Petri Dish with Evolution of "Hardiness" and "Strength"

The experiment successfully replicated the characteristic fan-shaped evolution patterns observed in the LTEE, demonstrating that agents evolved increased resistance over time through genetic descent. This validated the simulation's ability to model real-world evolutionary dynamics.

Conclusion: We accepted the alternative hypothesis (H1), showing that the agents exhibited an increase in 'hardiness' over time and formed distinct fan-shaped growth patterns.

#### 5.2.2 Experiment 2: Effect of Agent Aging on Speciation and Elitism

This experiment demonstrated that introducing aging reduced elitism. Aging fought against elitism by facilitating generational turnover, preventing a few individuals from dominating the population over extended periods.

Conclusion: We accepted the alternative hypothesis (H1), indicating that aging resulted in more distinct speciation patterns and reduced elitism.

### 5.2.3 Experiment 3: Effect of Map Complexity on Agent Diversity

The experiment showed that increasing map complexity had a significant impact on agent diversity, but the relationship was not straightforward or linear. Different types of environmental complexity led to varying effects on speciation and diversity.

Conclusion: We rejected the null hypothesis (H0) and the straightforward version of the alternative hypothesis (H1). The findings supported the alternative hypothesis (H2), indicating that map complexity affects diversity, but the relationship is complex and requires further investigation.

### 5.2.4 Experiment 4: Effect of Food Simulation on Speciation

The experiment showed that introducing food simulation significantly impacted agent behavior and evolutionary dynamics. Agents evolved to minimize metabolism and reproduction thresholds, leading to slower growth rates, fewer speciation events, and higher mortality rates.

Conclusion: We rejected the alternative hypothesis (H1). The findings did not support an increase in the rate of speciation or diversity within the allowed time period. The data indicated that food simulation slowed down the evolutionary process, but this does not provide information about the maximum potential diversity.

## 5.3 Limitations and Future Work

While the study had several significant findings, it also identified some limitations in the simulation as implemented. The simulation framework, despite its ability to model various environmental and biological factors, does not capture the full complexity of real-world ecosystems. The interactions between environmental variables could be explored further to reflect more detailed dynamics. All permutations of the existing parameters could not be fully explored in this paper, and except for the gridlocked experiment 1, the simulations did not run long enough to see the diversity reach a steady state, which limited the ability to definitively define how changing each parameter affected the maximum allowed diversity by the simulation parameters, only the diversity observed in a window of 1000 time steps.

The dominance of the Strength gene observed in the experiments was due to the lack of mechanisms to check the maximization of beneficial traits. In reality, there are trade-offs for any trait, such as a strong individual requiring a higher caloric intake. Future work could introduce such trade-offs to force the agents to exhibit more nuanced adaptations.

The dependence on parameter tuning could limit the generalizability of any findings across different scenarios. Additionally, the largely predefined and deterministic agent behavior patterns (even when some of the behaviors depend on random outcomes) might oversimplify the randomness and unpredictability of natural evolution.

Future research should address these limitations by improving simulation complexity and detail, such as changing resource distributions by biome and adding multiple species that serve different roles. Sexual reproduction with crossover could also be added, with a genetic basis selecting one reproductive strategy over another. And enhancing agent decision-making algorithms with stochastic elements or machine learning techniques could yield more detailed evolutionary dynamics. Introducing an energy budget model to limit attribute expression and create broader metabolic constraints and trade-offs may provide a richer understanding of biodiversity's emergence.

Finally, comparative studies with other evolution simulation frameworks would help benchmark the results, validate the observed patterns, and offer insights into optimizing the framework for practical ecological research or education.

# Acknowledgements

I would like to thank the Ministry of Education, Culture, Sports, Science, and Technology (MEXT) for the scholarship that enabled me to pursue my master's studies in Japan and fulfill a lifelong aspiration.

I am grateful to the University of Tsukuba for providing academic and social support throughout my research.

Great thanks goes to my advisor, Sakurai Tetsuya, for his mentorship and support, which helped make this research possible.

My deepest gratitude to Professor Claus Aranha, whose guidance, knowledge, and unwavering support through health worries, family concerns, and other life challenges enabled me to complete my master's courses in Japan as I have always dreamed.

To my lab mates Felipe, Fabio, Mario, and Yuri, thank you for helping us fight through together. Its us!

Finally, I extend my heartfelt thanks to my wonderful and loving wife, Vivian, for her constant encouragement, patience during late nights, and strength during stressful times. Your belief in me gave me the strength and motivation to achieve this milestone.

# Bibliography

- [1] D.E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, 1989.
- [2] Ken Perlin. An image synthesizer. *SIGGRAPH Comput. Graph.*, 19(97–8930):287–296, July 1985.
- [3] Elizabeth Pennisi. The man who bottled evolution. *Science*, 342(6160):790–793, November 2013.
- [4] Omar Lamrabet, Mathilde Martin, Richard E. Lenski, and Dominique Schneider. Changes in intrinsic antibiotic susceptibility during a long-term evolution experiment with escherichia coli. *mBio*, 10(2):e00189–19, March 2019.
- [5] Vaughn S Cooper and Richard E Lenski. The population genetics of ecological specialization in evolving escherichia coli populations. *Nature*, 489(7417):513–518, October 2000.
- [6] Martin Gardner. The fantastic combinations of john conway’s new solitaire game ‘life’. *Scientific American*, 223(4):120–123, October 1970.
- [7] Bert Wang-Chak Chan. Lenia: Biology of artificial life. *Complex Systems*, 28(3):251–286, 2019-10-15.
- [8] A.E. Eiben and J.E. Smith. *Introduction to Evolutionary Computing*. Springer, 2003.
- [9] Diego Fontaneto and Timothy G. Barraclough. Do species exist in asexuals? theory and evidence from bdelloid rotifers. *Integrative and Comparative Biology*, 55(2):253–263, August 2015.
- [10] Kenneth O. Stanley and Risto Miikkulainen. NEAT: ”evolving neural networks through augmenting topologies”. pages 99–127, 2002.
- [11] Ken Perlin. Making noise. <http://www.noisemachine.com>, Archived from the original on October 8, 2007.