TJHSST Senior Research Project
Simulating Evolution
2007-2008

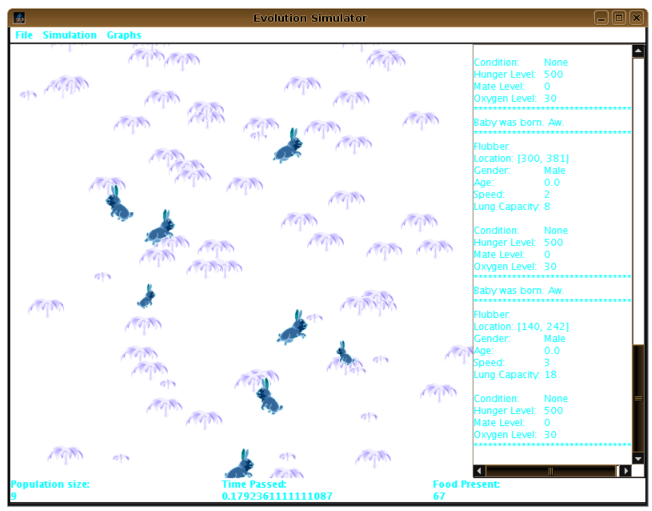
Tasha Wallage

June 4, 2008

Abstract

The main purpose of this program is to accurately simulate the genetic evolution of a species.It will attempt to do so using the most common evolutionary device known as genetic drift, a means of microevolution. This project is actually one of Agent-Based Modeling and Simulating (ABMS) in which the organisms are the agents that react with one another and their environment. It has successfully linked trait value changes with environmental disasters that create stress on a species.

Keywords: genetic algorithms(methods of representation, methods of selection, methods of change), evolutionary computation, genetic mutation, genetic drift, optimal state, microevolution, macroevolution, Darwanism, natural selection, genetic variation, recombination, gene flow, speciation



1 Introduction

1.1 Rationale

The computer will simulate an environment and the user can modify that envirnment. Modication of an organism environment can force it to adapt and in essence, better survive. Because of this, those that are not adaptable to the change will die off and those that are will live on and reproduce, thus creating a genetic drift in the species. This is the theory and according to this theory, one should be able to predict the changes in a species genetic make-up due to a change in its environment. If the environment becomes hotter, those creatures with higher temperature tolerance should be less affected than those without a high temperature tolerance and so one would expect to see the species evolve to have a greater tolerance to higher temperatures. My project tracks speed and lung capacity as the adaptable traits.

1.2 Purpose

Evolution often thought of as the changes that occur in an organism to better suit them to their envirnoment. However, this is not completely true. Evolution occurs in both positive and negative directions. It is completely random and the result could be in favor of the organism or it might not. It just so happens that the organisms resulting from an inefficient "evolution" do not survive and thus the species are left with the more fit species that will reproduce and populate. I am trying to simulate evolution and track the change in a species' traits to better understand how evolution really works and more importantly, to experiment with genetic aglorithms as represented in Computer Science.

2 Background

"Genetic changes do not anticipate a species' needs and those changes may be unrelated to the selection pressures on the species. Nevertheless, evolution is not a fundementally random process."[3]

Agent-Based Modeling

The actual evolution simulator is an ABMS with the Flubber class forming the 'agents.' An agent is "autonomous and self-directed." It can "function independently in its envirnoment and inits dealings with other agents." Mostly, an ABMS focuses on the interations between the agents. In this project, I will be observing both the interations between the agents and the interaction between the agents and their environment. [8]

Basic Concepts

A population of any given species is greatly affected by its environment. This is where an animal will get its food and raise its young. In order to do this, it has to be well adapted to the environment it lives, yet also able to change under stress (such as a change from the norm). This is when evolution will occur. The members of a species that are best able to handle stress are the ones that will live on to populate the species; therefore, their young will aquire the more desired traits and be able to live in the newly changed environment. The environment in which a population lives provides resources for the population such as food and shelter. If there is limited food, then the environment will only be able to support a given number of species, meaning that the population will have a max value or capacity. The function of the population over time should be logarithmic, approaching that max value. Howver, this is just a basic model of an environment, void of predators and many other factors that affect the size of the population. If there are predators, then the population size should oscillate in accord with the predators. When the population of the prey is alrge, the predators will have a plentiful food source availabe and breed more rapidly. This will cause the population size of the predators to increase. With more predators preying upon the other species, the population of the prey will diminish which, in turn, will cause the population of the predators to decrease because there will be less food availabe (not enough to sustain the large population). With less predators around, the prey will flourish, the population size will increase, and the cycle begins over again, leaving us with a sinusoidal relationship.[7]

1. Mechanisms that Decrease Genetic Variation[1]
   (a) Natural Selection
   Natural selection was introduced by Charles Darwin. It is when the frequency of the more prolific members of a species increases because they are better adapted to the environment.
   (b) Genetic Drift
   This occurs when the allele frequency changes (can allow mutant alleles drift into

fixation).

2. Mechanisms that Increase Genetic Variation[1]

   (a) Genetic Mutation
   This occurs when the gene sequence altered because the copy of "DNA" is corrupt.

   (b) Recombination
   This includes crossover of genes from the mother and the father to produce genes of the child(gene shuffling).

   (c) Gene Flow
   Gene flow occurs when genes drift into a population from a different population through mating. (This is not implemented in this project.)

3. Types of Evolution[1]

   (a) Macroevolution
   Macroevolution includes speciation, or the seperation of one species into two. It is an evolutionary change at or above the species.

   (b) Microevolution
   Microevolution is evolutionary change below the species level.

4. Types of Genetic Variation[4]

   (a) Variation Under Domestication

   (b) Variation Under Nature

3 Development Sections

3.1 Requirements, Overview, Limitations, Development Plan

The minimum requirement for this project would be an environment that includes the species under evaluation. There obviously must be a species to study and that species must have traits that can be altered, affecting its ability to survive. The environment must also provide a regenerating food source for the organism being studied else it will not live long enough to produce various generations leading to evolutionary results.

The project's main limitation is time. To singlehandedly write an accurate evolution simulator is very tedious and time-consuming especially since many problems occur, some of which cannot be fixed and a new method must be discovered. Also, the speed oft he computer is a limitation since running a simulation with millions of agents is a task that only a supercomputer can handle (or not, depending on how much each individual agent 'thinks' aka how much memory is alloted to each individual agent and how many methods are called from each during each cycle).

In order to effectively make progress on my project, I have done much work from home when unexpected errors occur and I have stuck to the timeline I plotted at the beginning of each quarter. When my first program turned out to be a failure, I spend extra time outside of class to create a new program that was more promising.

Software

1. Java

3.2 Research Theory and Design Criteria

Main classes

1. Blubber class

   - Description
   The sole purpose of the Blubber class is to provide food for the Flubber class. It stored the amount of energy it contains. As it gets older, it will grow bigger and thus the amount of energy it contains will increase. (If a Flubber eats an older Blubber it will gain more energy.)

2. Flubber class

   - Description
   The Flubber population is the species that is being studied. It contains changeable traits in its 'DNA' which determine the

characteristics of its children (unless they be claimed by mutation).

- States

  Each Flubber has four states: hungry, sexy, drowning, and curious. A Flubber bases its next action on its current condition. If it is hungry, it will search for food to eat or return to a nearby location from memory. If it is sexy, it will search for a mate of the opposite sexy that is also sexy. If it is drowning it will retrace its steps until it is no longer lacking in oxygen. Finally, if the Flubber is none of the afore mentioned conditions, it is curious. When a Flubber is curious, it will freely roam about its environment, noting the location of food that it 'sees.'

- Memory

  Flubbers have two basic memories. One is devoted to known food locations while the other is devoted to recent steps taken.

- Genes

  When two Flubbers mate, their genes are collected to determine the genetical makeup of the offspring, though randomly mutation will occur and the child will recieve a gene that is not of its mother or father.

3. Environment class

   - Description

     This class contains all the Flubber and Blubber objects in the environment and allows them to react with one another. It has two HashMaps; one for the Flubber population and one for the Blubber population. This class also provides the GUI for the simulation and is the portal for user input.

   - States

     The Environment class also has four different states. They are as follows:

     (a) Normal State
     This is the default state of the Environment in which the plants grow at a somewhat speedy rate and there is little or no stress on the species whatsoever. In the normal state, there may be water present, but only small lakes drawn in by the user.

     (b) Drought
     When a drought is present, about eighty percent of the food in the Environment will die and the regrowth and regeneration of the plantlife will be sluggish. Any water that was previously in the Environment will disappear (dry up) and the backround will change to a brownish color letting the user know a drought is present. When the Environment returns to a Normal State, the bodies of water will not return.

     (c) Flood
     A flood fills the entire Environment with water, leaving on the plantlife above the water. This means that the Flubber class can only survive by remaining on top of the Blubber plantlife. However, the regrowth and regeneration of the plantlife is increased while the Environment remains in this state.

     (d) Disease
     A diseased Environment will remove any individual Flubbers that do not meet the immunity required to survive. Depending on the severity of the disease and the diversity of the Flubber population, the disease may kill anywhere from no Flubber, to all Flubbers in the population.

Algorithms

1. Process for Recombination

   The process for creating a new organisms with a new combination of genes mixed from its parents (and sometimes randomly mutated) takes the traits from both parents and gives the child a trait that is either equal to one of the parents, or is a mix of the two (something in between). The assignment of the trait is semi-random.

2. Randomization for Mutation

   The process by which genes are mutated is completely random. In fact, it is double random because the swapping of genes is random and the chance that it is mutated is also random.

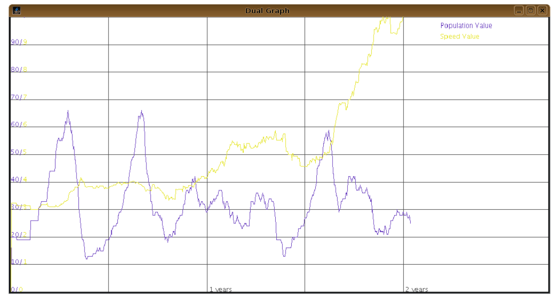
3.3 Testing and Analysis

In my first model, I included both predators and prey. This made the system very complex and unstable. If there were too many predators, they would end up consuming all the prey and then starve. If there were only a few predators, they would have a hard time catching the prey and would usually die before breeding. When the predators died out, the prey population would surge and the food from the environement could not grow fast enough to feed the gigantic herbivorous prey population. Eventually, the prey population would die out due to starvation. This also happened when the initail prey populaion was too large, thus when the prey died from starvation, the predators also died out.

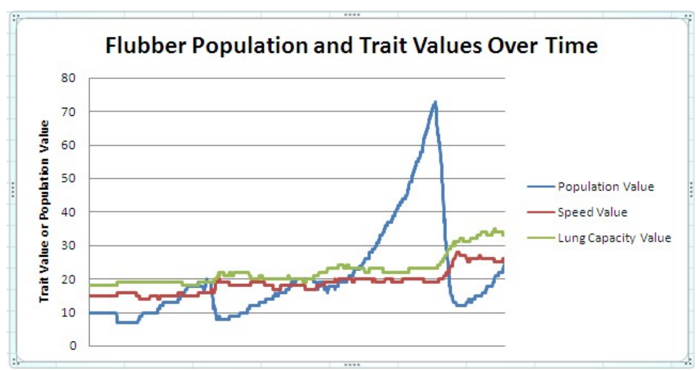
My current model has proved to be much more successful than its predecessor. Because there are no predators, the system is less succeptible to balancing problems. I have run this simulation consistently for 10 hours and the population of Flubber agents does not die out. Also, the Flubber class has gone through much testing because it is the main part of the simulation. I have had to test and correct all the methods, the most tedious being the breed and move methods. However, it now has proven to be very stable, allowing for actual testing on the species' survival to environmental factors. These factors include droughts, floods, and disease.

3.4 Visual Representation of Data and Results

I had originally created a Graph class and a DualGraph class which extended it. These classes provided a graphical display of the Flubber population changes and the change in the trait value (in this case, speed). A picture of such a graph is displayed below.



Now, however, I am using an upgraded version of the DualGraph class that, instead of only graphing two inputs, can graph as many inputs as the user specifies. This means that I can graph the average trait values alongside the population values of the Flubber population. Below is a graph of the speed, lung capacity, and population size of a Flubber popuation over a given time period (recreated in MSExcel from output data).



3.5 Development Procedures

Steps to Simulating Evolution

1. Create a basic environment with which a species may interact
2. Create a food source for the species
3. Create an herbivorous species with basic methods and designated traits to be tracked

4. Create a method that produces an idividual's "DNA" (a summary of the trait values)
5. Define how the species may evolve (genetic algorithms)
6. Create environmental factors that put stress on the species (flood, drought, etc.)
7. Create a graphical display to keep track of the output data
8. Track the changes in traits and make obsevations

4 Results, Discussion, Conclusion, Recommendations

With the new model in place, the program runs quite smoothly and the graphs provide a good graphical display of what is happening. It behaves almost as one would predict with an oscillating population value and a trait value that shows an upward trend (evolution for the better). From the DualGraph, you can see that the trait values usually show a rapid increase when the population is very high. Because the traits I am tracking are speed and lung capacity, this makes a lot of sense. As the population gets high, the availability of food decreases and the ability to get to food quicker becomes an important survival skill; therefore, the agents with greater speed will live on to reproduce while those with lesser speed will die out, increasing the trait average for the population. On the other hand, if there is a flood, the agents with the greatest lung capacity will live on while those with low lung capacity will die out. This means the average trait value of lung capacity will increase over time.

During the final stages of creating of creating the program, I introduced a new trait to the Flubber population. This trait was an immunity number. The immunity number of the initial Flubbers is a random number between one and one hundred. However, when the Flubbers reproduce, this number is treated as teh lung capacity or speed trait; it is determined by the parents or claimed by mutation. These numbers end up being quite diverse and no number is necessarily better than any other. When a disease is introduced into the environment, it comes with two parameters; two numbers, each between one and one hundred. If the immunity number of a particular agent is above the larger of teh two numbers or below the smaller of the two numbers, it dies from the disease. If the immunity number of the agent is in between the two numbers, it will have an immunity to the disease and, therefore, will not die or suffer and affects from the disease. This means that Flubbers with different immunity numbers will be immune to different diseases, but none will have full immunity from everything, much like in the real world.

When a disease strikes the population, a fraction will usually survive and reproduce, but the next few generations will have very similar immunity numbers, giving the population a very fragile existence. The species will have greater succes with a more diverse population.

An interesting detail I discovered while running this simulator for long periods of time is the overall trait averages fluctuate with the crisis the population encounters. For example, a flood will cause the overall trait average for lung capacity to rise, but if followed by a drought, the trait average may once again decrease back to where it was before the flood because that trait is no longer important for survival. Rarely are trait changes permanent. Just as it takes thousands of years for major evolutionary changes to occur in real life, it would take hours on a computer, even when running at really fast speeds.

This project leaves much room for improvement. One such area would involved expanding trait values to include other factors such as UV exposure (which would also include implementing the environment to include times of day with different sun exposure), metabalism (the speed at which the organism consumes its energy), and even muscle structure (to analyze evolution in motor techniques). However, there is a limit to evolution simulators. As of today, our computers can only handle so many calculations at one time. It is possible to go into so much detail that a computer will not be able to run the program efficiently, but programs like these usually require a team of hired professionals to create (such as the alife project). Overall, this field of study looks to be promising not only for simulating evolution, but also for other scientific and social studies. A program was made to simulate how people would react in a fire alarm in order to design a better escapse system, so programs such as these truly do have their uses.

5 Literature Cited

References

[1] Chris Colby, "Introduction to Evolutionary Biology", http://www.talkorigins.org/faqs/faq-intro-to-biology.html

[2] Charles Darwin, "The Origin of Species", http://www.talkorigins.org/faqs/origin.html

[3] John Wilkins, "Evolution and Chance", http://www/talkorgins.org/faqs/chance/chance.html

[4] Laurence Moran, "Random Genetic Drift", http://www.talkorgins.org/faqs/genetic-drift.html

[5] Jaroslaw Puszcynski, "Artificial life Portal", http://www.alife.pl/portal/main/e/index.html

[6] Jeff Smith, "Genetic Algorithms: Simulating Evolution on the Computer", http://www.developer.com/tech/article.php/964131

[7] Eric Turner, "Evolution Simulator", http://www.tjhsst.edu/ rla-timer/techlab07/TurnerProposal07.pdf

[8] Macal and North, "Tutorial on Agent-Based Modeling and Simulation", http://www.cas.uiuc.edu/networkreadings/north1.pdf