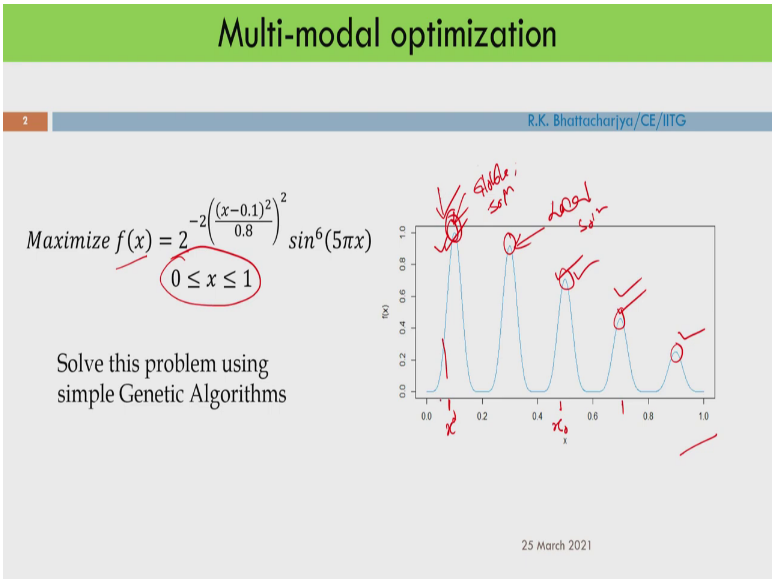
**Optimization Methods for Civil Engineering**
**Prof. Rajib Kumar Bhattacharjya**
**Department of Civil Engineering**
**Indian Institute of Technology, Guwahati**

**Lecture - 21**
**Multi-Modal Optimization**

Hello student, welcome back to the course on Optimization Methods for Civil Engineering. So, today we will discuss Multi-Modal function Optimization using genetic algorithm.

(Refer Slide Time: 00:43)




Here I would like to maximize this particular function the function is f of x it is a 1 dimensional function say 1 it is a 1 variable function and it is a function of x then f of x is 2 to the power minus 2 x minus 1 whole square by 8 then this is whole squared and then sin to the power 6 5 pi x ok.

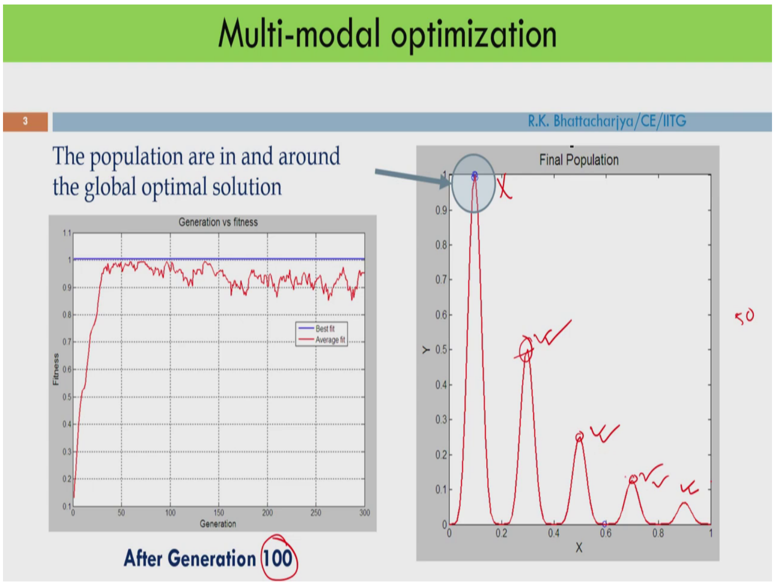
So, and the range of x is between 0 and 1 ok. So, now, if I plot this function so, I have shown you the plot here and you can see there are several optimal point ok. So, these are all optimal point and if I maximize. So, I can say that this particular solution is global solution and others are basically local solution. So, I can say these are all local optimal solution and one of them that this is the base solution and you can say that this solution is the global solution.

So, now if we apply genetic algorithm suppose I would like to solve this problem using genetic algorithm. So, in that case I will get the global optimal solution, that means, I should get this particular solution ok.

Now if I apply classical method as you know that classical in case of classical method you will get different local optimal solution based on the initial solution you have chosen. Suppose, if I show the initial solution somewhere here then suppose this is the x naught then probably you will get this solution, similarly if you do it here you will get it.

But if you are taking any solution in between this ah if you take any solution initial solution somewhere here. So, you will get the global optimal solution, but if you are applying genetic algorithm then you will get the global solution. So, I will get this particular solution.

(Refer Slide Time: 02:42)



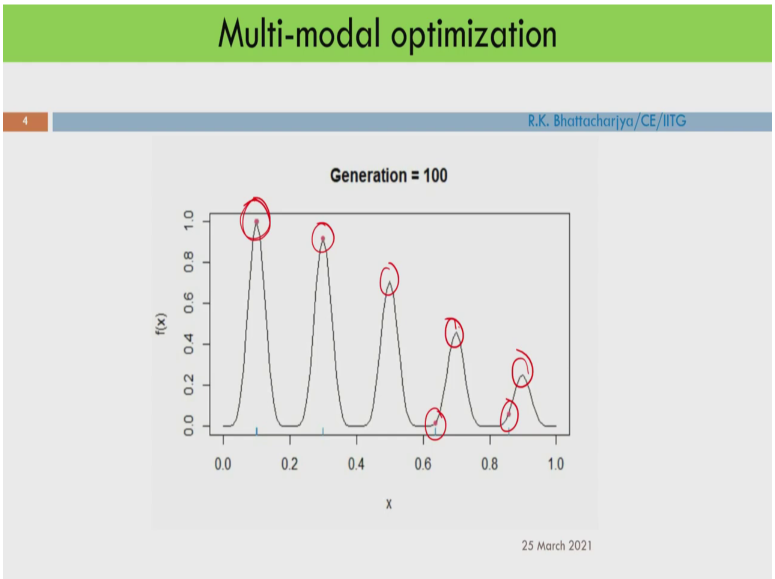

Now, let us apply genetic algorithm. So, I have shown you the results after 100 iteration. So, I have applied genetic algorithm here and you can see that. So, all the population suppose I have taken a population of 50 here, then all these populations are at the global optimal solutions. There is no solution on the local optimal solution.

So, all the population that mean all the solution are basically converging at the global optimal solution. Now, if I want to suppose this solution is not acceptable; that means, this global optimal solution is not acceptable.

So, I need a local optimal solution, then in case of genetic algorithm you will only get the global optimal solution. If suppose this particular solution is not acceptable then if I have any knowledge about the other local optimal solution. So, in that case we can take a decision ok this solution is not acceptable, but I can take this solution.

So, if you have any knowledge about the other local optimal solution ok. So, therefore, if I can find out all the local optimal solutions along with the global optimal solution. So, then it will be easier for me to take a decision. So, therefore, let us see if we can modify the genetic algorithm and can obtain the other local optimal solutions.

(Refer Slide Time: 04:02)

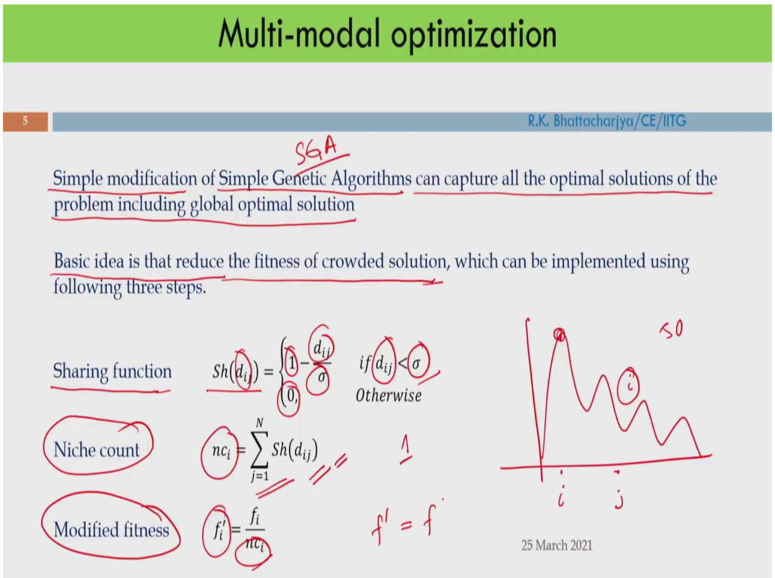



Here I have shown you the 100 generation of genetic algorithm. So, you can see that initially populations were all distributed between 0 and 1 between upper bound and lower bound and with the iteration you can see all these populations are converged to global optimal solutions.

So, there is no solution on the other local optimal solution. So, in 100 iteration you can see that most of the all the populations are at global optimal solution and few solution one solution is here and somewhere one is here and one is here and these solutions you are getting because of mutation ok. So, mutation is trying to search the other solution and therefore, you are getting some solution, but most of the solutions are at global optimal solution.

So, therefore, if we can; if you can modify our simple genetic algorithm and can obtain the other local optimal solution and then basically I can take a decision about the applicability of this optimal solution.

(Refer Slide Time: 05:04)



Now, what I can do? So, simple modification of simple genetic algorithm so, this is SGA Simple Genetic Algorithm can capture all the optimal solution of the problem including the global optimal solution. So, what I will explain here?

So, by with simple modification of the SGA. So, I can capture all the optimal solution of this problem. So, basic idea is that what is the idea basic idea is that you reduce the fitness of crowded solution. That means, if a solution is crowded then you reduce the fitness. So, what I will do basically suppose if I say. So, this is the problem, this is the problem and now what is happening all the populations are somewhere here.

So, in that case what will happen? So, this any solution if you take, so this solution is the crowded solution. Now what I will do? I do not need all the solutions at this particular point. So, I will reduce the fitness of this crowded solution. So, which can be implemented using the following 3 steps? So, what I will do? I have defined a sharing function here. So, this is S h dij. So, dij is the distance I will explain that one.

Now, distance between two solution suppose one solution is here and another solution is here. So, this is your i and this is j. So, distance between these two solutions a is dij. So, dij is the distance, now if the distance is less than sigma. So, sigma is a parameter you have to define that means, I will define that if the dij is less than sigma. So, in that case I will say that the particular solution is a crowded solution.

So, in that case what I will do. So, I will calculate the sharing function value which is equal to 1 minus dij by sigma. So, therefore, if dij is 0, that means, if the solution is two solution is somewhere here both the I distance between i and j is 0 then this function value will be 1 and now otherwise if it is suppose dij is greater than sigma. So, in that case I will put 0. So, I will not say that that particular solution is a crowded solution.

So, what I will do? I will say whether dij is less than sigma if it is less than sigma I will calculate the sharing function value using this equation that is 1 minus dij by sigma or otherwise if it is; if it is more than that more than dij is more than sigma. So, in that case say its that sharing function value is 0. So, I will say that the particular solution is not crowded solution.

So, now what I will do suppose I have total 50 solution, suppose 50 solution and for all these 50 solution I have taken a particular solution I. Now, this solution for this particular solution I i will calculate the distance for all these 50 solution ok. Now, I will get the distance and once I am getting the distance I can calculate the sharing function value and I can say that this solution is, whether this solution is crowded solution or not crowded solution.

After that I am calculating niche count what is niche count niche count is nothing but summation of all this sharing function value. Now, what this niche count is giving? If niche count is a large number or in that case what will happen? I can say that particular solution is a crowded solution. If niche count is large number if niche count is 1 then I can say that particular solution is not a crowded solution that is an isolated solution.

So, therefore, this niche count can be 1 or it can be more than 1. So, if the value is large then in that case I will say that this particular solution is a crowded solution and if niche count is 1 in that case I will say that this particular solution is not a crowded solution. So, now, what I will do basically then after that once I am getting the niche count then I am calculating the modified fitness. So, modified fitness is f dash i and which is basically fi by nci and niche count.

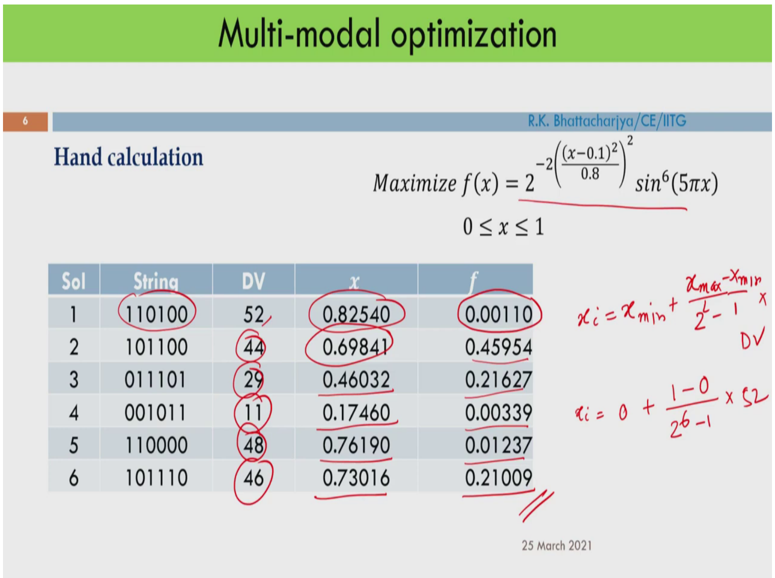
So, now what will happen? If niche count is 1, in that case the modified fitness equal to the original fitness. So, if niche count is 1; that means, that particular solution is not a crowded solution ok. So, therefore, fitness value will not reduce that is an isolated solution.

Now, if it is a crowded solution what I will do? And I will reduce the fitness values. So, what will happen? Eventually that if you are reducing the function value. So, when this solution will go for the cross go for the selection operator then this solution will not be selected.

Because you have reduced the fitness function of the crowded solution. So, the basic idea is that you reduce the fitness value of fitness value of the crowded solution. So, that once it is going to the selection operator this solution will not be selected. So, in comparison to the other optimal solution ok the basic idea is that you reduce the fitness value of the crowded solution.

So, that once the solutions are going to the selection operator this solution will not be selected ok. So, with this simple modification so, I can run my simple genetic algorithm and in that case I will be able to capture all the local optimal solution.

(Refer Slide Time: 10:51)




| Sol | String | DV | $x$ | $f$ |
|---|---|---|---|---|
| 1 | 110100 | 52 | 0.82540 | 0.00110 |
| 2 | 101100 | 44 | 0.69841 | 0.45954 |
| 3 | 011101 | 29 | 0.46032 | 0.21627 |
| 4 | 001011 | 11 | 0.17460 | 0.00339 |
| 5 | 110000 | 48 | 0.76190 | 0.01237 |
| 6 | 101110 | 46 | 0.73016 | 0.21009 |

Now, let us see the hand calculation of this particular problems. I would like to show you how you can reduce the function value of the crowded solution and how you are defining the crowded solution.

So, I have taken the same problem ok. So, for this same problem and in this case I have shown you binary coded, but anyway. So, this concept you can apply in case of real coded also there is no issue suppose these are the 6 solution I have considered.

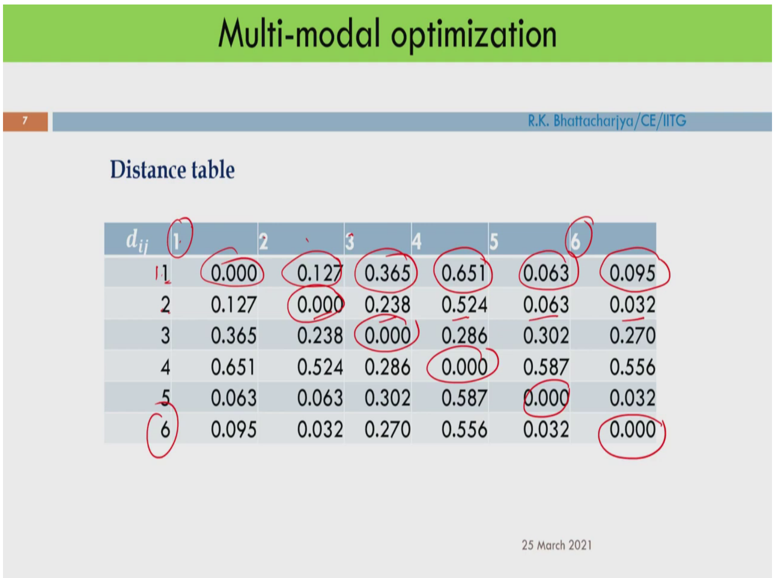
So, this is the string first string that is 1 1 0 1 0 0 and if I decoded value is 52 and the real value is your 0.82540. So, how we have calculated? That is your xi which is equal to x min x min plus this is x max minus x min divided by 2 to the power l minus 1 into decoded value of that particular string.

So, if I do that so, here for this particular means for particular solution 1. So, xi will be, so x min is 0 then plus and this is x max is 1 minus 0 2 to the power 6 now. So, we have 6 bit string minus 1 and decoded value is 52.

So, in that case I should get 0.82540 and for the second one decoded value is 44 and the x value is 0.69841 then third one decoded value is 29 and x value is 0.46032 then fourth one it is 11 and this is 1 0.17460 this is 48 and this is 0.76190 and this is 46 and this is 0.73016.

Now, if I calculate the fitness function that is your f value. So, this is a maximization problem. So, if I calculate. So, for the first solution the fitness value is 0.00110 for a second one 0.45945 third one 0.21 then fourth one 0.00339 and this is 0.01237 and for the last one this is 0.21009 ok. So, I am getting that one. So, now, what I will do? I will try to calculate the crowding distance ok or basically sharing function.

(Refer Slide Time: 13:32)

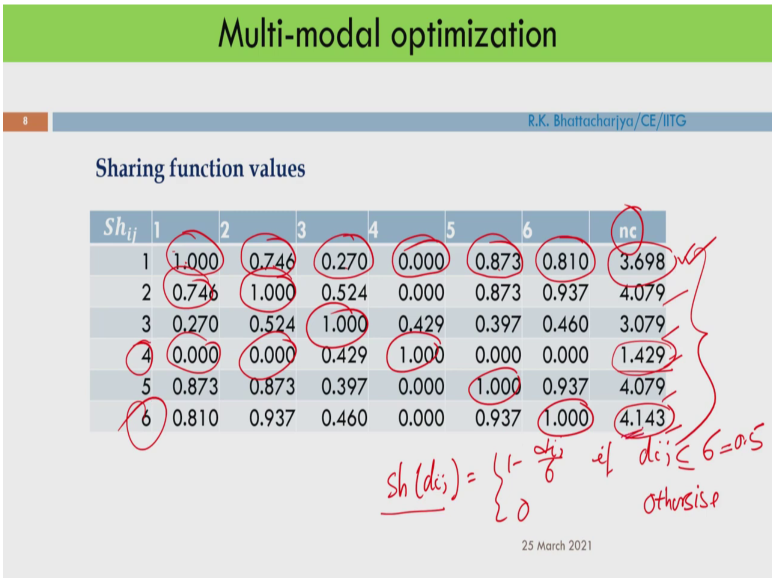



Now, the first step is the you calculate the distance ok. So, in this case I have 6 solution. So, this is 1 to 6 solution and similarly from here 1 to 6 solution and I am calculating the distance between 1 and 1 then 1 and 2 1 and 3 1 and 4 1 and 5 and 1 and 6. Similarly, I am calculating distance between 2 and 1 then 2 and 2 then 2 and 3 2 and 4 2 and 5 and 2 and 6.

So, similarly I am calculating the distance between dij ok. So, these are the distance between the solution. So, you can say that 1 1 distance is 0 and 1 2 distance is 0.127 1 3 distance is 0.365 and the distance between 1 and 4 0.651 distance between 1 and 5 is 0.063 and distance between 1 and 6 is 0.095.

So, you can see the diagonal values are 0 because distance between 2 and 2 is 0 distance between 3 and 3 is 0 4 and 4 is 0 5 and 5 is 0 and 6 and 6 are 0 ok. So, I am getting this distance. So, now, what I have to do basically. So, I can calculate this distance easily. So, there is I think this is not the that difficult, now after calculating the distance. So, you can calculate the sharing function value.

(Refer Slide Time: 14:54)




And that is S h dij equal to 1 minus dij by sigma if dij is less than equal to sigma or less than sigma this is 0 otherwise ok. Now, you can see that the distance between 1 1 1 is 0. So, therefore, dij is 0. So, you are getting assets dij is 1 and here also the distance is less than sigma.

So, in this case I have taken sigma equal to 0.5. So, here it is less than sigma. So, you are 0.746 here also it is less than sigma. So, you are getting 0.270 here it is more than sigma you can see that this value is more than 0.5. So, you can see this value is 0.651.

So, therefore, this particular solution is not a crowded solution with respect to this solution and therefore, assets value is 0 and here this is less than 0.5. So, therefore, it is 0.873 and here also it is less than 0.5 and therefore, this is 0.810. So, similarly I am calculating the sharing function value for other solution and you can see the diagonal values are one because distance is 0 ok. I am getting 0 and 0 here because distance is more than sigma, sigma is 0.5.

So, I have calculate the sharing function value for all these solution. Now, what is niche count nc this is the summation of all the sharing function values. So, therefore, if I add all this thing that 1 plus 0.746 plus 0.27 plus 0.873 plus 0.810.

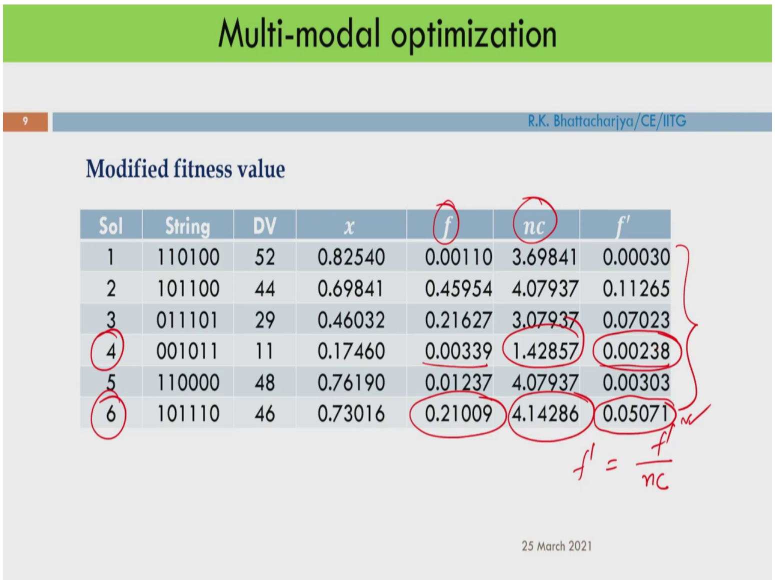
So, I should get 3.698. So, this is summation of sharing function value from 1 to 6 and for this you are getting 4.079 and here you are getting 3.079 here you are getting 1.429 and here you are getting you are getting 4.079 and here you are getting 4.143.

Now, looking at this niche count value so, you can say that solution 4 whose niche count is 1.429 and this solution is an isolated solution. So, it is not crowded solution, but you and what is the most crowded solution? The most crowded solution is the last solution 6 solution whose niche count value is 4.143.

So, therefore, looking at the niche count you can actually tell whether that particular solution is a crowded solution or not crowded solution, if niche count value is 1 certainly you can say that is not a crowded solution. So, there is no solution there to that particular solution.

And if the value is large, so in that case you can tell that is a crowded solution. So, therefore, the solution 4 is an isolated solution I would like to preserve that solution. So, I will reduce the fitness, but fitness will be reduced slightly, but the most crowded solution is for the solution 6 and fitness value of this particular solution will be reduced ok. So, what is next step? The next step is to get the modified fitness value. So, what I will do?

(Refer Slide Time: 18:30)



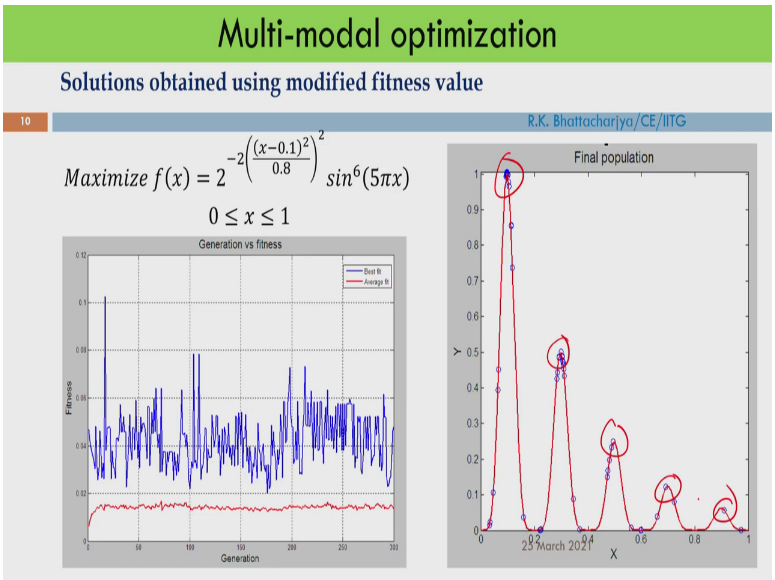

So, I have already calculated the fitness value. So, this is the actual fitness value of the solution and I am also calculating the niche count and now I am calculating the modified fitness value. So, that is equal to original fitness value divided by n c.

So, these are the reduced fitness value. So, you can see that original fitness of solution 6 is 0.21009 and this is a crowded solution and niche count value was very high that is 4.14286. So, now, the function fitness value is reduced to 0.05071. So, fitness value has been reduced for this and you can say that the fourth solution that is that was not a crowded solution and niche count is 1.42 and earlier it was 0.00339 and now you are getting the modified fitness value is 0.00238 ok.

So, fitness value of this particular solution is not reduced has not been reduced ok and for other crowded solution the fitness value has been reduced ok. So, I am calculating the modified fitness value now you apply the simple genetic algorithm. So, other steps are same what we have discussed; that means, you apply the selection operator you applied the crossover operator you apply the mutation operator elitism operators, so whatever. So, there is no sense on this particular operator.

So, only thing is that we have defined a modified fitness value. So, we have reduced the fitness value of the crowded solution and for the non crowded solution we have not reduced the fitness values this is the idea to preserve the other local optimal solutions.

(Refer Slide Time: 20:17)




So, now I have applied the genetic algorithm on this problem again and you can see that now I have some solution somewhere here. So, these are also optimal solution along with this along with the global optimal solution I could also preserve the other local optimal solution ok.

So, we have modified the genetic algorithm simple genetic algorithm in order to capture the other local optimal solution of the problem. So, you can also try these things only thing is that you have to define modified fitness function, as I have shown in the hand calculation.

Thank you.