CHALMERS
UNIVERSITY OF TECHNOLOGY

UNIVERSITY OF GOTHENBURG

Combining Reflexes and Reinforcement Learning in Evolving Ecosystems For Artifical Animals

Master's thesis in Computer science and engineering

Victor Skoglund, Hans Glimmerfors

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Gothenburg, Sweden 2021

Master's thesis 2021

Combining Reflexes and Reinforcement Learning in Evolving Ecosystems For Artifical Animals

Victor Skoglund, Hans Glimmerfors

UNIVERSITY OF
GOTHENBURG

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Computer Science and Engineering
Chalmers University of Technology
University of Gothenburg
Gothenburg, Sweden 2021

Combining Reflexes and Reinforcement Learning in Evolving Ecosystems for Artifical Animals

HANS GLIMMERFORS
VICTOR SKOGLUND

Supervisor: Claes Strannegård, Department of Computer Science and Engineering
Examiner: Torbjörn Lundh, Department of Mathematical Sciences

Master's Thesis 2021
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset in LaTeX
Gothenburg, Sweden 2021

Combining Reflexes and Reinforcement Learning in Evolving Ecosystems for Artifical Animals

Victor Skoglund, Hans Glimmerfors
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg

Abstract

As our world faces an increasing number of threats to its environment, it is becoming more important than ever to find ways to reduce our impact on Earth's ecosystems. Computer science may be able to help contribute to this cause by creating realistic simulations of nature such that scientists can analyze the impact climate change, pollution, hunting, etc. will have on an ecosystem.

In this paper we create a framework, now known as Ecotwin, which models artificial animals. The agents must manage both an energy need and a libido need in order to maximize their reward. The focus of our work lies in investigating the role played by evolution and learning in ecosystems, namely: is the combination of reinforcement learning and evolutionary algorithms needed for survival?

To investigate this, we construct an environment containing lethal food whose ingestion may be stopped by a specific gene. As a gene cannot be learned, we find that asexual reproduction is a less reliable reproductive method than sexual reproduction in dangerous environments. In environments with few threats, we instead find that asexual reproduction can develop a great set of genes through higher birth rates and higher internal competition.

In addition to our main focus on evolution, we also implemented a more realistic spread of plants – a food source for the prey species in predator-prey systems. The more advanced nature of these plants makes them more difficult to balance, however we find that with the correct parameters it remains possible to simulate population dynamics similar to those of stable oscillating three-species Lotka-Volterra equations. With different species of plants it is also possible to show the competition of plants adhering to these population dynamics.

Keywords: computer science, engineering, thesis, reinforcement learning, evolution, agent-based, simulation, animats.

Acknowledgements

We are very grateful for the help our peers Birger Kleve & Pietro Ferrari provided through their work by helping develop the framework used for simulations. We are equally thankful of Tobias Karlsson who provided us with help and insights from his parallel work. We would also like to thank Marcus Hilding Södergren for creating Ecotwin's original framework.
Thank you to our supervisor Claes Strannegård for guiding us in our work and for providing the groundwork for our animat model.

Hans Glimmerfors & Victor Skoglund, Gothenburg, June 2021

Contents

List of Figures

List of Tables

1

Introduction

Human society makes use of a large quantity of plants and animals in order to produce various items and foods. These animals may either be hunted in nature or raised as livestock, and the plants may similarly either be harvested in nature or on farms. Regardless of whether it is an organism or a natural resource which is extracted, the local ecosystem is disturbed and if the interference is too great, the ecosystem risks collapsing. Research shows that climate change will negatively affect animals' health [1], and as the over-exploitation of animals already poses one of the greatest risks of extinction to some species [2], it is all the more important to find solutions to reduce our impact on ecosystems.

Analyzing the safe rates of exploiting species with living creatures is a very time-consuming task and as natural habitats vary immensely, it is also a very complicated task. If we could instead recreate realistic and useful ecosystems in a digital environment, scientists would be able to analyze interactions between species and humans at a much faster pace – without first needing to grow plants and raise animals.

Historically, in order to analyze the stability of ecosystems, scientists have used mathematical models to view how species' populations interact and oscillate [3, 4, 5]. A flaw in this approach is that creatures are only considered on a population level rather than on an individual level.

With the advances in computer science over the last decades, both in machine-learning theory and in computing power, it is now possible to simulate predator-prey systems through the use of agent-based machine-learning [6]. Recent research has made use of this technique and concluded that training the predator and prey simultaneously increases the stability of the ecosystem [7, 8]. However, their implementations of reproduction have not related realistically to nature as sexual intercourse has been represented by algorithms unrelated to the agents' actions.

In this paper, we make the distinction between genotype and phenotype. Only the genotype is passed on from generation to generation whereas the phenotype is developed during an individual's lifetime. With this distinction, we aim to adhere closer to biology while also striving for stable population dynamics and intelligent behaviour based on a combination of inherited reflexes and reinforcement learning.

1.1 Research Questions

Our work aims to capitalize on agent-based simulations' presence of individuals *and* populations. This combination makes it possible to both investigate how population-wide changes affect its individuals, and to see how changes in an individual can change its population. In this paper, we will focus on answering the following research questions:

- Is there a purpose to death?
  - Does death cause faster evolution?
  - Is death particularly important in changing environments?
- Is sexual reproduction more advantageous for survival in some environments and asexual reproduction in others?
- Does a combination of learning and evolution make survival in dangerous* environment more likely?

*We define a dangerous environment as an environment where the following characteristics can be seen:

1. Suppose the edibility of food changes over an individual's lifetime. Any fixed policy is likely to die due to not knowing when the food is edible.
2. Suppose some food is edible and some food is deadly. Then any individual with no prior knowledge of the food's edibility is likely to die due to accidentally eating the deadly food.

In order to thrive in a dangerous environment, an individual should both be able to learn *when* to eat food and *what* food to eat. We will thus also study whether the combination of evolutionary algorithms and reinforcement-learning is able to overcome situations where the presence of only one aspect falls short.

2
Background

Our project is part of a larger work on animats (Ecotwin) as part of a research group at Chalmers University and Dynamic Topologies AB. The research group's common goal is to further the understanding of ecosystem simulations on individual- and population-wide levels, and to create an open-source platform for simulating multiagent-based simulations of ecosystems.

This paper starts with a section briefly describing the areas of reinforcement learning and evolutionary algorithms, as well as our animat model. Later sections go into more theoretical details, implementation details, analysis, and discussion regarding the simulations using the animat model.

2.1 Reinforcement Learning

2.1.1 What Is It?

Reinforcement learning (RL) is a subclass of artificial intelligence (AI) and its goal is to allow agents to perform tasks which will maximize cumulative rewards. These rewards will be given to an agent performing an action based on the problem at hand. For example, in Blackjack a player will lose if their hand adds up to higher than 21 or if their hand is lower than the dealer's hand. If the agent is given a reward only when winning against the dealer, then the agent should learn to stop drawing cards if the risk of going past 21 is too high - unless the dealer already has a higher hand.

RL resembles human behaviour in that an agent will be equipped with a neural network (NN) which will determine the best course of action, however the agent will sometimes choose to try another action in order to see what might happen. This could be likened to a football player shooting the ball with their left foot instead of using their better right foot - but if the goalkeeper is not ready for this, it might prove to give a better result! This is one of the key principles of RL: exploration; although the output from the NN suggests one action is better, the agent may still explore the possibilities in the hopes of discovering an unknown solution. The other key concept of RL is exploitation, where an agent makes use of what they have learned in order to take the best action as suggested by the NN.

RL can be a very powerful tool when looking for optimal behaviour in predictable

environments, however not all situations are predictable in nature. Suppose that an agent has lived all its life by feeding off red berries (strawberries, raspberries, red-currant, etc), and then it discovers a new red berry which happens to be poisonous. The previous experiences will tell the agent to eat the berry. Another example is food which is poisonous when unripe, but safe when ripe. An agent which has eaten only ripe fruit will not know of its side-effects and may die due to its previous experiences telling it that the fruit is good, despite it being unripe and deadly! To fight such situations, reflexes and instinct could prove useful – both of which can be provided by evolution.

2.1.2 How It Works

The decision-making which occurs in reinforcement learning works by connecting stimuli to actions. Depending on the outcome of the action, the agent is either rewarded or punished. This allows the agents to learn through trial-and-error by exploring its environment and experiencing which action is the best given a certain situation [9].

The agent receives a sensory input (State) from its environment. Based on the input, the agent then chooses an action and sends it back to the environment. As a reaction to the agent's action, the environment then provides the agent with a reward that can be either positive or negative along with a set of new observations [10]. This cycle is described in Figure 2.1.

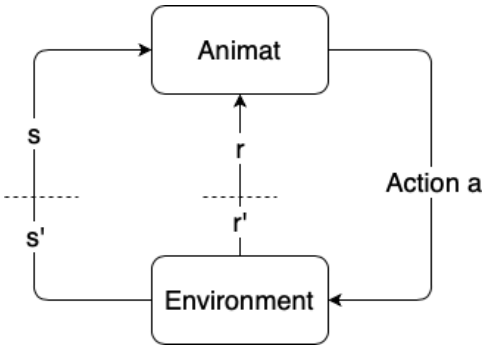


Figure 2.1: The cognition cycle of an animat.

The action which the agent picks given its current state s is sampled from the probability of the policy function Equation 2.1.

$$\pi(a|s) \tag{2.1}$$

Since Equation 2.1 is a probability function determining the probability of each action being picked the following equation must hold for all states:

$$\sum_{a \in A} \pi(a|s) = 1, \forall s \in S \tag{2.2}$$

The return can be calculated in every timestep t by calculating the sum of future rewards multiplied with a discount parameter γ given in Equation 2.3. γ determines

how heavily weighted future rewards should be in comparison to the current reward. An agent caring as much about future rewards as the reward it is receiving in its current state s_t would correspond to $\gamma = 1$ and an agent only caring about the currently experienced reward in state s_t would correspond to $\gamma = 0$.

$$G_t = \sum_{k=0}^{T} \gamma^k R_{t+k+1} \tag{2.3}$$

As stated earlier the goal of the agent is to maximize its cumulative reward, which corresponds to the expected return $\mathbb{E}[G_t]$. In order to calculate the expected return in state s, given that the agent follows policy π, we calculate the state-value function $v_\pi(s)$ as in Equation 2.4.

$$v_\pi(s) = \mathbb{E}_\pi[G_t|S_t = s], \forall s \in S \tag{2.4}$$

Since the agent samples an action from the policy function in Equation 2.1, the value of a state s for an agent following a policy π given an action a is of interest. With this function called the action-value function $q_\pi(s, a)$ it is possible to calculate the best possible action given the agent's current state and policy:

$$q_\pi(s, a) = \mathbb{E}_\pi[G_t|S_t = s, A_t = a], \forall s \in S, \forall a \in A \tag{2.5}$$

Finding the optimal policy π^* yielding the maximum cumulative reward now corresponds to maximizing the action-value function $q_*(s, a) = \max_\pi q_{\pi^*}(s, a)$.

In the scope of this thesis we will only consider the model-free, on-policy algorithm Proximal Policy Optimization (PPO).

2.1.3 Proximal Policy Optimization

PPO uses a policy gradient method in order to update its policy's parameters θ. Policy gradient methods update the parameters of the policy with gradient descent, changing the parameters in the direction of some gradient estimator $\nabla J(\theta)$ [9].

$$\theta_{t+1} = \theta_t + \alpha \nabla J(\theta) \tag{2.6}$$

PPO is based on an actor-critic algorithm which means that it learns to approximate both the value function and the policy. This is done by calculating a loss based on some advantage function A as in Equation 2.7 which determines the advantage of performing action a in the state s_t [11].

$$A_t(a) = q_\pi(s_t, a) - v_\pi(s_t) \tag{2.7}$$

This means that a loss can be calculated based on the current policy and some new policy π' performing action a in state s_t. The policy can then be updated in each iteration as in Figure 2.2.

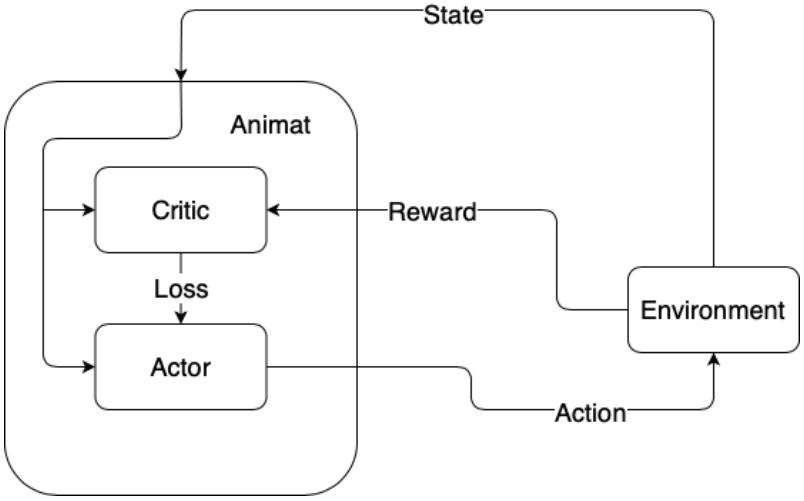


Figure 2.2: Figure 2.1 updated to better represent the actor-critic method used in PPO.

An issue with updating the parameters θ by updating the gradient based on this loss is the risk of taking too big steps and thus changing the policy drastically. PPO solves this by clipping the objective function of TRPO (Trust Region Policy Approximation). Firstly, we denote the probability ratio $r_t(\theta)$ in Equation 2.8.

$$r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)} \tag{2.8}$$

The probability ratio $r_t(\theta)$ is then used to calculate the clipped loss function in Equation 2.9. The left side of the minimum function is the objective function proposed in TRPO, the right side uses the hyper parameter ϵ to ensure that $r_t(\theta)$ stays within the range $[1-\epsilon, 1+\epsilon]$

$$L^{CLIP}(\theta) = \mathbb{E}_t[min(r_t(\theta)A_t, clip(r_t(\theta), 1-\epsilon, 1+\epsilon)A_t)] \tag{2.9}$$

PPO improves on the loss function further by incorporating the square loss of the value function $L_t^{VF}(\theta)$ as both the value function and policy might be used in the neural network architecture. Additionally they add an entropy bonus S to make sure that there is enough exploration for the policy to not too easily get stuck in some local minimum. Adding these three components results in Equation 2.10.

$$L^{CLIP+VF+S}(\theta) = \mathbb{E}_t[L_t^{CLIP}(\theta) - c_1 L_t^{VF}(\theta) + c_2 S[\pi_\theta](s_t)] \tag{2.10}$$

The hyperparameters c_1 is the value function coefficient and c_2 is the entropy function coefficient.
Lastly, PPO uses a version of generalized advantage estimation that is limited to T timesteps to calculate the advantage function A_t as shown in Equation 2.11.

$$A_t = \delta_t + (\gamma\lambda)\delta_{t+1} + ... + (\gamma\lambda)^{T-t+1}\delta_{T-1} \tag{2.11}$$

Where δ_t is defined as in Equation 2.12

$$\delta_t = r_t + \gamma v(s_{t+1}) - v(s_t) \tag{2.12}$$

2.1.4 Unity ML Agents

Unity is a game-engine designed to facilitate game- and software development. The engine gives developers access to tools for easy graphics rendering which allows us to visually show what happens during every timestep of a simulation. Furthermore, Unity also comes with access to numerous packages – libraries with functioning components, that is. ML Agents is one such package, designed to make reinforcement learning available to developers who do not have the theoretical background which may otherwise be needed. We choose to make use of Unity and ML Agents to save time on developing own frameworks for reinforcement learning and graphics rendering.

One great advantage of using ML Agents, and one of the reasons for why we do, is that ML Agents has built-in support for PPO. We are thus able to use a reinforcement learning algorithm designed to avoid getting stuck in local optima, without needing to implement it ourselves. There are a few reinforcement learning hyperparameters which are adjustable in the ML Agents configuration.

- `buffer_size`: Defines how many experiences should be collected before updating the policy.
- `batch_size`: The number of experiences used for one update of the gradient descent.
- `num_epoch`: The number of times the experience buffer is passed to the gradient descent.
- `learning_rate`: The learning rate α controlling how fast the policy's parameters change as in Equation 2.6.
- `learning_rate_schedule`: Defines whether there should be a constant learning rate or if the learning rate would drop linearly.
- `beta`: The entropy constant c_2 in Equation 2.10.
- `epsilon`: The clipping constant ϵ used for limiting the probability ratio in Equation 2.9
- `lambd`: Hyper parameter λ is used to control the weight of the current value estimate δ_t when calculating a new value estimate δ_{t+1} as in Equation 2.12.

2.2 Evolutionary Algorithms

Based on Darwin's theory of natural selection [12], evolutionary algorithms (EAs) solve and optimize problems by breaking down actions into genes. One generation will have a combination of genes which may or may not be a good solution to the given problem. Next, the better a solution is, the higher is the chance that the solution will be picked to create the next generation's solution. As the solutions are made up of genes, two solutions (or more) can be combined as is done with DNA in nature. If there were *only* the combination of genes, then eventually all solutions would converge to the point where all their genes are identical. Therefore, as in nature, mutation is used to alter a small number of genes randomly.

The idea of EAs is that features of good solutions will be passed on to following generations and that bad features will be lost. By combining EAs with RL, we can hope to recreate natural behaviour: reflexes and instincts inherited from parents (EAs), as well as learning over a lifetime (RL). In the example of deadly unripened fruit from subsection 2.1.1, one way that an agent may have adapted to prevent dying would be to regurgitate its food when it tastes something it did not expect. If the agent eats an unripe fruit which does not taste as the expected taste of the ripe fruit, then by regurgitating the food, the agent will survive and, if lucky, it may even be able to distinguish between unripe and ripe fruit in the future.

2.3 Animat

In order to distinguish biological animals and artificial animals, we make us of the contraction of animal and material: *animat* [13]. In our context, animats will be digital representations of different animals depending on the type of ecosystem we wish to simulate. For the purposes of this paper, we make the assumption that we are dealing with a carnivore predator and a herbivore prey.

2.3.1 Nervous System

Many bodily functions do not need to be controlled consciously [14]. For instance, we do not decide when to digest eaten food or check if we are hungry. Instead, the stomach is an example of an organ which is part of the autonomic nervous system. These organs will operate autonomously and signal the central nervous system which will in turn aid the organism in taking its decision on whether to rest, eat, hunt, flee, etc.
For the purpose of full ecosystem simulations, it would be too resource-demanding to create animats with nervous systems made out of autonomous organs. Therfore, our research team has made some assumptions to find a compromise between nature and processing power, where an animat's nervous system is built as follows:

- Policy Network:
  The animat's brain, the component which decides what the animat will do. Implemented with reinforcement learning.
- Reward Network:
  The animat's amygdala, the part of the brain which regulates the animat's feelings. Implemented with mathematical functions.
- Reflex Network:
  A collection of the animat's reflexes, all actions which are to be forced or hindered in a certain situation. Implemented with evolutionary algorithms.
- Prediction Network:
  The animat's prefrontal cortex, the part of the brain which is able to plan what will happen as a consequence of a certain action. Not yet implemented.

We refer to all these components as networks to reflect the nervous systems' intended implementation in computer science: artifical neural networks. However, as stated

in the networks' descriptions, not all the components are currently neural networks in our model.

2.3.2 Genotype & Phenotype

A *genotype* refers to the ensemble of a gamete's or zygote's [15]. It is thus a collection of all characteristics passed down from a parent to its offspring. Genes affect an individual's life in various ways, for example, genes may increase the risk of certain diseases whereas some genes may promote faster muscle growth or a higher metabolism. Genes never change during an individual's lifetime, and so if an individual is born with poor genes, natural selection will favor the genes of other individuals (through better hunting skills, choices of mates, etc.).

A *phenotype* refers to all the observable characteristics of an individual. An individual's phenotype *does* change over its lifetime but are in some ways largely defined by the individual's genotype. For example, an individual may have the genes for growing tall, but factors such as malnutrition and injury may hinder the individual from growing fully. Phenotype is thus a combination of the individual's genotype and the individual's interactions with its environment.

2.4 Predator-Prey Systems

Predator-prey systems refer to ecosystem simulations, either analytical models or agent-based computer simulations, where there is an interaction between simulated prey and predators which hunt the prey. Multiple predator-prey models exist, however the most well-known is known as the Lotka-Volterra equations.

2.4.1 Lotka-Volterra

Today referred to as Lotka-Volterra, this ecological model has been around for a long time, created independently by two mathematicians: Lotka (1925) [3] and Volterra (1926) [4] and is based on the interactions between predator and prey species. We can describe these interactions as the hunting and eating of the prey species, and thus such an interaction will on the one hand determine the growth of the predator species, and on the other hand, the decline of the prey species. In this model, the populations of the predator and prey species are described as follows:

$$\begin{cases} \dfrac{dx}{dt} = \alpha x - \beta xy \\ \\ \dfrac{dy}{dt} = \delta xy - \gamma y \end{cases} \tag{2.13}$$

Here, x and y are the number of prey and predators in the model respectively. The prey reproduce at a rate of α in each step t and are killed at a rate β from each interaction with a predator. Meanwhile, the predators reproduce at a rate δ from each interaction with a prey and die at a rate of γ. This model assumes that the

prey have an unlimited access to food and that there are no deaths due to accidents or old age. On the other hand, the predators are dependent on the prey to survive and reproduce, and their deaths are due to starvation or old age.

The Lotka-Volterra model makes it possible to represent oscillating population patterns between species and predict if a species risks extinction. The model makes certain unrealistic assumptions such as there being no internal competition between prey/predators, but the work is still to this day used as a basis for developing more complex models. In the classical predator-prey systems, only the populations as a whole are considered. This means that it is possible to study changes in large populations, but it is not possible to study interactions on an individual level. But with the last decades of progress in computational power and efficiency, it is becoming possible to create individual-based simulations. By using an individual-based model, the hope is to achieve the population dynamics seen in non-agent-based models when the simulation's scale is increased to encompass the appropriate number of creatures.

2.4.2 Multiagent-based Simulations (Previous Research)

In recent years, researchers have already recreated the population dynamics of the traditional Lotka-Volterra equations using multiagent-based reinforcement learning [6], effectively introducing spatio-temporal characteristics to the predator-prey systems [8]. Although this is a great step in moving toward nature, there is still much work required to minimize assumptions and make the simulations adhere to nature's laws.

By looking closer at recent implementations of agent-based predator-prey systems, we see both similarities and differences in their approaches. The works we consider here are the papers by Yang et al. (2018) [6], Wang et al. (2019) [8] and Yamada et al. (2020) [16]. These works propose different techniques for implementing population dynamics of predator and prey, but all share the similarity of making use of a grid-based world representation for orthogonal movement where no agent may share the same position at one time.

2.4.2.1 Yang et al. (2018)

In addition to opening up the discipline of predator-systems to reinforcement learning, Yang et al. (2018) [6] develop a grouping behaviour in predators. The model's prey are divided into rabbits and sheep, where only rabbits can be captured by lone predators – sheep require multiple predators. Thus, as part of the predators' action space are the actions: *join group* and *leave group*.

In order to capture a prey, a predator must be within the prey's *capture area*. A prey can only be shared among predators of the same group. If there are multiple groups inside a prey's capture area, then the group with the largest number of predators present is the group that catches the prey. The authors claim that the predators will self-regulate the numbers of members in groups as if the number of members

in a group becomes too high, then the reward for each individual will be divided among too many predators. In this way, predators will sometimes leave a flock in order to form new flocks and potentially find higher rewards.

A downside to this model is that only the predators are trained as agents. The model contains predators, prey and obstacles; however the prey are implemented as static objects, which have a small probability of spawning each timestep, as opposed to modelling the prey as agents. The predators' spawn are also dependent on a constant rate, which reflects well to Lotka-Volterra, however this way of reproduction does not aid in bringing the model closer to an individual level based on sexual reproduction.

2.4.2.2 Wang et al. (2019)

As opposed to solely training the predators in the ecosystem, Wang et al. (2019) [8] propose the *co-evolution* of predator and prey. Similarly to Yang et al. [6], agents share intelligence via a common neural network, but this time the researchers grant the prey a neural network just like the predator species. Not only is it more realistic to assume that all animals have a form of intelligence, their results also show that training predators and prey simultaneously offers more stable populations.

This work introduces stochastic interactions between agents. As opposed to Yang et al.'s [6] work, predators hunting prey is not always a given. In this model, each cell in the environment is randomly matched with one of its neighbours, and depending on the cells' occupants, an interaction occurs as follows:

$$\begin{aligned} X + \emptyset &\rightarrow \emptyset + X \\ X + Y &\rightarrow X + X \\ X + Y &\rightarrow \emptyset + X \\ Y + \emptyset &\rightarrow Y + Y \\ Y + \emptyset &\rightarrow \emptyset + Y \end{aligned}$$

where X denotes a predator and Y denotes a prey. When choosing neighbouring cells all 8 directly- and diagonally neighbouring cells are considered, thus the probability of a predator eating a neighbouring prey is only 1/8. Predators need to eat in order to survive, so with this mechanic a predator could potentially starve when it is in reach of prey. On the other hand, predators are highly rewarded for adjacent and close prey so they will seek to stay in range to eat, which should render accidental starvation negligible.

Similarly to the predators, the prey are rewarded for avoiding nearby predators. The difference between the reward functions of predator and prey is that predators are rewarded for staying close to prey within a larger distance, meaning that a predator may be following a prey without the prey's knowledge.

The addition of trainable prey, which reproduces asexually (albeit automatically), adds a great realistic aspect to the model. As opposed to Yang et al.'s [6] reproduction which spawns new prey anywhere in the world, Wang et al. [8] make prey

spread locally and thus better approximating nature's reproduction. However, an aspect which is still missing from this model is that prey do not need to eat to survive.

2.4.2.3 Yamada et al. (2020)

Yamada et al. [16] combine some of the ideas from the previous works but also go further in exploring the evolutionary aspect of agents in predator-prey systems.

Firstly, similar to the idea of the capture area used by Yang et al. [6], this work features a predation square which in this case refers to a radius around each predator. During each timestep, the prey closest to the predator within the predation square will be consumed.

Secondly, like Wang et al. [8], prey are implemented as trainable agents. However, the reward function used for the prey is entirely reliant on the shared intelligence across the species and generations. In fact, the reward function, which gives only a reward for reproducing and a punishment for dying, offers no way of learning how to survive during the prey's lifetime. This raises the question of instinct and whether it is needed for survival. Even if the prey were to use the same reward function as Wang et al. [8], there would be no explanation for why the prey inherently escape the predator.

Next, Yamada et al. [16] are first to introduce sexual reproduction among both predator and prey. Although this is not implemented through a conscious action and rather by a probability based on two agents' (of the same species) distance to one another, agents are still able to consciously decide to mate by moving closer to another agent.

Alongside the mating mechanism, the researchers also propose a separate environment with an evolutionary algorithm containing parameters for *speed*, *attack*, and *resilience*. In this environment, a proportion of the species is spawned each timestep with two random agents of that species being picked as the parents (regardless of their positions). By incorporating the evolution of the agents speed and the predator's attack and the prey's resilience, the researchers find that – just like according to Darwin's theory of natural selection [12] – the agents with the traits best adapted for survival are the ones which remain in the species' populations.

The results of this evolution show that, as time goes on, the predator's attack and the prey's resilience both increase in order to overcome its adversary. However, as the prey's reward function does not provide any lifetime-learning, the prey's speed sees no growth: the prey only learns that without resilience, it is likely to die. As a consequence of this, the predator's speed only sees initial growth as all its food is practically stationary.

Although the work's reward functions pose a problem as they do not promote any lifelong-learning, Yamada et al. [16] make a great contribution to the field by intro-

ducing evolutionary algorithms. Furthermore, their analyses of the aforementioned environments concluded that both environments displayed Lotka-Volterra-like population dynamics.

3
Theory

3.1 The Animat Model

In this thesis, the terminology *animat* will be used to describe a simulated animal. Similarly to previous implementations of agents in predator-prey systems [6, 8, 16], animats have energy levels which must be maintained above 0 for the animat to survive. The animats regulate these energy levels by consuming food. These kinds of consumable food vary depending on the animat, but in this work we consider an omnivore predator which preys on a herbivore prey, and a herbivore prey which in turn consumes plants. To help the animats survive, animats make use of a set of senses and observations to help decide which action to take.

3.1.1 Senses

In nature, animals can observe their surroundings and choose to forage, flee from a predator or find a mating partner based on various senses. For many animals these senses are sight, hearing, smell, touch and taste. We simulate senses in animats by treating the sensory signals as observations which are to be passed to the policy network.

In our model we leave out the sense of hearing and the sense of taste in order to limit the amount of redundant observations.
Both sound and taste could in some extent be compared to smell. The former, could be compared to a smell which disperses very quickly and the latter to a smell that is only experienced when an object is eaten. In combination with the reward received for eating a consumable and the smell that the animat experiences at its own position is a sufficient model for smell. Overall, the taste could be compared to the following experiences for consuming an edible object:

- good food changes the animat's energy positively → tastes good
- bad food changes the animat's energy negatively → tastes bad

3.1.2 Energy

Although real animals have multiple critical needs for their survival such as energy, water and light, we choose in this work to only investigate the need of energy as a common need for the animats. Energy is gained by eating and used up by taking actions such as moving and mating, and also passively to represent a Basal Metabolic

Rate. Additional energy cost is also consumed when giving birth to a new animat connected to the orphant's body mass in comparison to the parent. Lastly, aminats which has not yet reached a maturity age and are still growing consumes energy for growing corresponding to the change in its weight. Matching the energy acquisition and consumption would result in the following equation.

$$C = M + R + G + E \tag{3.1}$$

Where C is the energy consumed, M is the energy used for maintenance (BMR and performing actions), R is the energy used for giving birth to new animats and G is the cost of growth. This way of calculating energy acquisition and consumption goes well together with the energy budget model presented by Sibly et al. (2013) [17]. The last variable E would correspond to the excess energy which contributes to the energy reserve. However we only assume that an animat has a maximum energy level, all energy it consumes it can store and use whenever it needs to, but it can not store an unlimited amount of energy.

The maintenance energy cost representing the animats' BMR and cost for moving is correlated to the animats' mass and work done respectively. According to Kleiber's Law larger animals are more effective than smaller animals as they produce less heat per mass unit. Kleiber proposes a power of 3/4 when comparing the heat produced to the animal's mass [18]. The cost for movement will in this thesis be equal to the work done by an animat while moving.

It should be noted that although we only consider energy in our work, animals are not reliant on only one kind of nutrition. Animals require a combination of water, protein, carbohydrates, sunlight, etc. to survive. Thus in future research, animats may be better represented with multiple needs which must be managed in parallel.

3.1.3 Reproduction

Different species reproduce differently in nature. There are various different categories of asexual reproduction, as well as various mating behaviours during sexual intercourse. For simplicity, we make the distinction between asexual and sexual reproduction only by whether one or two partners are needed to create offspring. We also consider the female to always be the partner to give birth to offspring.

Unlike automatic reproduction mechanisms such as the one used by Yamada et al. (2020) [16], we consider a reproduction mechanic based on conscious decisions. By using its decision networks, an animat will need to make the decision that it wishes to reproduce. For sexual reproduction, two animats will need to approach each other and simultaneously make this decision to reproduce. For asexual reproduction on the other hand, an animat may choose to create offspring at any time if the animat is fertile.

During sexual reproduction, a subset of the parent(s)' genotypes are mutated and

passed on to the offspring. For clarity, given the mutation function M, the relation between the offspring's genotype G, and its parents' genotypes G' and G'' is seen below in Equation 3.2.

$$\{g \in G \mid G \subseteq M(G') \cup M(G'')\} \tag{3.2}$$

In the case of asexual reproduction, G is produced using the same relations between offspring and parent, however for $G' = G''$.

3.1.4 Growth

When introducing reproduction, the animats should also be able to grow so that they are not born as large and fertile as their parents. The arguments for implementing growth into the ecosystem is not only because it is realistic. Rather, in order to more easily preserve the energy within the system by adding new animats with a smaller mass and also so that a new born animat will not reproduce with its parents or siblings as soon as it is born.

We consider a $t_{maturity}$ to be the time needed to reach full maturity. Until this moment is reached, an animat's phenotype will be found through $G \cdot m(t)$ where G is the animat's genotype and the maturity ratio $m(t)$ is given by:

$$m(t) = \begin{cases} t/t_{maturity}, & \text{if } t/t_{maturity} < c_{born} \\ c_{born}, & \text{otherwise} \end{cases} \tag{3.3}$$

We only consider a constant c_{born} to avoid infinitely small values in the phenotypes which could hinder the animats' learning due to not being able to interact with the environment.

3.1.5 Decision-making

The key idea behind the project that this thesis is based on is that the animats makes decisions by utilizing four different networks: a policy network, a reward network, a reflex network and a prediction network. In the scope of this thesis we will use and combine all but the prediction network. Depending on the animat configuration and the environment an animat will react to its surrounding by combining the output signals of these networks combined. The structure of the animat and its decision network is presented in Figure 3.1.

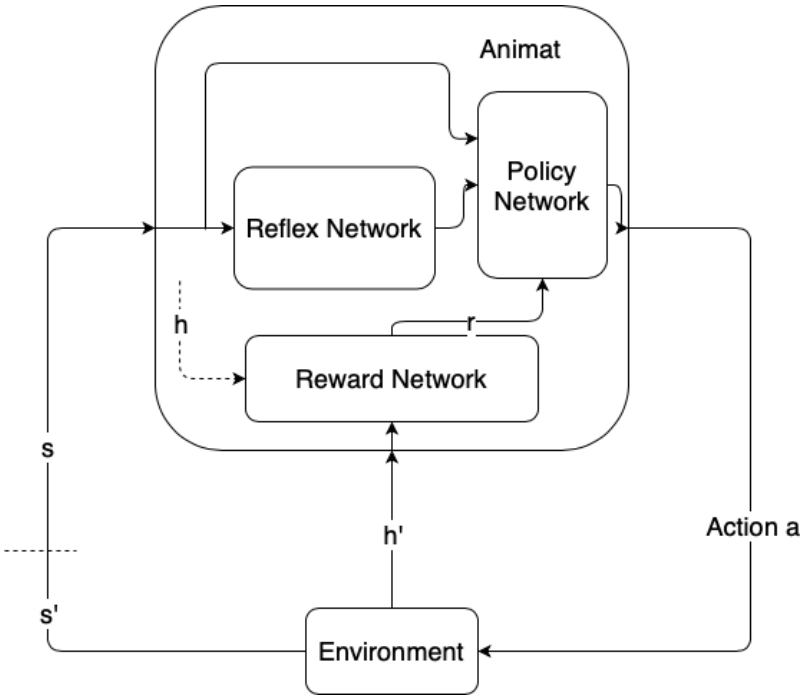


Figure 3.1: An overview of an animat's cognitive network structure

When an animat first receives observations from its environment it first checks if a reflex triggers. If a reflex triggers the reflex network communicates to the policy network to behave accordingly. Depending on what action the policy network decides that the animat takes the animat will either be rewarded or punished based on the output of the reward network which calculates the reward from the animat's homeostatic variables.

3.1.6 Policy Network

The role of the policy network is to choose an action to take depending on the observations the animat's sensor receives from its surroundings. The set of observations s are then used as an input to a neural network which predicts the best action a based on the set of observations or the animat's state. The predicted action will then be performed by the animat which receives new observations based on both the performed action and the new environment state.

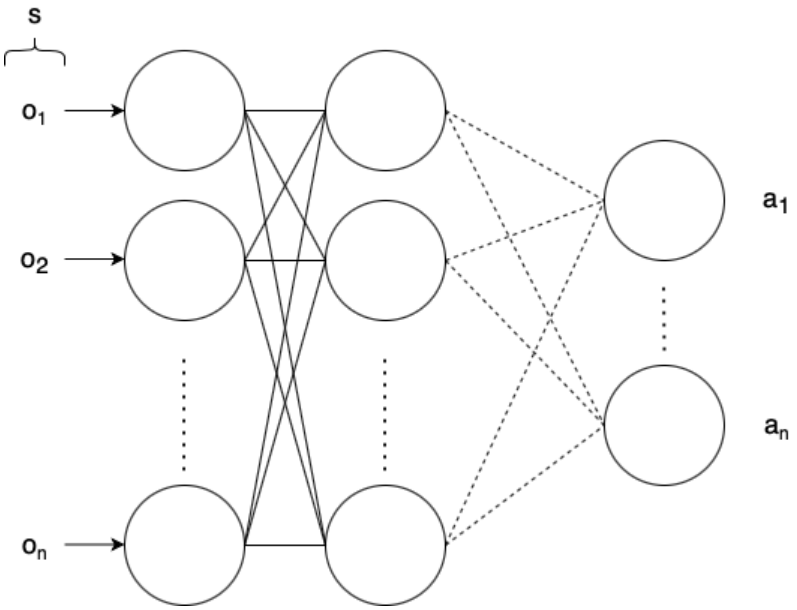


Figure 3.2: Policy Network Sketch

3.1.7 Reward Network

The focus of one of our research partner groups [19], working alongside us on developing this software model, was the animats' reward network. The goal of the reward network is to calculate the reward that an animat should get in each time step based on the animats homeostatic senses as shown in Figure 3.3. The reasoning behind this approach is that an agent does not need to be rewarded for every task it accomplishes, but instead be continuously rewarded as a consequence of accomplishing tasks.

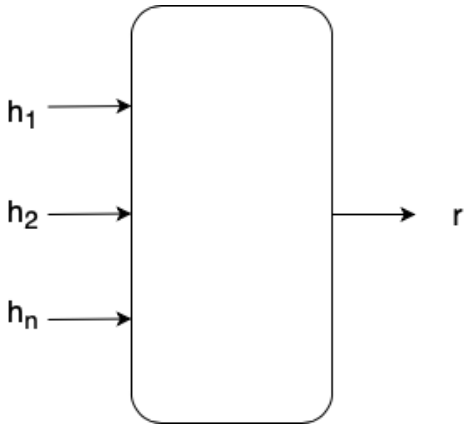


Figure 3.3: Reward Network Sketch

The reward that an animat receives is given by the difference in happiness given in Equation 3.4.

$$r_{t+1} = happiness_{t+1} - happiness_t \tag{3.4}$$

Where $happiness_t$ is the homeostatic state that is received from combining different utility functions as in Equation 3.5.

$$happiness_t(H_t) = \prod_{h \in H} (a_h + w_h u_h(h_t)) \tag{3.5}$$

The parameter H_t is a set of homeostatic variables $\{h_1, h_2, ..., h_n\}$ at time step t and u_h is a utility function corresponding to the homeostatic variable h. The weights w_h are used to change how big of an impact a certain homeostatic variable h has on the happiness of an animat. Finally, the variable a_h is a constant for each $h \in H$. In the scope of this thesis we consider only the logarithmic and linear utility functions as presented by Kleve and Ferrari (2021) [19].

The difference between using this reward network compared to using extrinsic rewards may appear subtle, but it allows us to more closely recreate the endorphins excreted by the brain for feeling good and happy. As endorphins are not exclusively released upon finishing a task, but rather continuously, we wish to simulate this in the animats. The basis for the reward network is thus happiness, and happiness is in turn a function of regulating the animat's homeostatic needs.

3.1.7.1 Homeostasis

The homeostatic variables that are used to compute happiness may be either critical or non-critical. The happiness of an animat depends on how far the perceived values of the homeostatic variables are from the desired states. A non-critical homeostatic variable that is far from its desired state will cause low happiness in an animat, but a critical homeostatic variable which is too far from its desired state may even cause death.

In our animats we consider two homeostatic variables, one critical and one non-critical. We consider `energy` as a critical need which may be increased through eating, and decreases over time or by taking certain actions. We also consider `libido` as a non-critical need which increases over time and decreases only if the animat mates. The domains of energy and libido are $[0, 1]$, with the desired levels: $D_{energy} = 1$ and $D_{libido} = 0$. For libido we use the linear utility function, this will make an animat increasingly unhappy as its libido grows. For energy we use the logarithmic utility function which will increase an animat's happiness more if the animat is hungry, whereas its happiness will increase less if its hunger is already satiated.

To reach maximum happiness, an animat must thus have maximum energy and minimum libido. Furthermore, by setting the homeostatic weights w_h to different values, the homeostatic variable which an animat must decide to regulate becomes even less binary. We consider energy to be more important to the animat's happiness as it is a critical value which is needed for survival, we thus set $w_{energy} > w_{libido}$.

3.1.8 Reflex Network

The reflex network takes an animat's observations as input and then either forces or prohibits one or more actions. By using a one-hot encoding from input to output, an animat can shrink its action space by masking the available actions when it receives a certain input. Based on the observations given from the animat's senses, certain actions are either forced or prohibited for the policy network to choose. The reflex network is part of the animat's genotype and so remains unchanged throughout the animat's lifetime.

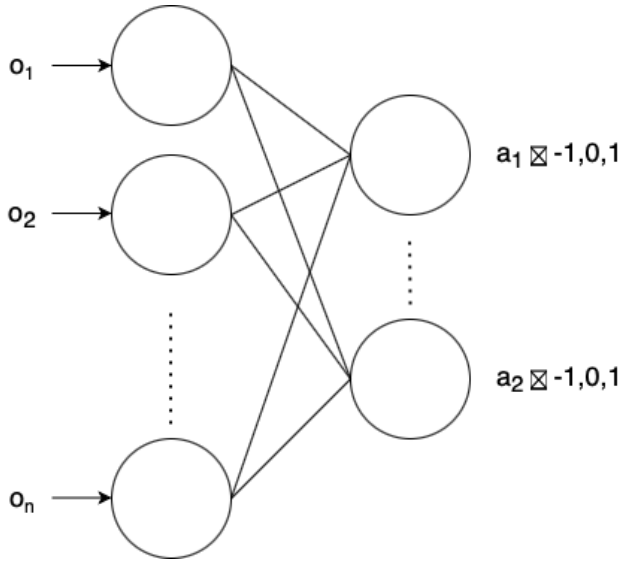


Figure 3.4: Reflex Network Sketch. The output {-1, 0, 1} refers to forbidden, unmodified, and forced actions, respectively.

Before a new action is chosen for the animat, it checks if it has some reflex corresponding to the current observation. If the observation made is connected to a reflex, the reflex network masks the actions as the corresponding status (forbidden/allowed/forced). As only one action is chosen per each time step t, forbidding an action means that the policy network can not pick that specific action at time step t and forcing and action means that the policy network can only pick that action at t.

In fact, an output suggesting that an action should be forced is equivalent to forbidding all other actions. For the action space A, the action mask can thus be represented as a set $M \in \{0,1\}^A$. The mask can now be easily used to recompute the action probabilities for the available actions – note that a maximum of one action can be forced at a time. To find the recomputed action probability $P(a)$ for an action $a \in A$, we need only do the following:

$$P_{masked}(a) = \frac{P(a) \cdot M_a}{\sum_{b \in A} P(b) \cdot M_b}, \forall a \in A. \tag{3.6}$$

The action masks from the reflex network thus open up an easy way for controlling an animat's behaviour. This also creates the possibility of simulating reflexes seen in mammals, such as the diving reflex prohibiting an animal from breathing underwater, the patellar reflex (knee-jerk), or pharyngeal reflex (gag reflex).

4
Methods

4.1 Animat Design

4.1.1 Predator and Prey

In our environments we have two different kinds of animats, a wolf predator and a goat prey. They differ slightly, but are in many ways very similar. All animats reproduce, grow and finally die in the same way. However, there are differences in the set of objects which can be observed by each species and which actions the animats can perform. The species specific differences are described in the upcoming subsections.

4.1.1.1 Prey

The prey are modelled as goats that consume different inanimate plant objects. The set of observable objects for the goat are given in Equation 4.1

$$\mathcal{O}_{goat} = \{F, G, W\} \tag{4.1}$$

Where F are fruit/plant objects, G are other goats and W is the wolf predator species. These observables are used for detecting objects by smell, sight and touch. The actions that the goat can perform is given by $\mathcal{A}_{goat}$ in Equation 4.2.

$$\mathcal{A}_{goat} = \{I, E, F, B, L, R, M\} \tag{4.2}$$

Where I is the idle action (animat does nothing). E is the eat action allowing the animat to take a bite of a consumable. The following four actions, F, B, L, R are used to move forward, backward and turning left and right. The last action M is the reproduce action (mate).

4.1.1.2 Predator

The predators are modelled as wolves. They hunt and prey on the goat animats. The set of observable objects for the wolves are of similar structure as $\mathcal{O}_{goat}$ but the specific objects differs as given by $\mathcal{O}_{wolf}$ in Equation 4.3.

$$\mathcal{O}_{wolf} = \{M, W, G\} \tag{4.3}$$

The consumable objects detectable by the wolves are the meat objects M. And similarly to the goat, the wolves can observe friendly animats W (other wolves) and

hostile animats G (goats). The wolves' actions are also similar to the ones of the goats but differs in one important matter as seen in Equation 4.4:

$$\mathcal{A}_{wolf} = \{I, E, F, B, L, R, M, A\} \tag{4.4}$$

The only way the wolves' set of actions $\mathcal{A}_{wolf}$ differs from the goat is the attack action A. This means that the wolves can attack hostile animats and doing damage to their energy level. Inspired by the attack and resilience traits evolved in Yamada et al.'s work [16], we change the damage done based on the animats' mass. A predator can evolve to deal more damage if it gets a higher mass, whereas a prey can evolve to receive less damage if it gets a higher mass. The damage dealt is given by $f_{attack} \cdot \frac{m_{predator}}{m_{prey}}$ where f_{attack} is randomly sampled between 0 and a preset value specific to the predator species.

4.1.2 Senses

Each animat has a set of senses which it can use to make observations regarding its environment. In this section the implementation of these senses are described.

4.1.2.1 Smell

The implementation for smell is designed for the animats to find the direction with the closest/most objects of a certain type. Every species in our model is able to sense the smell of a subset of all objects in the environment. The smell from an object of any other type is ignored.
The direction of smells is given by:

$$smell = \sum_{i \in \mathcal{E}} \frac{\hat{d}_i}{||d_i||^2}, \forall i \in \mathcal{O} \tag{4.5}$$

where $\mathcal{O}$ is the subset of observable object types sensible by the animat, d_i is the distance vector between the animat and the object i, and $\mathcal{E}$ is the set of objects in the environment. Thereafter the vector calculated in Equation 4.5 is separated into a vertical (d_{front}) and horizontal (d_{right}) component relative to the animat and a third component d_{above} telling the animat if it is on an object. This way the animat will be able to assess whether it needs to turn or keep its bearing to reach/avoid an object. In the rare case that an animat is equally far from all objects of a certain type, then the smell will not indicate in which direction the animat should move. However, since the directions of the objects are divided by $||d_i||^2$, this means that closer objects give a larger impact on where animats are more likely to find food, friends or foes.

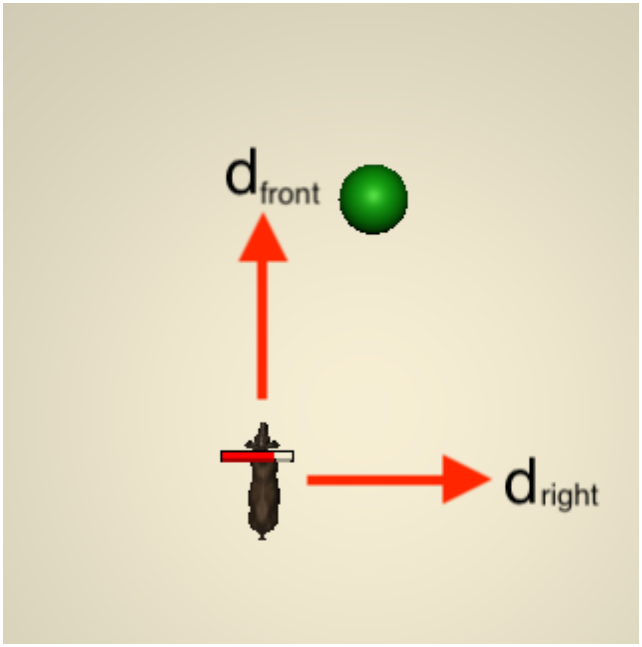


Figure 4.1: An animat smelling a plant object. Where d_{front} is the smell magnitude in front of the animat and d_{right} is the smell magnitude to the right of the animat.

Hence it could be said that the animat's sense of smell is greedy in that it may be interested in one close object in direction a, even if there are multiple objects further away in direction b. However, as odours disperse over time and distance, this approach is realistic in that it is not uncommon to better sense the smell of nearby objects as opposed to objects far away.

4.1.2.2 Sight

As natural sight relies on light bouncing off objects onto the retina, this means that we can simulate sight by using ray casts. In an environment considering all three dimensions, ray casting would be computationally expensive as animats would need to be able to see forward, to the sides, up and down. Luckily, as we only consider two dimensions for movement, animats only require sight forward and to the sides, i.e. only a few ray casts are needed per animat. We do not consider varying light levels in our simulations (e.g. day- and night-time) and so, an animat's field of view will always register sight.

The implementation of the ray casts are done using ML-agents' `Ray Perception Sensor 3D` component. This gives the animat information about which type of object it is perceiving and how far away the object is.

As only a handful of ray casts are used for each animat, it is possible that an object can hide in the "gaps" between each ray if an animat is too far away to be detected by the sphere radius which the ray uses to detect an object. Although this might be a realistic approach as this would simulate an increasing difficulty detecting objects that are far away.

In order for the predators to be able to see the prey in an environment that is full of plants, the plants have been masked out from the objects which the predators can observe as seen in Figure 4.2. The same assumption has been made for the prey regarding the meat objects as there is no reason for the prey to be able to see the meat. Every species' set of observable objects $\mathcal{O}$ for sight are the same as those used for the sense of smell.

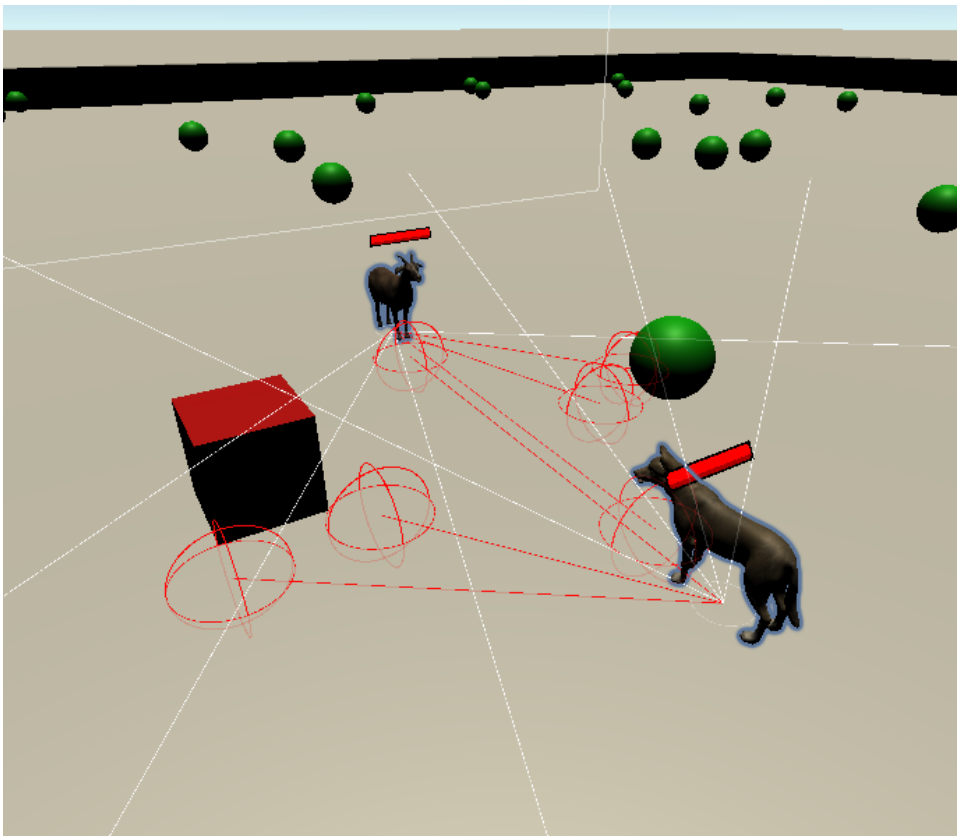


Figure 4.2: The animats' ray sensors. The wolf at the bottom right does not observe the green plant object and the goat at the top left does not observe the red meat object.

4.1.2.3 Touch

As a final external sense, we implement a sense of touch. Unlike the prior senses however, this sense rarely gives any useful observation to the animat. Touch sends an observation to the animat with an integer value of how many objects of a certain type it is currently touching. The reason why the sense of touch is designed in this way is that objects cannot be distinguished by material or shapes e.g. grass cannot be identified by its pointy texture. However, an animat may still find a correlation between the number of nearby objects and the outcome of a certain action. Touch can thus be a very important last chance for a prey to sense a predator, and furthermore the sense is vital for indicating when it is possible for the animat to eat or reproduce.

4.1.2.4 Homeostatic Senses

Lastly, an animat can also observe its homeostatic variables which are their energy level and their libido. The sought-after energy level is 1 and is decreasing for each time step and decreasing more rapidly when an action is performed. The libido's sought-after value is 0 and is instead increasing by a certain amount each time step, depending on mating season.

4.1.3 Evolution

Our animats make use of a simplified version of genotype and phenotype. An animat's phenotype depends only on time and its genotype, e.g. an animat with the genes for growing 100 metres tall *will* grow 100 metres tall unless it dies first. The only exception to this is the animat's policy network. By default the implementation of PPO in ML agents makes use of a shared policy network for each agent type. Thus each animat species shares a policy network which will be optimized over generations.

As for contents of an animat's genotype, there are three categories of variables which may be inherited and mutated: homeostatic weights, attributes (such as mass, maximum velocity, etc.) and reflexes. The homeostatic weights directly affect happiness by changing the impact a homeostatic variable has on the happiness (see subsection 3.1.7 and subsubsection 3.1.7.1). The genetically-connected homeostatic weights are the following:

- energy
- libido

The attributes that the animat has can often affect the animat's survival directly or the survival of its entire species. If a prey has a higher speed compared to its predator it might be able to escape a hunt. On the other hand an animat could ensure the entire species' survival by having multiple offspring for instance. The genetic attributes are:

- mass
- max velocity
- size
- fertility
- acceleration
- mating season
- smell radius
- offspring size
- number of offspring

Finally, an animat can inherit any number of reflexes from its parents. This can lead to an animat inheriting conflicting reflexes which inadvertently dooms the animat. For example, an animat could have one parent which, by reflex, is incapable of eating red food and one parent which is incapable of eating green food. If the animat inherits both these genes in an environment with only red and green food, the animat would be incapable of eating and starve immediately. On the other hand,

thanks to mutation, it is possible that the offspring of parents with reflexes is born without compromising reflexes.

4.1.3.1 Reproduction Cycles

Nature shows at least some degree of seasonality in reproductive periods [20]. Seasons vary depending on the species and multiple mating seasons may occur per year in some species, but in our model we consider one mating season per year.

4.1.3.1.1 Fertility

Each animat is born with a gene corresponding to its fertility. We consider a male animat to be equally fertile year-round whereas a female's fertility changes throughout the year. To simulate an estrous cycle using this idea, the fertility at timestep t is found by $f_t = f \cdot E(t)$ where $E(t)$ is the function of the estrous cycle. We model the seasonal estrous cycle according to Equation 4.6.

$$E(t) = \frac{1}{2}[\sqrt{\frac{1+b^2}{1+b^2+\cos^2(\frac{2\pi t}{t_{year}}+2\pi(S_s+S_i))}}\cos(\frac{2\pi t}{t_{year}}+2\pi(S_s+S_i))+1] \quad (4.6)$$

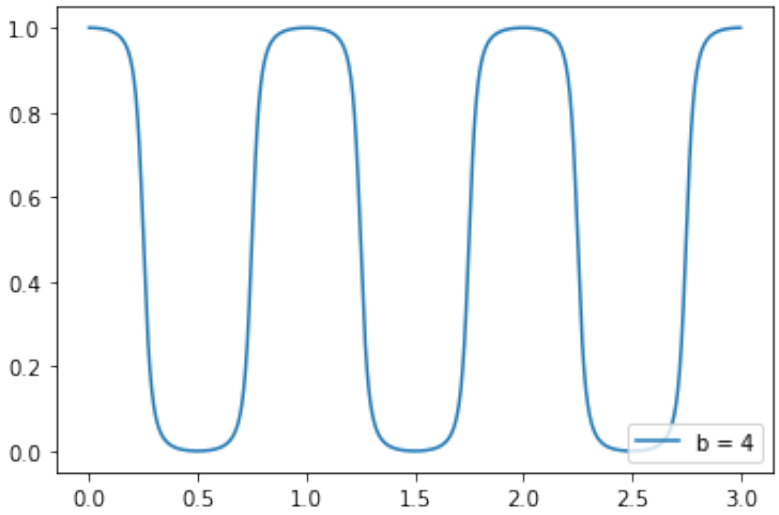


Figure 4.3: Estrous cycle $E(t)$ where S_s and S_i are the seasonal offset for the species and the individual according to Equation 4.6

We consider an estrous cycle $E(t)$ such that females are more fertile one half of the year and less fertile the other half e.g. highly fertile during summer and moderately fertile in late spring/early autumn. As $0 \leq f \leq 1$, using $E(t)$ from figure 4.3, we see that $f_t < \frac{1}{2}$ when the female is not in heat. To allow for generational shifts in the mating seasons, we introduce a seasonal offset S_i which is inherited and mutated at birth like all other genes. We also introduce the seasonal offset S_s which is used

for offsetting the species' fertility and libido changes (more in 4.1.3.1.2). This way we are able to make our prey mate during autumn/winter as goats do, and make our predator mate during summer as wolves do. However, we assume equal – but shifted – estrous cycles for all species to limit the complexity of the model and to give all species equal opportunities for mating. We also recommend a estrous cycle for mammals with monthly estrous cycles as seen in Equation 4.7

$$E'(t) = \max(\sin^3(\frac{2\pi t}{t_{month}} + 2\pi(S_s + S_i)), 0) \tag{4.7}$$

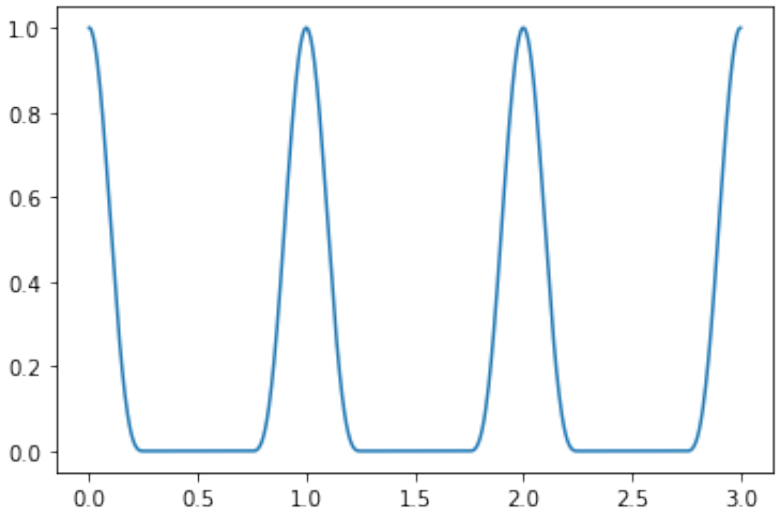


Figure 4.4: Another possible estrous cycle in Equation 4.7, more closely related to the estrous cycles in primates.

In animals with more frequent mating seasons such as primates, monthly peaks of fertility may more closely resemble the real-life estrous/menstrual cycles. We choose to not include the estrous cycle $E'(t)$ from 4.4 in our model as the shorter the mating seasons, the more difficult it is to identify seasonal reproductive behaviour.

4.1.3.1.2 Libido

In addition to the females' changing fertility, all animats' libidos change over time. To promote mating during the mating season and avoid mating off-season, the libido can either increase or decrease depending on the season. The libido changes in every timestep by

$$c_s \cos(\frac{2\pi t}{t_{year}} + 2\pi S_i)$$

where c_s is a constant for the species. This is true for every animat that has reached its maturity age $t_{maturity}$ and has not recently had an offspring (9 months in our experiments).

4.1.3.2 Sexual Reproduction

For an animat to maximize its happiness, it must handle all its critical needs as well as its non-critical needs (see 3.1.7.1). An animal's libido is a non-critical need, and in order to satiate its libido, the animat must participate in sexual intercourse.

In order to participate in sexual intercourse, both the animat itself, and its partner must take the `MATE` action when staying inside its partner's reproduction radius as seen in Figure 4.5. In case there are multiple available partners, as is often the case in nature, the female animat will choose the male partner with the highest product of energy and fertility.

Figure 4.5: Two animats of different sex staying within each others reproduction radius as seen in blue (male) and red (female).

To facilitate the mating, we allow animats to stay available for mating up to $t_{mating} = 5$ timesteps after taking the `MATE` action. We also synchronize the mating of animats to avoid situations where a less able male is chosen as the sexual partner when a more able male may still choose to mate. We accomplish this by flagging every action in each timestep, and only when the number of flags equals the number of animats present, we match each mating female with an appropriate male within a distance of d_{mating}.

When animats mate, they receive a reward from the reward network based on their libido to encourage a behaviour where the species reproduce and avoid extinction. Thereafter, both partners are made unavailable for mating to all other animats in order to restrict the number of sexual encounters per animat per timestep. In addition to rewarding the animats for limiting their libido, a number $0 \leq n \leq 1$ is sampled, and the female will create a number of offspring using $g(n)$ given by:

$$g(n, f_{parent1}, f_{parent2}) = \begin{cases} 1, & \text{if } n < f_{parent1} * f_{parent2} \\ 0, & \text{otherwise} \end{cases} \tag{4.8}$$

where f designates the fertility of a parent.

As the need for mating is driven by the animats' libido levels, upon mating these levels are to be reset. However in order to push the females to reproduce offspring, we only reset the libido of females if they are impregnated whereas a male's libido is reset regardless of impregnation.

4.1.3.3 Asexual Reproduction

As we use the same framework for both sexual and asexual reproduction, this has led to a very simplified mechanic for asexual reproduction. Making use of the same fertility and libido mechanics as in sexual reproduction, an animat may attempt to mate with itself in order to produce offspring. The key difference between asexual and sexual reproduction is that an animat reproducing asexually is only rewarded for successfully creating offspring. But if an asexual animat is thought of as a female sexual animat, the circumstance for reward is the same.

4.1.3.4 Mutation

Mutation works on all the animats' inherited characteristics. As the majority of these characteristics are floating-point numbers, this means that the inherited genes are altered by a random factor r_c between $1 - c_{mutation}$ and $\frac{1}{1 - c_{mutation}}$ according to Equation 4.9.

$$g = g \cdot r_c, \forall g \in Genotype \tag{4.9}$$

This holds for both the homeostatic weights and the genetically adjustable attributes. However, in the case of the reflexes there is a probability $1/c_{mutation}$ that a new reflex is created.

4.1.4 Age

An animat's age is defined as the difference between the number of time steps t passed since the simulation's beginning and the time step of the animat's birth t_{born}, i.e: $age = t - t_{born}$. The age is necessary for both newborn animats' maturity and later, its possible death from old age.

4.1.5 Growth

In order to preserve energy in the system and stop animats from directly mating with its parents or siblings, newborn animats are born infertile and smaller than their parents as in Figure 4.6.

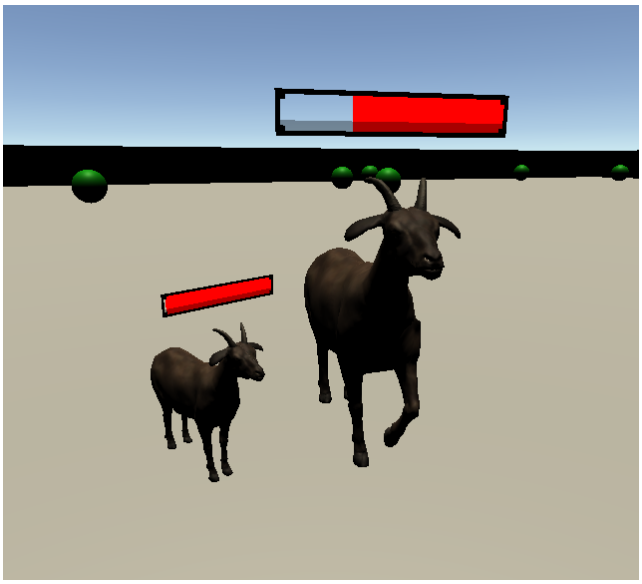


Figure 4.6: A newborn animat and its parent.

They then grow with a constant rate until they reach their maturity age. The cost of growing is the difference in mass Δm_t in each time step. At the same rate that the child's mass is increasing, a number of other attributes grow as well. These attributes are, among others, the animat's size, acceleration and maximum speed.

4.1.6 Death

The simulations have either one or two death criteria for an animat to terminate. In all environments, the animat starves if the energy level reaches 0. The other death criteria is if the animat reaches a maximum age. It is stated in the environment whether or not an animat can die of old age.

If a prey animat dies, it drops a meat consumable. This consumable is the food for the predator, thus the predator must either let the goat starve, die of old age or attack it before its energy level runs out.

4.2 Advanced Plant Modelling

Plants – i.e. a food resource for the prey – have already been implemented in a multiagent-based predator-prey system by Wang et al. (2020) [7]. However, according to the paper's plant specification, the plants spawn in any position without a plant – causing plants to cover the entire environment in only a handful of time steps. We improve upon their flawed implementation of food in two different ways: one basic probabilistic spawn, and one more advanced model.

We will use the basic generation of new plant objects in a limited amount of experiments as they are computationally cheap to use and easy to control. A plant has a small probability of spawning in each timestep, and thus by adjusting the

probability that a new plant will spawn, the environment can be filled with less or more food. We will refer to this kind of food object as static food.

In addition to the static food objects, we propose a modelling of plants which spreads gradually in the environment depending on the other plants in the environment. The first reason behind this is that the model would serve as a more realistic plant model for the plants to spread. The second reason is that herbivores might develop a more natural foraging behavior, as the spreading of their food is more closely related to how it spreads realistically.

Accurately simulating plants adds computational complexity to simulations, and therefore it is necessary to consider which aspects of nature may be simplified. Two assumptions we make for all our plants to reduce the environment's complexity are the following:

- Lifetime
  A plant of species s lives for t_s timesteps and then dies (unless eaten first). Upon death, the plant is no longer observable or edible.
- Genes
  All plants of a species s share a set of genes G_s. Thus, there is no evolution of plants through mutation or natural selection.

4.2.1 Competition

Plants of different species compete for resources from the soil and the sun, and thus we wish to model a competition between individual instances of plants. We define *grace radius* as all points within a Euclidean distance of d_{grace} from a plant which cannot be occupied by any new plants. Furthermore for interspecific competition, we define *hostile radius* as all points within a Euclidean distance of $d_{hostile}$ from a plant which cannot be occupied by new plants of other species.

4.2.2 Spread

Each plant has a parameter t_{ripen} during which the plant has not yet matured and is therefore unable to reproduce. After this, a plant will reproduce asexually to create a new plant with a probability $p_{fertility}$, where $0 < p_{fertility} < 1$ during each timestep. If a seedling would be restricted from spawning due to a grace/hostile radius, then no new position is chosen and instead the reproduction fails. This restriction makes the spread of plants slow down as the number of plants increases, thus avoiding explosive and sudden growth.

4.2.3 Grass

We make the assumption that grass may only spread through its roots. To model this spread, a grass object has a defined spread radius within which the grass can spread. To avoid that a grass object fails to spawn a seedling due to placing the

seedling within the grass object's own grace radius, we randomly pick a distance d such that $d_{grace} < d < d_{spread}$.

To represent the way that grass can be grazed without immediately killing the grass, we allow grass to be eaten by animats before recovering after $t_{recover}$ timesteps. A grass object in the process of recovering cannot be detected by an animat, and other plants are able to grow inside its grace and hostile radii – killing the recovering grass.

4.2.4 Dandelions

We make the assumption that dandelions only spread through its wind-carried seeds. Similarly to grass, we avoid sampling a position for the seedling within its parent's grace radius, however in order to make the spread of faraway of seedlings less likely, we instead pick a distance d^2 such that $d^2_{grace} < d^2 < d^2_{spread}$.

As dandelions do not recover like grass, dandelions are disfavored in the environment before simulations even begin. However, by giving dandelions a larger grace radius than grass, dandelions have the chance of stopping grass from growing in their vicinity.

4.3 Environment Designs

With the many theories used to create our model, we have chosen to perform our simulations in a number of environments in order to examine different aspects of the animats' behaviours.

4.3.1 Pre-training

Although our work's motivation is to investigate whether evolution is needed to correctly simulate the population dynamics in ecosystems, we make use of pre-training to a degree. Nature's evolution took millions of years, and thus it would be unreasonable to make use of nothing but evolutionary algorithms to find the optimal parameters for reinforcement learning. However, we limit the degree of pre-training such that our animats do not develop near-optimal behaviour before the main experiments.

Furthermore, pre-training allows us to observe if the animats are able to balance multiple needs with the use of the reward network. As the reward function is a key part of getting functionally learning animats, it is important to be able to validate whether we have designed a good reward function through the pre-training experiment.

Unlike in the main experiments, all pre-training animats are immortal. This means that they spawn again in case they die, and receive an additional penalty for dying. This design comes from the risk that an animat could develop strategies for regulating its needs by simply starving itself. Another difference between the pre-training

and the main experiments is that reproduction does not lead to any offspring as they would interfere with the two animats' pre-training (energy and libido levels are changed as normal).

In order to limit stochasticity in the animat itself, the pre-training animats do not mutate their attributes, homeostatic weights nor reflexes. Lastly, if not stated otherwise, every pre-training animat is generated with set value attributes and no reflexes.

4.3.1.1 Prey

During pre-training we use male and female herbivores in an environment with only static plant food – food that is randomly generated with some probability p_{food}. The reasoning for using static food is that the amount of food is easily controlled by adjusting p_{food}. With limited spread of food in the environment, the animats must learn to find food, as opposed with the advanced plant spread – whose spread is more difficult to restrict. The importance in using animats of different sexes is that they will be able to learn how to balance their two needs: their energy level and their libido.

A pre-trained prey is thus expected to be able to find food through exploration and it is expected to know when it must eat to survive. Additionally the prey is expected to, in some extent, learn how to keep their libido as low as possible without starving.

4.3.1.2 Predator

A pre-training predator is trained in a similar way to the pre-training prey. We introduce predators of different sexes in an environment with a limited amount of food. In order to teach the predators that it needs to first attack a prey and then eat it we split the pre-training into three parts.

In the first part of pre-training, we use goats that do not move and wolves that are forced to eat whenever meat has been generated after killing a goat. This is done using reflexes, the goats have reflexes that prevents them from performing any other actions than idling and wolves have reflexes that forces them to follow food smell and then eat the food whenever they can smell food or are close enough to eat. As a second part of pre-training, the wolves' reflexes are disabled after a set amount of steps, allowing the wolves to once again perform any action that the policy network provides for the individual.

The last part of the wolves pre-training consists of the wolves hunting goats that have pre-trained behaviors. This means that the goats are pretrained to forage and reproduce, leading to the wolves having to hunt for their prey instead of just collecting the prey. The reasoning behind this pre-training design is to allow the predators to faster learn that they need to eat their prey after it has been killed. Similarly to the prey, the wolves must also learn how to balance their energy level by hunting

and eating, and their libido by reproducing whenever their libido is high.

4.3.2 Main Experiments

In the main experiments we place a number of either pre-trained prey and predators in environments containing different types of food. Not all environments contain a predator species, but in the environments where they are concerned we can expect an initial advantage for the predator as they have experience in hunting the prey. Prey on the other hand may have initial difficulties in surviving when chased by predators, this is where the goal of natural selection's properties come into play: to adapt to its environment. The structure of each environment is described in this section, specific initial values can be found in Appendix A. All values were chosen by trial and error through multiple tests in order to get the best possible results with the smallest performance cost.

4.3.3 Lethal Food

To demonstrate the importance of evolution we construct an environment containing static food i.e. static objects which are spawned, moved and destroyed using heuristics. These food objects are separated into three categories:

- Good (green) food: animats gain energy upon eating
- Bad (yellow) food: animats lose energy upon eating
- Lethal (red) food: animats die upon eating

This environment contains only herbivorous prey animats, and through the use of the reflex network we investigate the need for reflexes to survive. The reflex network may contain reflexes to avoid eating a certain type of food after the animat has chosen to perform the `EAT` action. This test can in some way be compared to the evolutional development of a reflex like the dive reflex which hinders mammals to breathe underwater, saving them from possibly drowning. Through evolution, reflexes hindering the animats' survival should go extinct and only reflexes favoring the survival of the animats should remain. Similarly, the animats without reflexes should be unable to survive due to the presence of lethal food.

The animat can sense which food is which by its color which is then sent to the reflex network which decides whether the animat is allowed to eat the specific food it is standing next to. At the start of the simulation there are goats which cannot eat good food, goats which cannot eat bad food and goats which cannot eat lethal food. The good food is represented by its green color, the bad food by its yellow color and the lethal food by its red color as seen in Figure 4.7.

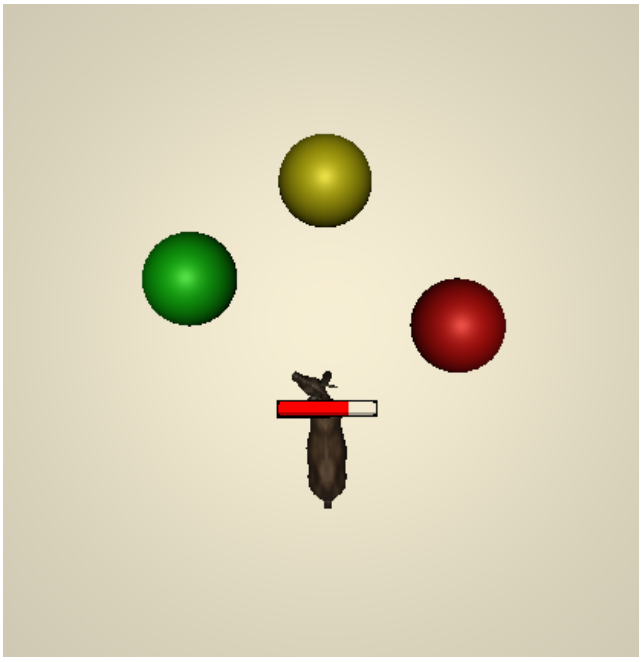


Figure 4.7: A prey animat with good (green), bad (yellow) and lethal (red) food.

4.3.4 Grass & Dandelions

Finally, we propose environments using the grass and dandelions described in section 4.2 and observed in Figure 4.8. We introduce this very simple abstraction of plants' spread in nature in order to model how animats affect their environments. In nature, plants are also affected by the animals which prey upon them. Thus, we investigate the population dynamics of not only the animats, but also the plants, to see if the simulations can give rise to multidimensional Lotka-Volterra equations.

We run the simulations with the advanced plant models in three different variations of this environment: one with only grass and dandelions, one with prey, and one with predator and prey. This enables the comparison of the plant populations when preyed upon: fluctuating numbers of prey, and growing numbers of prey. In the last experiment we want to examine how the predators affect the balance between plants and herbivores. Additionally we want to observe whether we will see behaviour resembling stable 3-system Lotka-Volterra equations.

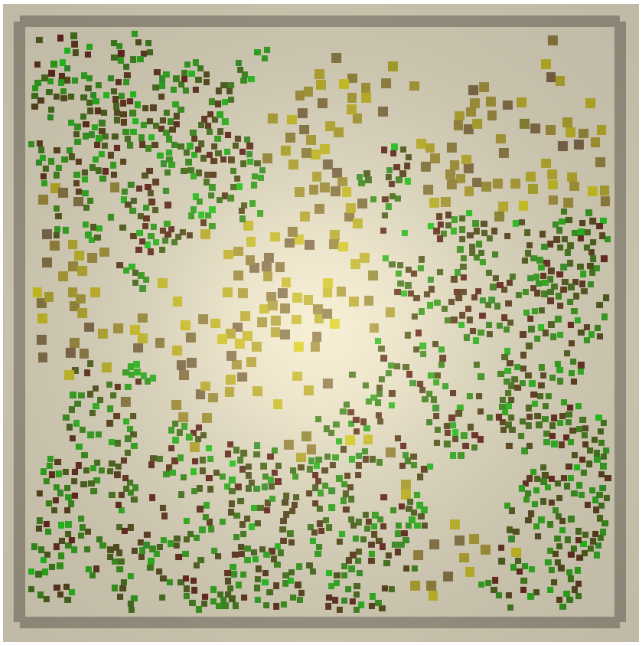


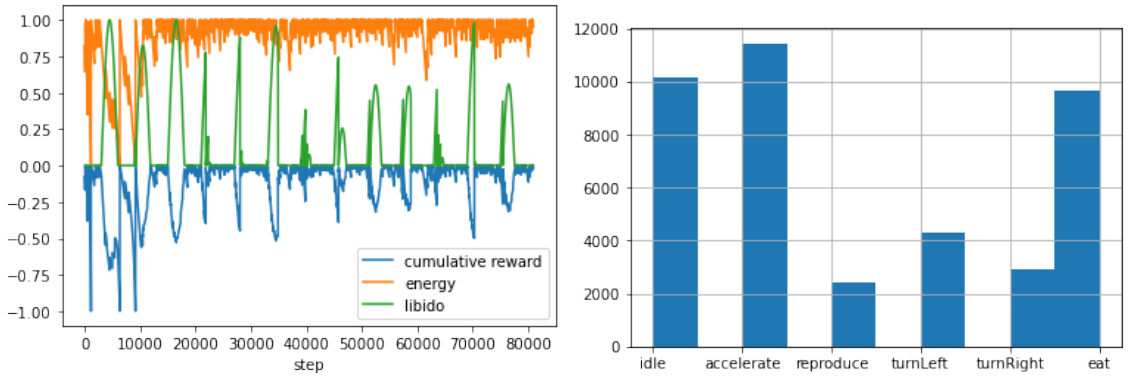
Figure 4.8: Environment with competing grass represented in green and dandelions represented in yellow

5
Results

This section covers the results collected from the pre-training and main simulations run. Parameters mentioned in this section are more detailed than in the earlier sections of the paper, however for more exact examples of parameters used, we refer to Appendix A.

5.1 Pre-training

The first goats to be trained were trained for a bit more than 80 000 time steps. In this environment the goats learned to forage really well and achieved some balance between their energy and their libido. The goats' energy levels never reached 0 after approximately 10 000 time steps and they learned to reproduce regularly after 30 000 time steps. However, the amount of food available in the environment was decreased as the simulation went on, making the access to food scarce in the later parts of the training. This caused the goats to learn a behaviour promoting idling in order to save resources and wait for new food to spawn which can be seen in Figure 5.1.



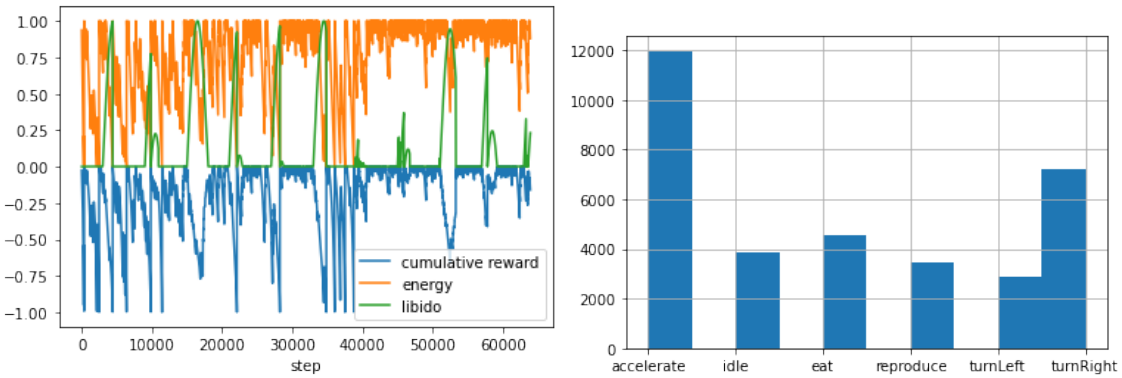
(a) Energy and libido's effect on reward

(b) Number of actions performed in the last 40 000 time steps

Figure 5.1: First pre-training goat showing idling behavior.

Another set of goats were trained which would not have to wait for new food in order to develop a more active foraging behavior. This set of goats was trained in a larger environment to better resemble the world they would actually be tested in which caused them to have greater difficulties surviving initially. However, after

approximately 40 000 time steps they achieved a behavior promoting both active foraging and regular reproduction as seen in Figure 5.2.

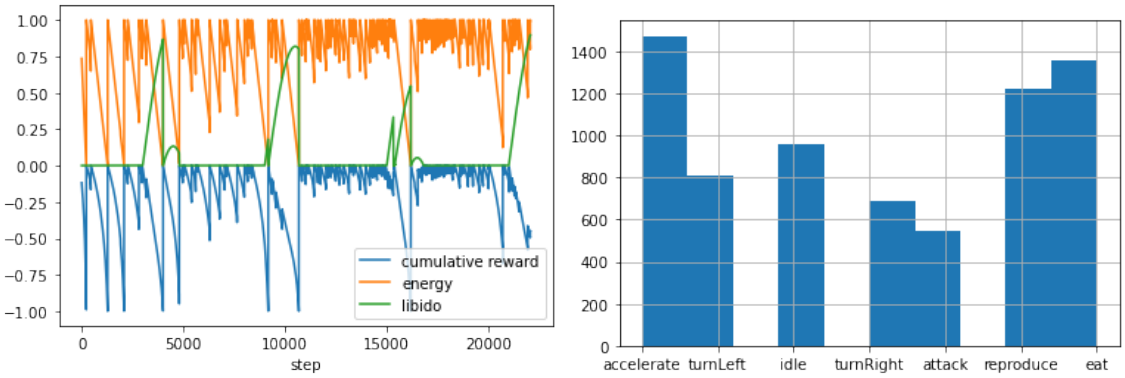


(a) Energy and libido's effect on reward

(b) Number of actions performed in the last 30 000 time steps

Figure 5.2: Second pre-training goat with active foraging and regular reproduction.

The first set of wolves was pre-trained on static, immobile goats. For the first 7 500 time steps they had reflexes forcing them to eat whenever they had captured a prey. This proved to help a lot with training times as they learned much faster not just to hunt, but also to eat what they captured. Also these wolves started to show tendencies of balancing, foraging, and reproduction as seen in Figure 5.3.

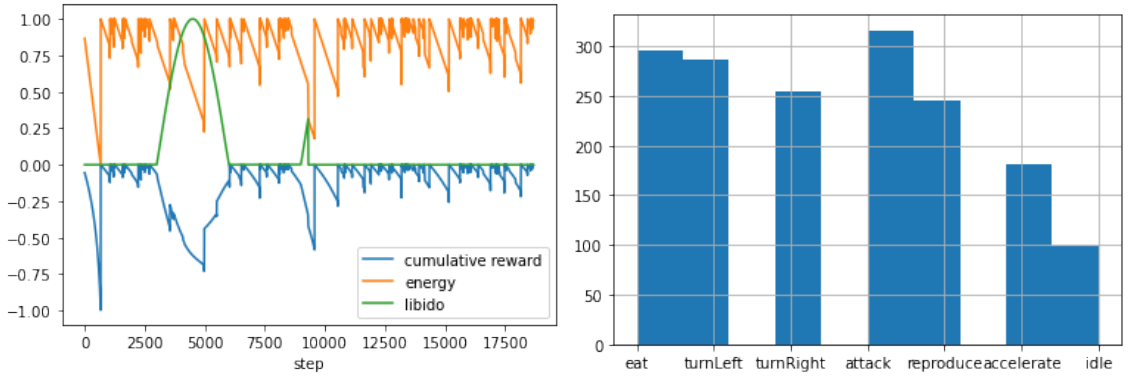


(a) Energy and libido's effect on reward

(b) Number of actions performed in the last 5 000 time steps

Figure 5.3: Pre-training wolves training on static goats.

When the wolves continued to train on the moving goats they had an easier time surviving due to their acquired skills from the first pre-training on static goats. Here they showed clear abilities to satisfy both their energy level and their libido seen in Figure 5.4.



(a) Energy and libido's effect on reward

(b) Number of actions performed in the last 2 000 time steps

Figure 5.4: Already pre-trained wolves training on goats with set policies

5.2 Main Experiments

5.2.1 Lethal Food

The lethal food experiment showed clear results on how the reflexes helped, or hindered, the animats' possibilities of surviving. In Figure 5.5 it becomes clear that the reflex preventing the animats from eating lethal food (red curve) is directly necessary for the animats' survival and the reflex preventing the animats from eating good food (green curve) is directly hindering. It also becomes clear that even if the reflex preventing the animats from eating bad food (yellow curve) could be considered a good reflex it is not necessary for their survival when comparing it to some arbitrary reflex without any effects in the environment (blue curve). Worth noting is that an animat can have multiple reflexes at the same time. For instance, an animat with a reflex preventing it from eating lethal food could also have a reflex preventing it from eating bad or good food.

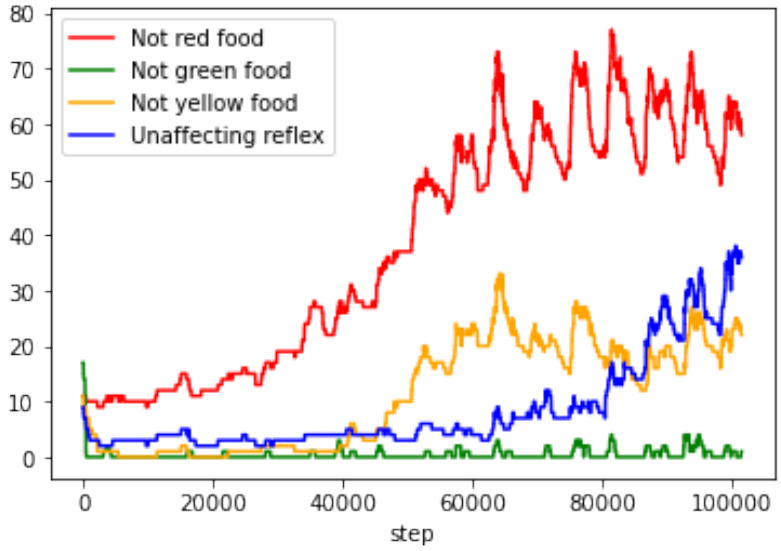
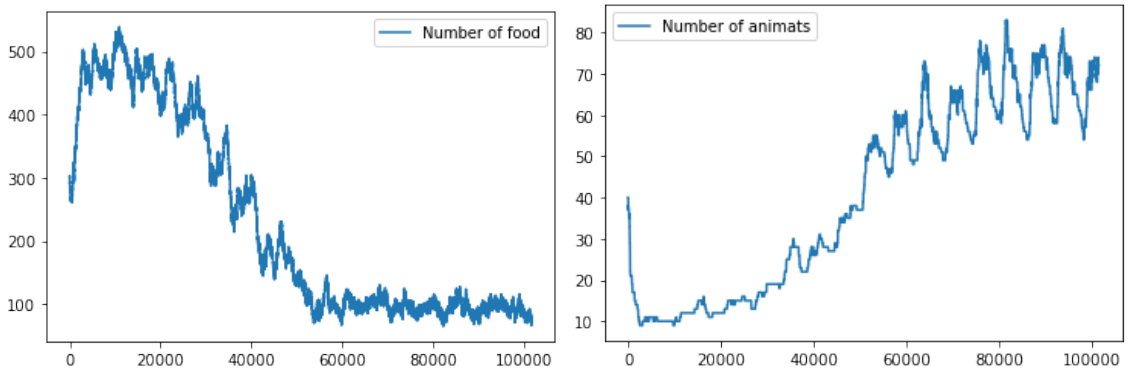


Figure 5.5: The prevalance of different types of reflexes in the prey population over time.

When comparing the amount of animats with specific reflexes in Figure 5.5 above to the total number of animats in Figure 5.6 below, we can see that almost all the animats have the gene connected to the reflex preventing them from eating red food. It is also clear that the fluctuation and limit in the goat population is due to the scarce amount of food in the environment. This is an indication that the population has reached its theoretical limit for which this specific environment supports.



(a) The number of food in the environment over time.

(b) The number of prey in the environment over time (plotted separately for clarity due to scale).

Figure 5.6: Amount of food and number of animats in the environment.

In Figure 5.7 we observe how the longest-living goat in the simulation manages to balance its needs. It never lets its libido reach 1 and dies after 80 000 steps. The near-death after 55 000 steps and the increasing difficulty to survive is most likely due to the lack of food in the environment seen in Figure 5.6.

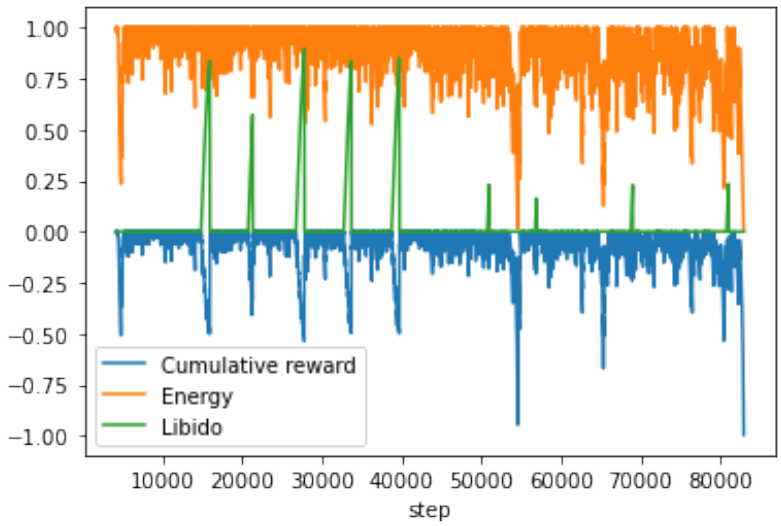
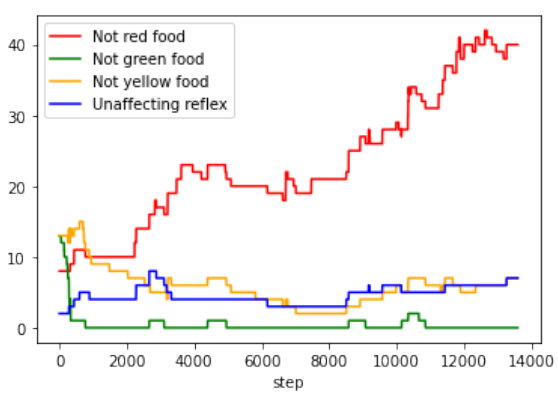


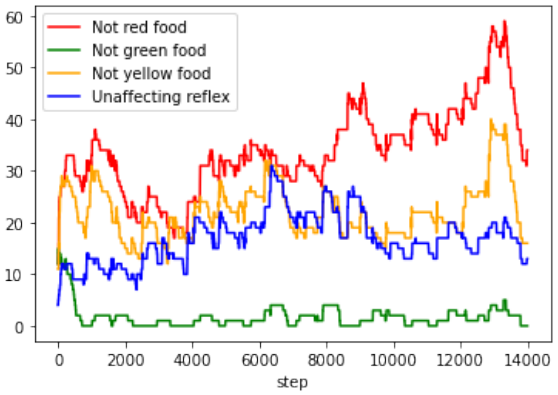
Figure 5.7: The cumulative reward and the homeostatic variables of the longest-living goat.

Animats that reproduce sexually were also compared to those animats that reproduce asexually in this environment in order to establish how well each reproductive method passes on genes. As seen in Figure 5.8, both reproductive methods lead to surviving animats. However, the asexual animats prove to grow faster and to a larger population, albeit with a less stable growth – also displaying periods of decreasing populations.

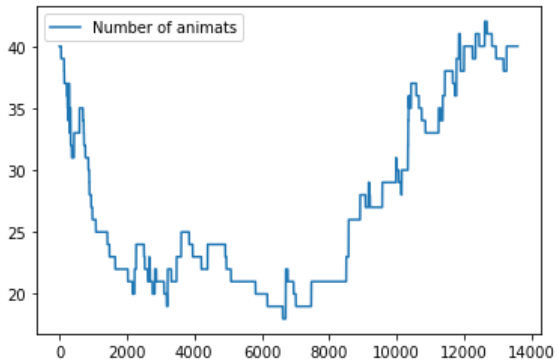
To further compare the reproductive methods, we create an even harsher environment (see results in Figure 5.9). We adjust the "bad" food to work in the same way as the lethal food. This new environment thus contains two species of lethal food objects. We initiate the animats' reflexes in the same way as in the previous simulation, thus any animat created initially could die due to a lethal food object.



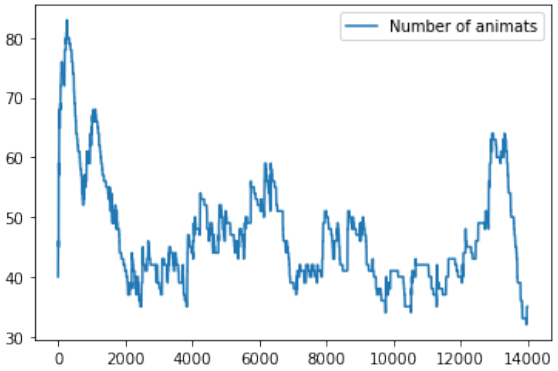
(a) The number of reflex genes in the animats alive, reproducing through sexual reproduction.



(b) The number of reflex genes in the animats alive, reproducing through asexual reproduction.



(c) The number of animats alive reproducing through sexual reproduction.

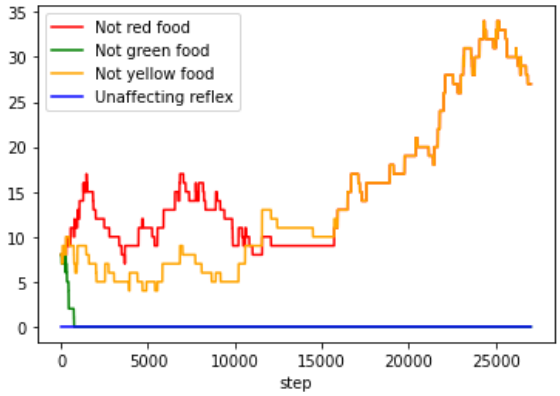


(d) The number of animats alive reproducing through asexual reproduction.

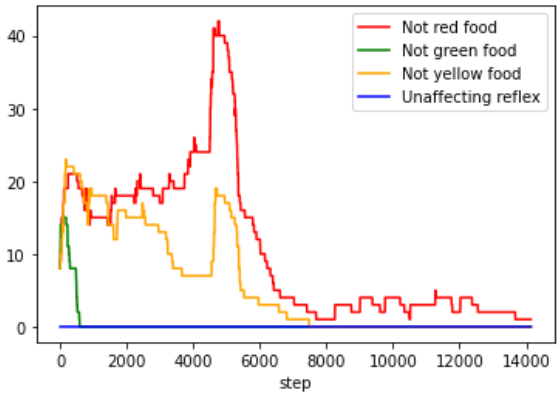
Figure 5.8: Amount of animats surviving in the lethal environment

As seen in Figure 5.9, there is now a clear difference in the population dynamics of the asexual animats and the sexual animats. As previously, the animats with the gene forbidding the eating of the good food (green), instantly starve to death. The sexual animats show a stable growth in population and the prevalance of important reflexes. After roughly 16 000 time steps, all animats in the sexual population have evolved to have both the reflex forbidding eating red lethal food, and the reflex forbidding eating yellow lethal food.

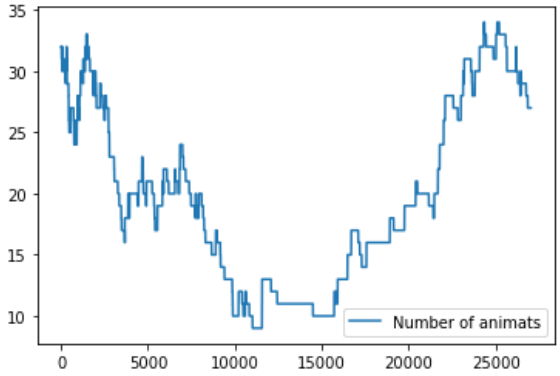
The asexual animats show the same tendency as before to reproduce more often than their sexual counterparts. However, the likelihood of surviving long enough to reproduce and to mutate the second lethal reflex for an offspring is very small. By looking at the peak in population at around 5 000 time steps, we see that out of the roughly 65 animats, around 40 animats have the red lethal reflex and around 20 have the yellow lethal reflex. There is no overlap between the reflexes which are needed for near-guaranteed survival, and this explains the following fall in population and finally the extinction of the asexual animats.



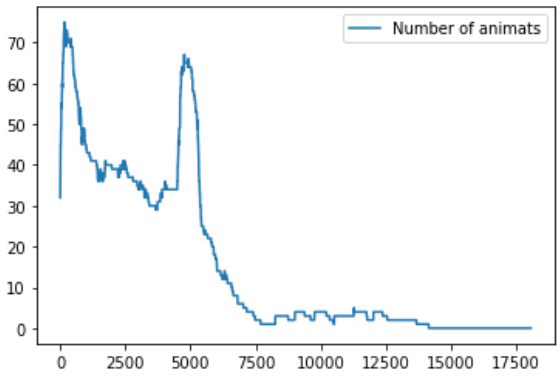
(a) The number of reflex genes in the animats alive, reproducing through sexual reproduction.



(b) The number of reflex genes in the animats alive, reproducing through asexual reproduction.



(c) The number of animats alive reproducing through sexual reproduction.



(d) The number of animats alive reproducing through asexual reproduciton.

Figure 5.9: Amount of animats surviving in the double lethal environment.

5.2.2 Grass & Dandelions

When running an environment with only grass and dandelions, the dandelions manage to conquer the environment and ultimately eradicate the grass. This is due to the dandelions' superior hostile radius, restricting grass' spread more than grass restricts the spread of dandelions. After only a few hundred time steps, the dandelion population outgrows the grass population and then slowly prevents the grass from spreading any further as observed in Figure 5.10.

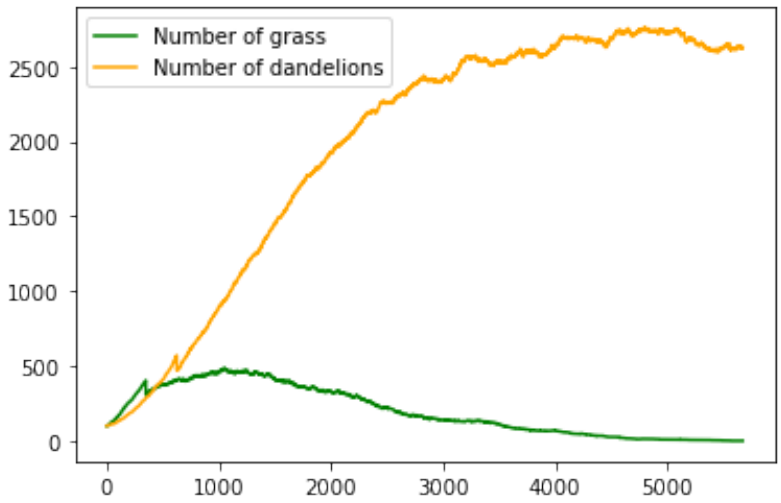


Figure 5.10: Competition between grass and dandelions.

When the goats are inserted into the environment, we instead see the opposite effect of competition between dandelions and grass. This is due to the fact that the grass plants grow from the roots and may grow back after being consumed. The dandelions which lacked this protection quickly died out due to the herbivores consuming enough of them which both limits the spread dramatically, and lets grass cover more areas to further restrict the dandelions' spread. Ultimately, if the herbivores consume too much of the grass too, then the species will perish as seen in Figure 5.11.

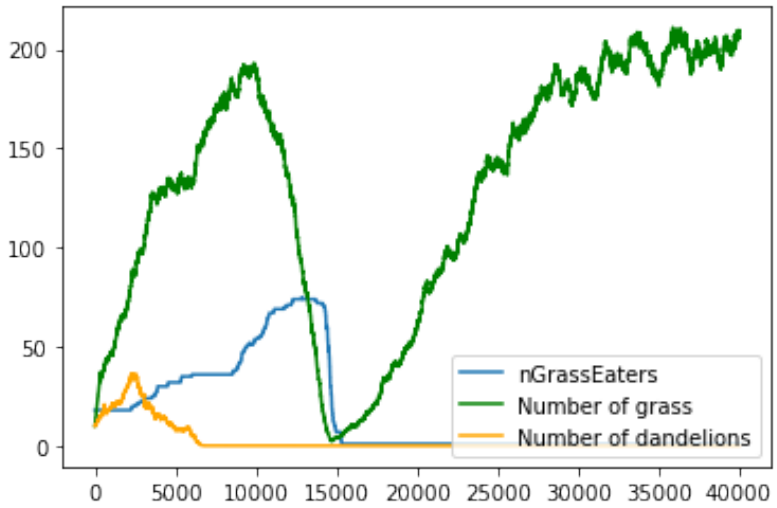


Figure 5.11: Population dynamics of grass, dandelions and herbivores. The plant populations are represented at a scale of 0.1 of their true numbers.

With the decline of the grass population, and the starving prey population, it is possible for the dandelions to make a recovery in the ecosystem. As the dandelions can spread further than the grass, the dandelions are able to quickly spread throughout the whole ecosystem and thus anew constitute a food source for the

herbivores. This way, we can observe oscillating populations resembling those of oscillating Lotka-Volterra equations. Such a result can be seen in Figure 5.12, however this simulation differs from that displayed in Figure 5.11 in that a small amount of plants are assumed to immigrate. With this assumption, the very last dandelion could be consumed by the goats before reappearing and thriving amidst the starving animats. However, this assumption is not enough to guarantee the survival of the herbivores and the oscillating nature of the populations. Which situation arises depends on the stochasticity of the experiments.

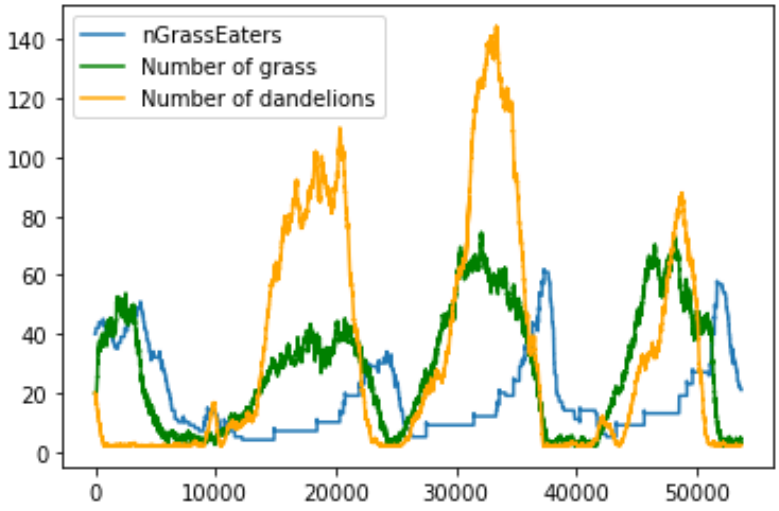


Figure 5.12: Population dynamics of grass, dandelions and herbivores. The plant populations are represented at a scale of 0.1 of their true numbers.

Generating stable ecosystems with the advanced plant models, herbivores and carnivores proved a difficult task. The ecosystems require careful parameter-tuning, which may still result in unstable environments. This is clear from Figure 5.13 where the predators have become too good at hunting, resulting in them eliminating the prey population.

Even though the predators become too skilled in hunting in some ecosystems, the opposite effect shows in others. In Figure 5.14 the prey has become well adapted to its environment and are able to resist the predators, thus removing any interesting population dynamics between the prey and predators. Instead we observe population dynamics similar to what is presented in Figure 5.12 between the herbivores and the plant species.

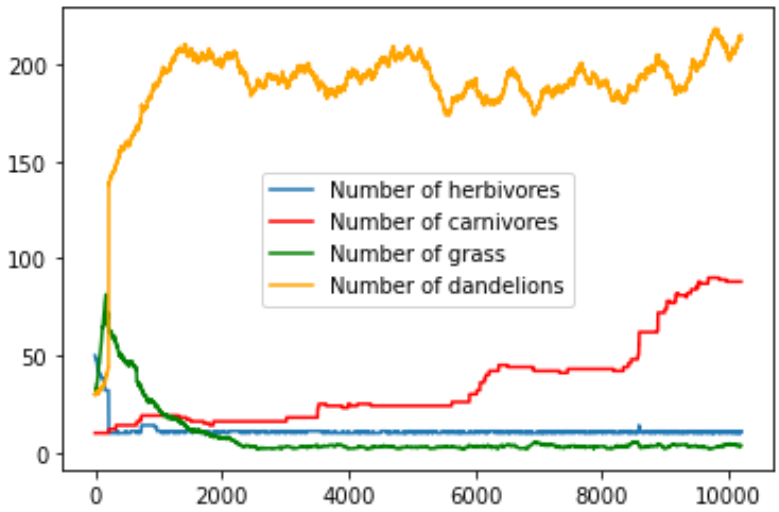


Figure 5.13: Population dynamics of grass, dandelions, herbivores and carnivores with a dominating carnivore population

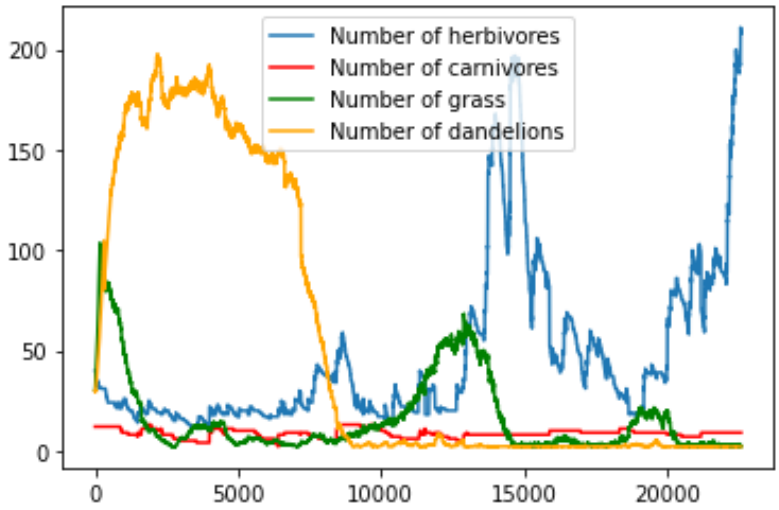
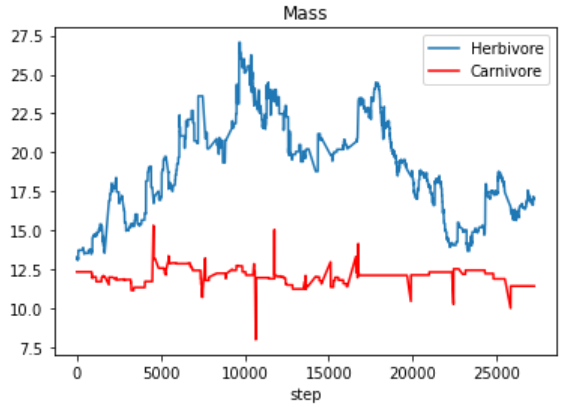
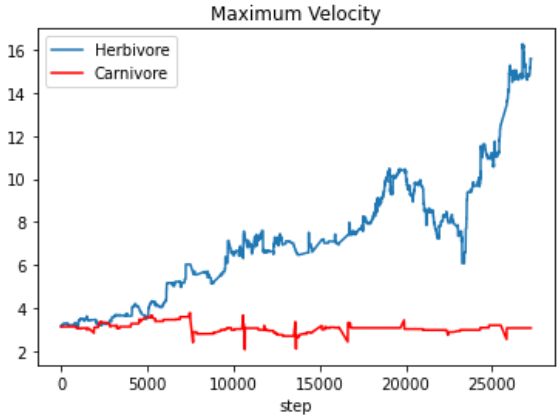


Figure 5.14: Population dynamics of grass, dandelions, herbivores and carnivores with a dominating herbivore population

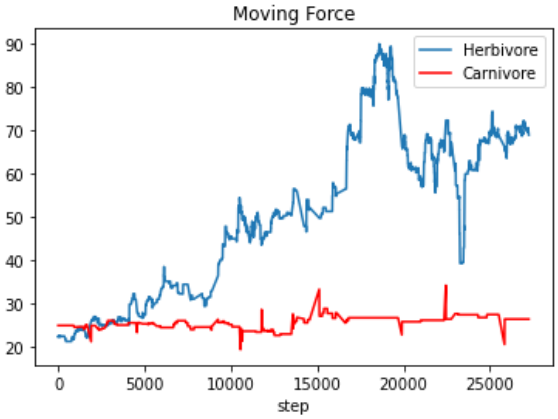
The success of the prey's survival is most likely due to its well adapted attributes seen in Figure 5.15 which have managed to evolve over time. The increase in mass makes it incredibly difficult for the predators to kill the prey as the attack damage done by the predators scale based on the proportion between the mass of the predators and the prey. Later in the simulation, maximum velocity and maximum force increase making it even harder for the predators to capture the prey as the predators will not be able to outrun the prey. When velocity and acceleration (moving force) have increased enough for the prey to outrun the predators, the mass decreases once again. The most probable cause for this is that as mass is no longer directly needed in order to survive, the prey evolves a lower mass in order to save energy and adapt to an environment with a scarce food supply.



(a) Genetically inherited mass compared between herbivores and carnivores.



(b) Genetically inherited velocity compared between herbivores and carnivores.



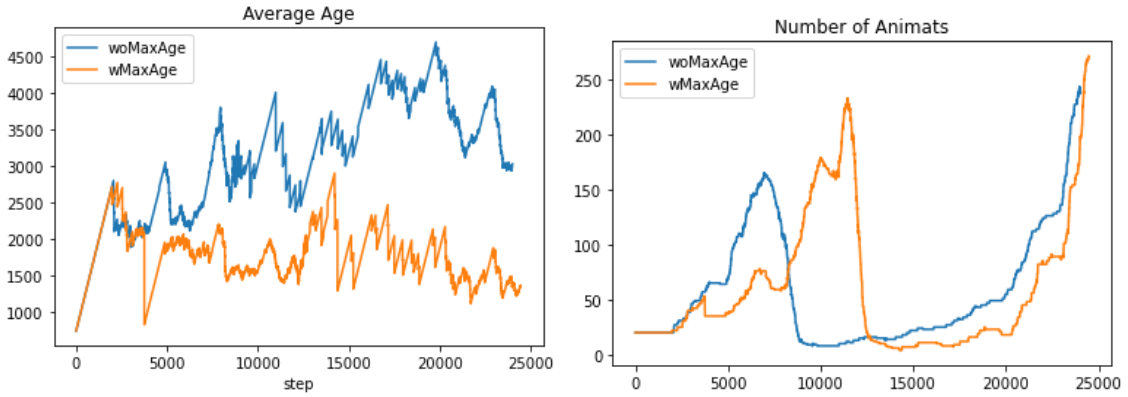
(c) Genetically inherited moving force compared between herbivores and carnivores.

Figure 5.15: Genetically inherited and mutated attributes of herbivores and carnivores.

The final data we analyze is how the age of the animats affect the species' evolution. In all prior experiments the animats could theoretically live an infinite number of time steps. The only factors to limit animats' lifespans were:

- Predation
- Starvation
- Lethal food

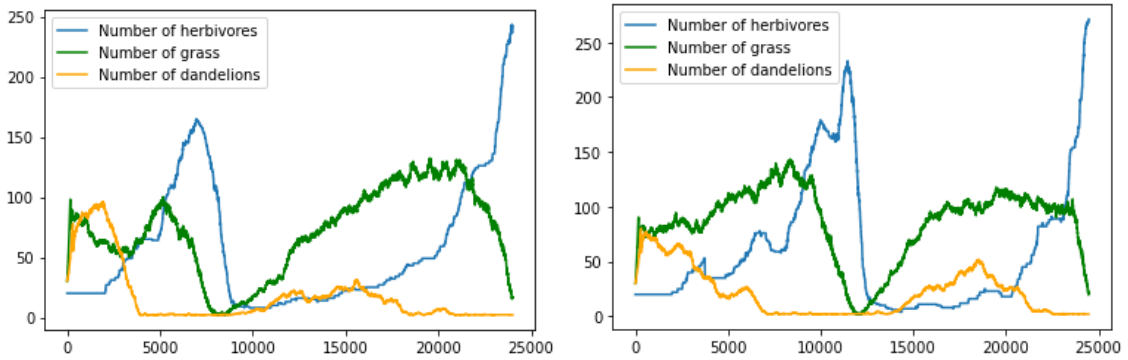
Our reasoning for introducing a maximum age for the animats is to inspect how death affects evolution. In Figure 5.16 and Figure 5.17 we can see how differently the same animat species performs when restricted or unrestricted by a maximum age. The simulations resemble one another on the whole, but show some slight differences in the genes in Figure 5.18.



(a) Average age compared between herbivores with- and without a maximum age limit.

(b) Number of animats alive compared between herbivores with- and without a maximum age limit.

Figure 5.16: Average age and population of herbivores with- and without an age limit



(a) Population dynamics in simulation without maximum age.

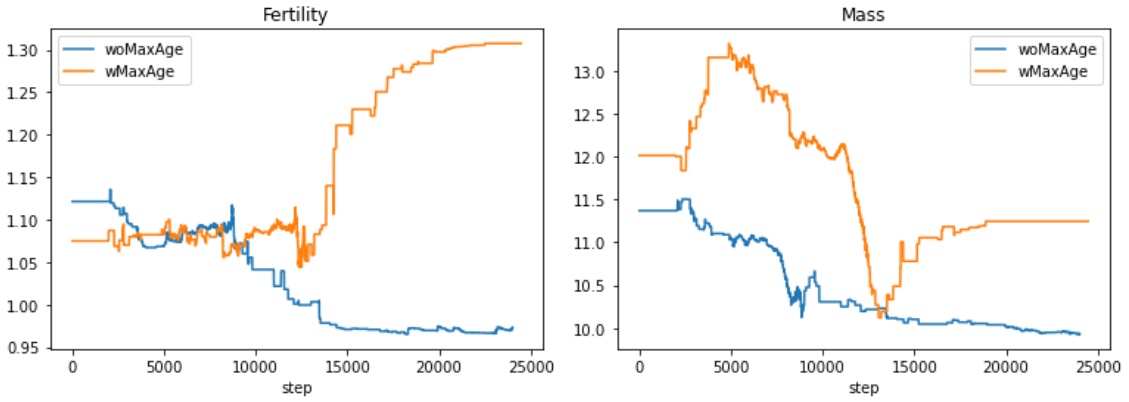
(b) Population dynamics in simulation with maximum age.

Figure 5.17: Population dynamics in simulations with- and without an age limit

As seen in Figure 5.17, when the animats are restricted by a maximum age, the population's growth during the mating seasons is not monotonic. The unrestricted animat population sees only growth from its new offspring. On the other hand, due to the value of the maximum age, the restricted animat population faces times of a loss in population – despite the presence of food. Regardless, the food is depleted equally and leads to the near-extinction of the species in both simulations. The second peak in populations also indicates that the times of decrease in population does not correlate with a higher maximum species population.

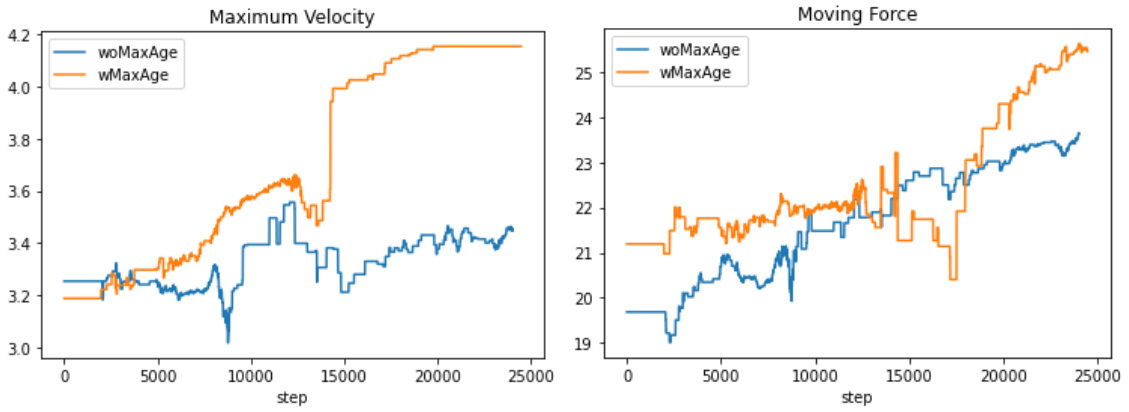
For comparison, below in Figure 5.18 a number of genes from the two runs can be observed. Although the genes are quite different, this may be due to stochasticity and is not necessarily due to the introduction of a maximum age. The sharp

increase/decrease in the genes of the restricted animat population around time step 14 000 can be explained by the large death toll, thus increasing the impact of the genes of a select few animats.



(a) Genetically inherited fertility compared between herbivores with- and without a maximum age limit.

(b) Genetically inherited mass compared between herbivores with- and without a maximum age limit.



(c) Genetically inherited velocity compared between herbivores with- and without a maximum age limit.

(d) Genetically inherited moving force compared between herbivores with- and without a maximum age limit.

Figure 5.18: Genes in run with herbivores compared with genes in run with herbivores restricted to a maximum age.

However, by looking closer at the fertility gene, this gene faces near exclusive growth leading up to the second population peak. The other genes seen in Figure 5.18 face both ups and downs. With a higher fertility, the species is more likely to remain in large numbers, despite a large death toll due to age. Fertility is also directly linked to the success of passing on genes to coming generations, and might suggest that death plays a role in evolution.

6
Discussion

6.1 Three-species Population Dynamics

Another master's thesis from our research group [21] focused on analyzing the population dynamics in three-species predator-prey systems. The results from those simulations showed that the multiagent-based ecosystem simulations of predator, prey and food may exhibit cyclic population dynamics similar to cyclic three-species Lotka-Volterra equations. However, our focus on implementing a more realistic reproduction mechanism based on animats' decision-making makes it very difficult to balance parameters to recreate similar results. Although Lotka-Volterra equations may display stable population dynamics, it is equally possible to express deteriorating ecosystems using Lotka-Volterra equations.

The failure to reliably reproduce cyclic population dynamics for three species as described by stable Lotka-Volterra equations may thus be due to the parameters used in our simulations. An example of cyclic population dynamics between predator, prey, and grass, can be seen in Figure 6.1. Despite the example of a stable ecosystem, population dynamics can be analyzed even in unstable ecosystems. Furthermore, there is nothing to stop the term "three-species" from designating several species on the same level in the ecosystems food-chain. Hence, we choose to discuss the population dynamics of grass and dandelions in our environments.

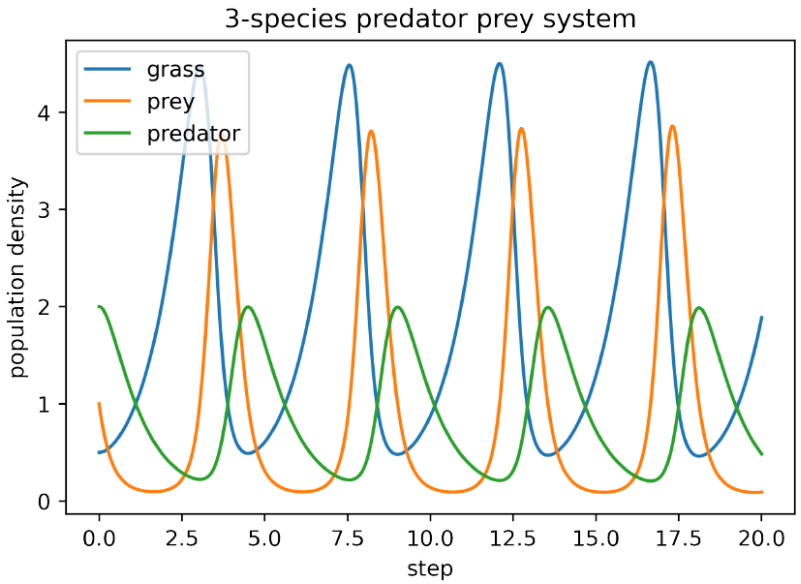


Figure 6.1: Example of 3-species Lotka-Volterra indicating cyclic population dynamics for predator, prey, and food (credit to Karlsson (2021) [21] for figure).

The competition between grass and dandelions as seen in Figure 5.10 is extremely one-sided and does not correspond very well to the behaviour of lawns in real life. Although dandelions do spread relatively quickly and can quickly be seen spreading in all parts of a lawn, grass is very rarely completely outcompeted. By making use of an artificial winter which limits spread, and making grass partially resistant to this winter, we make an attempt at stabilizing our environment. The resulting interaction between the species can be seen below in Figure 6.2.

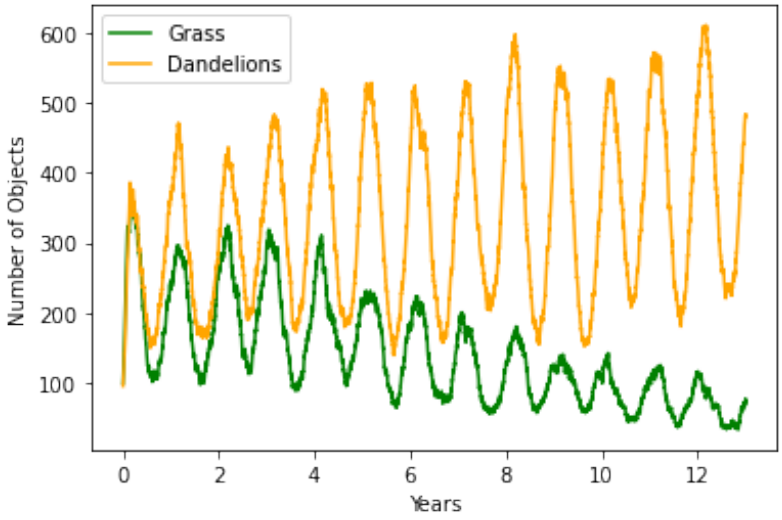


Figure 6.2: Example of the population dynamics of plants in the absence of prey. One year is modelled as 600 timesteps, with winters reducing plant spread by 55%.

The change in spread depending on seasons is implemented by the simple

$$spread = (1 - \sin(t/t_{year}) \cdot w) \cdot (1 - resistance) \cdot spread_{base} \tag{6.1}$$

where w designates by how much the spread will be reduced. As can be seen in Figure 6.2, this greatly helps the survival of grass in the environment, creating a more stable ecosystem. However, seeing as the dandelions faced extinction with the introduction of prey in the ecosystem, understandably the addition of winter does not help in protecting an already endangered species.

The fragile nature of our environments can be seen in subsection 5.2.2. Due to stochasticity, our herbivore may or may not push our plant species to extinction (see Figure 5.11 and Figure 5.12 for comparison). This goes to show that ecosystems can be extremely fragile, and in our case the design of our plants may not support a herbivore very well. The quick decline of dandelions in the presence of our prey species could be indicative of how a stable ecosystem may behave when a new invasive species is introduced to the environment (the environment seen in Figure 6.2 can be seen as relatively stable as the decline of grass is over more than a decade of in-simulation time). With our simulations, which produced arguably unstable ecosystems, we wish to highlight the importance of continued research into the topic in order to protect nature's unstable ecosystems.

It should be noted that our grass & dandelions environment demonstrated two-species population dynamics (see Figure 5.12) in the runs where stochasticity did not cause the extinction of the plants. As our dandelions and grass are on the same level in the food-chain, we do not claim that our model demonstrates three-species Lotka-Volterra population dynamics, as this might be misleading. Yet, we wish to highlight our results demonstrating similar population dynamics using the prey species and two different food populations.

6.2 Reproduction and Evolution

In order to get simulations which supply results within a somewhat reasonable time, a lot of assumptions were necessary. Evolution is very simplified in our model. We have referred to a gene in this paper as some variable that an animat has which can be mutated and inherited and corresponds to some attribute. However, a single gene might affect multiple attributes, or one attribute might be affected by multiple genes. This means that creating a fully realistic model of evolution would also mean doing a fully realistic model of an animal's genetic code which is not feasible in a simulated ecosystem (at least not with the computers of today).

When comparing many other papers on simulating population dynamics, assumptions are made that new agents are generated statistically, either with a set probability or based on the current population. In order to create more realistic simulations, or at least simulations that are more true to nature, it is inevitable to model reproduction between animats instead of having the environment create new agents. One

of the more important takeaways from this paper is that agent-based reproduction and inheritance appears to work very well for reinforcement learning agents and proposes a good and realistic approach to scaling a population in a multiagent-based model.

6.3 Reflexes

Based on the results from the experiment with lethal food and the figures 5.5 and 5.6 it is quite safe to state that reflexes are necessary. This is however not true for all environments as it depends heavily on what assumptions are made. Many ecosystem simulations are not complex enough so that reflexes that control locomotion matter, as locomotion is often assumed to depend on one single action such as "walk forward". Whether or not it is realistic to have a reflex not to eat a certain amount of food, the test case is a proof of concept that the reflex network we designed can work for multiple purposes. For one, it could simulate a reflex similar to the diving reflex, which prevents animals from breathing underwater and therefore reduces – the otherwise great – risk of drowning. Secondly, it could be used to model instincts and innate behaviours, similarly to how a dog may shake to get rid of excess water in its fur or how a baby seagull may pick its mother's beak for food.

However, there are more usages for the reflex network than this. For one, we showed during the pre-training that the reflex network can be used to help the agent explore specific states, possibly helping the agent to learn sought-after behaviors as seen in Figure 5.3. Furthermore, the reflex network can be used to create fully deterministic agents. By always forcing actions based on the observation, one could create an agent that is, either for its whole lifetime or for a limited time, controlled only by the reflexes triggered by the agent's observations.

6.4 Reward Balancing

A part of this thesis was to incorporate the reward network provided from one of the other master's thesis groups working on the same project. We used the reward network in our project to balance the reward between the agents' critical energy level and their libido. Even though the libido was not a critical need directly necessary for the individual's survival, most animats managed to balance their energy level and their libido. This shows that balancing multiple needs is possible with this reward model and could possibly be used to balance many more than two needs.

6.5 Asexual- vs Sexual Reproduction

Analyzing the experiments run in the first lethal food environment (described in subsection 4.3.3), we see clearly that the asexual animats performed better as a species. This can be explained by the ease of reproducing asexually as opposed to reproducing sexually. Considering that a large portion of the initial animats starve

due to the reflex avoiding good food, the sexual animats have some difficulties in finding mates quickly. The asexual animats have no such problem and therefore reproduce faster. However, due to the faster reproduction of the asexual animats, the animats face multiple phases of starvation after consuming a great portion of the environment's good food.

With the higher population and the starvation phases observed, an animat without the reflex for avoiding bad food is more likely to die than an animat with this reflex (as needing to learn to avoid bad food will bring the animat closer to starvation). In such circumstances, statistically over time the number animats with the secondary gene will grow whereas the number of animats lacking this gene will decrease (due to the starvation). Thus, by bringing the species closer to the environment's carrying capacity, the asexual animats are able to better develop genes suitable to its environment.

The sexual animats on the other hand, do not face the same level of internal competition for food as the species does not get as close to the environment's carrying capacity. As only the "lethal food gene" is vital for the survival of animats in the environment, this explains why the "bad food gene" does not spread throughout the population as quickly as in the asexual population. However the population grows more steadily than that of the asexual animats, and never runs into any starvation phases.

These simulations suggest that asexual- and sexual reproduction help animals adapt to their environments in different ways. Asexual reproduction may rely more on the ease of reproducing and survival of the fittest compared to sexual reproduction which may more safely guarantee the survival of the species through reduced internal competition.

When the animats are introduced to a harsher environment containing two lethal food species, the asexual animats face too much risk of dying, and their technique of high reproduction and internal competition is ineffective in finding a fitting genotype. The sexual animats' performance is almost unaffected, instead adapting to its environment as previously. Whether a species is more likely to perform better using asexual- or sexual reproduction is thus very dependent on its environment. If the environment faces a far greater threat to an animat's survival, sexual reproduction is more beneficial. If internal competition faces a nearly equal threat to an animat's survival, then asexual reproduction may instead be more beneficial for the well-being of the species.

6.6 Limitations

One of the biggest limitations of this project has been the game engine Unity on which the project is built. After all, building the ecosystem simulator in a game engine has both advantages and disadvantages. While we can observe what happens in a simulation in real-time and generate pretty environments using Unity's physics

engine, we are limited by the computation speed possible from running environments in Unity. Even if it is easier to spot errors or observe whether the environment is performing as expected, running longer simulations takes a lot of time. This was proven in the advanced plant model environments.

An advanced plant model environment and a larger animat population environment performed well for themselves, but when utilizing an advanced plant model environment with a larger population of animats the simulation became increasingly slower. But for any environment with many reinforcement animats (more than 100) the simulations would be incredibly slow. This becomes extremely limiting when researching population dynamics and reproduction as some species may require large populations and frequent reproduction.

6.7 Ethics

A great risk would be misinterpreting the results given from the simulations that can be provided from this framework, especially given the state that it is in now. If one would make conclusions from the simulations and act thereafter it could have dire consequences. It is very important to be reminded that even if we intend to make simulations which are as realistic as possible, there are still many assumptions being made. Furthermore, there is stochasticity not just in the environments that can be provided from this framework, but also in nature. Therefore conclusions from these kinds of simulations should be made very carefully.

An ethical dilemma much further down the line is whether it would be ethical to run simulations if the animats would be too intelligent and considered to have feelings. Assume there comes a point where an animal could be modelled perfectly digitally such that everything the animat could feel or think would be the same as for a real animal. Would it then still be more ethical to perform tests on the animats? Additionally, if we could model humans perfectly, does that mean that we should, or would we only cause our digital cousins suffering?

6.8 Future Work

As for the future work of this project there are a lot of ways to improve upon the already proposed ecosystem simulator. In our reproductive model we have not considered a pregnancy period during which an animat fetus needs to develop prior to its birth. This is a very important aspect of mammals however, and the research into how population dynamics are affected by pregnant animats is so far an untouched topic. Similarly, the instant births used in our simulations can be replaced with eggs which must be brooded and hatched.

Secondly, another natural step is to increase the amount of needs an animat needs to regulate to survive and/or receive a high happiness. An example of this could be through the introduction of water and thirst, limiting the amount of time the

animats can spend on foraging which might allow populations further down the food chain to more easily recover.

Finally, there is the possibility to model more complicated terrain. In our implementation, all movement is done on a 3-dimensional plane. This means that movement is essentially in 2D. However, there is nothing hindering the introduction of real 3D terrain. With the introduction of terrains such as mountain ranges or deserts, environments could be modelled to support different climates and enable a certain set of actions only in specific conditions.

Although we cannot be sure which direction this area of research will take following our work, the animat model and the ecosystem simulation project we have worked on is planned to continue development. Further information about the project can be found on www.ecotwin.se.

7
Conclusion

7.1 Research Questions

7.1.1 Is there a purpose to death?

Does death cause faster evolution?
If an animat lives for a longer time, it is able to pass on its genes for a longer time. An animat from an older generation is more likely to have genes which are less adapted to the current environment. Let us assume that a prey possesses genes which cause slow movement, and predator numbers have recently increased, then the prey's genes are ill-suited for survival. As predators in our model directly live off prey, it would therefore be preferable for the prey species that the older animat be stopped from passing on such genes in order to reduce the likelihood of another slow animat being born. Otherwise, the predators would have two potential targets (the parent and the child) as opposed to one.

In our experiment using maximum age in our herbivore, we found that the fertility gene increased far more than in the fertility gene of the animat unrestricted by age. The other genes which showed increase or decrease may be explained by needing to consume more energy to move due to a higher mass, or needing to move faster to compete with other animats. However, fertility has no direct effect on an animat's survival. On the species' survival however, fertility is highly important, and with natural deaths caused by a high age we suspect that the higher fertility is linked to the shorter lifespans. As a higher fertility will require mating fewer times to procreate, this analysis can hence be used to conclude that death does cause faster evolution.

Is death particularly important in changing environments?
The maximum age experiments shown in Figure 5.16 and Figure 5.18 assumed a maximum age of 4 500 time steps for the animats in the Grass & Dandelions environment. As there are no threats to the animats, introducing death due to age should pose no risk of extinction to the animats. If instead we consider the lethal food environment and the results seen in Figure 5.8 or Figure 5.9, given that our initial animats are created at an age of 750 time steps, this means that all initial animats would have died at the 3 750 time step mark (leaving 5-10 animats alive) if the experiment had been done using a maximum age. With the difficulty that the animats have of sexually reproducing when only a few animats persist in the

ecosystem, running simulations with the same maximum age settings would greatly increase the risk of extinction.

As only one of our environment designs supports introducing a maximum age, we thus have no frame of reference. The food spread in lethal food environment does not cause as much change to the environment as the Grass & Dandelions environment. We are thus unable to conclude whether death is particularly important in changing environments because we *only* investigate this property in changing environments.

7.1.2 (A)sexual Reproduction

> Is sexual reproduction more advantageous for survival in some environments and asexual reproduction in others?

Whether sexual or asexual reproduction is more advantageous for a species is a very difficult question to answer. Animals in nature have adapted to their surroundings in one way or another and have after millions of years come to a point where they either reproduce sexually or asexually. As our environments lack the complexity needed to simulate different predators faced over thousands of years and changing climates, our conclusions should not be taken as absolute truths.

Our conclusion, based on running the lethal food environment (see subsection 4.3.3) with asexual animats, is that sexual reproduction more effectively passes on instincts which are needed to survive – represented using reflexes in our work (see subsection 3.1.8). The danger of the lethal food environment is that there are multiple ways an animat may die: by eating lethal food, by eating multiple bad food, or by starvation. This means that the gene which forbids the eating of good food will doom an animat from birth.

Out of 40 initial animats, 10 of these are born with this deadly gene and will starve before the mating season. This is the same starting condition as the simulation run with sexually reproducing animats, however when reproducing the asexual animats are only able to pass on one useful gene to their offspring. The sexual animats may pass on two useful genes (the genes which forbid the ingestion of bad and lethal food). As there is never any combination of two parents' genes, the animats are equally likely to mutate a new gene which restrict the ingestion of good food as they are likely to gain the second useful gene through mutation.

Despite this, asexually reproducing animats are able to efficiently spread both the vital gene, *and* the non-vital gene, through increased internal competition and survival of the fittest. However, by increasing the danger level of the lethal food environment – by adding a second lethal food type – we find that the lack of all critical reflexes in the asexual animats puts the species at a much greater disadvantage than sexually reproducing animats.

The asexual animats in our experiments were thus more successful at passing on advantageous genes ("do not eat bad food"), but were unable to pass on all vital

genes ("do not eat lethal food"). From our experiments we can thus conclude: sexual reproduction is more advantageous for survival in dangerous environments. In safer environments we instead find that asexual reproduction develops a set of genes adapted to its environment at a faster rate.

7.1.3 Learning and Evolution

> Does a combination of learning and evolution make survival in dangerous environment more likely?

As can be observed in Figure 5.5 and Figure 5.6(b), when faced with an environment containing lethal food, all animats without the gene preventing the ingestion of lethal food die. Evolution is thus essential for the survival in a dangerous environment. Furthermore, by analyzing the prevalence of other reflexes in the population, we can see that a gene which affects the animat in no way whatsoever is equally present as a gene forbidding the ingestion of bad food. This shows that although the population has evolved to avoid eating lethal food, it is nonetheless necessary to learn the behaviour of avoiding bad food in order to survive. If animats were unable to learn to survive, this means that the "bad food gene" would be present in the whole surviving population, however as this is not the case, this means that animats are able to learn how to survive whilst still having the ability of eating bad food. From our lethal food experiment we can thus say with confidence that a combination of learning and evolution does make survival in dangerous environments more likely.

7.2 Contributions

In this work we have foregone the discrete grid-based environments often used to more realistically reflect the wide range of movement animals are capable of. As expected, the computational performance is not very well reflected by this improvement. Nevertheless, our project did not focus on optimizing simulation run-times and as such it is very possible that the environments can be improved and built upon for future research without the fear of computational bottlenecks.

Additionally, the core of our work concerned the reproduction of animats and the inheritance of genes. We have implemented a working mechanic for (a)sexual reproduction which depends on homeostasis and decision-making. By connecting sex to libido and only rewarding animats for regulating these libido levels, we can limit mating to seasons and limit animats from mating continuously. For reference, when the animats were offered an extrinsic reward for mating, they would develop a behaviour which promoted mating until starvation and death. With our model for reproduction and evolution, animats are able to pass on reflexes which offer their offspring an advantage for survival, and allow the population to develop attributes which provide the best chances for survival.

During this project, a secondary focus was made on creating a food source for the prey species which is more realistic than previous static implementations. The advanced plant model described in section 4.2 proposes a simple compromise between

realism and computational complexity. With a goal of creating a dynamic plant spread resembling that of a few select plants, the idea of investigating three-species Lotka-Volterra dynamics came naturally. Nevertheless, this goal proved extremely difficult to attain. Karlsson (2021) [21] estimated the impact of the parameters used in his experiments and their corresponding values in Lotka-Volterra equations in order to generate the desired population dynamics. Our model contains a large amount of additional parameters however, both due to the evolutionary nature of our animats and due to the more advanced plant implementation. We do not believe it to be impossible to recreate cyclic population dynamics observable in Lotka-Volterra, however we are of the belief that such a task requires a large amount of parameter-calibration.

Bibliography

[1] Lacetera N. "Impact of climate change on animal health and welfare" In: *Animal Frontiers 9.1* (2019), pp. 26–31. DOI: 10.1093/af/vfy030

[2] Jackson J.B.C. et al. "Historical Overfishing and the Recent Collapse of Coastal Ecosystems". In: *Science 293* (2001), pp. 629–637. DOI: 10.1126/science.1059199

[3] Lotka, A.J. "Elements of Physical Biology". (1925). *Williams and Wilkins.*

[4] Volterra V. "Fluctuations in the Abundance of a Species considered Mathematically". In: *Nature 118* (1926), pp. 558–560. DOI: 10.1038/118558a0

[5] Maynard Smith J. & Slatkin M. "The Stability of Predator-Prey Systems". In: *Ecological Society of America 54.2* (1973), pp. 384–391. DOI: 10.2307/1934346

[6] Yang Y. et al. "A Study of AI Population Dynamics with Million-agent Reinforcement Learning" In: *17th International Conference on Autonomous Agents and Multiagent System* (2018). arXiv: abs/1709.04511v4

[7] Wang X., Cheng J. & Wang L. "A reinforcement learning-based predator-prey model". In: *Ecological Complexity 42* (2020). DOI: 10.1016/j.ecocom.2020.100815

[8] Wang X., Cheng J. & Wang L. "Deep-Reinforcement Learning-Based Co-Evolution in a Predator–Prey System". In: *Entropy 21.8* (2019), pp. 773. DOI: 10.3390/e21080773

[9] Sutton R.S. & Barto A.G. "Reinforcement learning : an introduction". Second edition. (2018). *MIT Press.*

[10] Sutton, R.S. "Reinforcement learning architectures for animats". In: *From Animals to Animats: Proceedings of the First International Conference on Simulation of Adaptive Behavior* (1991), pp. 288–296.

[11] Schulman J. et al. "Proximal policy optimization algorithms". (2017) arXiv: abs/1707.06347

[12] Darwin C. "On the Origin of Species, By Means of Natural Selection, Or the Preservation of Favoured Races In the Struggle For Life". (1859). *John Murray*

[13] Wilson S.W. "Knowledge Growth in an Artificial Animal". In: *Adaptive and Learning Systems* (1985), pp 255-264. DOI: 10.1007/978-1-4757-1895-9_18

[14] Pratt O., Gwinnutt C. & Bakewell S. "The autonomic nervous system-—basic anatomy and physiology. Update in anaesthesia." In: *Education for anaesthetists 24* (2008), pp: 36—39.

[15] Johannsen, W. "The Genotype Conception of Heredity". In: *The American Naturalist 45.531* (1911), pp. 129–159. DOI: 10.1086/279202

[16] Yamada J., Shawe-Taylor J. & Fountas Z. "Evolution of a Complex Predator-Prey Ecosystem on Large-scale Multi-Agent Deep Reinforcement Learning". (2020). arXiv: abs/2002.03267

[17] Sibly R., et al. "Representing the acquisition and use of energy by individuals in agent-based models of animal populations". In: *Methods in Ecology and Evolution 4.2* (2013), pp. 151–161. Wiley Online Library. DOI: 10.1111/2041-210x.12002

[18] Kleiber M. "Body size and metabolic rate". In: *Physiological reviews 27.4* (1947), pp. 511–541. DOI: 10.1152/physrev.1947.27.4.511

[19] Kleve B. & Ferrari P. "A Generic Model of Motivation in Artificial Animals Based on Reinforcement Learning". (2021). Master's Thesis: Chalmers University of Technology

[20] Conaway C.H. "Ecological Adaptation and Mammalian Reproduction" In: *Biology of Reproduction 4* (1970), pp. 239–247. DOI: 10.1093/biolreprod/4.3.239

[21] Karlsson, T. "Multi-Agent Deep Reinforcement Learning in a Three-Species Predator-Prey Ecosystem". (2021). Master's Thesis: Chalmers University of Technology

[22] Frisk D. "A Chalmers University of Technology Master's thesis template for LaTeX". (2016). Unpublished.

A

Appendix 1

A.1 Environment Parameters

Table A.1: Parameters used for the Lethal food experiment with only sexually reproducing animats

Environment size	75×75
Number of time steps	100 000
Number of initial food	300
Initial goats with "not red" reflex	10
Initial goats with "not green" reflex	10
Initial goats with "not yellow" reflex	10
Initial goats without reflexes	10
Goat mutation constant	0.1

Table A.2: Parameters used for the Lethal food experiment with both sexually and asexually reproducing animats

Environment size	50×50
Number of time steps	14 000
Number of initial food	200
Initial goats with "not red" reflex	8
Initial goats with "not green" reflex	8
Initial goats with "not yellow" reflex	8
Initial goats without reflexes	8
Goat mutation constant	0.05

Table A.3: Parameters used for the Lethal food experiment with both sexually and asexually reproducing animats as well as two kinds of lethal food

Environment size	50×50
Number of time steps	14 000
Number of initial food	200
Initial goats with "not red" reflex	8
Initial goats with "not green" reflex	8
Initial goats with "not yellow" reflex	8
Initial goats without reflexes	8
Goat mutation constant	0.001

Table A.4: Parameters used for the Grass & Dandelions experiment without animats

Environment size	75×75
Number of time steps	5 000
Number of initial grass	100
Number of initial dandelions	100

Table A.5: Parameters used for the Grass & Dandelions experiment without predators

Environment size	75×75
Number of time steps	55 000
Number of initial grass	200
Number of initial dandelions	200
Initial goats	20
Ratio male:female goats	50 : 50
Goat mutation constant	0.1

Table A.6: Parameters used for the first Grass & Dandelions experiment with predators

Environment size	75×75
Number of time steps	55 000
Number of initial grass	300
Number of initial dandelions	300
Initial goats	20
Ratio male:female goats	50 : 50
Goat mutation constant	0.1

Table A.7: Parameters used for the first Grass & Dandelions experiment with predators

Environment size	75×75
Number of time steps	55 000
Number of initial grass	300
Number of initial dandelions	300
Initial goats	50
Ratio male:female goats	50 : 50
Goat mutation constant	0.1
Initial wolves	12
Ratio male:female goats	50 : 50
Wolf mutation constant	0.1

Table A.8: Parameters used for the second Grass & Dandelions experiment with predators

Environment size	75×75
Number of time steps	55 000
Number of initial grass	300
Number of initial dandelions	300
Initial goats	40
Ratio male:female goats	50 : 50
Goat mutation constant	0.1
Initial wolves	12
Ratio male:female goats	50 : 50
Wolf mutation constant	0.1

Table A.9: Parameters used for the Grass & Dandelions experiment without predators but with and without a max age

Environment size	75×75
Number of time steps	25 000
Number of initial grass	300
Number of initial dandelions	300
Initial goats	20
Ratio male:female goats	50 : 50
Goat mutation constant	0.1
Goat maximum age (if any)	4 500

A.2 Animats

A.2.1 Possible Reflexes

Prey types:

- Follow Food:
  Will cause movement toward food if there is nearby food (unless already in eating distance of food)
- Not Good:
  Forbids the eating of Good Food
- Not Bad:
  Forbids the eating of Bad Food
- Not Lethal:
  Forbids the eating of Lethal Food
- Regular:
  No forced/forbidden actions: the pre-trained agent

Clarification – Follow Food:

- Forces eating food, if standing within eating distance of food
- Otherwise, if food is seen, allows moving/rotating toward the seen food
- Otherwise, allows moving/rotating in the direction of the most food smelled
- Otherwise, allows any action

A.2.2 Animat Parameters

Table A.10: Parameters used for the Goats

Initial age (unless born during simulation)	750
Male expected # of mates	8
Female time to recover after birth	9 months
Maturity age	750
Min speed	-0.3
Max speed	10
Rotation	18
Moving force	3
Mass	12
Smell radius	10
Energy move	-0.0002
BMR	$-0.003 \cdot \text{mass}$

Table A.11: Parameters used for the Wolves

Initial age (unless born during simulation)	1000
Male expected # of mates	8
Female time to recover after birth	9 months
Maturity age	1000
Min speed	-0.3
Max speed	10
Rotation	18
Moving force	3
Mass	12
Smell radius	10
Energy move	-0.00005
BMR	$-0.003 \cdot \text{mass}$
Energy attack	-0.0005
Attack force	0.33
Attack range	2

A.3 Food Parameters

A.3.1 Grass & Dandelions

	Grass	Dandelion
Time to Ripen (months)	0.05	0.075
Time to Recover (months)	0.35	N/A
Time to Decay (months)	0.70	1.25
Expected Seedlings in Lifetime	2.75	2.35
Spread Radius	0.85	3.50
Grace Radius	0.525	0.775
Hostile Radius	0.65	0.90
Winter Resistance	10%	0%
Energy Reward (At Spawn)	0.05	0.03
Energy Reward (When Ripe)	0.075	0.09
Energy Reward (When Decayed)	0.025	0.04

Clarification – Expected Seedlings in Lifetime:
The probability of spawning a new plant per timestep is found by:

$$fertility = \frac{Expected_{seedlings}}{t_{decay} * t_{year}}$$

The actual number of seedlings created can thus be higher than the expected number (see environment, for length of year).

A.3.2 Lethal Food

	Good Food	Bad Food	Lethal Food
Energy Reward	+0.2	-0.2	-1
Chance of Spawning	85%	10%	5%

Food is always spawned according to these probabilities and is independent of the types of food already present in the environment.