PHY 150 - Project Proposal

Shane Celis

May 22, 2008

1 Introduction

I would like to investigate a question regarding genetic algorithms (GAs). I would like to answer this question: Are implicit fitness functions merely obscured explicit fitness functions? Explicit fitness functions are the functions typically used in GAs to define an objective, usually something to be maximized or minimized. An implicit fitness function is more like natural selection where there is an environment of replicators whose differential reproduction rates "imply" its fitness.

2 System

The system I was envisioning defined a relatively simple environment that would allow one to express an implicit fitness function. (I'm all for an even simpler one.) It would consist of a grid of some limited space (perhaps $x, y \in [-1, 1]$). It will have particles of a fixed radius. The particles will move around and collide. The behavior of the particles will be described by two second order differential equations, describing a vector field, which will be encoded in the particle's genes (second order so that collisions will be possible). Figure 1 shows what the world will look like with some arrows to indicate velocities.

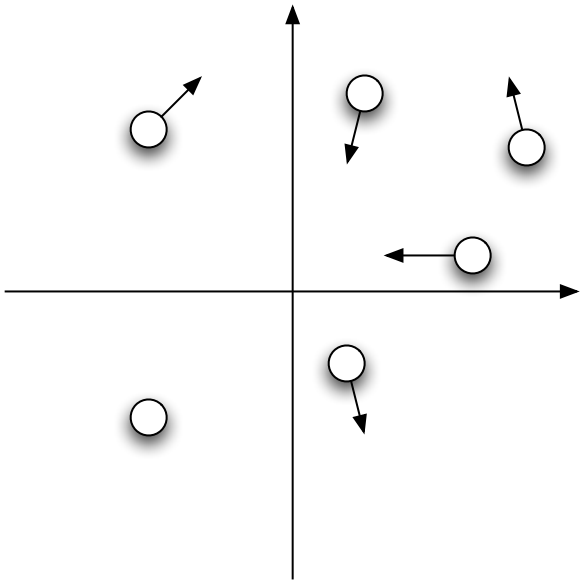

Figure 1: Grid environment

My thinking for the rules of this environment is that if a particle moves outside of the given range for x or y it will be eliminated. One can think of it as though the particle has fallen off an edge. The population as whole will perhaps be fixed at a constant number m. If a particle dies, it will be replaced by selecting a "living" particle at random and copying it with a chance of mutation. To provide a kind of competition, if a particle collides with another particle which causes or coincides with that particle falling off the edge, then the bumper particle will reap a benefit of being copied with mutation to fill the newly opened slot in the population. A natural death might be simulated by a random individual being removed at some time interval to avoid stagnation, perhaps proportionate to the time the individual has been in the population.

Here is an idea of what the vector field expressed by the genes may look like. It would allow for the genes to exhibit linear and non-linear behavior.

$$\begin{aligned} \ddot{x} &= \textstyle\sum_{i=1}^{n} a_i \; x^{b_i} \; y^{c_i} \\ \ddot{y} &= \textstyle\sum_{i=1}^{n} d_i \; x^{e_i} \; y^{f_i} \end{aligned} \tag{1}$$

The exponents of x and y would be integers with some relatively small bounds, and n is the number of terms to use for the vector field. The genes

would be $\{a_1, b_1, c_1, d_1, e_1, f_1, a_2, ...f_n\}$ and maybe another value to define a particle's color purely for visual purposes of seeing heredity at a glance.

One reason that I like this idea of the grid is that one might imagine that these particles might settle on a trivial strategy. For instance, they may just adopt a vector field that is zero everywhere, or one that has a restoring force to just one point, such that no particles move. This is a stable strategy only if everyone adopts it. If there is one mutant that doesn't, that say moves around in a circle it will knock the other particles from rest and potentially replicate itself generously. In that sense, it doesn't allow for a "conspiracy of doves" as evolutionary biologists would call it.

2.1 State Space

The state space for this system seems to be quite large. Perhaps they could be decomposed into the state space for the GA and for the dynamic it uses. If we look only at the genes, the state space is the space of all possible genes. Since a_i, d_i are real and the rest are integers, the size of state space would be $(R^{2n} + Z^{4n})$. The state space of the grid environment would include the gene state space plus the position and velocity of each particle which are both real two dimensional vectors.

3 Properties

I would like to run the simulation in two different fashions. One with an explicit fitness function, perhaps with a fitness function that maximized the speed of the particles within some fixed limits of the grid. I expect it will provide an average fitness that is monotonically increasing, maybe exhibiting epochs and innovation peaks. Then I would like to run the simulation with the environment which would define an implicit fitness function. Measuring averages of various parts of the system such as, average speed, average lifespan, average position, see if any of them exhibit the same qualities as the fitness graph when done with an explicit fitness function.

4 Methods

I would implement this in Python. The genes would be implemented as a list of numbers. The mutation operators can be defined by a gaussian

pertubations of the genes values. (It could also be done as a bit string. If there are existing libraries for this, I'd be happy to use them.) The Runge Kutta 4 integrator makes it pretty easy to simulate the trajectory of the particles. Circles are a relatively easy shape to collide. (I've implemented collisions before.) The display of the grid as it changes could be done using the vpython libraries.

5 Hypothesis

My guess is that I will find properties that exhibit what looks like an explicit fitness function. However, I don't know whether those will be stable. Perhaps, for some time interval t_1 there's a property that appears to be being "selected for", but perhaps at some later time interval t_2 it changes to something else.

One risk is that I might not measure the right things. I'm not sure I can measure the system exhaustively such that I can clearly state that there is no property that looks like a typical fitness graph. Also what if the fitness is hidden in a strange composition of its properties.

6 Steps

1. Write simulator (4 days)
2. Exercise simulator and collect data (3 days)
3. Write report (4 days)

This is an admittedly aggressive schedule considering the date that I've put this document together (May 21st, 2008). There are essentially 14 days left until the last day of presentations. However, I have buffered my estimates to account for having classes and final preparation in these times. I have worked as a software developer and that provides me with some confidence in my estimate as to what it will take to write the simulator. However, if there is perhaps a subset of this project that would be worth investigating, I would be open to hearing about it if this project looks too risky.