Chapter VII

Genetic Algorithms

Introduction

The methods in this chapter were developed in response to the need for general purpose methods for solving complex optimisation problems. A classical problem addressed is the **travelling salesman problem** in which a salesman must visit each of *n* cities once and only once in an optimum order, that which minimises his travelling. While not typical of a problem encountered in a computer game context, the problem of optimising responses or strategies clearly is applicable. There are two traditional methods for tackling optimisation problems:

- **Enumeration:** Basically we test the whole search space with respect to some optimisation function to find the best point with respect to this function. This can be a long process for substantial problems.
- **Calculus-based methods:** We may divide these into:
  - *Direct methods,* which use the gradient at each point in the space to guide in which direction to search next. This is comparable to the error descent

methods we used earlier and is effective in well-behaved problems (problems for which there are no local minima and which have continuous cost functions).

- *Indirect methods* which attempt to solve the nonlinear set of differential equations to get those points where the gradient is zero that is, stationary points of the evaluation function. Such solutions are often difficult or even impossible to find.

Because of these drawbacks, alternative methods which may be characterised as involving guided random searches have arisen. Such methods are based on enumerative techniques but allow some knowledge of the domain to permeate the search process. We will look at one of these types of methods which is based on processes which seem to involve the solution of difficult problems in the real world.

Genetic Algorithms

The attraction of simulating **evolution** as a problem solver is similar in many respects to the attraction of simulating neurons: evolution seems to offer a robust method of information processing. We will begin by briefly examining natural evolution and then consider the aspects of such evolution which are deemed to be important when we simulate it.

Natural Evolution

Evolution is a process of change for living beings. But whereas learning is a process which happens to an individual in the space of a lifetime, evolution is a process which changes species over a much longer timescale. Notice that the individual is unaware of evolution; evolution is happening on a different timescale to the individual's time-scale. In addition, evolution has no memory itself. It does not matter what happened last generation or the generation before, the only material with which evolution has to play is the current version of life.

It is important to be clear about the fact that evolution acts upon **chromosomes** rather than on living beings. We can view chromosomes as a code which determines life. Therefore we require a process which will decode the chromosomes. The individual parts of a chromosome are **genes**. It is the positioning of certain genes in specific positions in the chromosome which determine the features of the resulting life. In natural evolution, there is an alphabet of only four values; GAs typically use a binary 1/0 alphabet.