**CI Assignment #4**

**Krishangee Bora ,Panther id : 002421280**

**Kbora1@student.gsu.edu**

**Below with are steps outlined :-**

First lets define as to what are **genetic algorithms ?**

* Genetic algorithms are basically software programmes that learn in an evolutionary manner similar to the way biological systems evolve.
* It is a search method that follows a process which simulates evolution in a computer
* “survival of the fittest “solution, it works on large populations of solutions that are subjected to selection pressure.

The whole concept along with the problem was executed in **MATLAB** . According to the question we have to minimize the equation of f(x,y) subject to the following equations given .

**STEP 1 :-** In the **decode.m** file we try to put this condition across matlab. It is as provided :-

% Decoding

function [x,Genes] = Decoding(lb,ub,Genes)

[row, gens] = size(Genes);

count = length(lb);

x = zeros(count,row);

Genes\_count = int32(gens/2);

for n = 1:count

d = double(zeros(row, 1));

for i=1:Genes\_count

d = d + double(Genes(:,(n-1)\*Genes\_count+i))\*double(2^(Genes\_count-i));

end

x(n,:) = lb(n) + d \* (ub(n)-lb(n)) /double( 2^Genes\_count) ;

end

index = 1;

for n = 1:row

if x(1,n)^2-x(2,n)+1<=0 && 1-x(1,n)+(x(2,n)-4)^2<=0

index = n;

break;

end

end

for n = 1:row

if x(1,n)^2-x(2,n)+1<=0 && 1-x(1,n)+(x(2,n)-4)^2<=0

index = n;

else

Genes(n,:) = Genes(index,:);

x(:,n) = x(:,index);

end

end

end

In the lines marked by the red boundary it is highlighted out.

* The next step is that it is bounded by 2 values in the boundary which is of upper and lower boundary whereby

X= [0 ,10] and y =[0,10]

* In the next step split the gens vector into 2 vector to solve x,y value

X = 0 0 1 0 0 1 0 0 1 1 0 1 1 0 0 1 1 0 0 1

Y =0 1 0 1 1 0 01 1 1 0 0 1 0 0 1 0 1 0 1

In the next step we convert binary to digital value conversion :-

0 ^ 20 + 0^ 19 …

**STEP 2 :-**Now lets got the fitness.m file whereby we take to the equation we have to minimize :-

The code snippet is as follows :-

Fitness (objective function)

function f = Fitness(x1, x2)

f = (sin(2\*pi. \*x1).^3.\*sin(2\*pi.\*x2))./(x1.^3.\*(x1+x2));

end

**STEP 3 :-**Now lets go to the GA optimization file.The code snippet is as follows:-

% Parameters

clear variables; clear all; close all; clc;

Pc = 0.1; % Probability of CrossOver

Pm = 0.08; % Probability of Mutation

PopSize = 1000; % Population Size

Gen = 1000; % Generation

Len1 = 20; % Bits of x1

Len2 = 20; % Bits of x2

lb = [0, 0];

ub = [10, 10];

tic

% Binary encoding initialization

Genes = logical(randi([0 1], PopSize, Len1+Len2));

BestGenes = zeros(Gen, Len1+Len2);

BestGenesEver = zeros(Gen, Len1+Len2);

BestGene = Genes(1,:);

BestGeneEver = Genes(1,:);

Fits = zeros(PopSize, 1);

BestFits = zeros(Gen, 1);

BestFitsEver = zeros(Gen, 1);

BestFit = 0;

BestFitEver = 0;

GenEver = 0;

point1 = zeros(Gen, 1);

point2 = zeros(Gen, 1);

% GeneticAlg

for i=1:Gen

% compute fitness

[M\_X, Genes] = Decoding(lb,ub,Genes);

x1 = M\_X(1,:);x2 = M\_X(2,:);

Fits = Fitness(x1,x2);

% save the best gene

BestFit = min(Fits);

GeneRow = find(Fits==BestFit,1); % save only the first index

BestGene = Genes(GeneRow,:);

BestFits(i) = BestFit;

BestGenes(i,:) = BestGene(1,:);

if BestFit < BestFitEver

BestFitEver = BestFit;

BestFitsEver(i) = BestFit;

BestGeneEver(1,:) = BestGene(1,:);

BestGenesEver(i,:) = BestGene(1,:);

GenEver = i;

point1(i) = x1(GeneRow);

point2(i) = x2(GeneRow);

else

BestFitsEver(i) = BestFitEver;

BestGenesEver(i,:) = BestGeneEver(1,:);

end

% Selection Operator

%Genes = RouletteSelection(Genes, Fits, PopSize);

Genes = TournamentSelection(Genes, Fits, PopSize);

% CrossOver Operator

Genes = CrossOver(Genes, Pc);

% Mutation Operator

Genes = BitMutation(Genes, Pm);

%Genes = IndividualMutation(Genes, Pm);

end

toc

% plot preparation

dense = 50;

x1 = linspace(lb(1), ub(1), dense);

x2 = linspace(lb(2), ub(2), dense);

F = zeros(40);

for i=1:dense

for j=1:dense

F(i,j) = Fitness(x1(i),x2(j));

end

end

num = find(point1~=0);

p1 = point1(num);

p2 = point2(num);

FitEver = Fitness(p1,p2);

t = length(num);

% plot

figure(1)

surf(x1,x2,F)

hold on

plot3(p1,p2,FitEver,'r.-')

plot3(p1(t),p2(t),F(t),'b\*')

xlabel('x1')

ylabel('x2')

zlabel('F(x)')

figure(2)

plot(BestFits)

hold on ;grid on

fprintf('F(x) = %0.4f\n',BestFitsEver(GenEver));

fprintf('x1 = %0.4f\n',p1(t));

fprintf('x2 = %0.4f\n',p2(t));

plot(BestFitsEver,'r')

plot(GenEver,BestFitsEver(GenEver),'r\*')

xlabel('Generation')

ylabel('F(x)')

title(['Generation = ',num2str(GenEver),' F(x) = ',num2str(BestFitEver)])

Here we have different parameters. We will run it across the various parameters .The parameters included are as follows :-

* probability of crossover
* probability of mutation
* population size
* generation
* length of bits of x1 or x
* length of bits of x2 or y

The various scenarios of studies are shown below :-

**CASE 1 :-** The various graphs were obtained and hence the observations and conclusions were drawn .These are all discussed below starting herewith

On passing for default values we obtain best fit as **-0.03** and best fit generation as **-0.03** for the range of x and y between 0 and 10 with default parameters passed .This is shown below

X=1.5

Y=3.8

The other parameters are as follows :-

Pc = 0.02; % Probability of CrossOver

Pm = 0.008; % Probability of Mutation

PopSize = 150; % Population Size

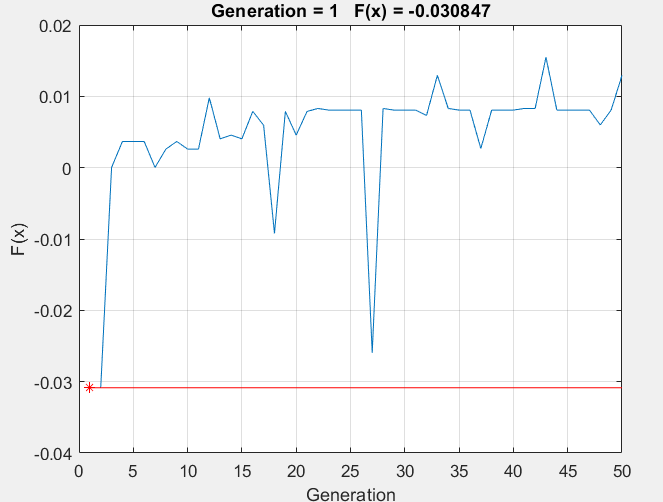
Gen = 50; % Generation

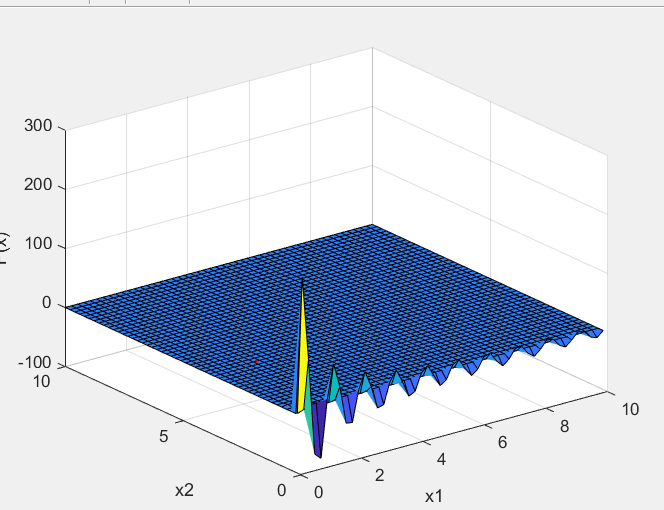
Len1 = 10; % Bits of x1

Len2 = 10; % Bits of x2

lb = [0, 0];

ub = [10, 10];





From the graphs best fit and minimized value of F(x, y )= -

0.03 and best fit generation is 1

**Case 2** :- The various graphs were obtained and hence the observations and conclusions were drawn .These are all discussed below starting herewith

X or x1 = 1.2262,

Y or x2 =3.7523

The other parameters values are as follows:-

Pc = 0.1; % Probability of CrossOver

Pm = 0.08; % Probability of Mutation

PopSize = 1000; % Population Size

Gen = 1000; % Generation

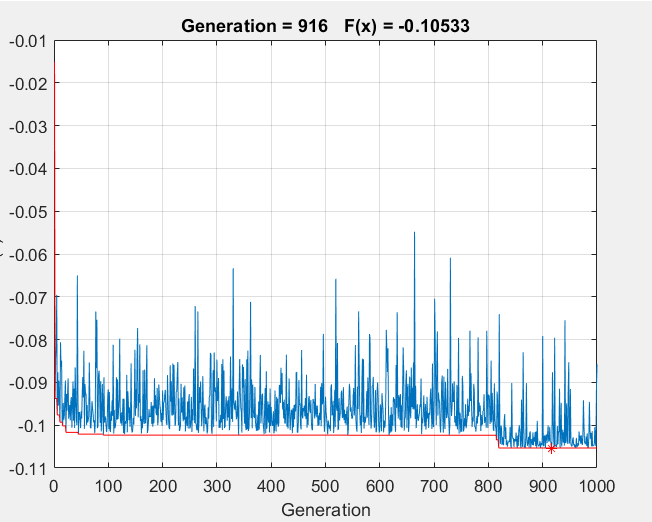
Len1 = 20; % Bits of x1

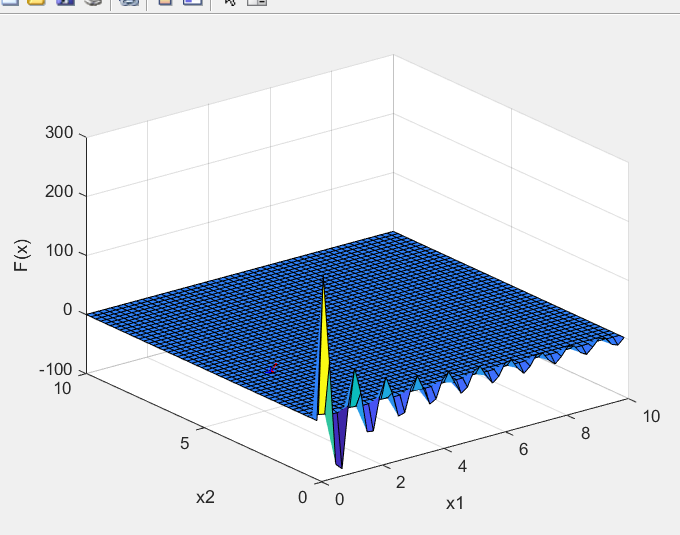
Len2 = 20; % Bits of x2

lb = [0, 0];

ub = [10, 10]

Now the fit function of f(x,y) which hereby has been processed as f(x1,x2) or F (X) is shown by the generation of the graphs in matlab as follows :-





From the graphs best fit of F(x, y )= -0.10533 and best fit generation is 916 .

**Case 3:-** The various graphs were obtained and hence the observations and conclusions were drawn .These are all discussed below starting herewith

X or x1 = 1.2764,

Y or x2 =3.7821

The other parameters values are as follows:-

Pc = 0.01; % Probability of CrossOver

Pm = 0.02; % Probability of Mutation

PopSize = 500; % Population Size

Gen = 20; % Generation

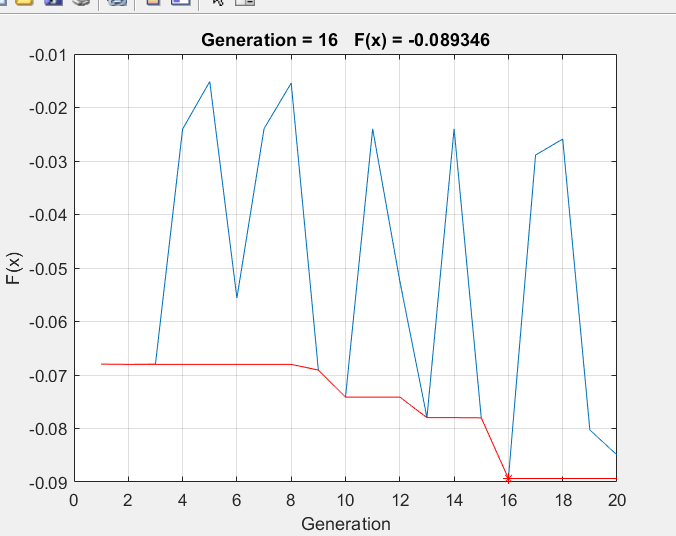
Len1 = 20; % Bits of x1

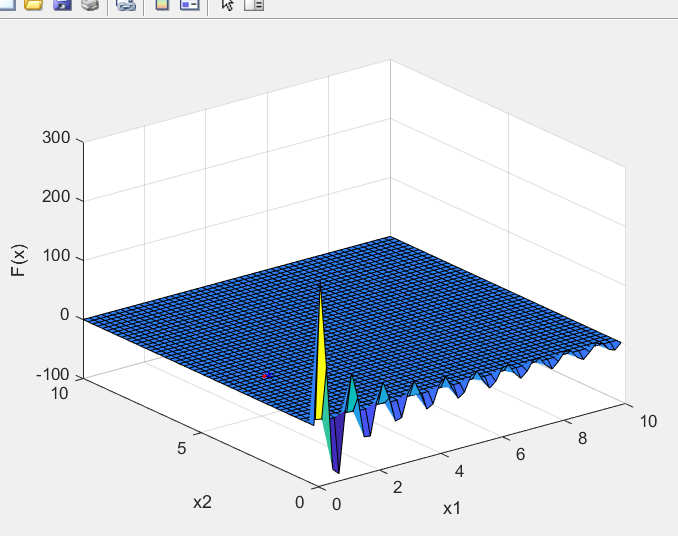
Len2 = 20; % Bits of x2

lb = [0, 0];

ub = [10, 10]

Now the fit function of f(x,y) which hereby has been processed as f(x1,x2) or F (X) is shown by the generation of the graphs in matlab as follows :-





From the graphs best fit and minimized value of F(x, y )= -0.089 and best fit generation is 16

**CASE 4 :** The various graphs were obtained and hence the observations and conclusions were drawn .These are all discussed below starting herewith

X or x1 = 1.2284,

Y or x2 =3.7446

The other parameters values are as follows:-

Pc = 0.02; % Probability of CrossOver

Pm = 0.04; % Probability of Mutation

PopSize = 5000; % Population Size

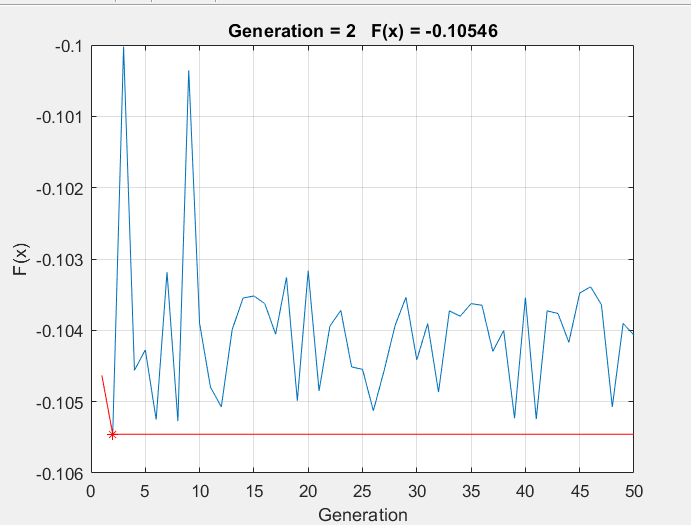
Gen = 50; % Generation

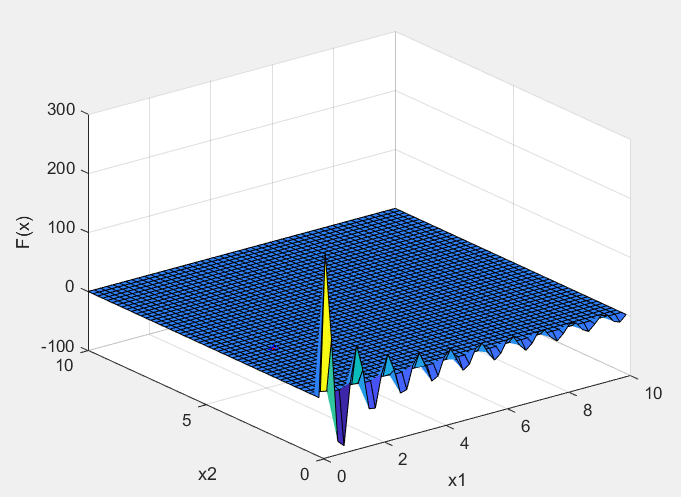
Len1 = 20; % Bits of x1

Len2 = 20; % Bits of x2

lb = [0, 0];

ub = [10, 10]





From the graphs best fit value and minimized value of F(x, y )= -0.10546 and best fit generation is 2

**CASE 5:** The various graphs were obtained and hence the observations and conclusions were drawn .These are all discussed below starting herewith

X or x1 = 1.2228,

Y or x2 =3.7526

The other parameters values are as follows:-

Pc = 0.02; % Probability of CrossOver

Pm = 0.04; % Probability of Mutation

PopSize = 5000; % Population Size

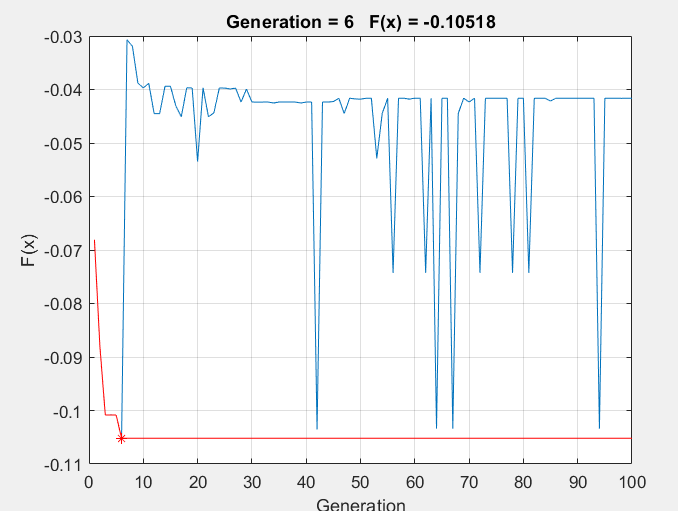
Gen = 50; % Generation

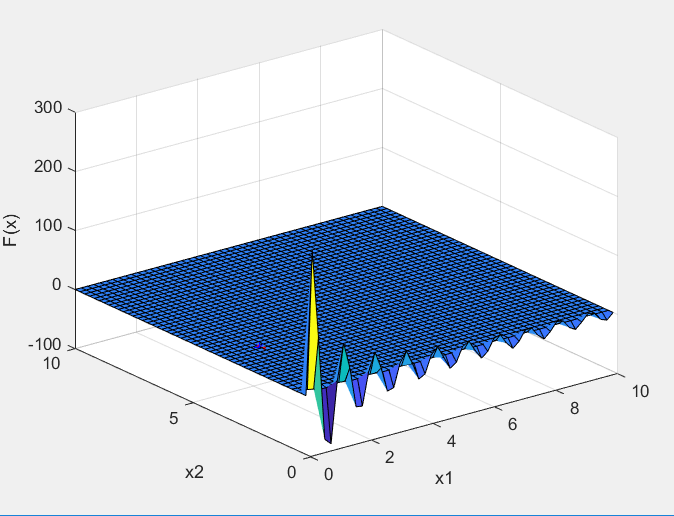
Len1 = 20; % Bits of x1

Len2 = 20; % Bits of x2

lb = [0, 0];

ub = [10, 10]





From the graphs,best fit value and minimized value of F(x, y )= -0.1052 and best fit generation is 6

**CASE 6 :** The various graphs were obtained and hence the observations and conclusions were drawn .These are all discussed below starting herewith

X or x1 = 1.2304,

Y or x2 =3.7688

The other parameters values are as follows:-

Pc = 0.8; % Probability of CrossOver

Pm = 0.04; % Probability of Mutation

PopSize = 50; % Population Size

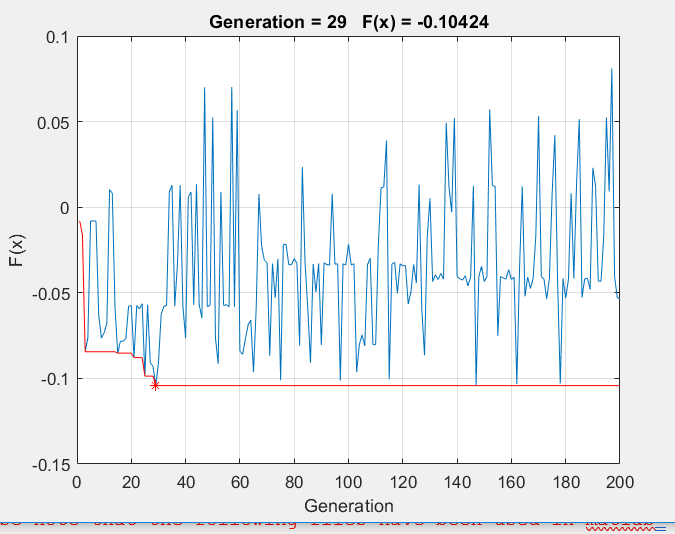
Gen = 200; % Generation

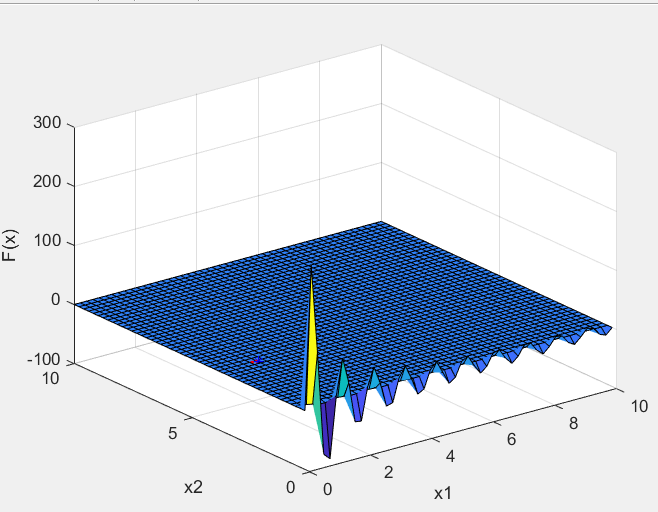
Len1 = 20; % Bits of x1

Len2 = 20; % Bits of x2

lb = [0, 0];

ub = [10, 10]





From the graphs best fit and minimized value of F(x, y )= -

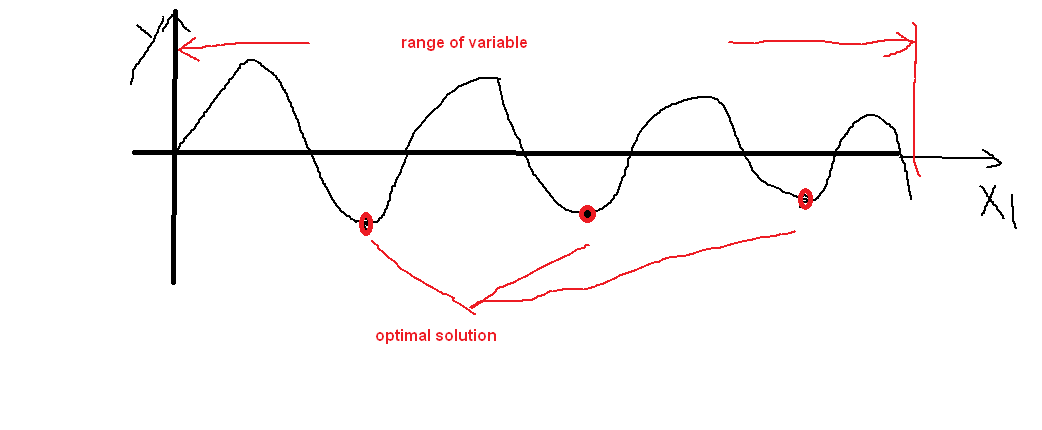
0.10424 and best fit generation is 29

**Observations and conclusions :-**

**From the above it is seen and can be concluded from the above factors that**

* The **minimum value of F(x,y) for 0>= x<= 10 and 0>= y<= 10 is -0.10546 and values for x is 1.2 and y is 3.7**
* **population size** — if your population is too small, the improvement per iteration in the fitness function will be low (measured as the best candidate solutions or the average of solutions). if you increase the population size and the fitness function increases faster then its a sign its suboptimal. also there is a point where increasing the population size does not improve the rate of increase in the fitness function.
* **mutation rate** — if its too low the algorithm will not cover the search space much. if its too high then good candidate solutions will be perturbed and pushed into worse solutions. again basically a rough measure is the rate of improvement in the fitness function.
* **number of generations** - the number of generations is related to improvement in the fitness function. a fitness function usually shows major improvement in early generations and then asymptotically approaches an optimum. after enough trend you can determine what the theoretical optimum is from fitting/extrapolating this line, and how much improvement per time is possible.

**To summarize in gist optimal solution can be explained by graph :-**

****

**EXTRA NOTES USED TO SOLVE**

Please note that the following files have been used in matlab .However for better understanding of stepsthey have been added.. Here for more clarification there are 3 files whose codes have been included but not mentioned explicitly.These are as follows:-

* Roulette
* Tournament
* Individual mutation.

The codes for each of the following are as follows:-

**Roulette code :-**

% Roulette Wheel Selection Operator

function NewGenes = RouletteSelection(Genes, Fits, PopSize)

% Cumulative Probability

CumuProb = zeros(1,PopSize);

for i=1:PopSize

CumuProb(i) = sum(Fits(1:i));

end

MaxProb = CumuProb(PopSize);

% New Genes

NewGenes = Genes;

for j=1:PopSize

P = rand \* MaxProb;

Ind = find(CumuProb > P,1); % return only the first index

NewGenes(j,😊 = Genes(Ind,😊;

end

end

**Tournament code :-**

% Tournament Selection Operator

function NewGenes = TournamentSelection(Genes, Fits, PopSize)

% Binary tournament

NewGenes = Genes;

for i=1:PopSize

r1 = randi([1 PopSize]);

r2 = randi([1 PopSize]);

if Fits(r1) > Fits(r2)

NewGenes(i,😊 = Genes(r1,😊;

else

NewGenes(i,😊 = Genes(r2,😊;

end

end

end

**Individual mutation code:-**

% Individual Mutation Operator

function Genes = IndividualMutation(Genes, Pm)

% Mutation on each individual

[row, col] = size(Genes);

for i=1:row

if rand < Pm

j = randi([1, col]);

Genes(i,j) = ~Genes(i,j);

end

end

end