ASSIGNMENT 3

Name: Jason Collier
Student Number: 214008258
Date: 28/08/2017

I declare that this is my own, original work.

Signature: _____

IMPLEMENTATION DETAILS

Stopping Conditions: STOP when t is GREATER THAN 10000 iterations OR when population fitness has not changed for 100 iterations

Initialization: Mass of each alloy initialized in the range [0, 59]

Values for population size investigated: 50, 100, 1000

Values for mutation rate investigated: 0, 0.15, 0.3, 0.6 (probabilities for uniform crossover)

Values for mutation magnitude investigated: random values in the range [0, α] where $\alpha = x_{max,j} - x_{ij}$ or $\alpha = x_{ij} - x_{min,j}$ (selected at random)

Values for selection method investigated: 5%, 10%, 50%, 100% of total population (tournament selection)

Values for crossover rate investigated: 0.5, 0.85, 1 (probabilities for uniform mutation)

RESULTS

Number of iterations (typically): 500 (P_m=0.15, P_c=0.85, population=1000, Selection=10%)

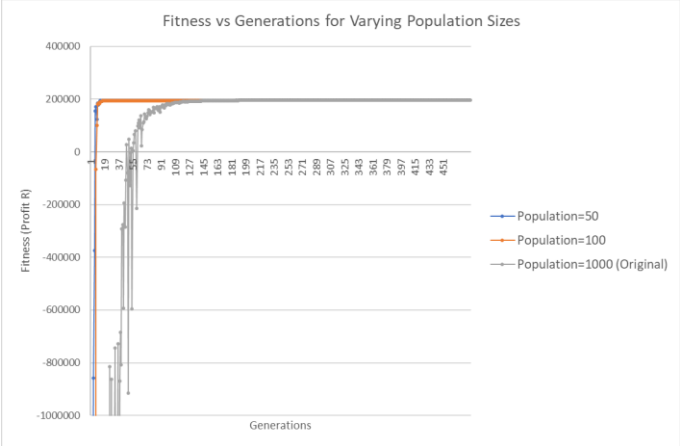
Largest Profit found: R195692.73

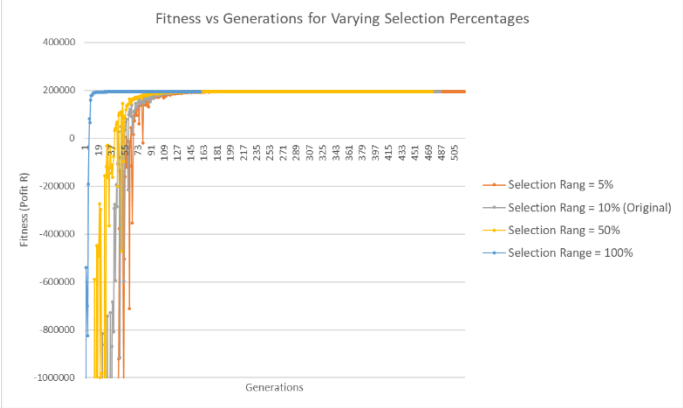
Optimal amount of each metal:

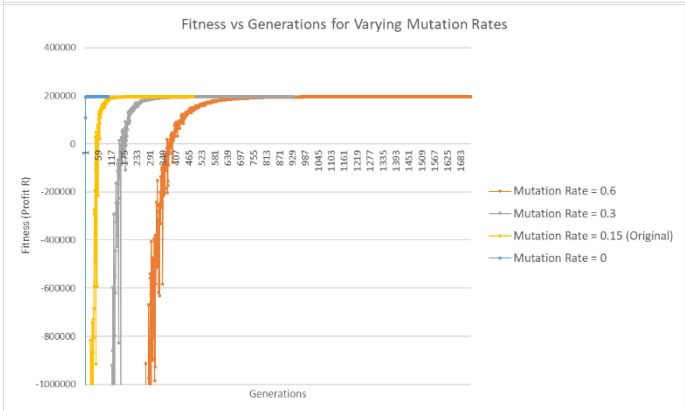
Adamantium	Unobtainium	Dilithium	Pandemonium
12 Kg	0 Kg	44 Kg	7 Kg

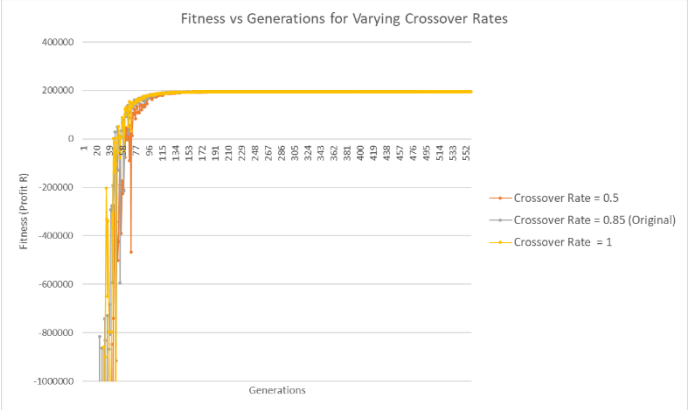
Figures 1 - 4: SSE vs generations for various settings

(See Next Page)


Fitness vs Generations for Varying Population Sizes
Fitness (Profit R)
Generations
Population=50
Population=100
Population=1000 (Original)


Fitness vs Generations for Varying Selection Percentages
Fitness (Profit R)
Generations
Selection Rang = 5%
Selection Rang = 10% (Original)
Selection Rang = 50%
Selection Range = 100%


Fitness vs Generations for Varying Mutation Rates
Fitness (Profit R)
Generations
Mutation Rate = 0.6
Mutation Rate = 0.3
Mutation Rate = 0.15 (Original)
Mutation Rate = 0


Fitness vs Generations for Varying Crossover Rates
Fitness (Profit R)
Generations
Crossover Rate = 0.5
Crossover Rate = 0.85 (Original)
Crossover Rate = 1

OBSERVATIONS

The final settings chosen included a population size of 1000 individuals, a tournament selection size of 100 individuals, a mutation rate of 0.15, a crossover rate of 0.85 and an initialization range of [0, 59]. These settings were chosen, based on the graphs on page 2, to maintain population diversity while keeping efficiency in mind due to time constraints. These are the settings that resulted in the optimum masses of each alloy and largest overall profit as written on page 1.

The algorithm stops when the population fitness does not change for 100 consecutive generations or when a maximum of 10000 generations is reached to prevent infinite loops from occurring. The results of different settings, using these stopping conditions, are graphed on page 2 and are discussed below.

A small population of 50 was used in order to quickly verify whether the code was working or not. Once verified, larger population sizes of 100 and 1000 were used. Larger populations than these were not considered as they slowed the algorithm down too much, so a population size of 1000 individuals was selected to maintain diversity and efficiency (see graph 1).

Various percentages of this population, used in tournament selection to select parents, were investigated, namely 5%, 10%, 50% and 100%. The selective pressures are too high for selection groups of 50% and 100% of the total population, and the takeover time is too long for a 5% selection group (see graph 2). So, a selection group of 10% of the total population was used.

The new population is selected by taking the best individual of the previous population and filling up the rest of the population with offspring.

Mutation rates of 0, 0.15, 0.3 and 0.6 were investigated (see graph 3). Higher mutation rates were not investigated as they made the converging time unnecessarily long. A mutation rate of 0 was not chosen because it does not add diversity to the population and a mutation rate of 0.6 was not chosen because it added too much time to the algorithm process. A mutation rate of 0.15 was decided on as it is big enough to add diversity and small enough not to distort the good solutions.

Varying the crossover rate does not have much of an effect on the algorithm (see graph 4), but a crossover rate of 0.85 was used over the other rates tested in order to add an element of randomness and lower the selective pressure slightly, while not slowing the algorithm too much.

While there can be many answers that will correctly optimize this companies profit, none achieve a profit higher than R197000. It is then safe to say that this algorithm, which achieved a solution to obtain a profit of R195693, has solved the problem satisfactorily.