1b)

The first diagram is a plot showing the average fitness of the fittest member of the population at each generation as requested. The second diagram shows the average fitness of the fittest member (as requested) as well as the fitness of the fittest member in each run.

We were able to obtain the global maximum roughly 3 to 5 times out of the 30 times in each of the simulations we ran.





c)

If the fitness function can take on negative values, it would interfere with the selection\_R algorithm. This is because we choose a parent with probability .

However, if is negative, then the probability would not be correct and could potentially be negative. To fix this problem, find the most negative fitness value an individual can take and add that value to the function. This guarantees that each fitness value can only be non-negative.

d) We would like the algorithm to exploit local searching more, meaning the crossover point should occur more frequently at lower bits than higher bits. Since each string consists of two numbers and , we want the crossover point to occur more frequently at the lower bits of the two numbers (to the left of bits 7 and 14).

e) Since and can take on any integer from , We need to search through 128 numbers for each value. This means to make sure we find the optimal solution, we need to evaluate times.

2 b)



With this particular fitness function, tournament selection seems to work better consistently. One advantage of roulette is that the fittest individuals will consistently have higher probability of producing descendants. This means that if you have a series of bits that produces higher fitness, this series of bits will more likely be preserved over generations. In other words, children will tend to converge to similar bits. This is beneficial if the optimal solution is centered in an area because it will more likely fine-tune the solution to reach the optimum. This may become a disadvantage when there are local maximums scatter throughout, because tournament increases the variability and randomness to the children generation process.

3a)

To convert the minimization problem to a maximization problem, we simply want high values in the function to become low values in the function and vice versa, without introducing negative values. The easiest way to achieve this would be taking the reciprocal of the function, that is,

3b)



3c)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Mean/Average | Standard Deviation | # trials that found opt |
| GA: selectionT | 898576353.9 | 15183619.7 | 21 |
| GA: selectionR | 902501886.6 | 16171807.8 | 7 |
| SA: 4b | 914415586.7 | 18478494.7 | 4 |
| SA: 4c | 912259469.4 | 22359638.3 | 3 |

3d)

Since we only have thirty simulation runs, we only generated thirty initial conditions. This is too small a pool of samples for much statistical significance.