Computer Systems Lab, 2006-2007
The Implementation of Genetic Algorithms to Locate Highest Elevation

Harry Beddo

January 23, 2007

Abstract

My main areas of interest within Computer Science are machine learning and artificial intelligence.
Keywords: genetic algorithms

1 Introduction - Problem Statement and Purpose

The program works like a general genetic algorithm program. First, the parameters, cost, and cost function is defined. Then, the initial population is created. The cost is evaluated for each individual in the population. Pairs are selected to reproduce. Then there is mutation in the population. The resulting population is then tested and if the desired result is obtained the program stops. Else, it will start over with the cost evaluation step.

2 Genetic Algorithms

Genetic Algorithms are search techniques that find approximate solutions to search problems. The idea of genetic algorithms is inspired by evolutionary biology, "only the strong survive." Every individual in the population is made up of genes. Two individuals are matched in order to mate and exchange data. The idea is that the offspring of the two parents will be better than the parents. Through each generation, only the best individuals survive according to a cost function. The process continues until the population begins to converge.

3 Procedures

3.1 Creation

The initial population is put into an array consisting of 24 random chromosomes. Each chromosome is made up of fourteen random ones and zeros: the first seven digits refer to the x-coordinate, and the last seven refer to the y-coordinate. These values refer to discrete values of longitude and latitude on the map. There is also another array of costs. The values in the cost array

are the elevations at the locations of the corresponding chromosomes. The cost is negated in order to put it into the form of a minimizing algorithm. A large initial population will provide the genetic algorithm with a large search space. Usually, not all of the chromosomes from the initial population will survive the subsequent steps.

Next, the costs and their associated chromosomes are sorted from least to greatest. Only twelve chromosomes go through the iteration process. The top six are kept for each successive round and the bottom six are discarded. From the initial population, the top six from the original twenty-four are kept.

3.2 Pairing

The next step in the iteration process is to pair two chromosomes from the six remaining. There are many different ways the pairing process can be done. This is the step that I will have to experiment with to see which option is best. There are three options I will explore. First, is just simply pairing the first two, second two, and so on. Second, I could randomly select the pairs based on weights, with the lowest chromosome having the most weight to be selected as a parent. Third, I could use a tournament style selection. This style will select a small subset of any chromosome in the population and the chromosome with the lowest cost function of that subset is chosen.

3.3 Mating

Once the chromosome pairs are determined, the mating process begins. A crossover point is selected for the two parent chromosomes (p1, p2) to exchange bits and form two offspring (o1,o2). First, p1 passes its binary code from the left of the crossover point to o1. Next, p2 passes its binary code from the right of the crossover point to o1. Then, p1 passes its binary code from the right of the crossover point to o2 and p2 finally passes its code to o2. This process results in each offspring carrying portions of code from each parent. The population is doubled from the mating and goes to a size of twelve.

3.4 Mutation

Mutations then occur in the resulting population. A mutation will change a single bit from a 1 to a 0 and visa versa. Only .05 of all of the bits in the population and none from the best chromosome will undergo a mutation. A larger number of mutations will allow the population to search more outside the convergence path into new territory, while a smaller number of mutations converges the population quicker.

3.5 Checking

After the mutations take place, the new costs of the offspring are determined and the array is sorted again. This process is iterated again until a designated point. After a certain number of iterations, the population would not change if it were not for mutations. At this point, the search needs to be stopped and the current best chromosome should be the location of the highest elevation point.

3.6 Language and Algorithms

The main algorithms I will be researching is:

1. Genetic Algorithm

Computer language I'll use:

1. Python

4 Discussion

The overall purpose of this program is to find the highest elevation peak without evaluating every location on the map. If worked correctly, the program should only have evaluated a fraction of the total points available and given a result close to the maximum height.

References

[1] Haupt, Randy L., and Sue Ellen Haupt. "The Binary Genetic Algorithm." Practical Genetic Algorithms. 2nd ed. Hoboken, New Jersey: Wiley-Interscience, 2004. 27-50.