

# EVCO Open Assessment

Y6375144

December 15, 2013

# Chapter 1

## Introduction

### 1.1 Genetic Algorithms

For some problems it can be difficult to find an optimal solution. This is because the search space grows at a much faster rate (exponentially, for example) than the input space. The Travelling Salesperson problem is a perfect example of this - for a search space with  $n$  cities, there are  $n!$  paths that the salesperson could take.

Evolutionary algorithms do not present a solution to this problem. However, by taking inspiration from biological evolution, they do offer an alternative approach that will often lead to a ‘good-enough’ solution in a much more reasonable number of calculations.

Genetic algorithms are a type of evolutionary algorithm that typically represent population members (i.e. candidate solutions) in bitstrings. The population of strings is repeatedly updated through the following process:

1. Select - A number of the existing population members are selected to continue through to the next round. Typically these would be the fittest members in the current population.
2. Crossover - To generate new population members, existing ones are created by crossing two (or more) population members over. See figure N for a visual representation.
3. Mutate - The population is mutated at random. This typically involves passing over every bit in a string, and with a random probability flipping the bit from a 0 to a 1 (or vice versa).
4. Evaluate - A fitness function is run on each member of the population, giving an indicator of how ‘good’ a solution is.

Genetic algorithms hold much potential, but in order to get the most from them you must carefully choose your representation and parameters. The representation is the bit string typically, but it can also take other forms. There are many parameters that can be fine tuned, such as the number of offspring to breed (at the crossover stage), or the type of crossover to perform.

### 1.2 The Problem

## Chapter 2

# The Solution

2.1 Representation

2.2 Selection

2.3 Crossover

2.4 Mutation

## Chapter 3

# The Results

### 3.1 Conclusion