Assignment : 2

Name : Fule Ankit Ashokrao
Roll no. : 204103111
Subject : Soft Computing in Engineering
Topic : Genetic Algorithm

➢ Introduction :

In this coding assignment, we are going to implement Genetic Algorithm using MATLAB code. Genetic Algorithm is a population based search and optimization technique. It works on the Darwin's principle of natural selection i.e survival of fittest. GA solution are represented in coded form and depending upon coding scheme several versions of GA are available such as Binary coded GA, Real coded GA, Micro GA etc. In this assignment, we are going to solve the problem with the help of Binary coded GA.

Every GA consist of some steps which are as follows.

- Initialize a population of solution
- Calculation of fitness value of each solution in the population
- Reproduction
- Crossover
- Mutation

In Binary coded GA, solutions are represented in binary form i.e in 0 and 1 form.
GA is generally used in solving maximization problem but our given problem has to be solved for minimization. Therefore, to solve minimization problem using GA we have to first convert it to maximization problem.

Following are the ways to convert given minimization problem into maximization problem.

- Max Y ' = 1/f(x) ; for $f(x) \neq 0$
- Max Y ' = 1/(1+f(x)) ; for $f(x) \geq 0$
- Max Y ' = 1/(1+f(x)^2)
- Max Y ' = -f(x)

➢ Methodology :

Step 1 :

We are taking objective function from user. In given problem objective function is given by,

Minimize $f(x1, x2) = x1 + x2 - 2 * x1^2 - x2^2 + x1 * x2$
In the range $0 \leq x1 \leq 0.5$; $0 \leq x2 \leq 0.5$

Step 2 :

We are also taking following parameters from user

- Population Size (N)
- Crossover Probability (pc)
- Mutation Probability (pm)

Step 3 :

- Specifying maximum number of generation for which we are carrying out algorithm (200)
- String length for both the decision variable is taken = 20 each. Therefore, length of solution = 40.
- Now initializing population of size N randomly.

Step 4 :

- Calculating fitness value of each solution by using fitness function. Here fitness function is taken as $\frac{1}{(1+f(x1,x2))}$

Step 5 :

- Performing Reproduction using Roulette wheel selection. Here chances of Solution getting selected in mating pool is proportional to its fitness value.

Step 6 :

- Now Crossover is performed with probability of crossover = pc.(0.9-1.0)
- For this mating pairs consist of two solution are selected from mating pool generated in reproduction randomly.
- In this we are performing two-point crossover where two crossover sites are selected to perform swapping of bits in solution.
- Crossover sites are selected randomly.

Step 7 :

- After these steps we are going to perform Mutation on the obtained solution set.
- Mutation probability is selected as low as possible. Here we are taking it from user = pm (less than 0.1)
- Mutation is performed bitwise
- Mutation operator changes 0 to 1 and vice versa

Step 8 :

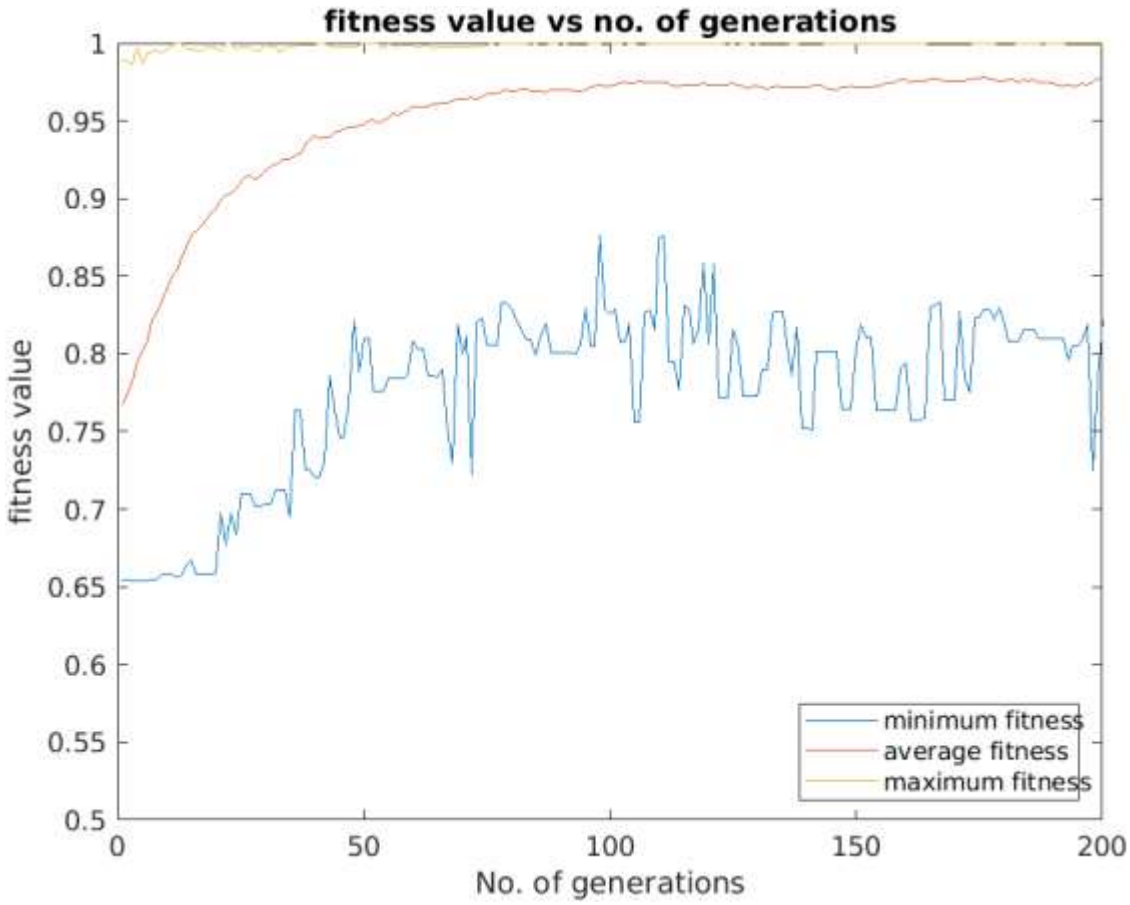
- 1^{st} generation is completed.
- Now repeat steps 4 to 7 to obtain optimal solution.

➢ Result and Discussion :

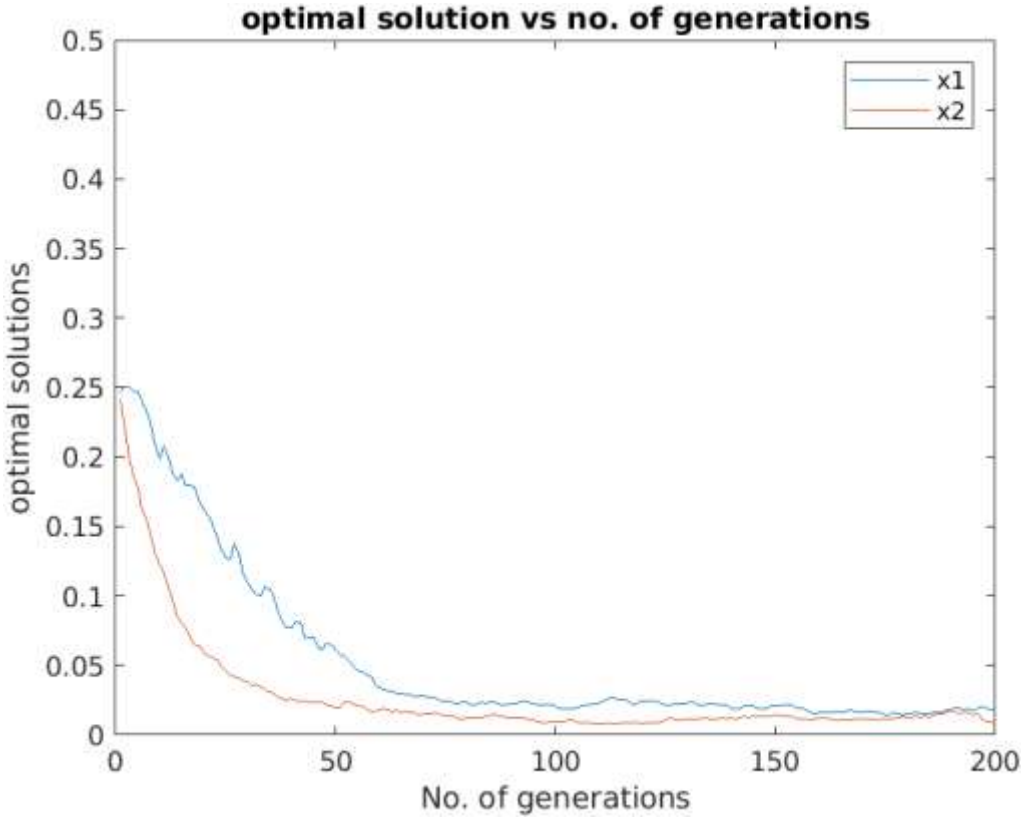
By running the algorithm for 200 iteration we are able to get optimal solution. Here the plots attached are obtained for following values of different parameters.

- Population size = 1000
- Crossover probability = 0.95
- Mutation probability = 0.001
- String length = 40

❖ Plot between fitness value vs number of generations :



❖ **Plot between optimal solution and number of generations :**



➢ **Conclusion** :

- By using higher population size problem is more likely to converge.
- Mutation probability also affects convergence of the problem. lesser the mutation probability more is the chances of convergence.
- Binary coded GA needs higher number of bits to store values of decision variable and it increases the computation time of the algorithm so to avoid this Real coded GA's are used.