Authors: Mohamed Kelai, Oussama Zaidi, Yasser El-jarida
Emails: mohamed.kelai@um6p.ma, oussama.zaidi@um6p.ma, yasser.eljarida@um6p.ma
Supervised by: Pr. Julien Randon Furling
Date: 6-12-2024

1 Introduction

As part of our course on Mathematical Models of Complexity and Simulation, we developed an agent-based model to explore predator-prey dynamics. Agent-based modeling is a powerful tool to simulate and understand complex systems by representing individual agents and their interactions. In this project, we aimed to build and analyze a predator-prey simulation, focusing on how different initial conditions influence ecosystem outcomes such as predator extinction, prey extinction, or coexistence.

The primary challenges addressed in this project include:

- **Understanding Population Dynamics:** How initial population ratios and densities impact predator-prey survival.
- **Creating Effective Visualizations:** Developing a simulation interface and phase diagrams that clearly communicate results.
- **Improving Model Accuracy:** Iteratively refining the simulation model based on preliminary results.

The following sections describe the development, implementation, and progression of our model, including the iterative improvements made to enhance accuracy and visualization quality.

2 Simulation Development and Progress

The project evolved through three main stages, each building upon the insights gained from previous iterations. We aimed to improve both the simulation's performance and the accuracy of its visual representations.

2.1 Stage 1: Initial Visual Simulation

The first version of the predator-prey simulation was built using Python and with the Pygame library, allowing us to observe population dynamics visually in real time.

- **Initial Setup:** A 20x20 grid was created where prey and predators interacted as agents.
- **Movement and Interaction:** Prey moved randomly, while predators sought out prey. Predators lost energy each step and gained energy when they successfully ate prey.
- **User Interface:** A visual interface with sliders for setting initial prey and predator populations, simulation speed, and buttons for controlling the simulation (setup, run, pause, reset).

The initial visual simulation, as shown in Figure 1, provided a foundational understanding of how populations interacted. However, it was computationally intensive, lacked scalability, and did not easily allow systematic exploration of parameter space.

2.2 Stage 2: Phase Diagram of Initial Populations

In the second stage, we aimed to quantify the population outcomes by creating a phase diagram using initial prey and predator populations as axes.

- **Initial Population Range:** Prey and predator populations were varied from 10 to 100 in steps of 10.
- **Outcome Analysis:** For each combination of initial populations, 100 simulations were conducted, and the most frequent outcome was recorded.
- **Outcomes:** Results were categorized as:

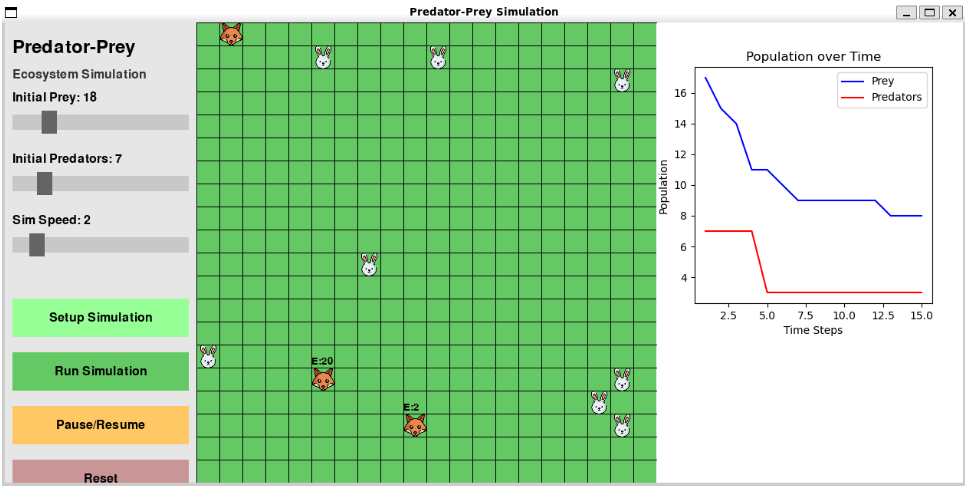

Figure 1: Initial predator-prey visual simulation using Pygame.

- **0:** All Prey Died.
- **1:** All Predators Died.
- **2:** Coexistence.

This approach allowed us to generate a phase diagram that provided insights into how different starting conditions influenced population dynamics (see Figure 3). However, due to the discrete steps in initial populations, the resulting plot appeared fragmented and lacked smooth transitions.

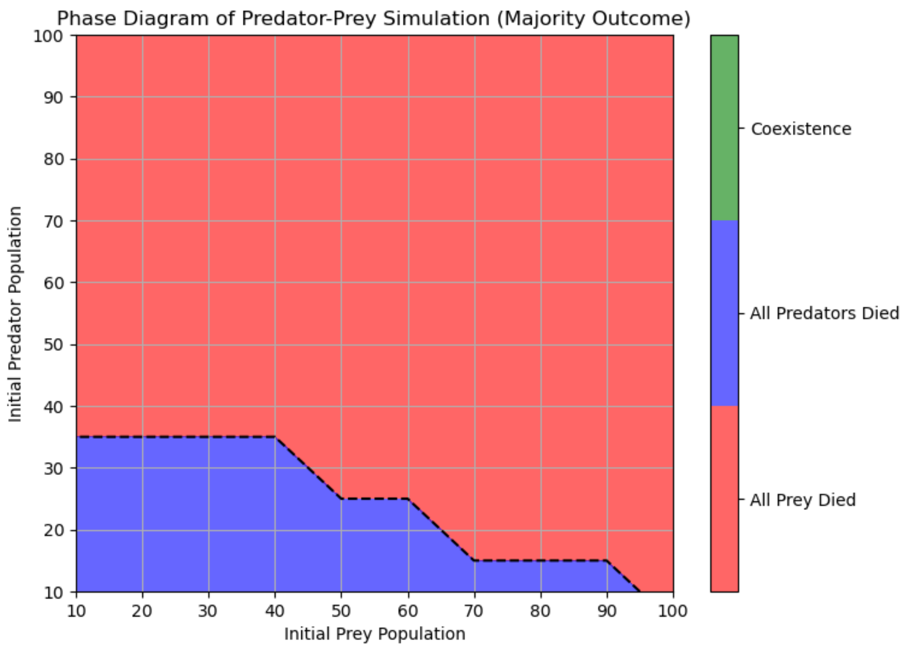

Figure 2: Phase diagram of the simulation outcomes using initial populations as axes.

2.3 Stage 3: Ratio and Density-Based Phase Diagram

In the third stage, we refined our approach by using a more representative metric for population dynamics. Instead of using absolute initial populations, we used:

- **Ratio (x-axis):** The ratio of prey to predators, ranging from 0 to 10, with steps of 0.02.

- **Density (y-axis):** The density of agents (total prey and predators per grid cell), ranging from 0 to 1, with steps of 0.01.

2.4 Running the Simulation

We conducted the simulations in two main stages to ensure feasibility and accuracy of our results.

2.4.1 Smaller-Scale Simulation

Initially, a reduced version of the simulation was run with fewer iterations per initial condition (e.g., 100 simulations) and a coarser grid for ratio and density values. This smaller-scale simulation allowed us to quickly assess the feasibility and behavior of the model under different initial conditions.

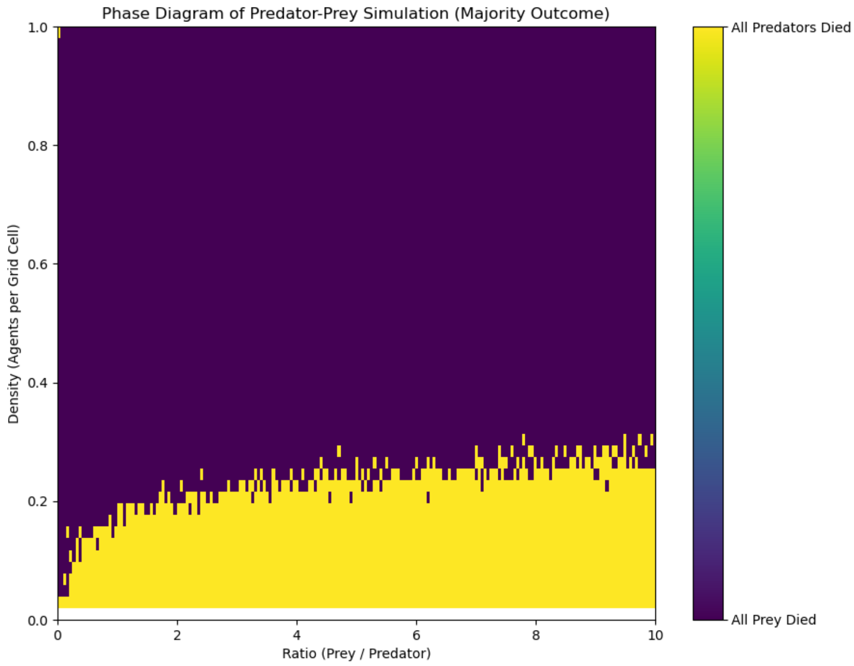

Figure 3: Phase diagram from the smaller-scale simulation using prey/predator ratio and agent density. Boundaries are less defined due to fewer simulations and larger step sizes.

Observations: The smaller-scale diagram provided insight into trends such as predator extinction and prey extinction zones. However, due to the coarser resolution and fewer iterations, the boundaries between outcomes were less defined, prompting the need for a higher-resolution version.

2.4.2 Full-Scale Simulation on the Toubkal Cluster

After assessing the preliminary results, we conducted a full-scale simulation on the Toubkal cluster at UM6P. This involved approximately 24.5 million iterations, utilizing smaller step sizes for both ratio (0.02) and density (0.01) to ensure higher resolution and detailed phase boundaries.

The simulation took about 1 hour and 17 minutes to complete, achieving a rate of approximately 5270 iterations per second, aided by parallel processing on the cluster's resources. The high-resolution results provided well-defined boundaries between different population outcomes, such as prey extinction, predator extinction, and coexistence.

The comparison between Figures 3 and 4 highlights significant improvements in the smoothness and accuracy of the phase boundaries. This iterative process showed that increasing the resolution and computational effort is essential for capturing precise and reliable ecosystem dynamics.

The evolution from absolute population values to ratio and density-based metrics allowed for smoother and more insightful visualizations. The refined phase diagram reveals:

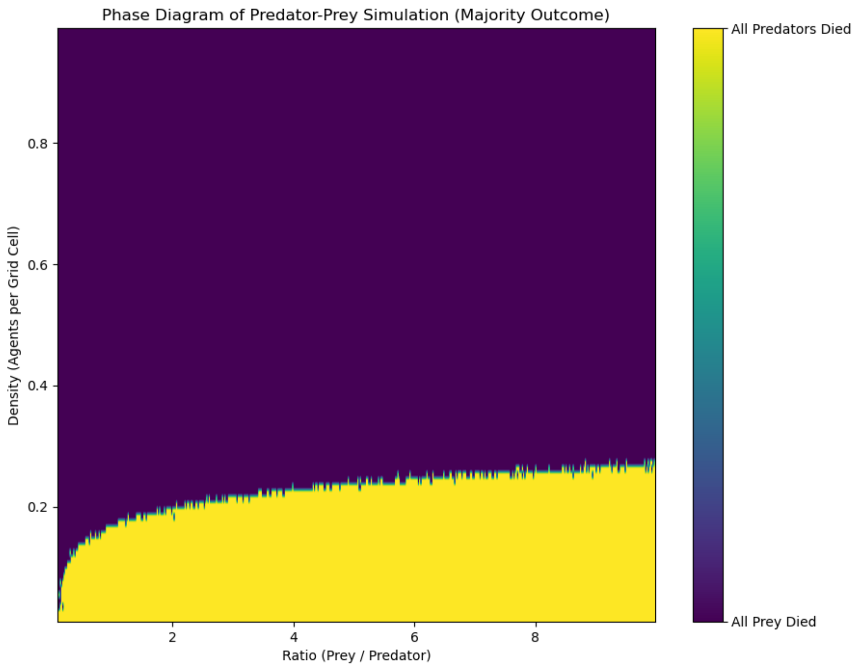

Figure 4: Full-scale phase diagram of predator-prey simulation using prey/predator ratio and agent density as axes. Well-defined transitions between different outcomes are observed.

- **Low Density Region:** At low densities, predators commonly went extinct, highlighting the low likelihood of encounters between predators and prey.
- **Low Ratio Region:** A very low ratio of prey to predators ($r < 1$) leads to more prey extinction, which ultimately resulted in the extinction of predators. However, for this analysis, we focus on which population went extinct first.
- **Critical Transitions:** The boundaries on the phase diagram indicate critical points where the system shifts, especially at higher density values, from predator extinction to prey extinction, demonstrating the sensitivity of predator-prey balance to initial conditions.

2.5 Exploring Prey Reproduction to Achieve Coexistence

As part of our iterative refinement process, we introduced a reproduction factor for prey to assess its impact on the overall dynamics of the ecosystem. The motivation behind this addition was to explore whether prey reproduction could create conditions favorable for a stable coexistence state between predators and prey.

Implementation:

- A reproduction probability (`SHEEP_REPRODUCE`) was assigned to prey. Each prey had a 15% chance of reproducing during every simulation step.
- Reproduction occurred by generating a new prey in a randomly chosen neighboring empty grid cell, if available.
- Predators retained their energy-based survival and predation dynamics, ensuring that prey reproduction directly influenced the availability of food for predators.

Simulation Setup:

- The simulations were conducted under the same conditions as previous iterations, varying initial ratios of prey to predators (0.1 to 10) and agent density (0.01 to 1.0).

- The total number of simulations was increased to approximately 2.45 million, leveraging the Toubkal HPC cluster to ensure high-resolution and reliable results.
- The large-scale simulation took approximately 3 hours and 53 minutes, running at a speed of 175.12 iterations per second.

Observations:

- **Coexistence Zones:** The introduction of prey reproduction expanded the zones of coexistence significantly.
- **Predator Survival at Low Densities:** Even at low densities, predators were able to persist for longer durations due to periodic surges in prey populations caused by reproduction. However, this effect diminished as density approached its lower limit ($d < 0.05$).

Insights from Phase Diagram: The inclusion of prey reproduction modified the boundaries on the phase diagram (see Figure 5), particularly in regions with moderate densities and ratios. Coexistence states became more pronounced, indicating the stabilizing influence of prey reproduction on predator-prey dynamics.

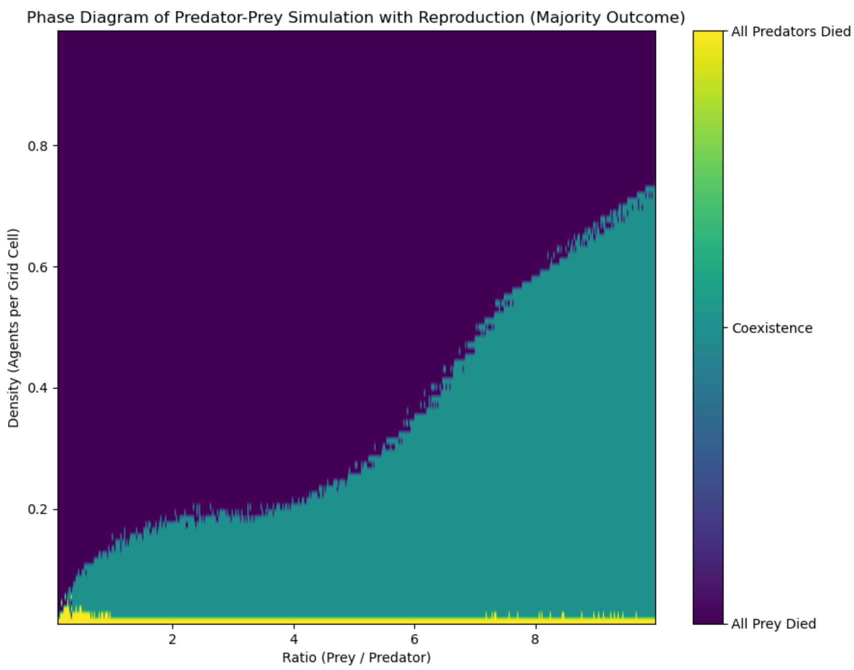

Figure 5: Phase diagram of predator-prey simulation with prey reproduction. Expanded coexistence zones and refined boundaries are observed, based on simulations run on the Toubkal HPC cluster.

2.6 Markov Chain Representation of Predator-Prey Dynamics

To further understand the dynamics of the predator-prey system, we can represent it as a Markov chain, especially without the reproduction factor for simplicity.

In this microscopic Markov chain representation, each agent (predator or prey) is represented by:

- **Nature:** '0' for prey, '1' for predator.
- **State:** '0' for dead, '1' for alive.
- **Position (x, y):** The coordinates on the grid.
- **Energy:** The energy level of the agent.

Each state transition depends on factors such as movement, hunting, and starvation. For movement, the effect on energy depends on the type of agent. For prey, the energy does not change during movement, whereas for predators, the energy decreases by a certain value. The general formula to calculate the probability of being in a certain state S at time t is the following:

$$P(S,t) = \sum_{S'} P(S', t-1) \times P(S \mid S') \tag{1}$$

Where:

- $P(S,t)$: Probability of being in state S at time t.
- S': All possible states at time $t-1$ that can transition to state S.
- $P(S \mid S')$: Transition probability from state S' to state S.

Transition probabilities depend on factors like:

- **Movement:** Probability of moving to a neighboring cell is generally uniform, e.g., $\frac{1}{N}$, where N is the number of available neighbors. The energy change is calculated by multiplying the energy cost by the type of agent: for prey, it is multiplied by 0 (no energy loss), and for predators, it is multiplied by 1 (energy decreases).
- **Hunting:** If a predator and prey end up in the same cell, the probability that the predator hunts the prey is 1.
- **Starvation:** If a predator's energy reaches zero, it will transition to the dead state with probability 1.

This Markov chain representation helps capture individual-level interactions and provides a way to calculate the probability of different outcomes over time. It gives a microscopic view of how agents transition between states, which allows for a more thorough understanding of predator-prey dynamics.

3 Conclusion

In this project, we developed and refined an agent-based predator-prey model to explore ecosystem dynamics under different initial conditions. We used both ratio and density-based metrics and included a prey reproduction mechanism to examine the impact on coexistence states. Our findings highlighted the critical role of initial population ratios, densities, and reproduction dynamics in determining whether coexistence, predator extinction, or prey extinction occurs. Additionally, a Markov chain representation was used to understand the transition dynamics, especially in the absence of reproduction. The use of high-resolution phase diagrams, generated using the Toubkal HPC cluster, provided deeper insights into the complex interplay between these variables and the overall stability of the ecosystem.

References

[1] Source Code, *Github*, `https://github.com/YasserElj/Predator_Prey_Agent_Besed_Model`

[2] Pygame Community, *Pygame Documentation*, `https://www.pygame.org/docs/`

[3] Toubkal Sypercomputer, *Home Page*, `https://toubkal.um6p.ma/`