

## **Ecosystem Simulation**

Christopher Palazzolo (0721414)

Department of Computer and Information Science, Trent University

COIS 4470H: Modeling and Simulation

Prof. Wenying Feng

April 14, 2023

**Contents**

1. Introduction.....	2
1.1. Related Work .....	2
2. System Description .....	2
2.1. Simulation Language Selected.....	2
2.2. Conceptual Model.....	2
2.3. Specification Model .....	4
2.4. Key Output Measures and Alternative Configurations.....	6
3. Experiment and Comparison.....	6
3.1. Results from Simulation .....	6
3.1.1. Configuration 1 .....	6
3.1.2. Configuration 2 .....	8
3.1.3. Configuration 3 .....	8
3.1.4. Configuration 4 .....	9
3.2. Comparison and Discussion.....	11
3.3. Statistics Analysis .....	11
4. Conclusions and Recommendations .....	13
5. References .....	14

## 1. Introduction

The application area which was selected is ecosystem simulation. The purpose of the project is to understand sustainability in nature by determining the ratio of different plants and animals required to keep an ecosystem stable. Every effort has been taken to accurately simulate both the plants and animals, however, in some cases information from credible sources was not available, and either an assumption had to be made, or a less than reliable source of information was used. This paper will make it clear when such a case arises. Furthermore, if reliable information can be found, the configuration of the simulation can be updated to reflect that.

### 1.1. Related Work

The evolution component of the simulation is based on the Canonical ES algorithm, which was published in 2018 in the field of neural and evolutionary computing. The paper was called “Back to Basics: Benchmarking Canonical Evolution Strategies for Playing Atari”, where they proposed using a much simpler evolution systems algorithm instead of traditional reinforcement learning.

## 2. System Description

The system works by simulating the consumption and reproduction of wildlife on a day-by-day basis in addition to the growth of plants. The parameters of the system are simply the number of each plant and animal which will interact with each other.

### 2.1. Simulation Language Selected

The simulation was implemented in Python and is comprised of two scripts. The first script is the simulation itself (*main.py* which can be found in *Palazzolo-Report-Code.zip*). It runs the ecosystem and manages the animals within. Every 5 simulated days, it dumps the current state of the simulation to a JSON file, which contains the number of animals of each species at the end of every fifth day. Using tools like the profiler, performance bottlenecks in the code were identified and dealt with.

The second script takes that JSON file and hosts a small web server (*display.py* which can be found in *Palazzolo-Report-Code.zip*). By visiting the home page of the web server, the contents of the JSON file are visualized. It displays the current day and line graphs of the counts per species using the ChartJS library. The web page refreshes the data every 10 seconds. The web server backend is written in Python.

### 2.2. Conceptual Model

At a high level, the simulation starts with the species of animal at the bottom of the food chain and works backwards to the top. For each species, the algorithm starts with the animal who has the highest mass. They have the first pick of everything for the current day. To simulate consumption, the algorithm finds the animal of an edible species with the highest mass, but with a speed less than the animal who is looking for something to eat. If such an animal can be found, it will be eaten. This process repeats until the animal whose turn it is to eat is full. If an animal eats and has a child in its care, the animal will give the calories to each child instead of itself, only consuming what is left over.

Next, the algorithm checks if the animal is eligible for procreation. In the simulation, an animal will either be an “alpha” or a “beta”, with a 50% chance of being born as either. Both one alpha and beta animal are required to reproduce. The algorithm determines if an animal is ready for procreation by checking if the animal does not have any children in its care, and the gestation period for that animal has passed since its last procreation. If both of these conditions are true, then the animal will attempt to locate another animal of its species who has the highest mass, is ready for procreation, and is the opposite of them (in other words, if the current animal is an alpha, then it must search for a beta). If such an animal can be found, procreation can occur. If the children survive birth, they are put in the care of the alpha who is responsible for feeding them. Both parents must wait for the gestation period before procreating again.

Finally, the algorithm checks if the animal has reached its maximum age, or if the animal has not consumed a sufficient number of calories. In both cases the animal dies. For simplicity, the carcass is removed from the simulation and cannot be used as a food source for a predator. The simulation will run until one of the species goes extinct from the ecosystem, at which point, the program will terminate as the ecosystem needs animals of every species to sustain every other species, and to limit them from growing out of control and consuming all of the available prey faster than the prey can reproduce.

If the animal is considered a child, and in the care of another animal, it will not be able to obtain food on its own and it will be unable to procreate. Instead, the child starts with a very low mass which grows at a linear rate each day until they reach their full mass. At this point, they would no longer be considered a child and would leave the care of the alpha. Children are not exempt from being eaten by a predator.

Animals at the bottom of the food chain will eat grass. Simulating millions of individual blades of grass is impractical, but necessary for the level of detail used in this simulation. To get around this, the algorithm keeps a count of the number of blades of grass. For each blade that is eaten, this counter is decremented, and the number of blades enters a queue, and will reach the top of the queue in the amount of time it takes for grass to grow from a seed into a full blade. At that point, the number is added to the total grass available. This method allows grass to be a finite but renewable resource without a high amount of overhead.

The simulation determines how many calories each animal should consume per day from the following formula. This formula only works for domesticated cats and dogs; however, it is the only formula which could be found to calculate the daily calorie requirements of an animal. To calculate, first the resting energy requirement (RER) is needed. This is the energy required to “...perform essential body functions like digestion, respiration, heart functions, brain functions, etc. ... which can be calculated by multiplying the animal’s body weight in kilograms raised to the  $\frac{3}{4}$  power by 70.” (*Dog And Cat Calorie Calculator / OSU Veterinary Medical Center*, n.d.). The RER must then be multiplied by some factor to represent the energy spent on physical activities when the animal is not at rest. According to OSU Veterinary Medical Center, active and working dogs have a multiplier of 2-5. From this information, the following formula can be created.

$$\text{Daily Calories} = \text{Multiplier} \times 70(\text{weight})^{\frac{3}{4}}$$

If the calories which the animal consumed over their lifetime divided by the number of calories, they should have consumed drops below 10%, the simulation considers the animal to have starved.

When a child is born from its parent, it is not an exact copy. A slight mutation will be applied to the weight, speed, RER multiplier and minimum hunger to eat in order to simulate evolution and natural selection within the ecosystem. The mutation is generated from a standard normal distribution with some multiplier to have detailed control over the rate of evolution. The weight, speed, and RER multiplier uses a mutation multiplier of 0.1, and the minimum hunger to eat has a multiplier of 0.01 (1%). Using a multiplier on a standard normally distributed random number comes from the Canonical ES algorithm proposed in the paper “Back to Basics: Benchmarking Canonical Evolution Strategies for Playing Atari”.

Lastly, the starting animals in the ecosystem are created as adults who must wait for one gestation period before they are ready to procreate. All animals (regardless of if they are born during the simulation or are an original animal) start with one day’s worth of calories. In other words, they are treated as if they are completely full and have no need to eat right away. This prevents them from starving immediately on birth.

### 2.3. Specification Model

The food chain which will be simulated is modeled after the following. Grass will act as the primary producer, which will be consumed by a grasshopper followed by a frog, a snake, and finally a hawk which will be the apex consumer (Mukherjee, 2023). For simplicity, the simulation will not consider the sun or decomposers, and due to performance issues, which will be discussed in section 4, the hawk will not be included in the simulation.

For each species in the simulation, the system needs to know the number of calories which would be gained for each kilogram of animal consumed, the age (in days) when the animal can be considered an adult, a list of prey, weight, speed, maximum age, gestation period, and children born per procreation. The following table shows all of the information which will be used. The grass is modeled after Annual Ryegrass, the grasshopper is modeled after the White whiskered Grasshopper, the frog is modeled after the Common Frog, and the snake is modeled after the Brown Tree Snake. These animals were selected because the prey they are supposed to eat in the simulation is part of the diet of that specific type of animal.

	White whiskered Grasshopper	Common Frog	Brown Tree Snake
Calories Gained	2,570/kg ( <i>How Many Calories Are in Grasshopper</i> , n.d.)	730/kg ( <i>FoodData Central</i> , n.d.)	930/kg ( <i>The Nutrition of Snake / Nutrition / Healthy Eating</i> , n.d.)
Adult Age	25-30 days ( <i>targetstudy.com</i> , n.d.)	3 months ( <i>Croft</i> , 2023)	2-4 years ( <i>Snake Life Cycle : Lifespan &amp; Reproduction / Critter Control</i> , n.d.)

Weight	Males = 0.00011 kg Females = 0.00031 kg (United States Department of Agriculture, 1994)	0.0227 kg ( <i>Frogs Facts, Types, Lifespan, Classification, Habitat, Pictures</i> , 2020)	0.525 kg ( <i>The Animal Facts</i> , 2023)
Speed	3.048 m/s (Piercy, 2022)	4.4704 m/s (Herpetologist, 2022)	5.89408 m/s ( <i>Here Be Answers</i> , 2022)
Lifespan	12 months ( <i>Grasshopper Lifecycle</i> , n.d.)	Unknown. Could be 1 day to 30 years. (NATURE, PBS, 2021)	10-15 years ( <i>Boiga Irregularis (Brown Catsnake, Brown Tree Snake)</i> , n.d.)
Gestation Period	Lays eggs about every 30 days ( <i>A Grasshopper's Life Cycle</i> , 2020)	2-3 weeks ( <i>Tree Frog Metamorphosis - King County</i> , n.d.)	28-45 days (Nmr, 2016)
Children Born Per Procreation	3-5 eggs (Natural Heritage Program and Montana Fish, Wildlife & Parks, 2023)	Up to 4000 eggs ( <i>Common Frog / the Wildlife Trusts</i> , n.d.)	3-12 eggs ( <i>Boiga Irregularis (Brown Catsnake, Brown Tree Snake)</i> , n.d.)
Probability of Child Surviving Birth	50% (targetstudy.com, n.d.)	As low as 1% ( <i>Spawn &amp; Tadpoles FAQs</i> , n.d.)	Assumed to be 100% No information could be found

In terms of grass, it has an average of “33 calories per 100 grams” (Girl, 2021). To convert this to calories per blade, the average mass of a blade of grass is required. Unfortunately, there are no credible sources on the exact weight of a single blade of grass, however, after checking many different online forums (which is by no means a credible source) the general consensus seems to be around 0.1 to 0.01 grams. An assumption of 0.055 grams per blade will be used, which is the average of the range found. Therefore, the simulation will be run with the assumption that eating a single blade of grass will give the consumer 0.01815 calories.

In terms of the grass growth, depending on the type of grass “...it can take anywhere between five and 30 days for grass seed germination to begin. After that, it takes another three to four weeks before the grass is long enough to mow.” (Mymove, 2022). The assumption will be made that “long enough to mow” refers to the blade being an adequate height to be eaten. The grass in the ecosystem is modeled after Annual Ryegrass. Ryegrass “typically germinates in five to 10 days” (Mymove, 2022). Between the germination and growth periods, it will take an average of 26 to 38 days for the grass to regrow from being eaten.

In the event that a value is a range, a normal distribution is used instead. The mean is the average of the minimum and maximum, and the standard deviation is set to a number where both the minimum and maximum values are two standard deviations from the mean.

## **2.4. Key Output Measures and Alternative Configurations**

The goal of the simulation is to find a ratio of animals in a food chain in order to create a stable ecosystem. For the purposes of this simulation, a stable ecosystem is one where no species go extinct. Unfortunately, this simulation cannot be used to prove an ecosystem will be stable forever. If the simulation was run for a year without failure, it still could not be concluded that the ecosystem is stable because it could fail after two years. The simulation can only show that an ecosystem fails after a given amount of time. The longer an ecosystem remains stable, the better the selected ratio. The simulation uses a high amount of random number generators of various distributions so running the simulation with the same parameters and ratio of species can produce wildly different results.

After a considerable amount of experimentation, it was found that 10 animals of a species have enough alpha and beta animals to start the population, regardless of the lifespan, gestation period, and children per procreation (assuming there is a sufficient amount of food to sustain the animals). For that reason, the apex consumer of the ecosystem will start with a population of 10. From the ratios provided to the simulation, the number of starting animals per species can be calculated.

## **3. Experiment and Comparison**

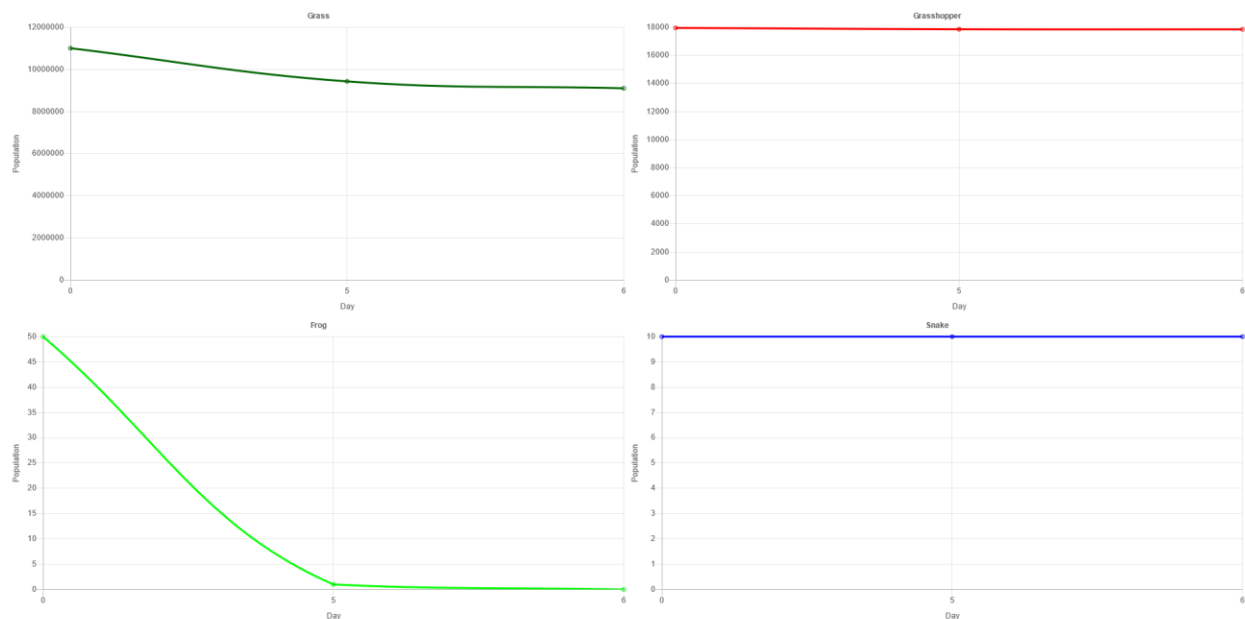
This section lists the ratios of animals in a simulation and compares the results to other configurations in an attempt to find a ratio which leads to a stable ecosystem.

### **3.1. Results from Simulation**

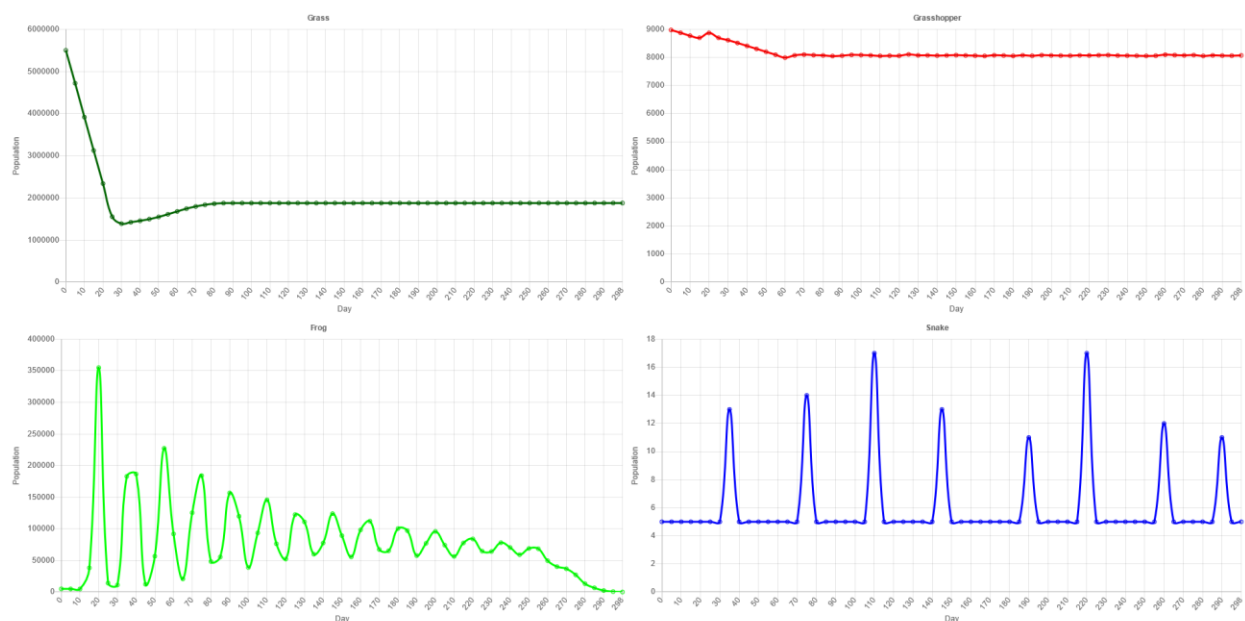
This section outlines the results of simulations with different ratios of animals in the ecosystem and shows the number of days which the ecosystem lasted before one of the species went extinct. All other parameters discussed in the previous section are the same. The image included in each section shows the visualization of the simulation after the extinction.

#### **3.1.1. Configuration 1**

Using the ratios calculated in section 3.3, the following simulation was run with 10 snakes, 5 frogs per snake, 359 grasshoppers per frog, and 613 blades of grass per grasshopper. Therefore, the starting counts were 10 grasshoppers, 50 frogs, 17,950 grasshoppers, and 11,003,350 blades of grass.

**Day: 6 - Christopher Palazzolo 0721414 - Ecosystem Simulation - COIS 4470 Modelling & Simulation - Trent University**

After only 6 days, the entire frog population died out and the simulation ended. This is because the frogs were unable to reach their first procreation cycle to average out with a stable population. To give the system a fair attempt, the initial number of frogs was multiplied by 200 (the average number of offspring per frog during each cycle) to start the simulation with the proper number of adults. Therefore, the starting counts were 5 snakes, 5000 frogs, 8975 grasshoppers, and 5,501,675 blades of grass.

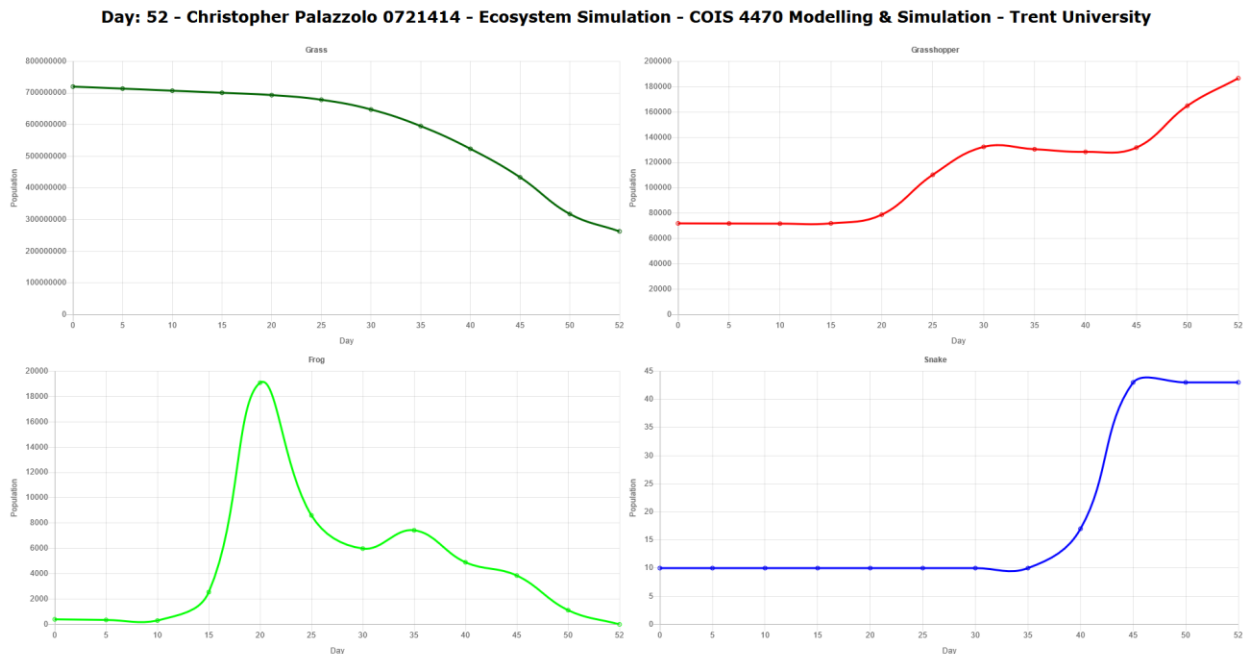
**Day: 298 - Christopher Palazzolo 0721414 - Ecosystem Simulation - COIS 4470 Modelling & Simulation - Trent University**



This simulation ran for 298 days, at which point the frog population died out. Interestingly, all of the charts level out over time and become very consistent except for the frogs whose population fluctuates wildly and then around day 260, begins to drop.

### 3.1.2. Configuration 2

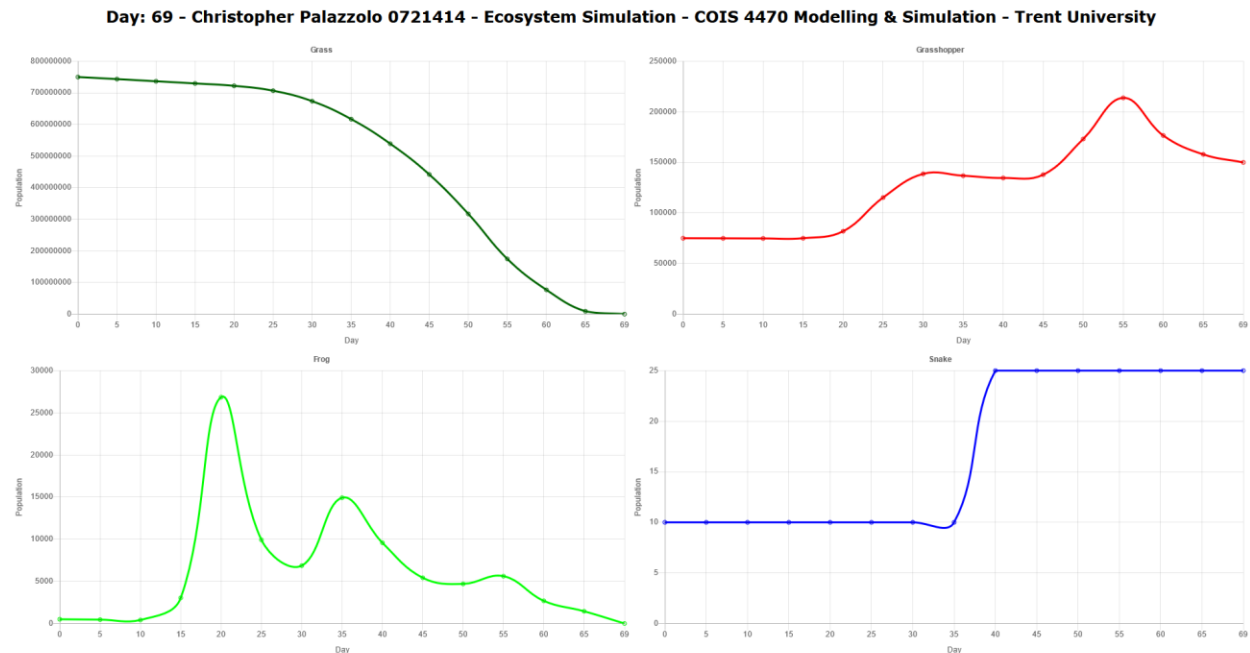
The following simulation was run with 10 snakes, 40 frogs per snake, 180 grasshoppers per frog, and 10,000 blades of grass per grasshopper. Therefore, the starting counts were 10 snakes, 400 frogs, 72,000 grasshoppers, and 720,000,000 blades of grass.



After 52 simulated days, the frog population went extinct. Given that as day 52 was approaching, the grasshopper population was at an all-time high, it is reasonable to say the cause of the frog extinction was because of too many predators and not due to insufficient prey.

### 3.1.3. Configuration 3

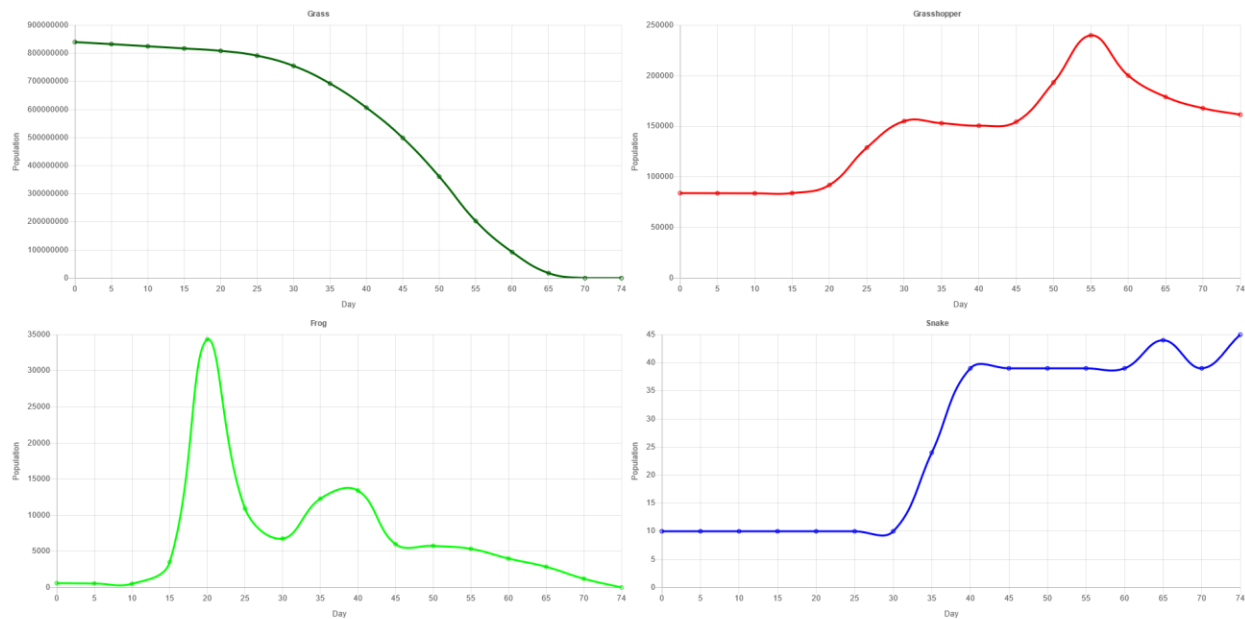
The following simulation was run with 10 snakes, 50 frogs per snake, 150 grasshoppers per frog, and 10,000 blades of grass per grasshopper. Therefore, the starting counts were 10 snakes, 500 frogs, 75,000 grasshoppers, and 750,000,000 blades of grass.



After 69 simulated days, both the frogs went extinct, and the environment ran out of grass. While the grass can regrow despite having no fully grown blades left, the population can recover from the blades which have not fully regrown yet. The same can not be said about the frog population, so the simulation terminated. From these results, it seems that the frog population was unable to limit the grasshopper population in order to ensure enough grass is available to sustain the grasshoppers. Like with the first simulation, the frogs went extinct despite having more than enough food, so it is once again reasonable to say the cause of the frog extinction was because of too many predators.

### 3.1.4. Configuration 4

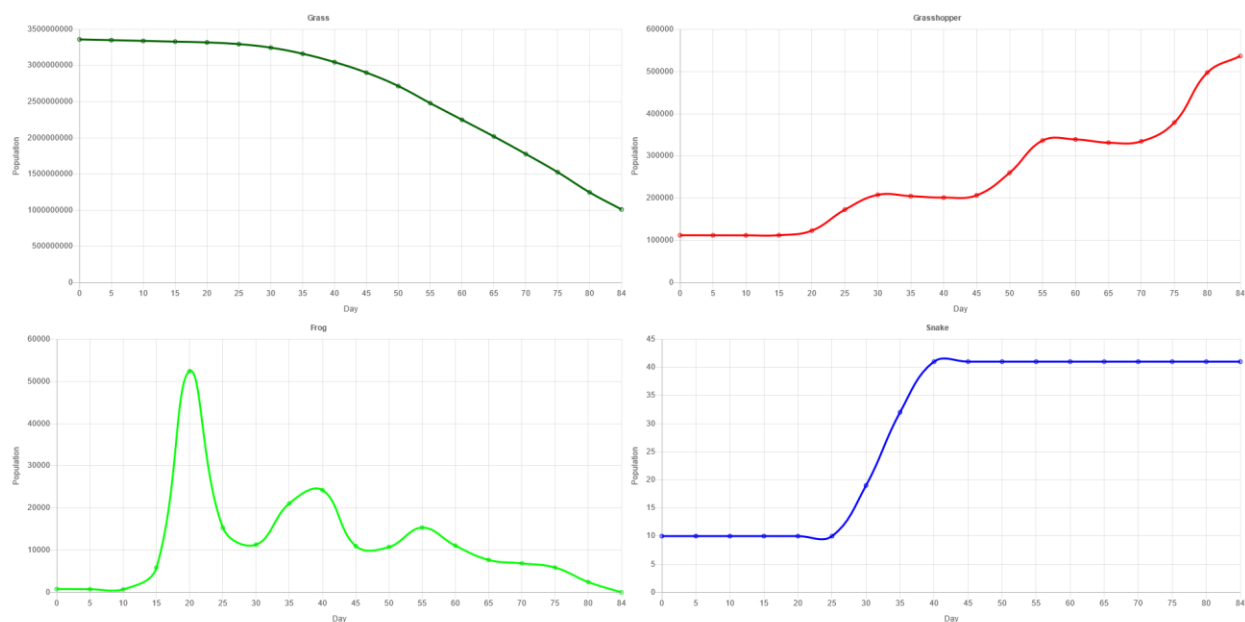
The following simulation was run with 10 snakes, 60 frogs per snake, 140 grasshoppers per frog, and 10,000 blades of grass per grasshopper. Therefore, the starting counts were 10 snakes, 600 frogs, 84,000 grasshoppers, and 840,000,000 blades of grass.

**Day: 74 - Christopher Palazzolo 0721414 - Ecosystem Simulation - COIS 4470 Modelling & Simulation - Trent University**

Like in the previous experiment, after 74 simulated days both the frog and grass population died out. This indicates the ratio of frogs to snakes, and grass to grasshoppers needs to be increased.

### 3.1.5. Configuration 5

The following simulation was run with 10 snakes, 80 frogs per snake, 140 grasshoppers per frog, and 30,000 blades of grass per grasshopper. Therefore, the starting counts were 10 grasshoppers, 800 frogs, 112,000 grasshoppers, and 3,360,000,000 blades of grass.

**Day: 84 - Christopher Palazzolo 0721414 - Ecosystem Simulation - COIS 4470 Modelling & Simulation - Trent University**

After 84 simulated days, the frog population died out. With each configuration, the length of the simulation increases which indicates that the ratios selected are getting better but are not quite at the ideal values to create a stable ecosystem.

### 3.2. Comparison and Discussion

Many simulations using a wide range of ratios had been run and each and every one of them eventually failed due to at least one species going extinct. Interestingly, in most cases it was the frogs who died out. After a close inspection of the graphs and data, it seems that after some time, the frogs get eaten by the snakes at a faster rate than they can reproduce which causes their population to die out. The grasshopper and grass populations almost always fluctuate and then stabilize. If the snake population is able to make it to its first gestation period, that population always seems to stabilize too. It is the frog population which is almost always volatile despite the ratios that start with. This is likely due to the random nature of the simulation. If every random number generator returned its mean value each time, then the second simulation from section 3.1.1 would have been stable. Unfortunately, the generators did not. In that simulation, the generators would have generated ideal numbers but at some point, they would have stopped producing ideal numbers which leads to the decline of the frog population.

Because the simulation is so random, the expected result can be calculated ahead of time, but this result is only an average and can be wildly different than the result of a simulation. The simulation would need to be run thousands of times in order to approach the expected result. Unfortunately, this is not feasible because the simulation is very slow.

### 3.3. Statistics Analysis

The average number of animals required to sustain their predators can be calculated. It should be noted that running these numbers through the simulation may not cause a stable ecosystem because there is so much volatility and variance with the simulation. As was previously stated, a simulation with an identical configuration can have wildly different results. Calculating the ratios also assumes all children are able to survive to adulthood. If a predator eats a parent with children in its care, then the children will be unable to feed themselves and will most likely starve. The calculation process does not account for this, which is why there may be so much variance between what is computed and what is simulated.

The first step is to calculate the calorie requirements of the species per day. The mass of each animal is the same in the first generation, so this step is easy. Note that due to the evolutionary nature of the simulation, subsequent generations will not have this constant mass. In addition to the mass, a multiplier is also needed to use the resting energy requirement into a daily energy requirement. The simulation uses a uniformly distributed random number between 2 and 5. The expected value of a uniform distribution is simply the mean of the upper and lower bound, so the expected multiplier can therefore be calculated as 3.5. From there, the expected daily calories can be calculated as follows.

$$\text{Daily Calories} = 3.5 \times 70(\text{weight})^{\frac{3}{4}} = 245(\text{weight})^{\frac{3}{4}}$$

Next, the number of calories gained from eating its prey is needed. The calories gained per kilogram for the prey is constant for all animals of a species. As with the predator, the mass of the prey is constant in the first generation but evolves in subsequent generations. The following formula can be used. If the prey in question is grass, then the algorithm would forgo the formula and use the 0.01815 calories per blade calculated in section 2.3.

$$\text{Calorie Gain} = \text{Calories Per Kg} \times \text{weight}$$

Finally, the number of preys which must be consumed to fill the daily calorie requirement can be calculated with the following formula.

$$\text{Prey Per Day} = \frac{\text{Daily Calories}}{\text{Calorie Gain}} = \frac{245(\text{Predator weight})^{\frac{3}{4}}}{\text{Calories Per Kg} \times \text{Prey weight}}$$

This alone is not sufficient to create a sustainable ecosystem. This formula alone implies that only 27 grasshoppers are required to sustain a frog. This simply calculates the number of grasshoppers which the frog will consume per day. To create a sustainable ecosystem, the number of grasshoppers needs to grow at a rate of 27 adults per day to sustain the average frog. The 27 will be eaten and the rest will need to produce the 27 which will be consumed on the next day. This is where calculation becomes trickier. Assuming every alpha animal procreates as soon as its children leave its care, the average children raised per day can be calculated as follows.

$$\text{Children Per Day Per Animal} = \frac{\mu_{\text{children per procreation}}}{\max(\text{time to adulthood}, \mu_{\text{gestation}})}$$

The time it takes a child to grow to adulthood is constant. Both the children per procreation and gestation periods are normally distributed random numbers, so the expected value can be used for the purposes of calculation. The maximum is used in the denominator because both the children need to grow to adulthood, and the gestation period needs to pass. To find the average number of preys to support a single predator, a number is needed to multiply Children Per Day Per Animal so that it equals Prey Per Day. This can be rearranged to isolate that multiplier to get the final formula.

$$\text{Prey Needed} = \frac{\text{Prey Per Day}}{\text{Children Per Day Per Animal}} = \frac{\frac{245(\text{Predator weight})^{\frac{3}{4}}}{\text{Calories Per Kg} \times \text{Prey weight}}}{\frac{\mu_{\text{children per procreation}}}{\max(\text{time to adulthood}, \mu_{\text{gestation}})}}$$

From this model, the ideal ratios can be calculated and can be found in the following table. All numbers have been rounded up to the nearest integer. Note that because the probability of a new grasshopper being an alpha is the same as it being a beta, both of which have a different mass, the average of both masses will be used in the calculation, which is 0.00021.

Animal	Grass	Grasshopper	Frog	Snake
Amount to sustain one predator	613	359	5	N/A

#### **4. Conclusions and Recommendations**

No ratios could be found to produce a stable ecosystem. Unfortunately, simulations with good ratios take a considerable amount of time to be run. If optimizations could be made to the simulation to run faster, algorithms like Canonical ES could be used to identify the ideal ratios through evolution learning and eventually converge on the perfect set of ratios to run a long simulation. The problem with Canonical ES is that at least 10 simulations would need to be run per generation, and many generations would need to pass before the algorithm begins to find good ratios. Even if run in parallel, each simulation typically takes a few hours to complete, so it could take around a week of uninterrupted running to find the ratio. Even once found, the perfect ratio could be thwarted by being unlucky with the random number generators.

The simulation is a slow process. It generally takes a few hours to simulate 50 days. Optimizations in both the algorithm and implementation will allow for more species, and the possible simulation of a complex interconnected food web where each predator can eat more than one species. These food web simulations will allow for models which more closely reflect the real world. While the current algorithm does allow for multiple food sources of a species, it would only consider the second species as a food option if it was unable to eat anything of the first species. This is also the reason why the hawk was excluded from the simulation. The ecosystem would have required a large number of snakes to sustain 10 hawks. The large number of snakes would require an even larger number of frogs, which propagates down to the individual blades of grass.

The simulation can only be as accurate as the parameters and algorithms used within it. In the case of this simulation, many assumptions and inaccuracies were introduced due to a lack of information available. As more information about calorie requirements of animals becomes available, the simulation can be modified to reflect that and more closely model the real world.

## 5. References

- A Grasshopper's Life Cycle*. (2020, November 19). Pets on Mom.com.  
<https://animals.mom.com/grasshoppers-life-cycle-6298.html>
- Boiga irregularis (Brown catsnake, Brown Tree Snake)*. (n.d.). Animal Diversity Web.  
[https://animaldiversity.org/accounts/Boiga\\_irregularis/](https://animaldiversity.org/accounts/Boiga_irregularis/)
- Chrabaszcz, P., Loshchilov, I., & Hutter, F. (2018). Back to Basics: Benchmarking Canonical Evolution Strategies for Playing Atari. *arXiv (Cornell University)*.  
<http://export.arxiv.org/pdf/1802.08842>
- Common frog* / *The Wildlife Trusts*. (n.d.). <https://www.wildlifetrusts.org/wildlife-explorer/amphibians/common-frog>
- Croft, A. (2023). Frog Lifespan: How Long Do Frogs Live? *AZ Animals*. <https://a-z-animals.com/blog/how-long-do-frogs-live/>
- Dog and Cat Calorie Calculator* / *OSU Veterinary Medical Center*. (n.d.).  
<https://vet.osu.edu/vmc/companion/our-services/nutrition-support-service/basic-calorie-calculator>
- FoodData Central*. (n.d.). <https://fdc.nal.usda.gov/fdc-app.html#/food-details/168148/nutrients>
- Frogs Facts, Types, Lifespan, Classification, Habitat, Pictures*. (2020, March 28). Animal Spot.  
<https://www.animalspot.net/frogs>
- Girl, L. E. D. F. (2021, July 21). How to Feed Grass to Penned Chickens. *Fresh Eggs Daily® with Lisa Steele*. <https://www.fresheggdaily.blog/2012/08/grass-free-nutritious-food-source.html>
- Grasshopper lifecycle*. (n.d.). <https://www.dudleyzoo.org.uk/grasshopper-lifecycle/>
- Here Be Answers. (2022). Average Speed for a Slithering Snake. *Here Be Answers / Helping You Find the Answers You Need*. <https://herebeanswers.com/average-speed-for-a-slithering-snake.html>
- Herpetologist, D. M. (2022). How Fast Are Frogs? *Toads N' Frogs*.  
<https://toadsnfrogs.com/how-fast-are-frogs/>
- How many calories are in Grasshopper*. (n.d.).  
[https://www.fitday.com/webfit/nutrition/All\\_Foods/Beverages/Grasshopper.html](https://www.fitday.com/webfit/nutrition/All_Foods/Beverages/Grasshopper.html)
- Mukherjee, S. (2023, February 17). *Food Chain – Definition, Parts, Types, and Examples*. Science Facts. <https://www.sciencefacts.net/food-chain.html>
- Mymove. (2022). How Long Does it Take for Grass to Grow? *MYMOVE*.  
<https://www.mymove.com/lawn-garden/how-long-for-grass-to-grow/>

- Natural Heritage Program and Montana Fish, Wildlife & Parks. (2023, April 13). *MTNHP*.  
<https://fieldguide.mt.gov/speciesDetail.aspx?elcode=IIORT60010>
- NATURE, PBS. (2021, May 6). *Frog Fact Sheet | Blog | Nature | PBS*. Nature.  
<https://www.pbs.org/wnet/nature/blog/frog-fact-sheet/>
- Nmr. (2016, January 1). How to Breed Snakes. *Animal City*.  
<https://animalcityinc.com/blog/37897/how-to-breed-snakes>
- Piercy, K. (2022, August 3). How Far and High can Grasshoppers Jump? (Who Jumps Best?).  
*Meadowia*. <https://meadowia.com/how-far-high-can-grasshoppers-jump/>
- Snake Life Cycle : Lifespan & Reproduction | Critter Control*. (n.d.).  
<https://www.crittercontrol.com/wildlife/snakes/snake-life-cycle>
- Spawn & tadpoles FAQs*. (n.d.). Amphibian and Reptile Conservation. <https://www.arc-trust.org/spawn-tadpoles-faqs>
- targetstudy.com. (n.d.). *Grasshopper - life cycle of grasshopper, behaviour, diet, types, scientific classification, physical features, reproduction of Grasshopper*.  
<https://targetstudy.com/nature/animals/grasshopper.html>
- The Animal Facts. (2023, March 20). *Brown Tree Snake - The Animal Facts - Appearance, Diet, Habitat, Lifespan*. <https://www.theanimalfacts.com/reptiles/brown-tree-snake/>
- The Nutrition of Snake / Nutrition / Healthy Eating*. (n.d.). <https://www.fitday.com/fitness-articles/nutrition/healthy-eating/the-nutrition-of-snake.html>
- Tree frog metamorphosis - King County*. (n.d.).  
<https://kingcounty.gov/services/environment/animals-and-plants/biodiversity/defining-biodiversity/species-of-interest/herps/treefrog.aspx>
- United States Department of Agriculture. (1994). Whitewiskered Grasshopper. *Wyoming Agricultural Experiment Station Bulletin*.  
<https://www.ars.usda.gov/ARSUserFiles/30320505/grasshopper/Extras/PDFs/Species%20Fact%20Sheets/Whitewhs.pdf>