Lab #2: Genetic Algorithms

Student ID _____ Section # _____

Marks _____

Introduction

Objective:

In this Lab students will become familiar with implementing and experimenting with genetic algorithms. For the course lectures you should be fairly familiar with non deterministic optimization. In this lab you will become familiar with some techniques that you can use to make probabalistic decisions. The process is not difficult but may be somewhat unfamiliar.

There are lots of examples of applets demonstrating genetic algoritms on the web. Not as popular as more traditional approaches to solving problems in the real world, whereever that is, but kind of neat.

A web site can be found at http://cs.felk.cvut.cz/~xobitko/ga that covers some genetic algorithm examples. It is kind of phat.

In this lab a problem suitable for solution by a genetic algorithm is investigated.

The following suggests two problems somewhat toy that can be investigated.

1) **Recommended Problem:** The first is an allocation problem where the object is to allocate components into two racks of equipment minimizing the degree of interconnect. Review the course notes for a detailed example albeit very small. In this lab if you choose this problem make sure you use a considerably larger problem. A motivation for doing this problem is that it is similar to the problem for Lab 3. In addition, this problem is of historical significance as it was one of the first problems addressed by a non deterministic optimization algorithm at IBM in the late 60s. I have included an adjacency matrix in the appendix that can be used as a benchmark. Complete problem specification can be found in the notes.

2) Another problem for possible consideration in this lab is to use a genetic algorithm to solve a problem where the solution is an unknown sequence of 8 colors (red, yellow, blue, or green). The initial population of solutions is selected randomly. Each candidate is evaluated against the solution in terms of number of correct colours and the number in correct positions. e.g. 2 correct colors, 1 correct position is equal to 3 phat points. Basically this is the Mastermind game, or survival of the phatist.

3) A problem of your choice. The difficulty associated with GAs are encoding the problem.

Only one problem needs be attempted by your genetic algortihm. But again you should vary the problem instance size to see how well it performs, complexity wise.

Preparation:

Overview the section in the notes covering the genetic algorithm section.

Questions:

1) Did the algorithm run as expected? How did you encode a solution?

2) What was your maximum score solution. What was the percent improvement from and original or initial "solution".

3) How long did the algorithm take to run? If you doubled the size of your problem did the running time scale linearly, quadratically or by some other means?

4) Vary the initial population size. How did this affect the running time?

5) How might your basic algorithm be improved?

6) What was the mutation operation you used?

7) What was the performance as a function of problem size?

Optional Problem:

As an alternative to the above you may consider doing a lab about "Daisyworld". It is a model, algorithm, or simulation that is used to help understand evolutionary strategies and stability. It is optimization but of a different variety than you are used to seeing.

Report

Report Submission Instructions

The lab report is due at the start of the next lab, and may be submitted either as a hardcopy (printout or typed) or electronically by email.

- Each student is required to hand in a separate report.
- In the body of the lab report answer each of the questions posed in the lab.
- Write the corresponding section title and step number with each of your answers.
- Document all Java Programs with comments in the program code.

Electronic Submission

Labs may be submitted electronically by e-mail to the TA as follows:

- Lab write ups will only be accepted in plain text or PDF format[1].
- Programs should be in plain text.

- By email:
  - Submissions will only be accepted from your regular **CCU** / **ECE** account.
  - In the body of the email include full name, student number and lab number.
  - Lab write-up and required programs, sent as attachments.

Report for Lab 2

- Lab write-up indicating all questions and their numbers.
- As Appendices:
  - Code you have written, and documentation.

1. CCU offers a postscript to PDF conversion utility: ***ps2pdf***

Appendix: Adjacency list of component connectivity. c_{ij} represents the number of connections between component i and component j.

```
0 0 7 4 0 0 3 0 0 8 0 0 6 0 0 0 9 0 0 1
0 0 6 0 0 5 2 2 0 0 0 9 0 0 8 7 0 7 0 0
7 6 0 0 0 3 0 0 0 8 0 6 0 0 9 0 0 0 0 5
4 0 0 0 2 0 0 0 9 0 8 0 0 4 0 8 0 0 9 0
0 0 0 2 0 0 3 0 0 4 0 4 6 0 0 0 5 0 6 0
0 5 3 0 0 0 0 4 0 0 0 9 0 0 2 0 0 0 0 0
3 0 0 2 3 0 0 0 0 0 7 0 7 0 0 7 0 0 2 3
0 2 0 0 0 4 0 0 0 0 0 4 0 0 5 0 0 6 0 0
0 0 0 9 0 0 0 0 0 7 0 0 0 8 0 0 0 0 5 0
8 0 8 0 4 0 0 0 7 0 6 0 6 0 0 6 0 5 0 7
0 0 0 8 0 0 7 0 0 6 0 0 0 4 0 0 5 0 0 0
0 9 6 0 4 9 0 4 0 0 0 0 0 7 0 0 0 8 0 0
6 0 0 0 6 0 7 0 0 6 0 0 0 0 0 9 0 5 0 7
0 0 0 4 0 0 0 0 8 0 4 7 0 0 0 0 0 0 0 0
0 8 9 0 0 2 0 5 0 0 0 0 0 0 0 0 8 0 0 0
0 7 0 8 0 0 7 0 0 6 0 0 9 0 0 0 0 0 5 9
9 0 0 0 5 0 0 0 0 0 5 0 0 0 8 0 0 0 0 3
0 7 0 0 0 0 0 6 0 5 0 8 5 0 0 0 0 0 5 0
0 0 0 9 6 0 2 0 5 0 0 0 0 0 0 5 0 5 0 0
1 0 5 0 0 0 3 0 0 7 0 0 7 0 0 9 3 0 0 0
```