Christopher Goes

Assignment 2 – Genetic Algorithms – Solving a simple Substitution Cipher

|  |  |
| --- | --- |
| **Parameter** | **Final value used** |
| Genetic Algorithm | Steady State |
| Fitness Function | Bhattacharyya |
| Crossover Function | PMX |
| Crossover Probability | 0.8 |
| Mutation Function | Single swap |
| Mutation Probability | 1.0 |
| Population Size | 256 |
| Fitness Evaluations | 100,000 |
| Tournament Size | 3 |
| Elitism | Yes |
| Local Search | No |

I experimented with varying the crossover probability, fitness function used, mutation probability, number of fitness evaluations, population size, mutation function used, and punishment.

Fitness Function

Varying the fitness function used to Euclidian did not significantly change the deciphering accuracy. In light of this, I decided to stick with Bhattacharyya. This was due to its ease of reading the fitness values, and a slightly faster running (~100-200ms as measured by Linux’s “time” tool).

Punishment

Testing Method

**Compilation**

Compiled using GCC 4.4, 4.8, and 4.9.

Flags used: “-O3 -lm”