Building a simulation of an artificial ecosystem.

Markus Pfundstein 10452397
Koen van der Kijl 10555900
Robin Bakker 10548017
Joseph Weel 10321624

June 27, 2014

INTRODUCTION

For our research, we were inspired to investigate Artificial Intelligence from the bottom-up perspective of evolutionary biology and artificial life (Langton 1997). We wanted to identify the necessary components for a simulated world in which small organisms fight for survival. To accomplish this, the following question had to be answered: What are the necessary components of an artificial ecosystem? Darwins theory of natural selection is based on the idea of survival of the fittest (Darwin and Beer 1951). Organisms must adapt to their environment in order to gain advantages, such as better fitness or faster reproduction. We hypothesize that the minimally required components for a working simulation consist of basic behaviour, evolution and balance. To simulate evolution we have added a genetic algorithm of crossover breeding. Reflexive agents possessing five basic behaviours and multiple properties serve as the organisms.

THE WORLD

In order to answer our research question, we developed a program to simulate a world containing cells (see Figure 1). In this world cells can interact with other cells by using various programmed behaviours. There are several fundamental laws that govern the simulation:

- The world is a simulated island. Cells cannot leave this island.
- If the energy of a cell reaches zero, the cell dies.
- A cell can move or stay at its current location. Both actions will decrease energy.
- A cell can be attacked by other cells, resulting in the decrease of the target's energy level. If the target dies, the attacker will eat it and gain energy.
- A cell's DNA determines its properties such as strength, vision, and speed. Strength determines if a cell will attack another cell or flee. A higher strength value will also lead to a faster decrease of energy. Vision is the perception radius of a cell, a higher vision leads to more responsiveness. Speeds determines which cells move first. Furthermore, speed represents the amount of tiles a cell can move.
- There can be multiple types of cells, indicated by different colours.
- Two cells of the same type can mate to create a new cell. The DNA of the new cell is calculated as a function of the DNA strings of its parents.

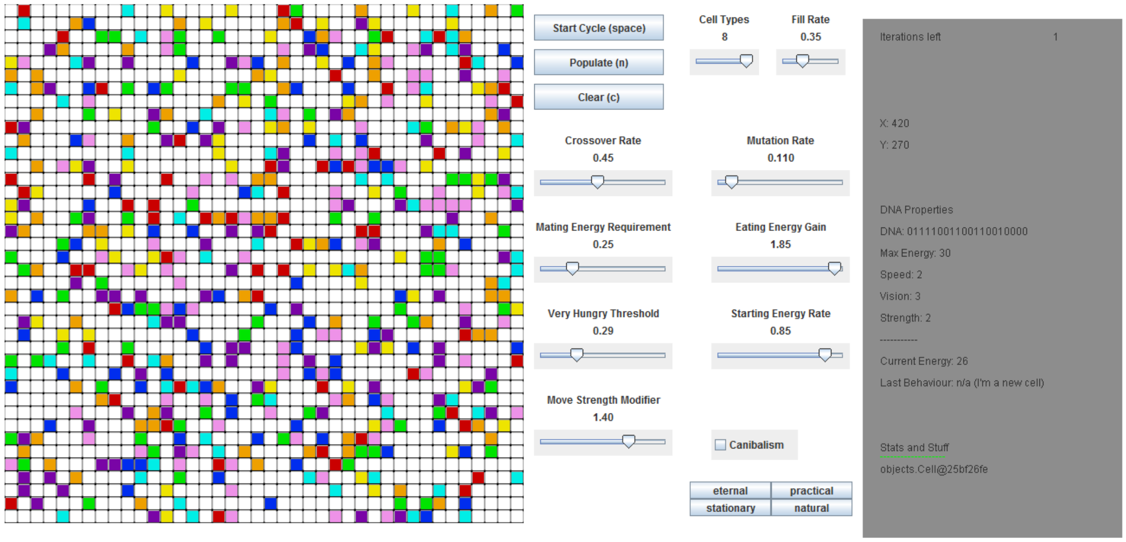

Figure 1. Interface

PROPERTIES AND GENETIC CODE

Each cell has a set of properties based on their DNA. Each property is encoded by the DNA as substrings of 0's and 1's with variable lengths. Properties such as the vision are decoded from a substring of the DNA to a numeric value by using a base 2 numeral system. For other properties that desire a low maximum value, such as sight, the value is based on the number of 1's in the substring.

When cells mate, they create two new DNA strings that are based on the DNA strings of the two cells. This process makes use of a genetic algorithm. The process takes the DNA strings of both cells and creates a new string with a chance of crossover. Furthermore, there is a slight chance of mutation in the new string. In case of crossover, the two strings take a part of the other string at a random selected bit (Mitchell 1998). If mutation occurs, some of the bits in the DNA string swap their value. This results in the creation of stronger cells.

BEHAVIOURS

Every cell is a reflexive agent: It will choose the most rewarding behaviour depending on the environment and its own state. Every turn, the agent can choose one of the following behaviours:

Mating behaviour

The cell looks for another cell of the same type. If it finds one and both cells have a sufficient amount of energy, they mate.

Hunting behaviour
The cell scours its surroundings for cells of different types. If it finds a target that it can defeat, the cell moves towards the target and tries to attack it. Famine may cause a cell to turn into a cannibal, causing it to attack cells of its own type. Cells may also cooperate with other cells of the same type to take down stronger cells.

Fleeing behaviour
The cell scours its surroundings for stronger cells. If the cell detects any dangerous cells, it tries to position itself as far away as possible from them.

Wandering behaviour
The cell moves to a random non-occupied tile within its moving radius.

Staying behaviour
The cell holds its position. Even if it does not move, its energy is decreased.

OBSERVATIONS

In order to find the best set of values concerning the worlds variables, we ran several simulations with different values. We noticed that mutation rate and energy requirement for mating have the biggest influence on the population. Furthermore, the cannibal toggle and energy gain when eating also have a noticeable effect on the development of the cells.

Mutation

The mutation rate determines the probability of change in the cell's DNA sequence. This can have tremendous effects on the properties of a child cell.

On the assumption, that a cell's initial bit value for strength is 0010, corresponding to a real value of 2, a change of the most significant section would result in a new strength value of 1010 and thus create a cell with a real strength of 10. When this cell breeds, all subsequent children have a high chance of inheriting this strength value. Therefore, the mutation rate has a huge influence on the development of the world. We also want to note that mutation is important for keeping the variation in DNA among cells steady at a high rate. If the mutation rate is set to zero, the variations will converge. Thus, the DNA of the cells will end up equal (see Figure 2 and Figure 3).

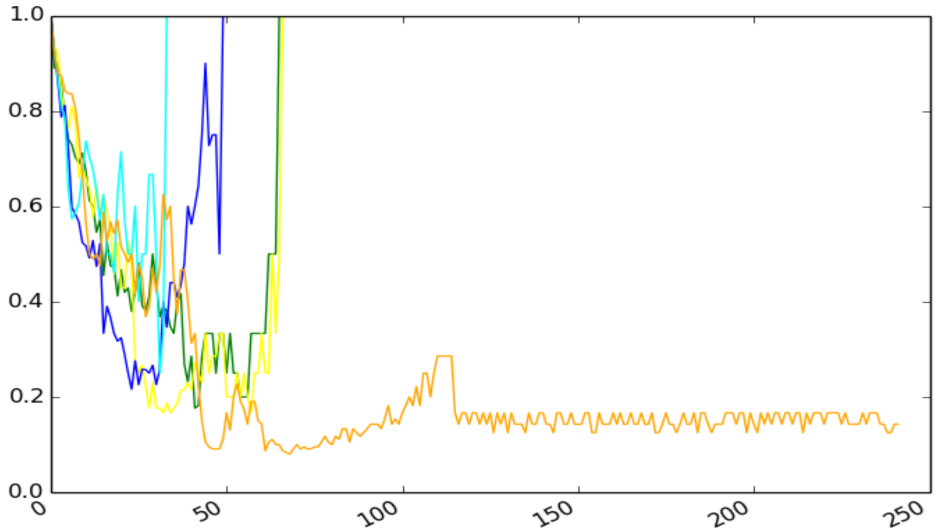

Figure 2. DNA variation without mutation (1.0 = 100% variation)

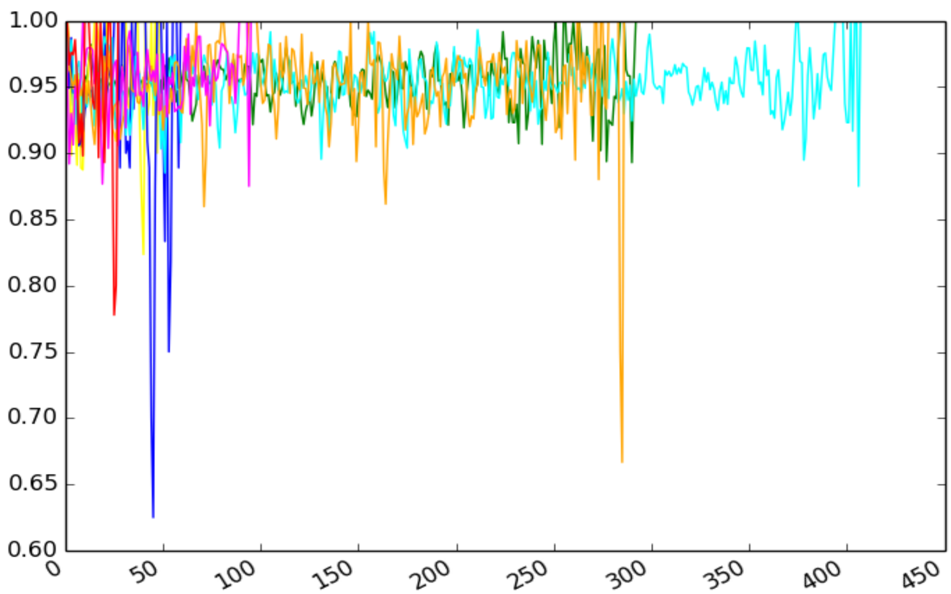

Figure 3. DNA variation with mutation (1.00 = 100% variation)

Mating requirement
Our initial idea was to let cells lose more than half their energy when mating. This way, cells have to eat and regain energy before mating again. We soon discovered that this method leads to rapid extinction. To overcome this, we ran several simulations to find the best mating requirement.
When testing with a requirement of 30%, all cells went extinct in about 40 turns. When we decreased the requirement to 25%, the amount of turns for complete extinction went up to 100. We found 21% to be the highest acceptable value. Here we observed initial rapid extinction with at most 2 or 3 surviving cell types. These cells then expanded and had to fight for control of the world. After 150 to 200 turns one type would usually evolve in such a way that it could kill all other types. Then it would go extinct due to a lack of food. Of course the amount of energy cells gain when eating also influences this result. (see Figure 4).

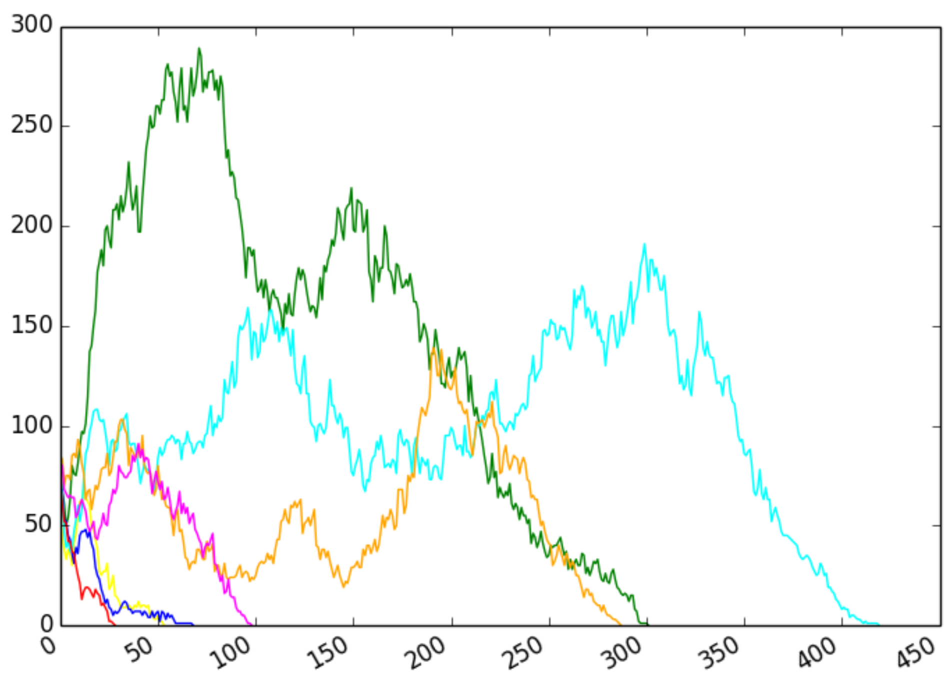

Figure 4. Live cycle cells in balanced ecosystem

Energy gain from eating
The energy gain from eating shows what percentage of the eaten cell's energy is added to the attacker's energy. If this number is high , the cell with the highest

strength will spread quickly due to a high amount of prey. The easy gain of energy can then be used to mate, resulting in the fast spread of his type.
In one test we turned the energy gain up to the 200%, which is the highest amount possible. Here we saw the strongest cell taking over in 130 turns. In a second test with equal variables, we turned the energy gain down to 50%. After 130 turns, there were still 3 types of cells alive and competing for domination of the board. It wasn't for another 100 turns before there was a victorious type.

Cannibalism

If cannibalism is toggled on, cells will also try to eat their own type if their energy levels get below a set threshold. It is possible to create a *perpetuum mobile* by giving the cells a low mate requirement and a high energy gain from eating. In this situation, a cell would mate until hungry, eat one of its babies and continue mating. In doing so, the cells are spawning more cells than they kill, effectively filling the world (see Figure 5).

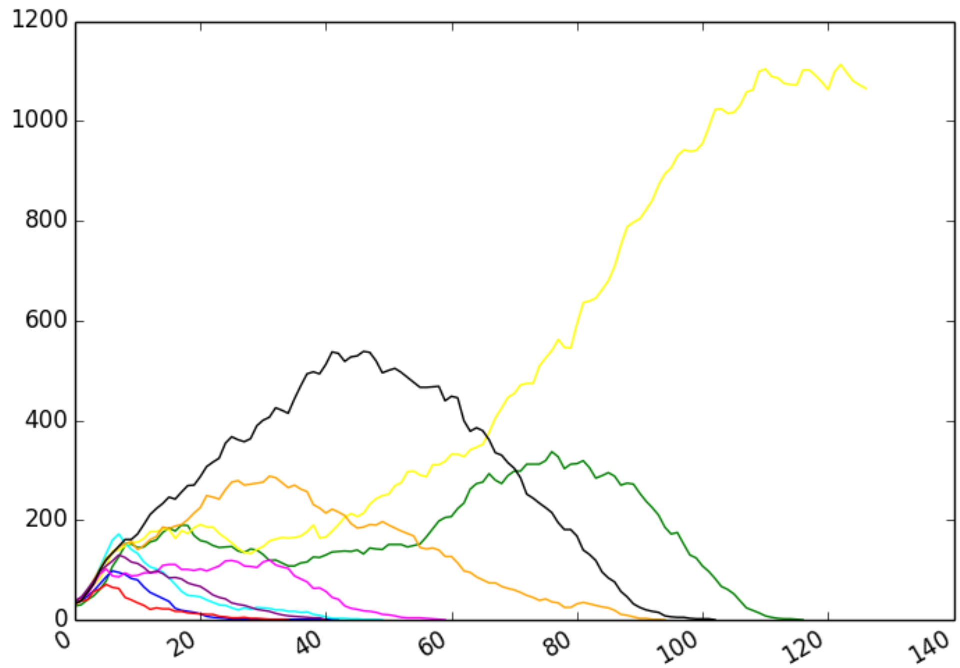

Figure 5. Live cycle cells in ecosystem with cannibalism

THE FUTURE

The simulation was implemented into an easily extensible framework, which means that new laws can be added without having to know the complex inner workings of the simulation. New behaviours, cell types and DNA properties are easy to add. In addition, the variables that influence the simulation can be changed with sliders in the GUI.

Since we only had one week to complete this project, we had to restrict ourselves to the implementation of the most basic features. Below are some possible directions for future implementations.

Energy cycle

The current system allows the amount of energy present in the world to fluctuate. Multiple explanations can be given for this factor. Firstly, the mating procedure could require less or more energy than the new cell receives as starting energy. Secondly, when a cell eats another cell, it could gain more energy than it can store. Lastly, there exist moves, wandering for example, where the energy is not put back into the system. All these factors disturb the balance in our ecosystem.

To create more balance, an energy cycle system could be added. In this system, the starting energy of new cells is based on the amount of energy put into mating by the parent cells. If the mating energy requirement is higher than the cells maximum energy, the surplus is added to an energy pool. The surplus energy from eating a cell and the energy loss due to an action are also added to this pool. The energy from the pool is released, at a later iteration, by creating cells incapable of acting. These cells function as a easy type of food for weaker cells.

Improvements of the behaviour system

At this point, the behaviour is static. Each cell has the same set of behaviours and requirements to execute a certain behaviour. Therefore, all our agents are of equal intelligence. This does not look like nature, where each organism has its own strategy for survival.

The behaviour system could be extended by varying the order in which types execute their behaviour or even restrict them to certain patterns. In doing this, different intelligence levels can be simulated. To imitate nature even more, one could develop a dynamic system where cells can learn new behaviour based on experience. This could eventually lead to a cell type that plans its own actions. The question is if this kind of cell would have an advantage over its strictly reflexive counterparts.

CONCLUSION

Even though we were able to develop a simulation that can exhibit some very interesting behaviours, it quickly became clear that creating a stable ecosystem is difficult and complex. Variables influence each other and minor changes can have consequences for the entire system. For example, altering the requirements for procreation of cells by 5% could turn overpopulation into rapid extinction. The basic building blocks that we have put together allow us to observe a bit of behaviour that can be found in the real world. However, more experiments and more sophisticated simulations will be necessary, in order to understand what it takes to develop evolutionary AI systems.

References

Darwin, C. and Beer, G. (1951). *The origin of species*. Oxford University Press Oxford.

Langton, C. G. (1997). *Artificial life: An overview*. Mit Press.

Mitchell, M. (1998). *An Introduction to Genetic Algorithms*. MIT Press.

APPENDIX

Source Code

`https://github.com/MarkusPfundstein/ZSB-Evolution`

Software and Hardware used

- Lenovo Thinkpad W530, Windows 8.1, 64bit
- Java 1.7
- Eclipse Keppler
- Github